Adaptive Robust Model Predictive Control for Nonlinear Systems

by

Brett T. Lopez

S.M., Massachusetts Institute of Technology (2016)
B.S., Aerospace Engineering
University of California - Los Angeles (2014)

Submitted to the Department of Aeronautics and Astronautics
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2019

Signature redacted

Author ....
Department of Aeronautics and Astronautics
May 23, 2018

Signature redacted

Certified by............... ....
Jonathan P. How
R. C. Maclaurin Professor of Aeronautics and Astronautics, MIT
Thesis Supervisor

Certified by................. Signature redacted ......
Jean-Jacques E. Slotine
Professor of Mechanical Engineering and Information Sciences, MIT

Certified by................................. Signature redacted
Nicholas Roy
Professor of Aeronautics and Astronautics, MIT

Accepted by .... Signature redacted
Sertac Karaman
Associate Professor of Aeronautics and Astronautics
Chair, Graduate Program Committee

Adaptive Robust Model Predictive Control for Nonlinear Systems

by

Brett T. Lopez

Submitted to the Department of Aeronautics and Astronautics
on May 23, 2018, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Modeling error and external disturbances can severely degrade the performance of Model Predictive Control (MPC) in real-world scenarios. Robust MPC (RMPC) addresses this limitation by optimizing over control policies but at the expense of computational complexity. An alternative strategy, known as tube MPC, uses a robust controller (designed offline) to keep the system in an invariant tube centered around a desired nominal trajectory (generated online). While tube MPC regains tractability, there are several theoretical and practical problems that must be solved for it to be used in real-world scenarios. First, the decoupled trajectory and control design is inherently suboptimal, especially for systems with changing objectives or operating conditions. Second, no existing tube MPC framework is able to capture state-dependent uncertainty due to the complexity of calculating invariant tubes, resulting in overly-conservative approximations. And third, the inability to reduce state-dependent uncertainty through online parameter adaptation/estimation leads to systematic error in the trajectory design. This thesis aims to address these limitations by developing a computationally tractable nonlinear tube MPC framework that is applicable to a broad class of nonlinear systems.

Thesis Supervisor: Jonathan P. How
Title: R. C. Maclaurin Professor of Aeronautics and Astronautics, MIT

Acknowledgments

I am so incredibly lucky to have had the opportunity to pursue my masters and doctorate degree at MIT. Oddly enough, the road to this point started in a small town near Anchorage, Alaska where I was trying to make a junior hockey team, of all things. It has been a long, difficult, and incredibly fun road from that point to now; with so many people I have had the privilege of working with and who have helped me along the way. I will do my best to recognize them here.

The first person I would like to thank is my advisor Professor Jonathan How. I cannot thank you enough for giving me the opportunity to come to MIT and work on so many exciting projects. I have a tremendous amount of respect for you and cannot express enough gratitude for training me to be the researcher I am today. No matter how busy you were, you always made time to meet with me and discuss new ideas. Very few professors are as dedicated as you are to your students. Thank you for all your hard work, believe me when I say it did not go unnoticed.

Thank you to Professor Jean-Jacques Slotine and Professor Nick Roy for serving on my thesis committee. Having the opportunity to work with both of you individually was incredibly rewarding. Thank you for always making time to meet one-on-one to discuss new research ideas. Your genuine excitement for my work and constant encouragement motivated me to become a better researcher. Many thanks to my thesis readers Dr. Kaveh Fathian and Dr. Markus Ryll for providing great feedback. Hopefully that exercise was not too painful...

To everyone in the Aerospace Controls Lab, it was awesome to work with and get to know you. Thank you Dr. Shayegan Omidshafiei, Dr. Kasra Khosoussi, Michael Everett, Kris Frey, and Parker Lusk for making those long, stressful nights working so much more enjoyable and providing advice on research and life.

A special thank you to my two closest friends Byron Patterson and Jon Pounds. Byron, you always provided the comic relief I needed to get through the day. Thanks for sharing so many fun hours in the gym, running around Boston, and on the phone talking about new ideas and life in general. Jon, whether it was lifting, playing hockey, or just getting food, I could always count on you for a good laugh and fun debates on random topics. Keep pushing

toward your goals and everything will work out.

Big thank you to the MIT men's hockey team for letting me get back on the ice. It was an awesome feeling to play competitive hockey again, especially with such a good group of people. Special thanks to the coaching staff Tommy Hopkins, Dave Hunter, and Nick Sisler and teammates Keith Galli, Aaron Goodman, Henry Merrow, Jon Pounds, and George Roch. I will definitely miss being out there with you guys but maybe now one of you will win in showdown.

Special thanks to Professor Ann Karagozian and Dr. Tama Hasson for providing the opportunity to do research at UCLA during my undergraduate. Another special thanks to Arturo Hernandez, director of the MESA center at El Camino Community College. I definitely would not be where I am today without the help, guidance, and encouragement you all provided.

Thank you to my family for always supporting my endeavors. Whether it was pursuing hockey or my doctorate, you were always there to support me in any way possible. In particular, to my grandparents Richard and Dolores Dominetta and Gus and Mary Lopez, thank you for doing everything you could to make sure us grandkids had every opportunity to be successful, for always being so positive, and sharing your wisdom, especially during the hard times. To Suzanne Gates, Amy Dominetta, and Carol and Dave Mazzotti, thank you for always believing in me, helping me even when I did not know I needed it, and keeping me on the road to success. To my parents Linda Dominetta and Dave Lopez, thank you for teaching me the importance of hard work and to not give up when things got hard.‘ I hope I have made every one of you proud.

Last, and by no means least, I would like to thank my wonderful partner Kimiko Kossler. Thank you for always supporting me in whatever endeavor I pursued, and especially during the stressful times of my degree. Being able to come home to you and our pups, Koda, Penny, and Franklin, made my life so much more enjoyable. I cannot wait to take the next step in this wonderful adventure with you.

This work was supported by the National Science Foundation Graduate Research Fellowship under Grant No. 1122374, by the DARPA Fast Lightweight Autonomy (FLA) program, by the NASA Convergent Aeronautics Solutions project Design Environment for Novel Vertical Lift

Vehicles (DELIVER), and by ARL DCIST under Cooperative Agreement Number W911NF-17-2-0181.

If there is no struggle, there is no progress.

Frederick Douglass

...so that his place shall never be with those cold and timid souls who neither know victory or defeat.

Theodore Roosevelt

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Model Predictive Control (MPC) has emerged as a fundamental control strategy because of its ability to handle constrained multi-variable systems. Using a model of the dynamics, MPC solves a constrained optimal control problem to generate both a sequence of open-loop control actions and future (i.e., desired) states. By repeatedly performing the optimization with the current state of the system, MPC defines an implicit control law with robustness and stability properties. MPC can then be viewed as simultaneously generating a reference trajectory and feedback control law. This key attribute, and the development of fast optimization solvers, has made MPC a staple of robotics [1, 2], aerospace [3, 4], process control [5], operations control [6], and many other engineering/non-engineering fields.

MPC heavily relies on a model to predict the future state of the system, making it susceptible to unknown external disturbances and modeling error. Consequently, severe performance degradation or instability can occur if uncertainty is not directly accounted for in the optimization [7]. Robust MPC (RMPC) was developed to explicitly account for uncertainty by optimizing over *control policies* instead of control actions but at the expense of increased computational complexity [7–9]. A number of approximate solution strategies for RMPC have been proposed to regain tractability but few are applicable to nonlinear systems with real-time constraints. In this thesis, a robust MPC strategy that accounts for uncertainty, both from external disturbances and modeling error, is developed for a broad class of nonlinear systems while maintaining computational tractability for real-world applications.

1.2 Problem Statement

Due to the computational demands of RMPC, the controls community has primarily focused on a decoupled strategy, known as tube MPC, that involves solving the nominal MPC problem online and designing a robust tracking controller offline [7, 10–12]. The controller is designed to keep the system in a tube, invariant to uncertainty, centered around the desired trajectory. While this decoupled design is computationally tractable, it is inherently suboptimal since the nominal MPC optimization is unable to modify the tube geometry to satisfy changing constraints or objectives [13–16]. Further, many state-of-the-art tube MPC algorithms are unable to leverage knowledge of state-dependent uncertainty (i.e., parametric modeling error) due to the complexity of constructing invariant tubes, leading to overly conservative approximations.

Independent of explicitly capturing state-dependent uncertainty, the inability to update the prediction model online leads to systematic error in the trajectory design; ultimately limiting the performance of MPC. Ideally, if the uncertainty is parametric, an estimation scheme could be used to obtain a more accurate model. The difficulty, however, lies in ensuring state/actuator constraints remain satisfied as the model is being updated online [17]. This is particularly challenging for nonlinear systems where the separation principle does not hold and constructing a stabilizing controller for *all* model realizations is difficult. While some work has been done in adaptive nonlinear MPC [17–21], the results are mostly theoretical and intractable for real-time systems.

This thesis addresses the current limitations of tube and adaptive MPC for nonlinear systems. Specifically, a framework for nonlinear systems is developed that can: 1) optimize the tube geometry to satisfy changing constraints and objectives; 2) explicitly capture state-dependent uncertainty to design more intelligent trajectories; and 3) estimate model parameters to reduce model uncertainty while maintaining stability and recursive feasibility. The framework is applicable to feedback linearizable (see Chapter 2 for a thorough discussion on feedback linearization), minimum-phase, and cascaded nonlinear systems which encompasses many real-world systems. There were several conceptual and computational challenges that had to be addressed in developing this framework:

- *Tube Geometry (Controller) Optimization:* The formal procedure to account for uncertainty in MPC is to simultaneously generate a trajectory and sequence of control policies. Optimizing over arbitrary functions is not computationally tractable, leading to a decoupled trajectory and controller design strategy that is inherently suboptimal [8, 9, 11, 12]. One approach to address the duality gap introduced by the decoupled design is to optimize over a set of parameterized linear controllers [13–15]. However, constructing a set of linear controllers that maximizes the region of attraction for a nonlinear system is computationally intensive and requires knowing the desired trajectory *a priori* [22–24]. Selecting a nonlinear control synthesis strategy that was easily parameterized and applicable to a variety of nonlinear systems was paramount to obtain a tractable strategy that best approximated the original robust formulation.

- *State-Dependent Uncertainty:* Explicitly considering state-dependent model uncertainty in tube MPC is challenging because it requires deriving an explicit relationship between the uncertainty and tube geometry. Existing tube MPC algorithms circumvent this complexity by simply assuming the uncertainty is bounded, leading to overly conservative solutions. Recently, scenario-based MPC [25], which entails forward simulating a finite number of realizations (or scenarios) of the uncertainty and checking for constraint violation, has been used to capture state-dependent uncertainty [26]. However, the number of samples required to statistically verify the constraints are satisfied can be substantial [25, 26]. Selecting a nonlinear control strategy that lead to a simple relationship between uncertainty and tube geometry was required to reduce conservativeness and computational complexity when compared to existing techniques.

- *Model Estimation:* Capturing state-dependent model uncertainty in MPC can significantly reduce conservativeness but better performance can be obtained by updating the prediction model online with adaptation/estimation. Incorporating adaptation, based on execution error, in nonlinear MPC is a challenging proposition since recursive feasibility of the optimization requires predicting how the model parameters will evolve along the trajectory. Minimax optimization has been the only proposed method that achieves recursive feasibility for nonlinear adaptive MPC but is too computationally expensive for real-time use [17, 18, 20, 21]. Alternatively, model parameters can be estimated by executing specialized trajectories to maximize parameter observability. However, nonlinearities in the dynamics and measurement model pose significant challenges for computing and optimizing observability, often leading to approximate approaches [27–31]. Set membership identification [32–35], which is closely related to the concept unfalsification [36, 37], is a different strategy that constructs an "unfalsified" set of model parameters. This approach has several nice properties, such as guaranteed recursive feasibility and robustness to unmodeled disturbances, but has only been applied to linear systems. Selecting a nonlinear control and adaptation/estimation strategy that ensured stability and recursive feasibility for constrained nonlinear systems was needed to address the deficiencies of current approaches.

1.3 Existing Gaps

This section presents a brief literature review of relevant material. A more thorough review can be found in Section 1.4.

1.3.1 Tube MPC

Robust Model Predictive Control (RMPC) is one strategy that accounts for uncertainty by simultaneously generating a trajectory and sequence of control policies [7, 10, 11]. As stated above, this approach is not computationally tractable because it entails optimizing over arbitrary functions. Tube MPC regains tractability by decomposing RMPC into an open-loop optimal control problem and design of an ancillary tracking controller [8, 12, 38]. The ancillary controller ensures the system remains in a tube, invariant to uncertainty, around the trajectory; ensuring feasibility of the open-loop problem given appropriate state/control constraint tightening. Various procedures for designing the ancillary control and computing the associated invariant tube for linear and nonlinear systems have been reported in the literature [12, 39–43]. Fundamentally, though, the proposed decomposition introduces a duality gap that can be non-trivial. Homothetic [14], Parameterized [13], and Elastic [15] Tube MPC have been proposed to bridge the duality gap by giving the optimizer additional flexibility to change control parameters. However, the feedback controller used in these works is only applicable for linear systems, and no advancements have been made for nonlinear systems. Furthermore, all of the aforementioned approaches are unable to capture state-dependent uncertainty, leading to overly conservative approximations.

1.3.2 Adaptive MPC

The effectiveness of RMPC (and its approximate solution strategies) is limited by the quality of the uncertainty description and prediction model. Consequently, the inability to update the prediction model degrades the performance of RMPC and its alternatives. Adaptive MPC addresses this deficiency by updating the model online using an adaptation law based on execution error. There are a plethora of linear adaptive MPC approaches, all of which rely on imposing periodicity of the control input (i.e., persistent excitation [44, 45]) to guarantee parameter convergence [46–51]. The difficulty of using adaptation in nonlinear MPC is that, in general, the separation principle does not hold so adaptation along the prediction horizon must then be considered to guarantee stability and recursive feasibility [17]. Adetola et al. [17, 21] proposed a minimax formulation where the cost function was maximized with respect to the model parameter set and minimized with respect to control policies. Maximizing with respect to the model parameter set ensures the state constraints are satisfied for the worst possible state realization. However, minimax optimization is too computationally intensive for real-time use. Furthermore, even though the lack of robustness in adaptive control is well documented [52], existing approaches must neglect the presence of unmodeled dynamics and disturbances to prove stability, establish recursive feasibility, and guarantee parameter convergence. Adaptive tube MPC [48, 53], which treats the adaptation as a disturbance, has shown promise but has yet to be applied to constrained nonlinear systems.

1.4 Literature Review

1.4.1 Robust Model Predictive Control

Model Predictive Control (MPC) relies heavily on a model to accurately predict the future state of the system, making it susceptible to unknown external disturbances or modeling error. Severe performance degradation and instability can occur if uncertainty is not directly accounted for. Methods that address the aforementioned issues can be broadly categorized as Robust MPC (RMPC) [56] or Stochastic MPC (SMPC) [57] techniques. The former, which is the focus of this thesis, assumes the uncertainty is bounded while the latter assumes a probabilistic description of the uncertainty. RMPC accounts for uncertainty by optimizing over control policies instead of control actions. While changing the decision variable is well motivated, optimizing over arbitrary functions is generally not computationally tractable. The remainder of this subsection presents the main strategies used to solve, exactly or approximately, RMPC problems.

Minimax MPC

Minimax optimization minimizes a cost function for the worst-case uncertainty realization, and was proposed to solve dynamic programming problems with bounded uncertainty [8, 9]. The close connection between MPC and dynamic programming, where the former is an approximation of the latter [58], has lead to similar formulations for solving RMPC problems. Minimax MPC techniques can be classified as either *open-loop* or *closed-loop*, the difference being whether control actions or control policies are the decision variables, respectively. *Open-loop* minimax MPC is more computationally tractable but suffers from feasibility issues since finding a sequence of control actions that satisfy state and actuator constraints for *all* realizations of the uncertainty is difficult [59, 60]. *Closed-loop* or *feedback* minimax MPC addresses the feasibility issues at the expense of computational complexity. Several results and procedures have been presented for linear systems in an attempt to regain tractability of feedback minimax MPC. Scokaert et al. [60] showed that the number of uncertainty enumerations can be reduced to only the extrema by leveraging convexity of linear dynamics, leading to a finite-dimensional optimization problem. In addition, a dual-mode minimax and linear controller was shown to stabilize the systems for arbitrary initial conditions. Bemporad et al. [61] showed that a set of linear controllers can be parameterized for online use if the system's dynamics and cost function are linear. Lofberg et al. [62] used semidefinite relaxations and robust optimization techniques to obtain approximate minimax optimization solutions. While all the aforementioned works present nice theoretical results, they are still too computationally complex for real-time use and are restricted to linear systems. Further, results for nonlinear systems predominately use a combination of the above techniques but with linearized dynamics [63] and therefore suffer from the same computational complexity issues.

Tube MPC

Tube MPC is a strategy that decomposes RMPC into an open-loop MPC problem and design of an ancillary controller that provides robustness to uncertainty [7, 10]. The ancillary controller keeps the system in a robust invariant set, known as a *tube*, around the desired trajectory. Bertsekas et al. [8] were the first to present the idea of a tube and its role in stochastic dynamic programming. Intuitively, they reasoned that a stabilized system with uncertainty converges to a region (tube) around the desired state, as opposed to the standard notion of stability where the system converges to an equilibrium point. Because of the close connection between MPC and dynamic programming [58], tubes have become a powerful concept for developing strategies that enable MPC algorithms to account for uncertainty.

The computational complexity of RMPC is often not tractable because it entails optimizing over arbitrary feedback policies. The seminal papers by Rossiter et al. [64] and Langson et al. [38] instead proposed to restrict the feedback control policies to be the superposition of a feedforward term and linear feedback (ancillary) controller. Through this parameterization, the decision variables become control actions (i.e., feedforward) with the ancillary controller restricting the spread of trajectories [38]. The goal of tube MPC, then, is to design an ancillary controller that keeps the system in the smallest possible tube around the trajectory. Further, by tightening the state and control constraints to account for tracking error and control effort due to feedback, one can establish asymptotic stability [38]. Numerous papers, surveys, and books have been written discussing the stability, feasibility, and performance of linear tube MPC [10, 55, 59]. While this is an effective strategy to achieve robustness to uncertainty, decoupling the nominal MPC problem and controller design introduces a performance duality gap. Specifically, as first shown by Raković et al. [14], the region of attraction can be enhanced by parameterizing the problem with a feedforward term *and* tube size. Fundamentally, the authors proposed treating the state and control tubes as homothetic copies of a fixed cross-section shape, enabling the problem to be parameterized by the tube's centers (i.e., feedforward term) and a cross-section scaling factor; the approach was named homothetic tube MPC (HTMPC). The authors later extended their previous work to tubes with varying shapes, known as elastic tube MPC (ETMPC), but at the expense of higher computational complexity [15]. Both HTMPC and ETMPC possess strong theoretical properties that have the potential to significantly enhance performance. However, experimental results have yet to be published and nonlinear extensions have yet to be proposed.

Nonlinear tube MPC is significantly more challenging than its linear counterpart due to the complexity of designing a nonlinear controller and computing the associated invariant set. Nonetheless, several strategies have been proposed, with recent works leveraging new nonlinear control synthesis methods. Mayne et al. [65] proposed using a two-tier MPC architecture where the nominal MPC problem, with tightened constraints, is first solved followed by ancillary problem that drives the current state to the nominal trajectory. Linear reachability theory is another strategy that has been used to calculate tubes [66] and was one of the first methods to be tested in hardware on ground [67] and aerial vehicles [40]. Unsurprisingly, though, linearizing the dynamics and treating nonlinearities as bounded disturbances leads to overly conservative behavior. Because of its strong robustness properties, sliding mode control [68] has been proposed as a suitable ancillary controller for nonlinear tube MPC [39, 69–74]. The work by Muske et al. [69] is of particular interest because the parameters of the sliding surface were selected by the MPC optimizer to achieve minimum time state convergence. Majumdar et al. [41] constructed ancillary controllers for nonlinear systems via sum-of-squares (SOS) optimization that minimized funnel size (akin to a tube). The method, however, required a pre-specified trajectory library making it highly dependent on the choice of path/motion planner. Further, the method entailed an extremely time consuming offline computation phase (20-25 minutes per trajectory). Singh et al. [42] proposed using control contraction metrics [75], a control synthesis strategy that only requires the system to be stabilizable, to construct tubes and showed their approach increases the region of feasibility for the optimization. The several recent results listed above have begun to address the challenges of nonlinear tube MPC, but more work is required to expand the results to systems with state-dependent uncertainty and to address the optimality gap introduced by the decoupled design.

Constraint Tightening MPC

As the name suggests, constraint tightening MPC utilizes tightened state constraints to achieve robustness to uncertainty [76]. Similar to tube MPC, this paradigm uses a nominal model but monotonically tightens state constraints along the prediction horizon. The degree to which constraints are tightened depends on the convergence rate of the system and the magnitude of the disturbance. This procedure, in essence, captures the spread of trajectories due to uncertainty and alleviates the computational complexity of computing robust control invariant sets as required in tube MPC [77]. Chisci et al. [78] developed a state-space framework that, for any arbitrary stabilizing linear controller, guarantees convergence to the smallest possible invariant set around the origin. This work was extended by Richards et al. [77] to include time-varying linear controllers and showed the extra degree of freedom enlarged the feasibility region. Recent results have emerged for nonlinear systems where incremental stability [79] is leveraged to obtain a lower-bound on the convergence rate of the system [80]. While this approach shows promise, more work is required to expand the technique to a broader class of nonlinear systems.

Scenario-Based MPC

Instead of deriving an explicit bound on the system's state (as in tube MPC), scenario-based MPC relies on simulating randomly-sampled realizations (or scenarios) of the uncertainty to calculate the spread of trajectories [25, 81]. By sampling from the uncertainty set, the challenge of calculating invariant tubes for complex nonlinear systems is eliminated. For instance, Garimella et al. [26] sampled from the uncertainty set to estimate the dispersion of trajectories for high-dimensional nonlinear systems with state-dependent uncertainty. Until recently (e.g., see [16]), the results presented by Garimella et al. were not possible using tube MPC. The key drawback of this approach is that the number of samples required to statistically verify the constraints were satisfied can be substantial. This is especially problematic for computationally constrained systems with complex dynamics. While there has been several advances in intelligent sampling, any type of online sampling procedure is going to be too intensive without sophisticated computing techniques and hardware (i.e., parallel computing via a GPU).

1.4.2 Adaptive MPC

The inability to improve the prediction model in RMPC formulations fundamentally limits their performance. Adaptive MPC (AMPC) uses ideas from the adaptive control literature [44, 82] to improve the model through a parameter update law based on execution error; eliminating the suboptimal performance caused by modeling error. Linear AMPC heavily relies on the notion of persistent excitation (PE), which is a metric that describes the richness or periodicity of a signal and is a necessary condition to obtain accurate parameter estimates [44, 45, 82]. AMPC must balance two competing objectives: 1) regulate the system to achieve stability and 2) excite the system to obtain accurate parameter estimates. Simultaneous regulation and excitation is known as the dual control problem and is, in general, considered intractable [51]. Nonetheless, a number of approximate strategies have been developed. Genceli et al. [47] were the first to develop an AMPC framework by imposing a PE constraint on the control input to achieve accurate parameter identification. Their work was later extended by Shouche et al. [46] to deterministic auto-regressive moving average (DARX) processes to reduce the optimization's dimensionality. Marafioti et al. [51] showed that the hard PE constraint used by Shouche et al. could be relaxed, resulting in a formulation that did not strictly enforce periodicity of the control input. Hernandez et at. [48] used a linear tube MPC approach that treated the PE part of the input as a bounded disturbance. By using a tube MPC framework, Hernandez et al. were able to prove stability and recursive feasibility for constrained linear systems with PE; a result that other works were not able to obtain. Gonzalez et al. [49] developed a framework where the system converged to an invariant set inside which PE was safely applied. This work was extended by Anderson et al. [50] by explicitly considering parametric uncertainty (as opposed to simply additive uncertainty) and using probabilistic invariant sets to reduce conservativeness.

Nonlinear AMPC is significantly more challenging than its linear counterpart because 1) the separation principle does not hold and 2) the theoretical results for PE are, in general, not applicable to nonlinear systems. As a result, there has been little work on nonlinear AMPC. Mayne et al. [19] implemented a receding horizon estimation scheme that globally stabilized nonlinear systems. However, the authors assumed the estimate converges in finite time to prove stability and feasibility (i.e., constraint were not violated); performance guarantees, including feasibility, before convergence could not be established. DeHaan et al. [18, 20] and Adetola et al. [17, 21] instead used adaptation to shrink the uncertainty set, which contains the uncertain parameters, instead of estimating the parameters directly. Fundamentally, the adaptation scheme eliminates regions of the uncertainty set where the parameters are known *not* to lie. To prevent state and control constraint violation, the authors proposed using a minimax optimization where the effects of estimation were explicitly minimized. The computational complexity of the minimax optimization, which the authors admit, is too computationally intensive for real-time use. A tractable nonlinear AMPC approach that does not rely on unrealistic assumptions has yet to be developed.

1.4.3 Active Estimation

Proprioceptive Methods

Proprioceptive active estimation is concerned with executing maneuvers that improve the system's observability. In general, the separation principle for control and estimation does not hold for nonlinear systems making a coupled approach necessary for optimal performance. However, such an approach requires more than the observability rank condition used in classical control, and requires a *measure* of observability. Krener et al. [28] proposed using the minium singular values or the condition number of the observability gramian as measures of observability. The observability gramian is constructed by linearizing the dynamics and measurement model about a desired trajectory and integrating over a time horizon. The authors made a number of useful suggestions for practical use, such as scaling the state coordinates so the relative size of the singular values are comparable. Hinson et al. [29] further explored the properties of the observability singular values and their relation to the estimation covariance. An analytic solution to the observability optimization problem of a nonholonomic integrator was also presented. Hausman et al. [30] later used the observability gramian to develop an observability-aware trajectory optimization framework for self-calibration of nonlinear systems, and was shown to outperform other methods in terms of accuracy and computation time when tested on a simulated quadrotor with a GPS-IMU sensor suite. Priess et al. [31] expanded the work to consider observability throughout the entire operation (not just during calibration phase) and added obstacle constraints. More accurate estimates were obtained when tested on a simulated quadrotor using a loosely-coupled visual-inertial state estimator; a good platform choice since monocular visual-inertial systems are only observable with non-zero acceleration [83, 84]. The main limitation, though, of using the observability gramian is the necessity to linearize the dynamics and measurement model about a desired trajectory; leading to linearization error and requiring an iterative optimization framework. Further, as shown by Rafieisakhaei et al. [85], using the observability gramian, instead of using the estimation covariance, can generate less accurate estimates when initial, process, or measurement noise is present.

Exteroceptive Methods

Exteroceptive active estimation, more commonly known as belief space planning, leverages information about the environment to steer the system to areas that improve estimation accuracy. Bry et al. [86] developed the rapidly exploring random belief trees (RRBT) algorithm which considers state-dependent stochasticity in both dynamics and measurements. Simulation results demonstrated RRBT's ability to balance information gathering to reduce uncertainty and generate low-cost paths to a goal. Achtelik et al. [87] proposed an extended version of RRBT framework that accounted for complex dynamics while maximizing information gain. The approach was shown to reduce the estimation covariance both in simulation and hardware but was unable to run in real-time. Indelman et al. [88] presented a continuous-time formulation that was shown to generate smooth trajectories that maintained estimation uncertainty bounds. Costante et al. [89] developed a framework suited for vision based-systems where both geometric and photometric (i.e. texture) information about the environment were leveraged to find a minimum uncertainty path to a goal. Simulation and hardware results showed that longer but lower uncertainty paths where generated when compared to non-perception-aware planning methods. While all of the aforementioned methods have been shown to improve estimation accuracy, the need to have a detailed description of the environment is a severe limitation. For receding horizon applications, these methods will be most useful for when the vehicle revisits known areas. Furthermore, efficient techniques for extracting and storing pertinent information about the environment are needed to meet the requirements of real-time, receding horizon implementations.

1.5 Technical Contributions and Thesis Structure

1.5.1 Contribution 1: Combined Trajectory and Controller Design

A nonlinear tube MPC framework where both the trajectory and controller are designed *simultaneously* is developed to reduce the performance gap introduced through the standard decoupled approach. Previous works have focused on linear systems where optimizing over parameterized linear controllers was shown to improve performance. While conceptually this idea should be applicable to nonlinear systems, the complexity lies in finding a parameterization that is computationally tractable for online use and is generalizable to a broad class of systems. This contribution expands previous works [14, 15] to nonlinear systems by using boundary layer sliding control [68, 90] to construct parameterized robust nonlinear controllers.

The primary benefit of this contribution is the ability to dynamically change control performance and maximize responsiveness to changing operation specifications. This is especially important for energy-limited systems where precise tracking (i.e., high-bandwidth) control can limit operation time and may not always be needed (e.g., robot navigating through environment with varying obstacle density).

1.5.2 Contribution 2: Leveraging State-Dependent Uncertainty

A nonlinear tube MPC framework that leverages state-dependent uncertainty is developed to reduce the conservativeness introduced by simply assuming the uncertainty is bounded. Existing state-of-the-art nonlinear tube MPC algorithms are unable to establish an explicit relationship between uncertainty and tube geometry, leading to overly conservative simplifications. Unsurprisingly, the difficulty in establishing such a relationship is a direct result of the chosen control synthesis methodology. Recent work has shown that state-dependency can be leveraged via forward simulating different realizations of the uncertainty but at the expense of computational complexity. This contribution addresses the limitations of existing nonlinear tube MPC approaches by using boundary layer sliding control to construct an explicit relationship between uncertainty and the desired trajectory.

The primary benefit of this contribution is the ability to construct more intelligent trajectories by using a higher quality description of the uncertainty in the MPC optimization without a substantial increase in computational complexity. This is especially important for systems where acquiring a high-fidelity model is too time consuming or not realistic.

1.5.3 Contribution 3: Model Estimation with Set Membership Identification

A nonlinear adaptive tube MPC framework that estimates unknown model parameters is developed to improve the prediction model accuracy and reduce conservativeness. Previous works in linear and nonlinear adaptive MPC have relied heavily on persistent excitation to prove parameter convergence. However, these approaches ignore the presence of unmodeled dynamics/disturbance to achieve parameter convergence and often neglect state/actuator constraints. Adaptive tube MPC has shown promise, but has yet to be applied to nonlinear systems due to: 1) the difficulty in constructing stabilizing controllers for all model parameter realizations; and 2) the inability to derive an explicit relationship between the tube geometry and model uncertainty. This contribution addresses these two limitations by combining boundary layer sliding control and set membership identification to update the prediction model while maintaining recursive feasibility and computational tractability.

The primary benefit of this contribution is the ability to robustly reduce model uncertainty through set membership identification. This is especially important for systems where certain model parameters may be difficult to obtain *a priori* but have a significant impact on system performance.

1.5.4 Contribution 4: Simulation Experiments

The primary objective of this thesis was to develop a nonlinear adaptive tube MPC framework that addresses the deficiencies of existing approaches. To this end, simulation experiments of the proposed work are crucial for demonstrating the work's utility. Simulation experiments are conducted on a nonlinear drag, double integrator system performing collision avoidance and the nonlinear lateral dynamics of a high-performance aircraft. Model parameters, disturbances, and noise were chosen in an effort to generate realistic results.

1.5.5 Thesis Structure

The rest of the thesis is structured as follows:

- Chapter 2 presents the necessary mathematical background information for MPC, robust MPC, and tube MPC. Input-output and input-state feedback linearization, two common nonlinear control strategies, are also discussed.

- Chapter 3 introduces the Dynamic Tube MPC (DTMPC) framework that utilizes the robustness properties of boundary layer control to: 1) construct a parameterization that enables online optimization of the trajectory and control parameters (Contribution 1); and 2) derive an explicit relationship between the tube geometry and state-dependent model uncertainty (Contribution 2).

- Chapter 4 presents simulation results demonstrating DTMPC's ability to simultaneously optimize a trajectory/tube geometry and leverage state-dependent uncertainty. Test domains include collision avoidance of a nonlinear mechanical system and generating/executing aggressive maneuvers for an unstable, nonlinear model of the roll dynamics of a high-performance aircraft (Contribution 4).

- Chapter 5 introduces the Adaptive Dynamic Tube MPC (ADTMPC) framework that utilizes set membership identification (SMID) to reduce model uncertainty while maintaining computational tractability and recursive feasibility (Contribution 3).

- Chapter 6 presents simulation results showcasing how ADTMPC can robustly update the prediction model, leading to better overall performance. Test domains include collision avoidance of a nonlinear mechanical system and generating/executing aggressive manuevers for an unstable, nonlinear model of the roll dynamics of a high-performance aircraft (Contribution 4).

- Chapter 7 summarizes the thesis contributions and discusses directions for future work.

Chapter 2

Mathematical Preliminaries

2.1 Overview

This chapter contains the necessary mathematical background for MPC and the nonlinear control technique known as feedback linearization. Definitions and the mathematical formulations of traditional, robust, and tube MPC are first presented. Output and state feedback linearization, their properties, and limitations are then discussed.

2.2 Model Predictive Control

2.2.1 Overview

This thesis is concerned with developing a tractable robust MPC implementation for nonlinear, control-affine systems of the form

$$\dot{x}(t) = f(x(t)) + b(x(t))u(t), \tag{2.1}$$

where $x \in \mathbb{R}^n$ is the state and $u \in \mathbb{R}^m$ is the control input. The remainder of this section will discuss the conceptual and computational differences between traditional, robust, and tube MPC. For clarity, the time argument of the state and control input will be omitted except in problem definitions. Note that the term *uncertainty*, in the context of this thesis, means model error and/or external disturbances. Distinctions between the types of the uncertainty (i.e., model error or disturbances) will be explicitly stated.

2.2.2 Traditional MPC

A key characteristic of MPC, unlike more traditional control techniques, is the ability to explicitly handle state and actuator constraints. MPC entails repeatedly solving a nonlinear optimal control problem with the current state of the system and applying the resulting optimal control input open-loop. A typical nonlinear optimal control problem takes the following form.

Problem 1 – Traditional MPC

$$
\begin{aligned}
u^*(t) = \ & \underset{\breve{u}(t)}{\text{argmin}} & & J = h_f\left(\breve{x}(t_f)\right) + \int_{t_0}^{t_f} \ell\left(\breve{x}(t), \breve{u}(t)\right) dt \\
& \text{subject to} & & \dot{\breve{x}}(t) = \hat{f}(\breve{x}(t)) + \hat{b}(\breve{x}(t))\breve{u}(t), \\
& & & \breve{x}(t) \in \mathbb{X}, \quad \breve{u}(t) \in \mathbb{U}, \\
& & & \breve{x}\left(t_0\right) = x(t_0), \quad \breve{x}(t_f) \in \mathbb{X}_f
\end{aligned}
$$

where u^* is optimal open-loop control input; $\breve{\ }$ are the internal variables of the optimization; h_f and ℓ are the terminal and stage cost (typically quadratic in state and/or control); $\mathbb{X}$ and $\mathbb{U}$ are the state and actuator constraints; $\mathbb{X}_f$ is the desired equality/inequality terminal constraint; $\hat{f}$ and $\hat{b}$ are the nominal dynamics and input matrix; and $x(t_0)$ is the current state of the actual system. This procedure, shown in Fig. 2-1, can be viewed as an *implicit* feedback control law that attempts to stabilize (2.1). In some cases, an *explicit* control law, such as an LQR controller, can be obtained but typically requires the removal of the state and actuator constraints. A number of modified versions of Problem 1 can be found in the literature, with variations arising from the use of terminal constraints in place of a terminal cost function and vice versa. A summary on the different formulations, and their benefits, can be found in [7].

Stability, robustness to uncertainty, and recursive feasibility (i.e., ensuring the optimization remains feasible) of Problem 1 are important and well-studied concepts in the MPC community.

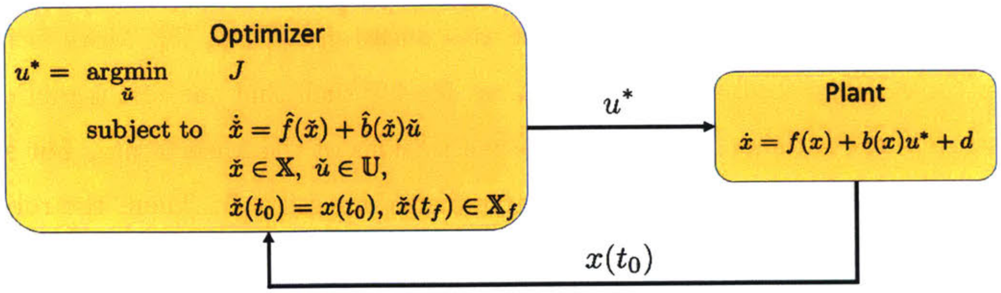

Optimizer
$u^* = \underset{\check{u}}{\text{argmin}} \quad J$
subject to $\dot{\check{x}} = \hat{f}(\check{x}) + \hat{b}(\check{x})\check{u}$
$\check{x} \in \mathbb{X},\ \check{u} \in \mathbb{U},$
$\check{x}(t_0) = x(t_0),\ \check{x}(t_f) \in \mathbb{X}_f$
u^*
Plant
$\dot{x} = f(x) + b(x)u^* + d$
$x(t_0)$

Figure 2-1: Nominal MPC architecture. An optimal control problem is repeatedly solved with the current state of the systems and the resulting control input is applied open-loop.

While there are a number of techniques to achieve stability and recursive feasibility, heavy reliance on a model leads to small robustness margins [7, 11, 65]. In practice, minimal robustness can have severe consequences when state/actuator constraints, such as not colliding with an obstacle or exceeding the physical limits of actuators, are not guaranteed to be satisfied. Further, in many situations, acquiring a high-fidelity model of the dynamics/disturbances is not practical. As a result, there has been substantial research in obtaining a robust and tractable MPC solution.

Before proceeding, it is important to highlight that the MPC framework should be viewed as more than just a stabilizing feedback controller. The framework is general enough that it can be used in almost *every* feedback loop that executes an action (i.e., u^*) when new information (i.e., $x(t_0)$) becomes available. This includes control, collision avoidance/trajectory generation/path planning, perception, or any other sequential decision making process [91]. The complexity lies in describing the physics of the system, getting constraints in a form suitable for optimization, and obtaining a computationally tractable problem.

2.2.3 Robust MPC

As mentioned above, traditional MPC is sensitive to modeling error and external disturbances, leading to poor performance in real-world applications. Fundamentally, this lack of robustness is a consequence of not explicitly considering uncertainty in Problem 1. Robust MPC, on the other hand, directly accounts for the uncertainty by optimizing over *control policies* instead of *control actions* (as in traditional MPC). The formal way to structure a robust

MPC problem originated in the dynamic programming community where the optimization selects the best control policy for the worst case uncertainty [8, 9, 58]. More formally, the optimizer constructs a feedback policy $\pi : \mathbb{X} \times \mathbb{R} \rightarrow \mathbb{U}$ such that the cost is minimized and the constraints are satisfied for all possible realizations of the uncertainty. For simplicity, assume (2.1) is subject to a bounded additive disturbance $d \in \mathbb{D}$. Then, the robust MPC problem is formulated as the following minimax optimization.

Problem 2 – Robust MPC

$$\begin{aligned}
\pi^*(x(t),t) = \underset{\pi(\check{x}(t),t)}{\operatorname{argmin}} \; \underset{d(t)}{\max} \quad & J = h_f\left(\check{x}(t_f)\right) + \int\limits_{t_0}^{t_f} \ell\left(\check{x}(t), \pi(\check{x}(t),t)\right) dt \\
\text{subject to} \quad & \dot{\check{x}}(t) = \hat{f}(\check{x}(t)) + \hat{b}(\check{x}(t))\pi(\check{x}(t),t) + d(t), \\
& \check{x}(t) \in \mathbb{X}, \quad \pi(\check{x}(t),t) \in \Pi, \quad d(t) \in \mathbb{D} \\
& \check{x}\left(t_0\right) = x(t_0), \quad \check{x}(t_f) \in \mathbb{X}_f
\end{aligned}$$

where Π is the allowable set of feedback policies and the remaining variable definitions are the same as that in Problem 1. The key difference, however, is that Problem 2 directly considers the uncertainty by optimizing over control policies.

Solving Problem 2 entails optimizing over arbitrary functions which, in general, is not computationally tractable. Discretization, as done in dynamic programming, is also not tractable because of the curse of dimensionality. For instance, Scokaert et al. showed that the computational complexity of linear robust MPC is exponential even after proving only the extrema of the uncertainty set has to be considered [60]. For nonlinear systems, the complexity is far worse since many more uncertainty realizations must be checked. The computational complexity of robust MPC has stimulated extensive research in finding a tractable alternative.

2.2.4 Tube MPC

The standard RMPC formulation involves a minimax optimization to construct a feedback policy π. However, optimizing over arbitrary functions is not tractable and discretization suffers from the curse of dimensionality. The standard approach taken in tube MPC [10] is

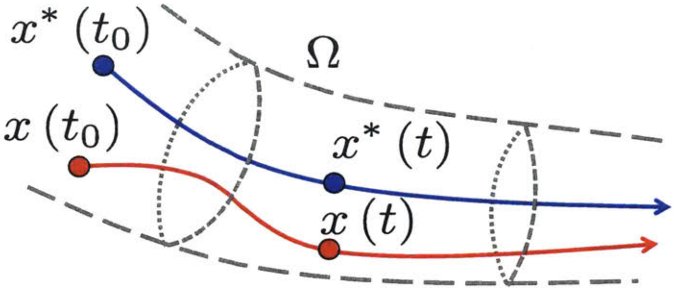


Figure 2-2: Illustration of robust control invariant (RCI) tube Ω centered around desired state x^*. If the state x begins in Ω then it remains in Ω indefinitely for all realizations of the model error or external disturbance.

to change the decision variable from control policy π to open-loop control input u^*. In order to achieve this re-parameterization, the following assumption is made about the structure of the control policy.

Assumption 1. The control policy π takes the form $\pi = u^* + \kappa(x, x^*)$ where u^* and x^* are the open-loop input and reference trajectory, respectively.

In the tube MPC literature, κ is known as the *ancillary controller* and is typically designed offline [12, 65]. The role of the ancillary controller is to ensure the state x remains in a *robust control invariant* (RCI) tube around the nominal trajectory x^*.

Definition 1. Let $\mathbb{X}$ denote the set of allowable states and let $\tilde{x} := x - x^*$. The set $\Omega \subset \mathbb{X}$ is a RCI tube if there exists an ancillary controller $\kappa(x, x^*)$ such that if $\tilde{x}(t_0) \in \Omega$, then, for all realizations of the disturbance and modeling error, $\tilde{x}(t) \in \Omega$, $\forall t \geq t_0$.

Fundamentally, RCI tubes are a mathematical object that describe how "close" the system remains to the desired trajectory for all realizations of the uncertainty. Fig. 2-2 provides a visualization of a RCI tube.

Given an ancillary control κ and associated RCI tube Ω, a constraint-tightened version of the nominal MPC problem can be solved to generate an open-loop control input u^* and reference trajectory x^*. Tube MPC is typically formulated as the following optimization.

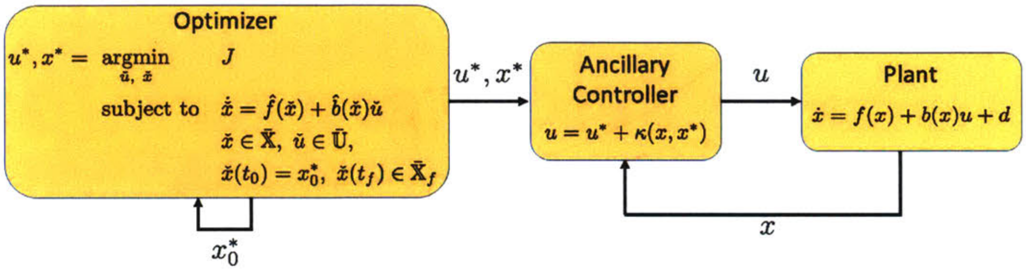

Optimizer
$u^*, x^* = \underset{\check{u},\, \check{x}}{\operatorname{argmin}} \quad J$
subject to $\dot{\check{x}} = \hat{f}(\check{x}) + \hat{b}(\check{x})\check{u}$
$\check{x} \in \bar{\mathbb{X}},\ \check{u} \in \bar{\mathbb{U}},$
$\check{x}(t_0) = x_0^*,\ \check{x}(t_f) \in \bar{\mathbb{X}}_f$
x_0^*
u^*, x^*
Ancillary Controller
$u = u^* + \kappa(x, x^*)$
u
Plant
$\dot{x} = f(x) + b(x)u + d$
x

Figure 2-3: Tube MPC architecture. A constraint-tightened optimal control problem is repeatedly solved with initial condition x_0^* that can be freely picked. The ancillary controller tracks the desired state while providing robustness to model error and disturbances.

Problem 3 – Tube MPC

$$\begin{aligned}
u^*(t),\ x^*(t) = \underset{\check{u}(t),\ \check{x}(t)}{\operatorname{argmin}} \quad & J = h_f\left(\check{x}(t_f)\right) + \int_{t_0}^{t_f} \ell\left(\check{x}(t), \check{u}(t)\right) dt \\
\text{subject to} \quad & \dot{\check{x}}(t) = \hat{f}(\check{x}(t)) + \hat{b}(\check{x}(t))\check{u}(t), \\
& \check{x}(t) \in \bar{\mathbb{X}}, \quad \check{u}(t) \in \bar{\mathbb{U}}, \\
& \check{x}\left(t_0\right) = x_0^*, \quad \check{x}(t_f) \in \bar{\mathbb{X}}_f
\end{aligned}$$

where $\bar{\mathbb{X}}$ and $\bar{\mathbb{U}}$ are the modified state and actuator constraints, respectively. Note that Problem 3 is initialized with x_0^* instead of $x(t_0)$, which is typically chosen to be somewhere along the last optimal solution (i.e., $x_0^* = x^*(\tau)$ where $\tau \in [t_0,\ t_f]$). Also, since Problem 3 is just a modified version of Problem 1, all the stability and recursive feasibility proofs for traditional MPC can be easily extended to tube MPC so long as the constraints are tightened appropriately. The complexity, then, of tube MPC is designing the ancillary controller κ and calculating the associated RCI tube Ω – both of which are done offline. The standard tube MPC architecture is shown in Fig. 2-3.

While tube MPC is a tractable alternative to robust MPC, it has three main drawbacks. First, the decoupled trajectory and control policy design (i.e., tube geometry) is inherently suboptimal. Second, it is extremely difficult to establish an explicit relationship between state-dependent uncertainty and the tube geometry; often leading to the simplifying assumption that the model uncertainty is simply bounded. Third, the difficulty of constructing a certainty equivalence controller for nonlinear systems inhibits the use of online adaptation/estimation to reduce model uncertainty. A solution for the first two issues is presented in Chapter 3 and the third issue in Chapter 5.

2.3 Feedback Linearization Control

2.3.1 Overview

A complete survey of nonlinear control theory would undoubtedly multiply the length of this thesis multifold. As a result, this section will focus on the nonlinear control technique known as *feedback linearization.* As will be rigorously shown, feedback linearization entails forcing a nonlinear systems to behave as a linear one by inverting the system dynamics. This inversion is sensitive to model error and disturbances, eliminating it as a suitable ancillary controller for tube MPC. However, many of the theoretical aspects of feedback linearization are central to the ancillary controller design in Chapter 3. The following material borrows heavily from results presented in [45, 92]. The following definitions will be used later in this section.

Definition 2. The equilibrium state $x = 0$ for a dynamical system is *stable* if for any $R > 0$ there exists $r > 0$ such that if $\|x(0)\| < r$ then $\|x(t)\| < R$ for all $t \geq 0$.

Definition 3. The equilibrium state $x = 0$ for a dynamical system is *asymptotically stable* if there exists $r > 0$ such that if $\|x(0)\| < r$ then $\|x(t)\| \to 0$ as $t \to \infty$.

Definition 4. The equilibrium state $x = 0$ for a dynamical system is *exponentially stable* if there exists $K,\ \lambda,\ r > 0$ such that if $\|x(0)\| < r$ then $\|x(t)\| \leq K\|x(0)\|e^{-\lambda t}$.

Definition 5. Let $h : \mathbb{R}^n \to \mathbb{R}$ and $f : \mathbb{R}^n \to \mathbb{R}^n$ be a smooth scalar function and vector field. The *Lie derivative* of h with respect to f is a scalar function defined by $L_f h = \nabla h f$.

Definition 6. Let $f,\ b : \mathbb{R}^n \to \mathbb{R}^n$ be two smooth vector fields. Then the *Lie bracket* of f and b is another vector field defined by

$$ad_f b = [f, b] = \nabla b f - \nabla f b,$$

where *ad* is commonly referred to as the *adjoint.*

Definition 7. A function $\phi : \mathbb{R}^n \to \mathbb{R}^n$ is a *diffeomorphism* if it is differentiable and bijective.

2.3.2 Input-Output Linearization

Consider the following single-input, single-output nonlinear system

$$\begin{aligned} \dot{x} &= f(x) + b(x)u \\ y &= h(x) \end{aligned} \tag{2.2}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}$, and $y \in \mathbb{R}$. The goal of input-output feedback linearization is to transform (2.2) into a linear system so concepts from linear control theory can be applied. This is accomplished by continuously differentiating the output until the input appears. For instance, assume that it takes $r \leq n$ differentiations of h for the input to appear. It can then be shown that

$$y^{(r)} = L_f^r h(x) + L_b L_f^{r-1} h(x) u, \tag{2.3}$$

where L_f and L_b are Lie derivatives (from Definition 5). Since $L_b L_f^{r-1} h(x) \neq 0$, then if the control law

$$u = \frac{1}{L_b L_f^{r-1} h} \left(-L_f^r h + v \right), \tag{2.4}$$

is applied to (2.2), the following *linear* system is obtained

$$y^{(r)} = v, \tag{2.5}$$

where v can be designed via pole placement, LQR, or other linear control techniques.

The number of r differentiations for the input to appear in the output is known as the relative degree of the system. If $r = n$, then the system is said to be *input-state linearizable* (to be discussed next). However, if $r < n$, then there is a subset of the dynamics that are not directly controlled. In order for these *internal* states to remain bounded, the stability of the internal dynamics must be verified. It can be shown that if the internal dynamics are stable when the output y is identically zero then the overall system is asymptotically stable [45].

In order to study these internal dynamics formally, the concept of *normal form* needs to be introduced. First, let

$$z = [z_1 \; z_2 \; \cdots \; z_r]^T = \left[y \; \dot{y} \; \cdots \; y^{(r-1)}\right]^T. \tag{2.6}$$

The normal form of (2.3) is given by

$$\dot{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_r \\ \sigma(z,\Psi) + \beta(z,\Psi)u \end{bmatrix}, \quad \dot{\Psi} = q(z,\Psi), \tag{2.7}$$

where

$$\sigma(z,\Psi) = L_f^r h(x), \quad \beta(z,\Psi) = L_b L_f^{r-1} h(x). \tag{2.8}$$

If it can shown that the internal dynamics

$$\dot{\Psi} = q(0,\Psi), \tag{2.9}$$

where $z = 0$ are stable, then the entire closed-loop systems is asymptotically stable. The following example outlines the general procedure.

Example 1. Consider the following system

$$\begin{aligned} \dot{x}_1 &= x_1^3 + x_2 + u \\ \dot{x}_2 &= -x_2^3 + 2x_1 \\ y &= x_1 \end{aligned} \tag{2.10}$$

with

$$u = -x_1^3 - x_2 + v. \tag{2.11}$$

For $y = 0$, then the internal dynamics are given by

$$\dot{x}_2 = -x_2^3, \tag{2.12}$$

which are exponentially stable so the closed-loop system is asymptotically stable. □

It is important to highlight that only asymptotic stability is guaranteed for the closed-loop system. This is a consequence of the peaking phenomenon [93], which is a result of studying the stability of the internal dynamics when the output is identically zeros. A natural question then arises: how does one stabilize (2.2) if the internal dynamics are unstable? Backstepping is one technique that entails designing v to stabilize the internal dynamics but requires the system to be in a particular form [94]. State-output linearization, discussed next, is another approach that attempts to construct h such that there are no internal dynamics (i.e., $r = n$).

2.3.3 Input-State Linearization

The goal of input-state linearization is to construct a diffeomorphism $\phi(x) : \mathbb{R}^n \to \mathbb{R}^n$ such that the general nonlinear system

$$\dot{x} = f(x) + b(x)u, \tag{2.13}$$

can be transformed into normal form

$$\begin{aligned} \dot{z}_1 &= z_2 \\ \dot{z}_2 &= z_3 \\ &\vdots \\ \dot{z}_n &= \sigma(z) + \beta(z)u \end{aligned} \tag{2.14}$$

where $z = [z_1 \; z_2 \; \cdots \; z_n]^T = [\phi_1(x) \; \phi_2(x) \; \cdots \; \phi_n(x)]^T = \phi(x)$. Note the absence of the output function $h(x)$ in (2.13) and the internal dynamics $\dot{\Psi} = q(z, \Psi)$ in (2.14). By transforming (2.13) to (2.14) and letting

$$u = \beta(z)^{-1}(-\sigma(z) + v), \tag{2.15}$$

linear control techniques can be used to design v without any concern about the stability of the internal dynamics. Constructive conditions for $\phi(x)$ can be obtained by simply using the definition $z = \phi(x)$ and differentiation n times

$$\begin{aligned}
\dot{z}_1 &= \nabla\phi_1(f(x) + b(x)u) = z_2 \\
\dot{z}_2 &= \nabla\phi_2(f(x) + b(x)u) = z_3 \\
&\vdots \\
\dot{z}_n &= \nabla\phi_n(f(x) + b(x)u) = \sigma(z) + \beta(z)u.
\end{aligned} \tag{2.16}$$

In order for (2.16) to hold, u must be eliminated from the first $n-1$ equations. This is achieved by setting

$$\nabla\phi_1 b(x) = \nabla\phi_2 b(x) = \cdots = \nabla\phi_{n-1} b(x) = 0, \quad \nabla\phi_n b(x) \neq 0, \tag{2.17}$$

or more concisely

$$L_b\phi_1 = L_b\phi_2 = \cdots = L_b\phi_{n-1} = 0, \quad L_b\phi_n \neq 0. \tag{2.18}$$

Recall that since $\dot{z}_i = z_{i+1}$, then $\dot{\phi}_i = \phi_{i+1}$ for $i = 1, \ldots, n-1$ so (2.18) can be expressed as a function of ϕ_1 only

$$L_b\phi_1 = L_bL_f\phi_1 = \cdots = L_bL_f^{n-1}\phi_1 = 0, \quad L_bL_f^{n}\phi_1 \neq 0. \tag{2.19}$$

Noting that if $L_bL_f^k = 0$ then $L_bL_f^k = ad_f^{k-1}b(x)$ for $k = 0, \ldots, n-2$, the following constructive conditions for ϕ_1 are obtained

$$\begin{aligned}
&\nabla\phi_1 ad_f^{k} b(x) = 0 \quad k = 0, \ldots, n-2 \\
&\nabla\phi_1 ad_f^{n-1} b(x) \neq 0
\end{aligned} \tag{2.20}$$

Observe that (2.20) is a (possibly nonlinear) partial differential equation for ϕ_1. In order for a solution to exist, two technical conditions must be satisfied. First, the vector fields $\{b(x),\ ad_f b(x),\ \cdots,\ ad_f^{n-1} b(x)\}$ must be linearly independent. This corresponds to (2.13) being fully controllable. Second, the set $\{b(x),\ ad_f b(x),\ \cdots,\ ad_f^{n-2} b(x)\}$ must be integrable. This condition is far more technical than the controllability condition but is necessary and sufficient for the solvability of (2.20). Interestingly, the physical interpretation of integrability is that (2.13) must be holonomic [95]. Consequently, nonholonomic systems are not input-state linearizable, which is why deriving controllers for these type of systems is so challenging.

2.3.4 MIMO Extensions

Consider the following MIMO system

$$\begin{aligned} \dot{x} &= f(x) + b(x)u \\ y &= h(x) \end{aligned} \tag{2.21}$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, and $y \in \mathbb{R}^m$. Same as in input-output linearization, each output y_i can be differentiated r_i times for $i = 1, \ldots, m$ until the input appears

$$y_i^{(r_i)} = L_f^{r_i} h_i + \sum_{j=1}^{m} L_{b_j} L_f^{r_i - 1} h_i u_j. \tag{2.22}$$

Each $y_i^{(r_i)}$ can be stacked to obtain

$$\begin{bmatrix} y_1^{(r_1)} \\ \vdots \\ y_m^{(r_m)} \end{bmatrix} = \underbrace{\begin{bmatrix} L_f^{r_1} h_1(x) \\ \vdots \\ L_f^{r_m} h_m(x) \end{bmatrix}}_{F(x)} + \underbrace{\begin{bmatrix} L_{b_1} L_f^{r_1} h_1(x) & \cdots & L_{b_m} L_f^{r_1} h_1(x) \\ \vdots & \ddots & \vdots \\ L_{b_1} L_f^{r_m} h_m(x) & \cdots & L_{b_m} L_f^{r_m} h_m(x) \end{bmatrix}}_{B(x)} u, \tag{2.23}$$

where $B(x) \in \mathbb{R}^{m \times m}$ is the decoupling matrix. The controller u then takes a similar form to that in (2.4), which requires $B(x)$ to be invertible. In the scenario where $B(x)$ is not invertible, dynamic extension of the inputs or re-defining the outputs are two possible strategies to make $B(x)$ non-singular. If the relative degree $r = r_1 + \cdots + r_m = n$ then there are no internal dynamics so (2.21) is input-state linearizable. Otherwise, the stability of the zero dynamics must be checked by setting $y = 0$.

2.3.5 Feedback Linearization Robustness

Fundamentally, both input-output and input-state feedback linearization are forcing the system to behave linearly by inverting the nonlinear dynamics. This inversion requires an accurate model of the dynamics, which may be difficult or costly to obtain. Since many real-world systems have some form of model error or are subject to external disturbances, it is desirable to construct a robust version of feedback linearization. Although derived independently from feedback linearization, sliding mode control provides *complete cancellation* of model error and external disturbances. In the context of tube MPC, using a sliding mode ancillary control to completely cancel model error and disturbances would reduce the tube geometry to zero – a very enticing proposition. However, this impressive level of robustness comes at the expense of high-frequency discontinuous control. Sliding mode control, its properties, and alternatives are discussed in detail in Chapter 3.

2.4 Summary

This chapter presented the mathematical preliminaries required for the rest of the thesis. Specifically, traditional, robust, and tube MPC were presented. Traditional MPC, which entails repeatedly solving an open-loop optimal control problem with the current state of the system, is susceptible to model error and disturbance leading to suboptimal performance in real-world scenarios. Robust MPC addresses this limitation by optimizing over control policies instead of control actions but at the expense of computational complexity. Tube MPC is a tractable alternative to robust MPC that uses an ancillary controller, designed offline, to keep the system in a tube centered around the desired trajectory, generated online. While this decoupled trajectory-controller design is tractable, it is inherently suboptimal.

The complexity of nonlinear tube MPC is in designing an ancillary controller and computing the corresponding tube. A number of different control designs can be found in the literature (see Section 1.4). For brevity, this chapter only presented two techniques, input-output and input-state linearization, because they are most relevant to the framework developed in Chapter 3. While the lack of robustness for both techniques is well documented, Chapter 3 will show that using a robust version of feedback linearization (i.e., sliding control) as an ancillary controller leads to several nice properties that address a number of limitations with existing nonlinear tube MPC approaches.

Chapter 3

Dynamic Tube MPC

3.1 Overview

This chapter presents the Dynamic Tube MPC (DTMPC) framework for nonlinear systems. DTMPC has two unique features: 1) the open-loop trajectory and tube geometry can be *simultaneously* optimized; and 2) the structure of the model uncertainty can be used within the optimization to produce more intelligent trajectories. The structure of this chapter is as follows. The problem formulation and necessary assumptions are first presented. Boundary layer sliding control, used as the ancillary controller in DTMPC, is then derived and discussed in detail. The main technical result of the chapter, proving the boundary layer sliding controller induces a robust control invariant tube, is then presented. Finally, the DTMPC framework is formally defined and discussed.

3.2 Problem Formulation

Consider a nonlinear, time-invariant, and control affine system given by (omitting the time argument)

$$\begin{aligned} \dot{x} &= f\left(x\right) + b\left(x\right)u + d \\ y &= h(x) \end{aligned} \tag{3.1}$$

where $x \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the control input, $d \in \mathbb{R}^n$ is an external disturbance, and $y \in \mathbb{R}^m$ are the states to be controlled.

Assumption 2. The dynamics f can be expressed as $f = \hat{f} + \tilde{f}$ where $\hat{f}$ is the nominal dynamics and $\tilde{f}$ is the bounded model error (i.e., $|\tilde{f}(x)| \leq \Delta_f(x)$).

Assumption 3. The elements of the input matrix b can be expressed as $b_{i,j} = \left(1 + \tilde{b}_{i,j}\right) \hat{b}_{i,j}$ for $i = 1, \cdots, n$ and $j = 1, \cdots, m$ where $\hat{b}$ is the nominal input matrix and $\tilde{b}$ is the bounded input matrix error (i.e, $|\tilde{b}(x)| \leq \Delta_b(x)$).

Note that the model error and input matrix bound in Assumption 2 and Assumption 3 are intentionally left to be state-dependent. As will be shown shortly, one of the key characteristics of DTMPC is the ability to leverage this dependency within the optimization.

Assumption 4. The disturbance d belongs to a closed, bounded, and connected set $\mathbb{D}$ (i.e., $\mathbb{D} := \{d \in \mathbb{R}^n : |d| \leq D\}$) and is in the span of the control input matrix (i.e., $d \in \mathrm{span}\,(b(x))$).

As discussed in Chapter 2, the standard RMPC formulation requires solving a minimax optimization to construct a feedback policy $\pi : \mathbb{X} \times \mathbb{R} \rightarrow \mathbb{U}$ where $x \in \mathbb{X}$ and $u \in \mathbb{U}$ are the allowable states and control inputs, respectively. However, this is not tractable even for simple linear systems where Scokaert et al. [60] showed the computational complexity is exponential. Tube MPC overcomes this challenge by re-parameterizing the control policy π (via Assumption 1) such that decision variable becomes the open-loop control input u^*. Robustness is achieved by designing the ancillary controller κ such that the state x remains in a robust control invariant tube Ω (Definition 1) centered around the desired trajectory. Both Assumption 1 and Definition 1 from Chapter 2 are repeated below.

Assumption 1. The control policy π takes the form $\pi = u^* + \kappa(x, x^*)$ where u^* and x^* are the open-loop input and reference trajectory, respectively.

Definition 1. Let $\mathbb{X}$ denote the set of allowable states and let $\tilde{x} := x - x^*$. The set $\Omega \subset \mathbb{X}$ is a RCI tube if there exists an ancillary controller $\kappa\,(x, x^*)$ such that if $\tilde{x}\,(t_0) \in \Omega$, then, for all realizations of the disturbance and modeling error, $\tilde{x}(t) \in \Omega$, $\forall t \geq t_0$.

Calculating a RCI tube for a given ancillary controller can be very difficult for nonlinear systems. Unsurprisingly, the chosen methodology for synthesizing the ancillary controller can dramatically influence the complexity of the tube geometry construction. Ideally, the controller and tube geometry could be parameterized such that an explicit relationship between the two can be derived; enabling the controller and tube geometry to be designed online within the optimization. Further, having a control strategy that is able capture state-dependent uncertainty and describe how it impacts the tube geometry would be desirable. While it may seem infeasible to find such a control synthesis strategy, Section 3.3 will show that boundary layer sliding control possesses both properties.

For now, assume that a control synthesis technique exists such that both the controller and tube geometry can be parameterized by $\alpha \in \mathbb{A}$ where $\mathbb{A}$ is the allowable set of parameters. By introducing the parameter vector α, a more precise definition of the ancillary controller is the mapping $\kappa : \mathbb{X} \times \mathbb{X} \times \mathbb{A} \rightarrow \mathbb{U}$. Also, how the state-dependent uncertainty and parameter vector α affect the tube geometry must be captured. This is achieved by introducing the tube geometry dynamics

$$\dot{\Omega} = \Upsilon\left(x^*, \alpha, \Omega\right) \tag{3.2}$$

where Υ is a function, possibly nonlinear, that describes how x^* and α influence the geometry. Therefore, the nonlinear tube MPC problem where the tube geometry and open-loop trajectory are simultaneously optimized can be formulated as the following optimization.

Problem 4 – MPC and Tube Geometry Optimization

$$\begin{aligned}
u^*(t),\ x^*(t),\ \alpha^*(t) = \underset{\breve{u}(t),\breve{x}(t),\breve{\alpha}(t)}{\text{argmin}} \quad & J = h_f\left(\breve{x}(t)\right) + \int_{t_0}^{t_f} \ell\left(\breve{x}(t), \breve{u}(t), \breve{\alpha}(t)\right) dt \\
\text{subject to} \quad & \dot{\breve{x}}(t) = \hat{f}(\breve{x}(t)) + \hat{b}(\breve{x}(t))\breve{u}(t), \\
& \dot{\Omega} = \Upsilon\left(\breve{x}(t), \breve{\alpha}(t), \Omega(t)\right), \\
& \breve{x}\left(t_0\right) = x_0^*, \quad \Omega\left(t_0\right) = |\tilde{x}(t_0)|, \\
& \breve{x}\left(t_f\right) \in \bar{\mathbb{X}}_f, \quad \breve{\alpha}(t_f) \in \mathbb{A}_f \\
& \breve{x}(t) \in \bar{\mathbb{X}}, \quad \breve{u}(t) \in \bar{\mathbb{U}}, \quad \breve{\alpha}(t) \in \mathbb{A}
\end{aligned}$$

where h_f and ℓ are quadratic terminal and stage cost; $\bar{\mathbb{X}} := \mathbb{X} \ominus \Omega$ the tightened state constraints; $\bar{\mathbb{X}}_f := \mathbb{X}_f \ominus \Omega$ the tightened terminal state constraints; $\bar{\mathbb{U}} := \mathbb{U} \ominus \Gamma$ the tightened control input constraints; $\mathbb{A}_f$ is the terminal constraint set for parameter α; and $|\cdot|$ is the element-wise absolute value.

3.3 Boundary Layer Sliding Control

3.3.1 Overview

This section reviews time-varying boundary layer sliding control [45, 90], provides analysis supporting its use as an ancillary controller, and shows how the DTMPC framework leverages its properties. As reviewed in Section 1.4, sliding mode control has been extensively used for nonlinear tube MPC because of its simplicity and strong robustness properties. Unlike other control strategies, sliding mode control completely cancels any bounded modeling error or external disturbance (reducing the RCI tube to zero). However, complete cancellation comes at the cost of high-frequency discontinuous control making it impractical for many real systems; a number of continuous control versions have since been developed. Note that the boundary layer controller was originally developed in [90] but, until now, has not been used within the context of tube MPC. Before proceeding the following assumption is made.

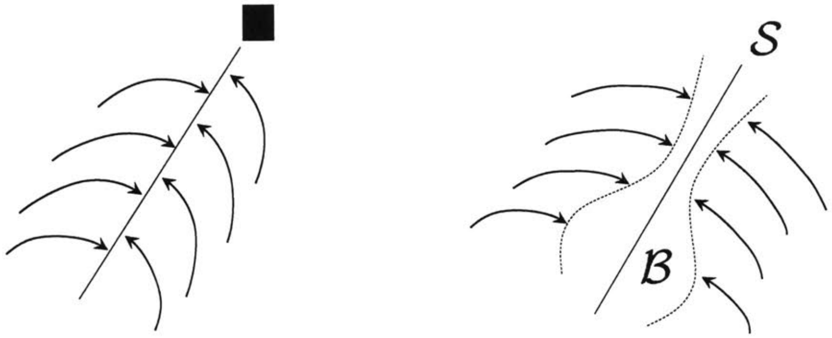
Assumption 5. The system given by (3.1) is either input-state or input-output linearizable with zeros that are stable or can be stabilized via backstepping.

Note that Assumption 5 requires system (3.1) to be fully actuated, minimum phase, or in cascaded form. While many systems fall into one of these categories, future work will extend DTMPC to more general nonlinear systems.

3.3.2 Sliding Mode Control

Let $\tilde{y}_i := y_i - y_i^*$ be the tracking error for output y_i for $i = 1, \ldots, m$. Then, for $\lambda_i > 0$, the sliding variable s_i for output y_i is defined as

$$\begin{aligned} s_i &= \left(\frac{d}{dt} + \lambda_i \right)^{r_i - 1} \tilde{y}_i \\ &= \tilde{y}_i^{(r_i-1)} + \cdots + \lambda_i^{r_i-1} \tilde{y}_i \\ &= y_i^{(r_i-1)} - y_{pi}^{(r_i-1)}, \end{aligned} \tag{3.3}$$



(a) Sliding manifold $\mathcal{S}$. **(b)** Boundary layer $\mathcal{B}$.

Figure 3-1: Sliding manifolds. (a): Sliding surface for traditional sliding mode control. $\mathcal{S}$ is invariant to uncertainty and disturbances but requires discontinuous control. (b): A time-varying region $\mathcal{B}$ around $\mathcal{S}$ can be constructed such that $\mathcal{B}$ is invariant through continuous control.

where

$$y_{pi}^{(r_i-1)} = y_i^{*(r_i-1)} - \sum_{k=1}^{r_i-1} \binom{r_i-1}{k-1} \lambda_i^{r_i-k} \tilde{y}_i^{(k-1)}, \tag{3.4}$$

and r_i is the relative degree of output y_i (defined in Section 2.3).

In sliding mode control, a sliding manifold $\mathcal{S}_i$ is defined such that $s_i = 0$ for all time once the manifold is reached (shown in Fig. 3-1a). This condition guarantees the tracking error converges to zero exponentially via (3.3). To ensure the sliding manifold $\mathcal{S}_i$ is attractive, the following condition for the sliding variable s_i for output y_i is imposed

$$\frac{1}{2}\frac{d}{dt}s_i^2 \leq -\eta_i |s_i|, \tag{3.5}$$

where η_i dictates the convergence rate of s_i to the manifold $\mathcal{S}_i$. Using the definition of s_i from (3.3) and performing the differentiation in (3.5), one obtains

$$s_i \left(y_i^{(r_i)} - y_{pi}^{(r_i)} \right) \leq -\eta_i |s_i|. \tag{3.6}$$

Recall from Section 2.3, the dynamics of the output variable is given by

$$y_i^{(r_i)} = L_f^{r_i} h_i(x) + \sum_{j=1}^{m} L_{b_j} L_f^{r_i-1} h_i(x) u_j + d_i. \tag{3.7}$$

Substituting into (3.6),

$$s_i\left(L_f^{r_i}h_i(x)+\sum_{j=1}^{m}L_{b_j}L_f^{r_i-1}h_i(x)u_j-y_{pi}^{(r_i)}+d_i\right)\leq-\eta_i|s_i|, \tag{3.8}$$

which can be stacked for each output y_i to obtain a more concise form

$$s*\left\{\underbrace{\begin{bmatrix} L_f^{r_1}h_1(x) \\ \vdots \\ L_f^{r_m}h_m(x)\end{bmatrix}}_{F(x)}+\underbrace{\begin{bmatrix} L_{b_1}L_f^{r_1}h_1(x) & \cdots & L_{b_m}L_f^{r_1}h_1(x) \\ \vdots & \ddots & \vdots \\ L_{b_1}L_f^{r_m}h_m(x) & \cdots & L_{b_m}L_f^{r_m}h_m(x)\end{bmatrix}}_{B(x)}u-y_p^{(r)}+d\right\}\leq-\eta*|s|, \tag{3.9}$$

where $*$ and $|\cdot|$ are element-wise multiplication and absolute value, respectively.

For clarity of the following derivation, $B(x)$ is assumed to be known so $\hat{B}(x)=B(x)$. The derivation for the sliding mode controller with an uncertain input matrix entails identical steps but is more tedious to follow. The equation for the controller with an uncertain input matrix will be presented following the derivation.

Let u take the form

$$u=\hat{B}(x)^{-1}\left(-\hat{F}(x)+y_p^{(r)}-K(x)\text{sign}(s)\right), \tag{3.10}$$

where K is the robust gain that must be found. Then, (3.9) becomes

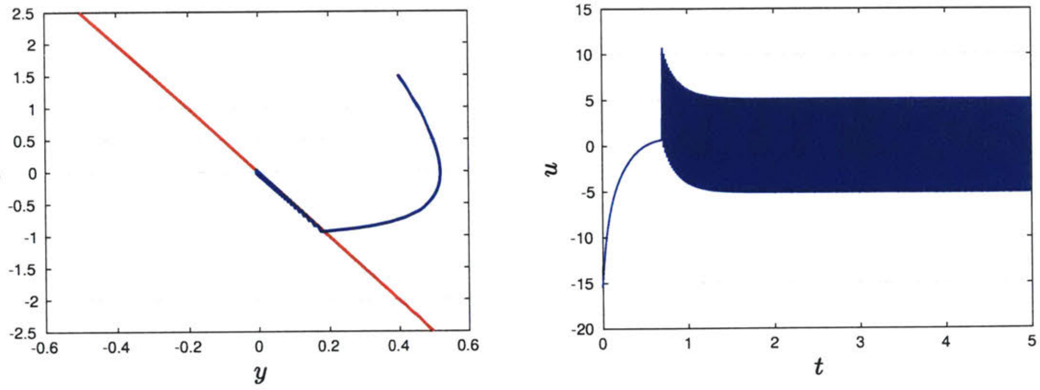
$$s*\left\{F(x)+B(x)\left[\hat{B}(x)^{-1}\left(-\hat{F}(x)+y_p^{(r)}-K(x)\text{sign}(s)\right)\right]+y_p^{(r)}+d\right\}\leq-\eta*|s|. \tag{3.11}$$

Since $B(x)=\hat{B}(x)$ (again only for clarity)

$$s*\left\{F(x)-\hat{F}(x)-K(x)\text{sign}(s)+d\right\}\leq-\eta*|s|. \tag{3.12}$$

Taking the element-wise absolute value of both sides and using Assumption 2 and Assumption 4,

$$|s|*\{\Delta_F(x)-K(x)+D\}\leq-\eta*|s|, \tag{3.13}$$



(a) State trajectories.

(b) Control input u.

Figure 3-2: Simple sliding mode control example for $\ddot{x} = -x - x^3 + \dot{x}^3 + u + d$, $y = x$, and $d = \sin(2\pi t)$. (a): State trajectory (blue) converges to and slides along the sliding surface (red) in finite time, irrespective of the disturbance. (b): Control input as the state trajectory reaches and slides along the surface. The sign$(\cdot)$ term in the controller leads to discontinuities in the control signal. This phenomenon is known as chattering.

which is satisfied if K is chosen to be

$$K(x) = \Delta_F(x) + D + \eta. \tag{3.14}$$

In the case where there is uncertainty in the input matrix then, using Assumption 3, K takes the more complicated form

$$K(x) = (I + \Delta_B(x))^{-1} \left[\Delta_F(x) + \Delta_B(x) \left|-\hat{F}(x) + y_p^{(r)}\right| + D + \eta\right]. \tag{3.15}$$

Hence, the sliding mode control law is given by (3.10) and (3.14) (or (3.15) if $\Delta_B(x) \neq 0$). Intuitively, the choice of the robust gain K is dictated by the level of uncertainty in the dynamics and magnitude of the disturbance. In order for the sliding manifold $\mathcal{S}$ to remain invariant, K must be large enough to compensate for both model error and disturbances.

As stated earlier, the construction of the sliding mode controller ensures that the affects of model error and external disturbance are *completely* canceled. However, this impressive robustness is achieved by having high-frequency discontinuous control which, among other things, can excite unmodeled high-frequency dynamics and shorten actuator life span. An example of the discontinuous control, commonly referred to as chatter, is shown in Fig. 3-2 for a simple second order nonlinear system.

3.3.3 Boundary Layer Control

One strategy to smooth the control input is to introduce a boundary layer around the sliding surface. Let the boundary layer be defined as $\mathcal{B}_i := \{x : |s_i| \leq \Phi_i\}$ (shown in Fig. 3-1b) where Φ_i is the boundary layer thickness. If Φ_i is time varying, then the boundary layer can be made attractive if the following differential equation is satisfied

$$\frac{1}{2}\frac{d}{dt}s_i^2 \leq \left(\dot{\Phi}_i - \eta_i\right)|s_i|, \tag{3.16}$$

where η_i dictates the convergence rate to the boundary layer $\mathcal{B}_i$ (as opposed to the sliding manifold $\mathcal{S}_i$ as before). Following the same steps, the boundary layer controller takes the form (again assuming $\Delta_B = 0$ for simplicity)

$$u = \hat{B}(x)^{-1}\left[-\hat{F}(x) - y_p^{(r)} - K(x)\text{sat}\left({}^{s}/_{\Phi}\right)\right], \tag{3.17}$$

where sat$(\cdot)$ is the saturation function and the division is element-wise. Then, for $|s| > \Phi$, the boundary layer is attractive if

$$K(x) = \Delta_F(x) + D + \eta - \dot{\Phi}. \tag{3.18}$$

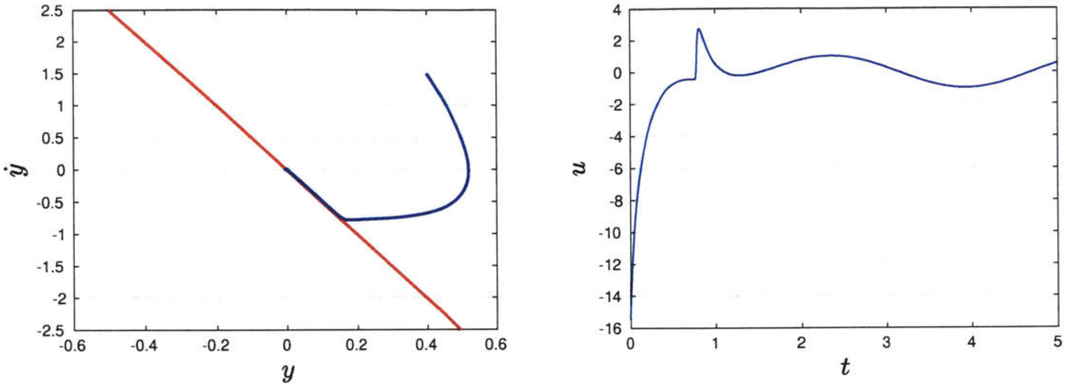
Additional information can be inferred by considering the sliding variable dynamics inside the boundary layer. After substituting (3.17) into the sliding variable dynamics one obtains

$$\dot{s} = -\frac{K(x)}{\Phi}s + F(x) - \hat{F}(x) + d, \tag{3.19}$$

where the division is element-wise. Alternatively, (3.19) can be written as

$$\dot{s} = -\frac{K(x^*)}{\Phi}s + \left(F(x^*) - \hat{F}(x^*) + d + O\left(\tilde{x}\right)\right), \tag{3.20}$$

which is a first order filter with cutoff frequency $\frac{K(x^*)}{\Phi}$. Let α be the desired cutoff frequency,



(a) State trajectory.

(b) Control input u.

Figure 3-3: Simple boundary layer control example for $\ddot{x} = -x - x^3 + \dot{x}^3 + u + d$, $y = x$, $\Phi = 0.1$, and $d = \sin(2\pi t)$. (a): State trajectory (blue) converges to a region (boundary layer) around the sliding manifold (red) in finite time, irrespective of the disturbance. (b): Control input as the state trajectory reaches the boundary layer. The sat$(\cdot)$ term in the controller eliminates the chattering phenomenon.

then, leveraging (3.18), one obtains

$$\frac{\Delta_F(x^*) + D + \eta - \dot{\Phi}}{\Phi} = \alpha, \tag{3.21}$$

or

$$\dot{\Phi} = -\alpha\Phi + \Delta_F(x^*) + D + \eta. \tag{3.22}$$

Thus, the final control law is given by (3.17), (3.18), and (3.24). In the case where $\Delta_B(x) \neq 0$, then, using Assumption 3, the robust gain K is

$$K(x) = (I + \Delta_B(x))^{-1} \left[\Delta_F(x) + \Delta_B(x) \left|-\hat{F}(x) + y_p^{(r)}\right| + D + \eta - \dot{\Phi}\right], \tag{3.23}$$

and the boundary layer dynamics are

$$\dot{\Phi} = -\alpha\Phi + \Delta_F(x^*) + \Delta_B(x^*) \left|-\hat{F}(x^*) + y_p^{(r)}\right| + D + \eta. \tag{3.24}$$

Fig. 3-3 illustrates the effectiveness of eliminating chatter from the control input. The state trajectory (shown in Fig. 3-3a) converges to a region around the sliding manifold in finite time without discontinuous control (shown in Fig. 3-3b). While boundary layer control was originally developed to eliminate chattering, the ability to explicitly capture the affect of uncertainty on the tracking error is the characteristic that makes it such a useful control strategy for tube MPC.

3.4 Main Result

The boundary layer sliding controller in (3.17) allows us to establish several key properties at the core of DTMPC. All of which are a consequence of the following theorem.

Theorem 1 (RCI Tube [16]). *Let $\tilde{z}_i = \left[\tilde{y}_i \ \dot{\tilde{y}}_i \ \cdots \ \tilde{y}_i^{(r_i-1)}\right]^T$ be the error vector for output y_i. The boundary layer controller induces a robust control invariant tube Ω_i where the tube geometry is given by*

$$\Omega_i(t) \leq e^{A_{c_i}(t-t_0)}\Omega_i(t_0) + \int_{t_0}^{t} e^{A_{c_i}(t-t_0-\tau)} B_{c_i}\Phi_i(\tau)d\tau, \tag{3.25}$$

where $\Omega_i(t) = |\tilde{z}_i(t)|$ and A_{c_i} and B_{c_i} are found by putting (3.3) into controllable canonical form.

Proof. Recalling the definition of s_i from (3.3), the error dynamics are given by the linear differential equation

$$\tilde{y}_i^{(r_i-1)} + \cdots + \lambda_i^{r_i-1}\tilde{y}_i = s_i. \tag{3.26}$$

With the error vector $\tilde{z}_i = \left[\tilde{y}_i \ \dot{\tilde{y}}_i \ \cdots \ \tilde{y}^{(r_i-1)}\right]^T$ and putting (3.26) into the controllable canonical form, the analytical solution is

$$\tilde{z}_i(t) = e^{A_{c_i}(t-t_0)}\tilde{z}_i(t_0) + \int_{t_0}^{t} e^{A_{c_i}(t-t_0-\tau)} B_{c_i} s_i d\tau. \tag{3.27}$$

Taking the element-wise absolute value $|\cdot|$, setting $\Omega_i(t) = |\tilde{z}_i(t)|$, and noting $|s_i| \leq \Phi_i$, (3.25) is obtained. Thus, by Definition 1, Ω_i is a RCI tube since $\tilde{z}_i$ is bounded. □

Remark 1. Theorem 1 proves that the tracking error $\tilde{y}_i$ for output y_i is bounded once the sliding variable reaches the boundary layer. It is important to highlight that Theorem 1 also establishes bounds on the derivatives of the tracking error $\dot{\tilde{y}}_i, \ \cdots, \ \tilde{y}^{(r_i-1)}$. Further, since

$y_i = h_i(x)$, bounds on the state tracking error $\tilde{x}$ can be obtained. The remainder of this thesis will adopt the convention of denoting the tracking error by $\tilde{x}$, and it should be understood that this bound is given by Theorem 1 and the relationship between the output variable y and state x.

Theorem 1 proves that the geometry of the RCI tube Ω_i is uniquely described by the boundary layer thickness Φ_i. Using the terminology introduced by Rakovič et al., the tubes in DTMPC are both homothetic and elastic. For this reason, and the ability to capture state-dependent uncertainty, the approach developed here is called Dynamic Tube MPC. Further, as briefly discussed in [42], a tighter geometry can be obtained if the *current* (as opposed to the predicted) tracking error is used in (3.25).

The importance of (3.22) (and (3.24)) cannot be understated. It gives a precise description for how the tube geometry changes with the level of uncertainty (from the model or otherwise). This is an incredibly useful relation for constructing tubes that are not overly conservative since, in most cases, the model error bound is typically picked to be a large constant because of the difficulty/inability to establish a relation like (3.22) (and (3.24)). By letting the uncertainty be state-dependent, the MPC optimizer (to be discussed in Section 3.5) is able to utilize a more accurate description of the uncertainty to maximize performance. This further underlines the importance of acquiring a high-fidelity model to reduce uncertainty and make the tube as small as possible without using high-bandwidth control.

Another interesting aspect of (3.22) (and (3.24)) is the choice of the cutoff frequency α. In general, α and λ are picked based on control-bandwidth requirements, such as actuator limits or preventing excitation of higher-order dynamics. It is clear that a larger α produces a smaller boundary layer thickness (i.e., high-bandwidth control leads to compact tubes). However, from (3.20), increasing the bandwidth also increases the influence of the uncertainty. Hence, the bandwidth should change depending on the current objective and proximity to state/control constraints for optimal performance (see Section 3.5).

3.5 Dynamic Tube MPC

3.5.1 Overview

This section presents the Dynamic Tube MPC (DTMPC) framework and discusses its properties. The novelty of DTMPC is its ability to change the tube geometry to meet changing objectives and to leverage state-dependent uncertainty to maximize performance. This section first presents a constraint tightening procedure necessary for recursive feasibility. Next, optimizing the tube geometry by adding the control bandwidth as a decision variable is discussed. Lastly, the non-convex formulation of DTMPC is presented. Before proceeding, the following assumption is made about the form of the state and actuator constraints.

Assumption 6. The state and actuator constraints take the form

$$||P_x x + q_x|| \leq c_x, \quad ||P_u u + q_u|| \leq c_u, \tag{3.28}$$

where $|| \cdot ||$ is the L_2 norm, and P_i, q_i, and c_i are constants dictated by the application.

Many physical systems possess these type of constrains so the above assumption is not overly restrictive. Note that state constraints are imposed on the full state of the system x and not just on the output variable y.

3.5.2 Constraint Tightening

State and actuator constraints must be modified to account for the nonzero tracking error and control input caused by model error and disturbances. The following corollary establishes the modified state constraint.

Corollary 1 (Tightened State Constraint). *Assume the control law (3.17) is used as an ancillary controller with associated RCI tube Ω and bounded tracking error $|\tilde{x}|$. Then, the following modified state constraint*

$$||P_x x^* + q_x|| \leq c_x - ||P_x \tilde{x}||, \tag{3.29}$$

guarantees, for all realizations of the uncertainty, the actual state satisfies the original constraint.

Proof. Recall that Theorem 1 established that the boundary layer controller induces a RCI tube Ω with bounded tracking error given by (3.25). Then, the state is always upper bounded by $x \leq x^* + |\tilde{x}|$. Substituting this bound into the state constraint (3.28) and using the triangle inequality, the result is obtained. □

Tightening the actuator constraints is more complicated since the control law in (3.17) depends on the current state x. However, the tracking error bound can be used to obtain an upper bound on the control input that is only a function of the boundary layer thickness, desired state, and dynamics. It is helpful to put the controller into a more useful form for the following theorem

$$u = B(x)^{-1}\left[y^{*(r)} - \hat{F}(x) - \sum_{k=1}^{r-1}\binom{r-1}{k-1}\lambda^{r-k}\tilde{y}^{(k)} - K(x)\mathrm{sat}\left({}^{s}\!/\!{}_{\Phi}\right)\right], \tag{3.30}$$

where the first term is the feedforward (and hence the decision variable in the optimization) and the last three are the feedback terms.

Theorem 2 (Control Input Upper Bound). *Assume that the control law is given by (3.30). Then, the control input is upper bounded, for all realizations of the uncertainty, by*

$$u \leq \bar{B}^{-1}\left[y^{*(r)} + \bar{F} + \sum_{k=1}^{r-1}\binom{r-1}{k-1}\lambda^{n-k}|\tilde{y}^{(k)}| + \bar{K}\right], \tag{3.31}$$

where

$$\bar{B}^{-1} = \max\left\{B^{-1}\left(\underline{x}\right), B^{-1}\left(\bar{x}\right)\right\}, \tag{3.32}$$

$$\bar{F} = \max\left\{\left|\hat{F}\left(\underline{x}\right)\right|, \left|\hat{F}\left(\bar{x}\right)\right|\right\}, \tag{3.33}$$

$$\bar{K} = \max\left\{K\left(\underline{x}\right), K\left(\bar{x}\right)\right\}, \tag{3.34}$$

$\underline{x} := x^* - |\tilde{x}|$, $\bar{x} := x^* + |\tilde{x}|$, *and* $\max\{\cdot\}$ *is the element-wise maximum.*

Proof. The tracking error bound can be leveraged to eliminate the state-dependency in (3.30). Specifically, the state is bounded by

$$x^* - |\tilde{x}| \leq x \leq x^* + |\tilde{x}|, \tag{3.35}$$

where $|\tilde{x}|$ is the solution to (3.25) when equality is imposed. It is clear from (3.30) that to upper bound u, the inverse of the input matrix B^{-1} and the last three feedback terms should be maximized. Define $\underline{x} := x^* - |\tilde{x}|$ and $\bar{x} := x^* + |\tilde{x}|$, then using (3.35), each term in (3.30) can be upper bounded by evaluating at $\underline{x}$ and $\bar{x}$ and taking the maximum, resulting in Equations (3.32) to (3.34) and hence (3.31). □

Remark 2. The bound given by Theorem 2 can be very conservative depending on the structure of the nonlinearities in the dynamics. Tighter bounds can be obtain, for instance, if the dynamics can be expressed as a sum of polynomial functions of a single variable (i.e., $F(x) = \sum_{i=1}^{r} p_i(x_i)$). If this is not the case, then convex approximations of (3.31) can be employed [96, 97].

The bound established by Theorem 2 can be put into a more concise form

$$u \leq \bar{B}^{-1}\left[u^* + \bar{u}_{fb}\right], \tag{3.36}$$

where $u^* := y^{*(r)} - \hat{F}(x^*)$ and $\bar{u}_{fb}$ is the sum of the last three terms in (3.31). Using Theorem 2, the following corollary establishes the tightened actuator constraint.

Corollary 2 (Tightened Actuator Constraint). *Assume the control law (3.17) is used as an ancillary controller with the associated RCI tube Ω and upper bound on input due to feedback $\bar{u}_{fb}$. Then, the following modified actuator constraint*

$$\|P_u \bar{B}^{-1} u^* + q_u\| \leq c_u - \|P_u \bar{B}^{-1} \bar{u}_{fb}\|, \tag{3.37}$$

guarantees, for all realizations of the uncertainty, the actual input satisfies the original constraint.

Proof. Theorem 2 established the upper bound on the control input to be $u \leq \bar{B}^{-1}\left[u^* + \bar{u}_{fb}\right]$.

Substituting this bound into the actuator constraint (3.28) and using the triangle inequality, the result is obtained. □

3.5.3 Optimized Tube Geometry

For many autonomous systems, the ability to react to changing operating conditions is crucial for maximizing performance. For instance, a UAV performing obstacle avoidance should modify the aggressiveness of the controller based on the current obstacle density to minimize expended energy. Formally, the tube geometry must be added as a decision variable in the optimization to achieve this behavior. DTMPC is able to optimize the tube geometry because of the simple relationship between the tube geometry, control bandwidth, and level of uncertainty given by (3.22). This is one of the distinguishing features of DTMPC since other state-of-the-art nonlinear tube MPC algorithms are not able to establish an explicit relationship like (3.22).

In Section 3.3, it was shown that the control bandwidth α is responsible for how the uncertainty affects the sliding variable s. Subsequently, the choice of α influences the tube geometry (via (3.22)) and control gain (via (3.18)). In order to maintain continuity in the control signal, the tube geometry dynamics are augmented such that α and Φ remain smooth. More precisely, the augmented tube dynamics are (ignoring input matrix uncertainty)

$$\begin{aligned} \dot{\Phi} &= -\alpha\Phi + \Delta_F(x^*) + D + \eta, \\ \dot{\alpha} &= v, \end{aligned} \tag{3.38}$$

where $v \in \mathbb{V}$ is an pseudo-control input that will serve as an additional decision variable in the optimization. It is easy to show that the above set of differential equations is stable so long as α remains positive.

3.5.4 Complete Formulation

With Corollary 1 and 2 establishing the tightened state and actuator constraints, Dynamic Tube MPC can be formulated as the following optimization.

Problem 5 – Dynamic Tube MPC

$$
\begin{aligned}
u^*(t),\ x^*(t),\ \alpha^*(t) = \underset{\breve{u}(t),\breve{v}(t),\breve{x}(t),\breve{\alpha}(t)}{\text{argmin}} \quad & J = h_f(\breve{x}(t_f)) + \int_{t_0}^{t_f} \ell(\breve{x}t), \breve{u}(t), \breve{\alpha}(t), \breve{v}(t))dt \\
\text{subject to} \quad & \dot{\breve{x}}(t) = \hat{f}(\breve{x}(t)) + b(\breve{x}(t))\breve{u}(t),\ \dot{\breve{\alpha}}(t) = \breve{v}(t), \\
& \dot{\Phi}(t) = -\breve{\alpha}(t)\Phi(t) + \Delta_F(\breve{x}(t)) + D + \eta, \\
& \dot{\Omega}(t) = A_c\Omega(t) + B_c\Phi(t), \quad \Omega(t_0) = |\tilde{x}(t_0)|, \\
& \breve{x}(t_0) = x_0^*, \quad \breve{\alpha}(t_0) = \alpha_0^* \ \ \Phi(t_0) = \Phi_0, \\
& \breve{x}(t) \in \bar{\mathbb{X}}, \quad \breve{u}(t) \in \bar{\mathbb{U}}, \quad \breve{\alpha}(t) \in \mathbb{A}, \quad \breve{v}(t) \in \mathbb{V}, \\
& \breve{x}\,(t_f) \in \bar{\mathbb{X}}_f, \quad \breve{\alpha}(t_f) \in \mathbb{A}_f,
\end{aligned}
$$

where $\bar{\mathbb{X}}$, $\bar{\mathbb{X}}_f$, and $\bar{\mathbb{U}}$ are the tightened state, terminal, and actuator constraints; $\mathbb{A}$ and $\mathbb{V}$ are the constraints for the bandwidth α and pseudo-control input v; $\mathbb{A}_f$ is the terminal constraint set for the bandwidth; A_c and B_c are the state and input matrix for the controllable canonical form; and h_f and ℓ are the quadratic terminal and stage cost. The output of DTMPC is an optimal control input, trajectory, and controller bandwidth.

DTMPC is inherently a non-convex optimization problem because of the nonlinear dynamics and (possibly) non-convex state/input constraints. However, non-convexity is a fundamental characteristic of nonlinear MPC and a number of approximate solution procedures have been proposed. The key takeaway is that Problem 5 is a nonlinear tube MPC algorithm that *simultaneously* optimizes the open-loop trajectory and tube geometry, eliminating the duality gap in standard tube MPC. Furthermore, conservativeness can be reduced since Problem 5 is able to leverage state-dependent uncertainty to select an open-loop trajectory based on the structure of the uncertainty and proximity to constraints. The benefits of these properties, in addition to combining the tube geometry and error dynamics, will be demonstrated in Chapter 4.

3.6 Summary

This chapter presented the Dynamic Tube MPC (DTMPC) algorithm that addresses a number of limitations of current nonlinear tube MPC algorithms. First, the open-loop MPC optimization is augmented with the tube geometry dynamics enabling the trajectory and tube to be optimized simultaneously. Second, DTMPC is able to utilize state-dependent uncertainty to design more intelligent trajectories, subsequently improving optimization feasibility. And third, the tube geometry and error dynamics can be combined to further reduce conservativeness. All three of these properties were made possible by leveraging the simplicity and robustness of boundary layer sliding control.

Chapter 4

Dynamic Tube MPC: Application and Analysis

4.1 Overview

This chapter presents simulation results demonstrating the two key features of DTMPC: 1) combining the trajectory and tube geometry design into a single optimization; and 2) capturing state-dependent modeling error to construct less conservative trajectories. DTMPC is applied to two domains, both with nonlinear dynamics and state/performance constraints. The first domain is a double integrator system with nonlinear drag performing obstacle avoidance. Results will separately highlight the ability of DTMPC to modify the tube geometry based on proximity to obstacles and leverage state-dependent uncertainty (unknown drag coefficient) to design less conservative trajectories. The second domain is a high-performance aircraft whose latitude (roll) dynamics are subject to oscillations; a phenomenon known as wing rock. Results will show DTMPC's ability to optimize the tube geometry (control bandwidth) to dynamically compensate for model uncertainty while satisfying other performance specifications.

4.2 Collision Avoidance

4.2.1 Overview

Collision avoidance is a fundamental capability for many autonomous systems, and is an ideal test domain for two reasons. First, enough safety margin must be allocated to prevent collisions when model error or disturbances are present. More precisely, the optimizer must leverage knowledge of the peak tracking error (given by the tube geometry) to prevent collisions. The robustness of DTMPC and ability to utilize knowledge of state-dependent uncertainty can thus be demonstrated. Second, many real-world operating environments have variable obstacle densities so the tube geometry can be optimized in response to a changing environment. The rest of this section presents the model, formal optimal control problem, results, and analysis.

4.2.2 Model

This section uses a double integrator model with nonlinear drag, which describes the dynamics of many mechanical systems. Let $r = [r_x \; r_y \; r_z]^T \in \mathbb{R}^3$ be the inertial position of the system that is to be tracked. The dynamics are

$$\ddot{r} = -C_d \left\| \dot{r} \right\| \dot{r} + g + u + d, \tag{4.1}$$

where $\| \cdot \|$ is the L^2 norm, $g \in \mathbb{R}^3$ is the gravity vector, C_d is the unknown but bounded drag coefficient ($0 \leq C_d \leq \bar{C}_d$), and d is a bounded disturbance ($|d| \leq D$). From (3.17), the control law is

$$u = \ddot{r}^* + \hat{C}_d \left\| \dot{r} \right\| \dot{r} - \lambda \dot{\tilde{r}} - K \mathrm{sat}\left({}^{s}/_{\Phi} \right), \tag{4.2}$$

where $\hat{C}_d$ is the best estimate of the drag coefficient, $s = \dot{\tilde{r}} + \lambda \tilde{r}$ is the sliding variable, and

$$K = \bar{C}_d \left\| \dot{r} \right\| \left| \dot{r} \right| - \bar{C}_d \left\| \dot{r}^* \right\| \left| \dot{r}^* \right| + \alpha \Phi, \tag{4.3}$$

$$\dot{\Phi} = -\alpha \Phi + \bar{C}_d \left\| \dot{r}^* \right\| \left| \dot{r}^* \right| + D + \eta, \tag{4.4}$$

are the robust gain and boundary layer dynamics, respectively, with $|\cdot|$ being the element-wise absolute value.

4.2.3 Collision Avoidance DTMPC

Let H, p_c, and r_o denote the shape, location, and size of an obstacle. The minimum control effort DTMPC optimization (for a given final time) with collision avoidance for system (4.1) is presented in Problem 6.

Problem 6 – Collision Avoidance DTMPC

$$
\begin{aligned}
u^*(t), r^*(t), \alpha^*(t) = \underset{\check{u}(t),\check{v}(t),\check{r}(t),\check{\alpha}(t)}{\text{argmin}} \quad & J = \tilde{\alpha}(t)^T R_f \tilde{\alpha}(t) + \int_{t_0}^{t_f} \left[\check{u}(t)^T Q \check{u}(t) + \tilde{\alpha}(t)^T R \tilde{\alpha}(t)\right] dt \\
\text{subject to} \quad & \ddot{\check{r}}(t) = -\hat{C}_d \left\|\dot{\check{r}}(t)\right\| \dot{\check{r}}(t) + g + \check{u}(t), \;\; \dot{\check{\alpha}}(t) = \check{v}(t), \\
& \dot{\Phi}(t) = -\alpha(t)\Phi(t) + \bar{C}_d \left\|\dot{\check{r}}(t)\right\| \left|\dot{\tilde{r}}(t)\right| + D + \eta, \\
& \dot{\Omega}(t) = A_c\Omega(t) + B_c\Phi(t), \quad \Omega(t_0) = |\tilde{r}(t_0)|, \\
& \check{r}\left(t_0\right) = r_0^*, \quad \Phi\left(t_0\right) = \Phi_0, \quad \check{r}\left(t_f\right) = r_f^*, \\
& \|H_i\check{r}(t) - p_{c,i}\| \geq r_{o,i} + \|H_i\tilde{r}(t)\|, \;\; i = 1, \cdots, N_o, \\
& |\dot{\check{r}}(t)| \leq \dot{r}_m - |\dot{\tilde{r}}(t)|, \quad \|\check{u}(t)\| \leq u_m - \bar{u}_{fb}, \quad |\check{v}(t)| \leq v_m, \\
& 0 < \underline{\alpha} \leq \check{\alpha}(t) \leq \bar{\alpha}, \quad \tilde{\alpha}(t) = \check{\alpha}(t) - \underline{\alpha},
\end{aligned}
$$

$\underline{\alpha}$ and $\bar{\alpha}$ are the upper and lower bounds of the control bandwidth, $\dot{r}_m$ is the peak desired speed, v_m is the max pseudo-control input, and N_o is the number of obstacles.

4.2.4 Simulation Environment

DTMPC was tested in simulation to demonstrate its ability to optimize tube geometry and utilize knowledge of state-dependent uncertainty through an environment with obstacles. The obstacles were placed non-uniformly to emulate a changing operating condition (i.e., dense/open environment). In order to emphasize both characteristics of DTMPC, three test cases were conducted. First, the bandwidth was optimized when both the model and

Table 4.1: DTMPC Collision Avoidance Simulation Parameters.

Param.	**Value**	**Param.**	**Value**
r_0	$[0\ 0\ 1]^T$ m	r_f	$[0\ 25\ 1]^T$ m
$\dot{r}_0$	$[0\ 1\ 0]^T$ m/s	$\dot{r}_f$	$[0\ 1\ 0]^T$m/s
λ	$[2\ 2\ 2]^T$ rad/s	R_f	$2I_3$
Q	$2I_3$	R	$0.1I_3$
$\underline{\alpha}$	0.5 rad/s	$\bar{\alpha}$	4 rad/s
u_m	5 m/s^2	v_m	2 rad/s^2
$\dot{r}_m$	2.5 m/s	D	0.5 m/s^2
$\hat{C}_d$	0.1 kg/m	$\bar{C}_d$	0.2 kg/m
η	0.1 m/s^2	N_o	5

obstacle locations were completely known. Second, the bandwidth was again optimized with a known model but the obstacle locations were unknown, requiring a receding horizon implementation. Third, state-dependent uncertainty is considered but control bandwidth is kept constant. Nothing about the formulation prevents optimizing bandwidth and leveraging state-dependent uncertainty simultaneously in a receding horizon fashion (as will be shown in Section 4.3), this decoupling is only for clarity. The tracking error (3.25) is used to tighten the obstacle and velocity constraint.

Problem 6 is non-convex due to the nonlinear dynamics and non-convex obstacle constraints so sequential convex programming, similar to that in [98], was used to obtain a solution. The optimization was initialized with a naïve straight-line solution and solved using YALMIP [99] and MOSEK [100] in MATLAB. If large perturbations to the initial guess are required to find a feasible solution, then warm-starting the optimization with a better initial guess (possibly provided by a global geometric planner) might be necessary. For the cases tested in this work, the optimization converged within three to four iterations – fast enough for real-time applications. The simulation parameters are summarized in Table 4.1.

4.2.5 Results and Analysis

Optimized Tube Geometry

The first test scenario for DTMPC highlights its ability to simultaneously optimize an open-loop trajectory and tube geometry in a known environment with obstacles placed

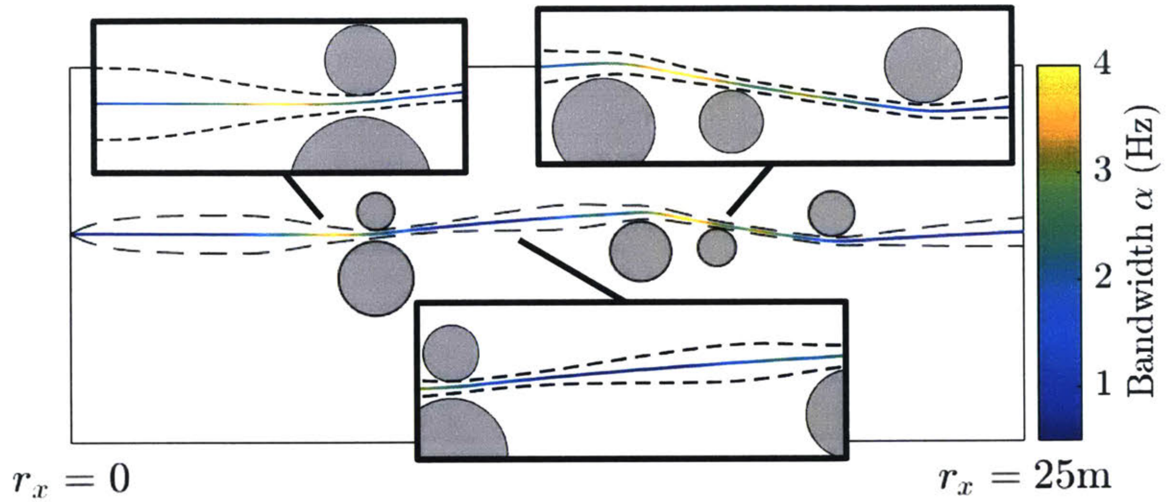


Figure 4-1: DTMPC simultaneously optimizing an open-loop trajectory (multi-color) and tube geometry (black) around obstacles (grey). High-bandwidth control (yellow) is used when in close proximity to obstacles while low-bandwidth control (dark blue) is used in open space.

non-uniformly. Fig. 4-1 shows the open-loop trajectory (multi-color), tube geometry (black), and obstacles (grey) when DTMPC optimizes both the trajectory and tube geometry. The color of the trajectory indicates the spatial variation of the control bandwidth, where low- and high-bandwidth are mapped to darker and lighter color, respectively. It is clear that the bandwidth changes dramatically along the trajectory, especially in the vicinity of obstacles. The inserts in Fig. 4-1 show that high-bandwidth (compact tube geometry) is used for the narrow gap and slalom and low-bandwidth (large tube geometry) for open space. Hence, high-bandwidth control is used only when the system is in close proximity to constraints (i.e., obstacles), consequently limiting aggressive control inputs to only when they are absolutely necessary. Thus, DTMPC can react to varying operating conditions by modifying the trajectory and tube geometry appropriately.

DTMPC is able to dictate how aggressively the controller compensates for model error/disturbances by optimizing the control bandwidth. One measure of aggressiveness is the closed-loop control effort $\mathcal{U} := \int_{t_0}^{t_f} \|u\|^2 dt$ (which can also be viewed as a measure of power consumption) as the controller tracks the desired trajectory. 1000 simulations with a white noise disturbance were conducted to compare the closed-loop control effort using static high-bandwidth control and the bandwidth generated by DTMPC. Table 4.2 shows that

Table 4.2: Closed-Loop Control Effort $\mathcal{U} = \int_{t_0}^{t_f} \|u\|^2 dt$.

$\mathcal{U}_{\alpha_{\max}}$	$\mathcal{U}_{\text{DTMPC}}$	**% Reduction**
4.2 ± 0.35	3.3 ± 0.34	21.4

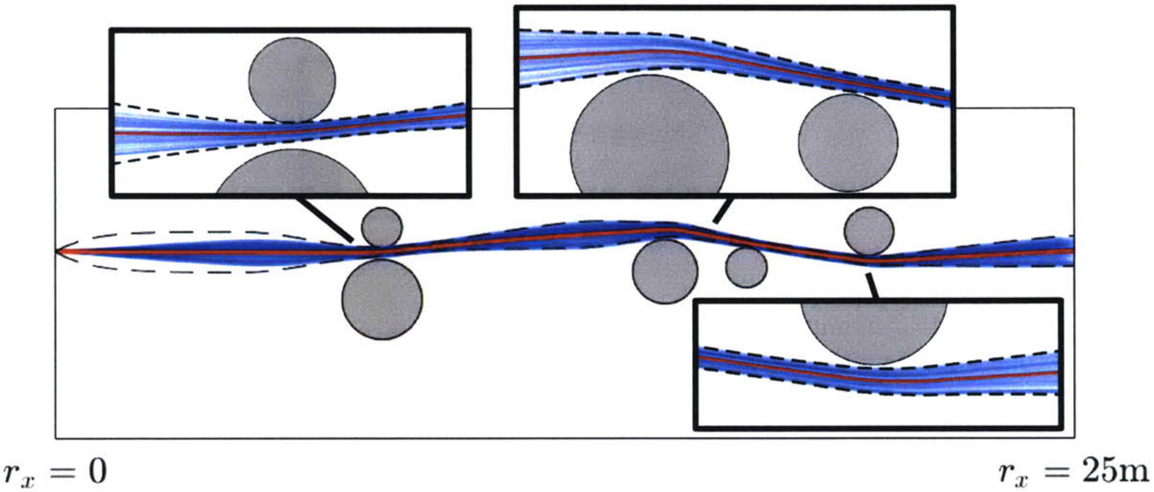


Figure 4-2: Monte Carlo verification that the time-varying tube in DTMPC remains invariant. The closed-loop system (blue) was simulated with a different disturbance profile uniformly sampled from the disturbance set.

DTMPC achieves a 21.4% reduction in $\mathcal{U}$, indicating the control signal is less energetic than that of the high-bandwidth controller. This shows that optimizing the control bandwidth within DTMPC leads to less aggressive/energetic feedback control inputs, which can have substantial benefits for energy-constrained systems.

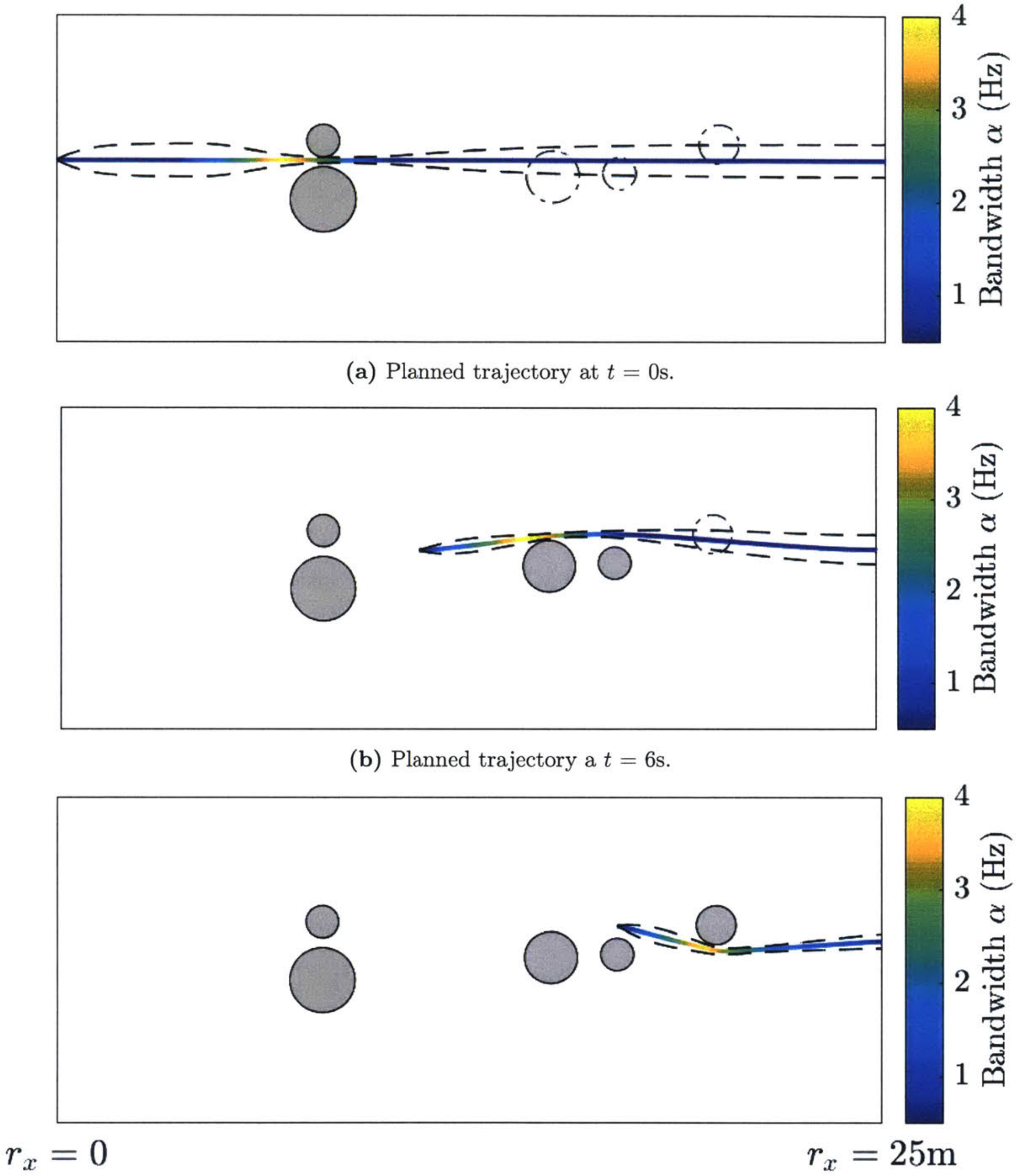
Since the tube geometry changes dramatically along the trajectory, it is important to verify that the tube remains invariant to uncertainty. This was tested by conducting 1000 simulations of the closed-loop system with a disturbance profile sampled uniformly from the disturbance set $\mathbb{D}$. Fig. 4-2 shows the nominal trajectory (red), each closed-loop trial run (blue), tube geometry (black), and obstacles (grey). The inserts show that the state stays within the tube, even as the geometry changes, which verifies that boundary layer control induces a RCI tube.

Receding Horizon Optimized Tube Geometry

In many situations the operating environment is not completely known and requires a receding horizon implementation. The second test scenario for DTMPC highlights its ability to simultaneously optimize an open-loop trajectory and tube geometry in a unknown environment. Fig. 4-3 shows a receding horizon implementation of DTMPC where only a subset of obstacles are known (dark-grey) and the rest are unknown (light-grey). The bandwidth along the trajectory is visualized with the color map where low- and high-bandwidth are mapped to darker and lighter color. The first planned trajectory (Fig. 4-3a) uses high-bandwidth at the narrow gap and low-bandwidth in open space. When the second and third set of obstacles are observed, Fig. 4-3b and Fig. 4-3c respectively, DTMPC modifies the trajectory to again use high-bandwidth when in close-proximity to newly discovered obstacles. This further demonstrates DTMPC's ability to construct an optimized trajectory and tube geometry in response to new obstacles.

State-Dependent Uncertainty

The third test scenario for DTMPC highlights its ability to leverage knowledge of state-dependent uncertainty, in this case arising from an unknown drag coefficient. From (4.4), the uncertainty scales with the square of the velocity so higher speeds increase uncertainty. Fig. 4-4 shows the open-loop trajectory (multi-color), tube geometry (black), and obstacles (grey) when DTMPC leverages state-dependent uncertainty. The color of the trajectory is an indication of the instantaneous speed, where low and high speed are mapped to darker and lighter color, respectively. It is clear that DTMPC generates a speed profile dictated by the proximity to obstacles. For instance, using the inserts in Fig. 4-4, the speed is lower (darker) when the trajectory goes through the narrow gap and around the other obstacles; reducing uncertainty and tightening the tube geometry. Further, the speed is higher (lighter) when in the open, subsequently increasing uncertainty causing the tube geometry to expand. If the state-dependent uncertainty is just assumed to be bounded, a simplification often made out of necessity in other tube MPC algorithms, the tube geometry is so large that, for this obstacle field, the optimization is infeasible with the same straight-line initialization as



(a) Planned trajectory at $t = 0$s.

(b) Planned trajectory a $t = 6$s.

(c) Planned trajectory at $t = 8$s.

Figure 4-3: Receding horizon implementation of DTMPC with known (dark-grey) and unknown (light-grey) obstacles. The bandwidth along trajectory (multi-color) varies, resulting in a dynamic tube geometry (black). (a): First planned trajectory and tube geometry when only the first two obstacles are known. (b): New planned trajectory when the next two obstacles are observed. (c): New planned trajectory when the last obstacle is observed.

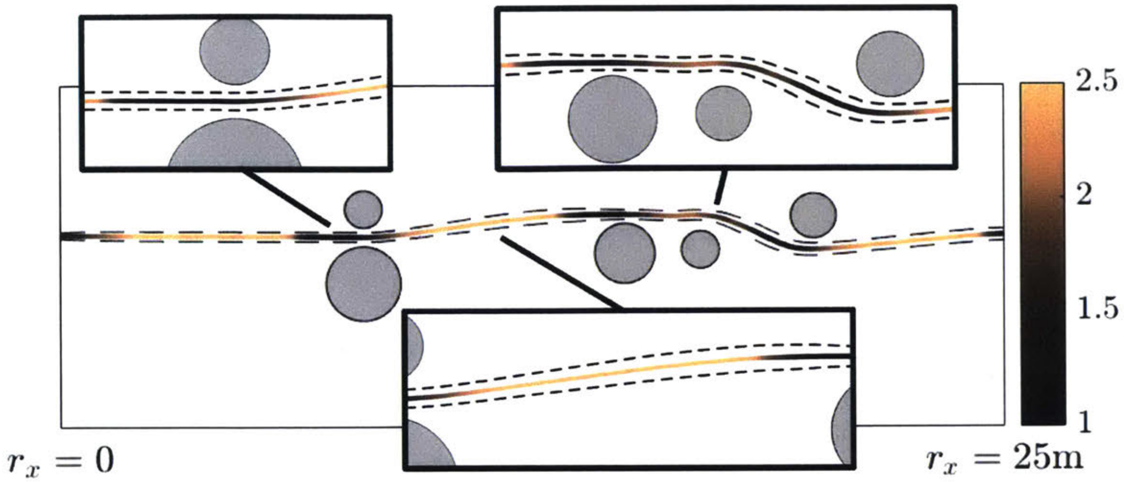


Figure 4-4: DTMPC leveraging state-dependent uncertainty to robustly avoid obstacles (grey). The speed along the trajectory, given by the color map, is low (dark) when in close proximity to obstacles and is high (light) in open regions. This causes the tube geometry (black) to contract and expand.

DTMPC. Hence, DTMPC is able to leverage knowledge of state-dependent uncertainty to reduce conservatism and improve feasibility.

4.3 Wing Rock

4.3.1 Overview

High-performance aircraft operating at high-angle of attack can exhibit wing rock, or limit cycle oscillations in the lateral (roll) dynamics [101–104]. These oscillations severely degrade the performance of the aircraft and can have catastrophic consequences if not compensated for. The aerospace and controls communities have devoted significant resources to modeling wing rock and developing robust controllers that damp out these oscillations. The highly nonlinear dynamics and large uncertainty in the aerodynamic coefficients makes wing rock an ideal test domain for DTMPC. The rest of this section presents the model, formal optimal control problem, results, and analysis.

4.3.2 Model

Let $\phi,\ \dot{\phi} \in \mathbb{R}$ be the roll angle and roll rate of a high-performance aircraft. The nonlinear roll dynamics for an aircraft susceptible to wing rock are

$$\ddot{\phi} = a_1\phi + a_2\dot{\phi} + a_3|\phi|\dot{\phi} + a_4|\dot{\phi}|\dot{\phi} + u + d, \tag{4.5}$$

where u is the input (typically aileron deflection), d is an unknown but bounded disturbance, and a_i are empirical aerodynamic coefficients.

The dynamics given by (4.5) converge to a stable limit cycle when the input u is identically zero. Hence, both roll and roll rate exhibit oscillatory behavior which can be difficult for a pilot or flight computer to compensate for. An example of this motion is shown in Fig. 4-5 where the roll angle and roll rate, shown in Fig. 4-6a and Fig. 4-6b, oscillate with a fixed amplitude (after the initial transients) and frequency. The phase plane, shown in Fig. 4-5c, is consistent with (4.5) being a stable nonlinear oscillator. This interesting nonlinear behavior has drawn significant attention both from the adaptive and nonlinear controls community, making it a standard adaptive/nonlinear control benchmark domain.

The coefficients a_i in (4.5) are typically uncertain so the boundary layer controller is given by

$$u = \ddot{\phi}^* - \hat{a}_1\phi - \hat{a}_2\dot{\phi} - \hat{a}_3|\phi|\dot{\phi} - \hat{a}_4|\dot{\phi}|\dot{\phi} - \lambda\dot{\tilde{\phi}} - K\mathrm{sat}\left({}^{s}/_{\Phi}\right) \tag{4.6}$$

where $\hat{a}_i$ is the best estimate of parameter a_i, $s = \dot{\tilde{\phi}} + \lambda\tilde{\phi}$ is the sliding variable, and

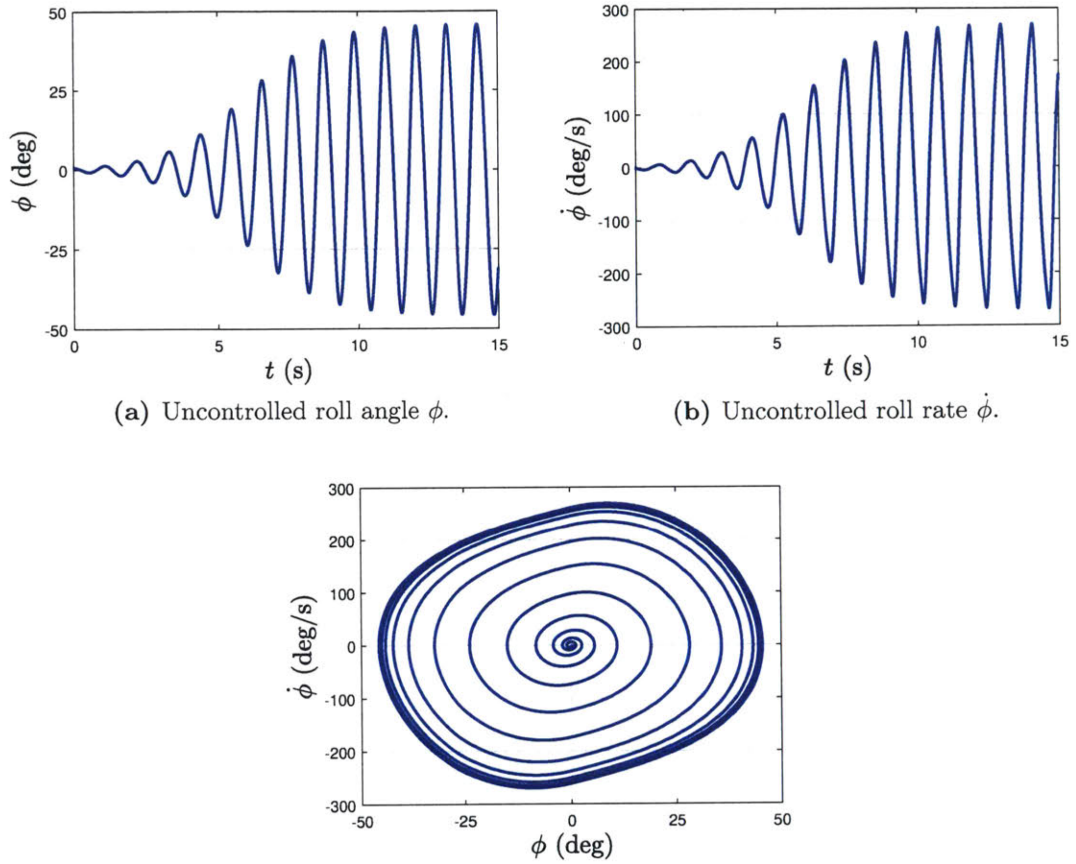
$$K = \left|\tilde{a}_1\phi^* + \tilde{a}_2\dot{\phi}^* + \tilde{a}_3|\phi^*|\dot{\phi}^* + \tilde{a}_4|\dot{\phi}^*|\dot{\phi}^*\right| + D + \eta - \dot{\Phi}, \tag{4.7}$$

$$\dot{\Phi} = -\alpha\Phi + \left|\tilde{a}_1\phi^* + \tilde{a}_2\dot{\phi}^* + \tilde{a}_3|\phi^*|\dot{\phi}^* + \tilde{a}_4|\dot{\phi}^*|\dot{\phi}^*\right| + D + \eta, \tag{4.8}$$

are the robust gain and boundary layer dynamics, respectively.

4.3.3 Wing Rock DTMPC

In Section 4.2, tube geometry optimization (via changing control bandwidth) and leveraging the structure of the model uncertainty were decoupled to highlight each feature. In this



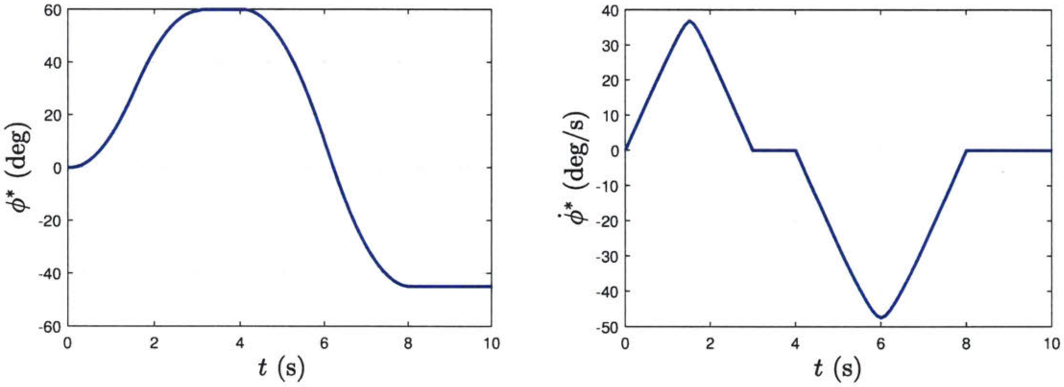
(a) Uncontrolled roll angle ϕ.

(b) Uncontrolled roll rate $\dot{\phi}$.

(c) Phase plane.

Figure 4-5: Natural oscillations of the latitude dynamics of a high-performance aircraft susceptible to wing rock. (a): Roll angle converges to oscillation of fixed amplitude and frequency. (b): Roll rate converges to fixed amplitude and frequency. (c): Phase plane of the roll angle and rate consistent with stable limit cycle.

section, DTMPC is used to optimize the control bandwidth in order to reduce the affect of model uncertainty and achieve specific performance criteria. In particular, suppose a high-performance vehicle wants to execute the trajectory shown in Fig. 4-6, which was generated without considering model uncertainty. Further suppose that the tracking error must be kept below a certain threshold but persistent high-bandwidth control is not desirable. DTMPC can be used to construct an optimal control bandwidth profile that satisfies the tracking error requirements by using high-bandwidth control to compensate model uncertainty only when necessary (i.e., near constraints). The wing rock DTMPC optimization is presented in Problem 7.



(a) Desired roll angle ϕ^*.

(b) Desired roll rate $\dot{\phi}^*$.

Figure 4-6: Desired trajectory for a high-performance aircraft susceptible to wing rock. (a): Roll angle. (b): Roll rate. Both roll angle and roll rate were generated without considering model uncertainty.

Problem 7 – Wing Rock DTMPC

$$
\begin{aligned}
\alpha^*(t) = \underset{\check{v}(t),\check{\alpha}(t)}{\operatorname{argmin}} \quad & J = \int_0^{t_f} \tilde{\alpha}(t)^2 dt \\
\text{subject to} \quad & \dot{\Phi}(t) = -\check{\alpha}(t)\Phi(t) + F(\phi^*(t), \dot{\phi}^*(t)) + D + \eta, \\
& \dot{\check{\alpha}}(t) = \check{v}(t), \quad \dot{\Omega}(t) = -\lambda\Omega(t) + \Phi(t), \\
& \Phi\,(0) = \Phi_0, \quad \check{\alpha}\,(0) = \alpha_0, \quad \Omega(0) = |\tilde{\phi}(0)|, \\
& \Omega(t) \leq \tilde{\phi}_m \quad |v(t)| \leq v_m, \quad 0 < \underaccent{\tilde}{\alpha} \leq \check{\alpha}(t) \leq \bar{\alpha},
\end{aligned}
$$

$$
\begin{aligned}
\text{where} \quad & F(\phi^*(t), \dot{\phi}^*(t)) = \left| \tilde{a}_1\phi^*(t) + \tilde{a}_2\dot{\phi}^*(t) + \tilde{a}_3|\phi^*(t)|\dot{\phi}^*(t) + \tilde{a}_4|\dot{\phi}^*(t)|\dot{\phi}^*(t) \right|, \\
& \tilde{\alpha}(t) = \check{\alpha}(t) - \underaccent{\tilde}{\alpha},
\end{aligned}
$$

$|\cdot|$ is the element-wise absolute value, $\underaccent{\tilde}{\alpha}$ and $\bar{\alpha}$ are the upper and lower bounds of the control bandwidth, $\tilde{\phi}_m$ is the maximum allowable tracking error, and v_m is the max pseudo-input.

Table 4.3: Wing Rock Aerodynamic Coefficients.

$\boldsymbol{a_1}$ (s^{-2})	$\boldsymbol{a_2}$ (s^{-1})	$\boldsymbol{a_3}$ (s^{-1})	$\boldsymbol{a_4}$ (-)
-32.7 ± 2.5	1.43 ± 0.5	-5.48 ± 1.2	0.1 ± 0.02

Table 4.4: DTMPC Wing Rock Simulation Parameters.

Param.	Value	Param.	Value
$\underline{\alpha}$	4 rad/s	$\bar{\alpha}$	10 rad/s
$\tilde{\phi}_m$	10 deg	v_m	5 rad/s^2
λ	2 rad/s	D	57 deg/s^2
$\Phi(t_0)$	15 deg/s	$\alpha(t_0)$	4 rad/s
$\Omega(0)$	0 deg	η	5.7 deg/s^2

4.3.4 Simulation Environment

Simulations were conducted in MATLAB using YALMIP [99] and MOSEK [100]. The coefficients a_i are taken from [103] and summarized in Table 4.3. Other simulation parameters can be found in Table 4.4. Even with the desired trajectory known beforehand, Problem 7 is nonconvex due to the product of α and Φ in the boundary layer dynamics. Sequential convex programming was again used to formulate Problem 7 as a convex optimization by linearizing the nonlinear dynamics about an initial guess. Re-linearizing and solving the convex optimization was repeated until convergence, which occurred only after three or four iterations.

4.3.5 Results and Analysis

The unique ability of DTMPC to both optimize the tube geometry (via control bandwidth) and capture state-dependent uncertainty means that the boundary layer controller can be designed to explicitly mitigate model uncertainty while satisfying other performance criteria. Specifically, high-bandwidth control can be utilized when the model uncertainty is large for a particular state, and vice versa. Fig. 4-7 shows the trajectory (multi-color) and corresponding tube geometry (black). The color map describes the temporal evolution of the control bandwidth along the trajectory; low/high bandwidth corresponds to darker/lighter color.

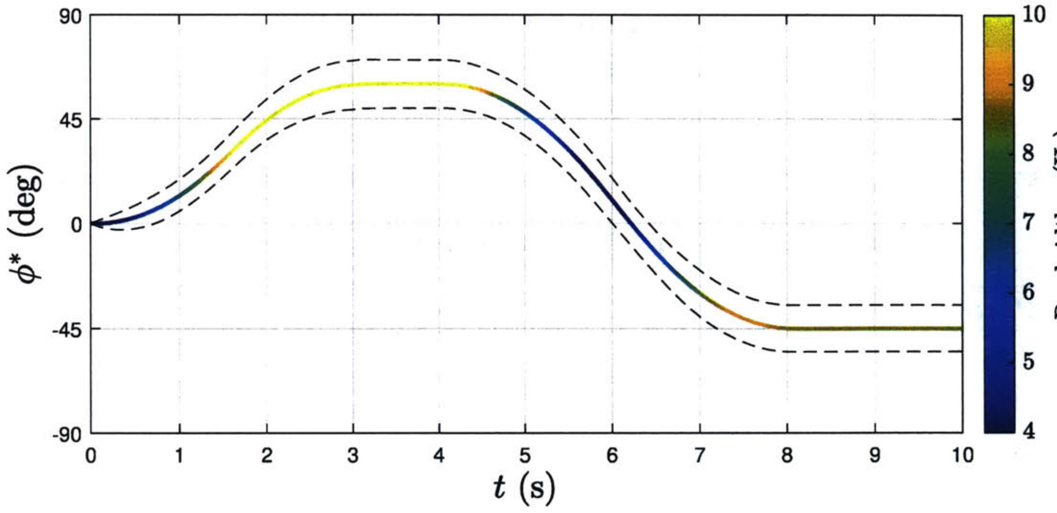
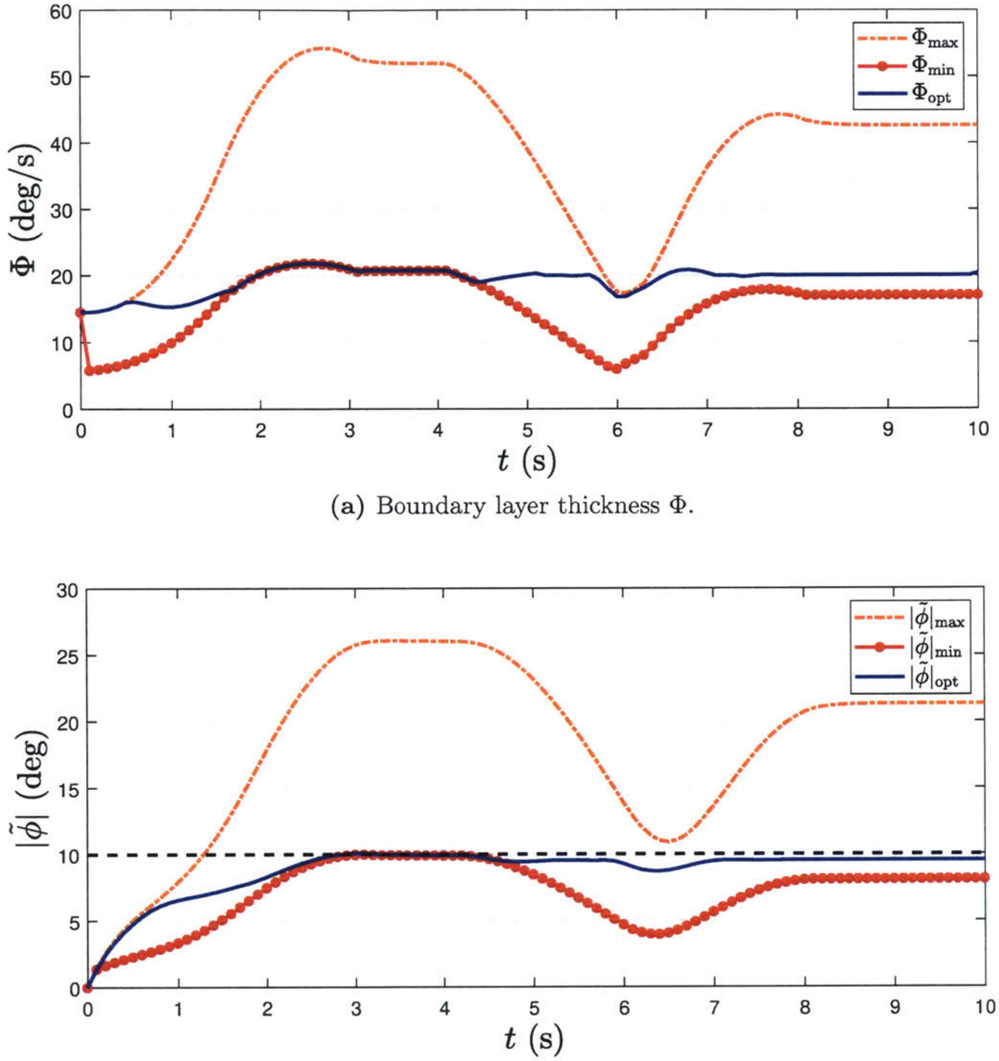


Figure 4-7: Desired roll angle (multi-color) and control bandwidth (color map) generated by DTMPC to compensate for model uncertainty. High-bandwidth control (lighter color) is used for large roll angles while low bandwidth (darker color) is used for small roll angles, indicating the model uncertainty is dominated by the roll angle. The unique feature of DTMPC is the ability compensate for model uncertainty by changing the control bandwidth; which is only possible since the structure of the uncertainty is directly considered in DTMPC. The tube geometry (black) is also shown.

High-bandwidth (lighter color) is used for large roll angles while low-bandwidth is used for small roll angles. This indicates the model uncertainty is largest for large roll angles, which is compensated for by high-bandwidth control.

To gain additional insight, the maximum Φ_{max} (orange), minimum Φ_{min} (red), and optimal Φ_{opt} (blue) boundary layer thickness is shown in Fig. 4-8a. Note that Φ_{max} and Φ_{min} correspond to $\alpha = \underline{\alpha}$ and $\alpha = \bar{\alpha}$, respectively. Both Φ_{max} and Φ_{min} are indeed largest for large roll angles, confirming that the roll angle term in (4.5) dominates the model uncertainty. The optimal boundary layer thickness Φ_{opt} is able to lie between Φ_{max} and Φ_{min} since the bandwidth of the controller changes along the trajectory. In addition, Fig. 4-8b shows the maximum $|\tilde{\phi}_{\text{max}}|$ (orange), minimum $|\tilde{\phi}_{\text{min}}|$ (red), and optimal $|\tilde{\phi}_{\text{opt}}|$ (blue) tracking error bound. Both $|\tilde{\phi}_{\text{min}}|$ and $|\tilde{\phi}_{\text{opt}}|$ satisfy the max tracking error constraint (black) but $|\tilde{\phi}_{\text{opt}}|$ uses less control bandwidth on average (as indicated by darker colors in Fig. 4-7). Hence, DTMPC is able to balance the two competing objectives of low tracking error and minimal bandwidth by leveraging the structure of the model uncertainty within the optimization. This again demonstrates the usefulness of capturing state-dependent uncertainty and optimizing the tube geometry (via control bandwidth), both of which are unique to DTMPC.



(a) Boundary layer thickness Φ.

(b) Tracking error bound $|\tilde{\phi}|$.

Figure 4-8: Boundary layer thickness and tracking error bound for different values of bandwidth α. (a): Maximum (orange), minimum (red), and optimal (blue) boundary layer thickness generated by DTMPC. The optimal boundary layer is able to lie in-between the max and min since the bandwidth can change along the trajectory. (b): Maximum (orange), minimum (red), and optimal (blue) tracking error bound generated by DTMPC. The optimal tracking error bound is able to satisfy the maximum allowable threshold (black) without excessive bandwidth by leverage the structure of the model uncertainty.

4.4 Summary

This chapter presented simulation results and analysis of DTMPC's performance in two domains: 1) a double integrator system with nonlinear drag performing collision avoidance;

and 2) the nonlinear latitude dynamics of a high-performance aircraft. The collision avoidance results separately demonstrated that DTMPC can optimize the tube geometry (via control bandwidth) to satisfy changing constraints and leverage state-dependent uncertainty to construct less conservative trajectories. Results of the nonlinear latitude dynamics of an aircraft susceptible to wing rock showed that DTMPC is able to optimize the control bandwidth to directly compensate for model uncertainty while satisfying other performance specifications (e.g., maximum allowable tracking error). All results are a direct consequence of the unique features of DTMPC and are not obtainable with current state-of-the-art nonlinear tube MPC frameworks.

Chapter 5

Adaptive Dynamic Tube MPC

5.1 Overview

This chapter presents the Adaptive Dynamic Tube MPC (ADTMPC) framework for nonlinear systems. The key feature of ADTMPC is the ability to reduce model uncertainty while maintaining recursive feasibility. The structure of this chapter is as follows. The problem formulation and necessary assumptions are first presented. Set membership identification (SMID), used to reduce model uncertainty in ADTMPC, is then derived and discussed in detail. The main technical result of the chapter, proving that the model uncertainty monotonically decreases even with disturbances/sensor noise, is then presented. Finally, the ADTMPC framework is formally defined and discussed.

5.2 Problem Formulation

Consider again the following nonlinear, time-invariant, and control-affine system

$$\begin{aligned} \dot{x} &= f(x) + b(x)u + d \\ y &= h(x) \end{aligned} \tag{5.1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ the control input, $d \in \mathbb{R}^n$ an unknown external disturbance, and $y \in \mathbb{R}^m$ are the elements of x to be controlled. As was shown in Chapter 2 and Chapter 3, by differentiating the output y_i until the input appears, (5.1) can be expressed as

$$y_i^{(r_i)} = L_f^{r_i} h_i(x) + \sum_{j=0}^{m} L_{b_j} L_f^{r_i} h_i(x) u_j + d_i, \tag{5.2}$$

or in vector form

$$y^{(r)} = F(x) + B(x)u + d, \tag{5.3}$$

where $F(x)$ and $B(x)$ are appropriately defined. Assumption 2 and Assumption 3 in Chapter 3 established bounds on the model uncertainty (i.e., $|\tilde{F}(x)| \leq \Delta_F(x)$ and $|\tilde{B}(x)| \leq \Delta_B(x)$). In this chapter, the following stricter assumptions are made about the structure of the model uncertainty.

Assumption 7. The dynamics F can be expressed as a linear function of parameter vector $\theta \in \mathbb{R}^p$, which consists of nominal value $\hat{\theta}$ and error term $\tilde{\theta}$. Specifically, the dynamics F takes the form $F = \varphi_F(x)^T \left(\hat{\theta} + \tilde{\theta}\right)$ where $\varphi_F(x) \in \mathbb{R}^{p \times r}$ is called the *dynamics regressor.*

Assumption 8. The input matrix B can be expressed as a linear function of parameter vector $\rho \in \mathbb{R}^{q \times m}$, which consists of nominal value $\hat{\rho}$ and error term $\tilde{\rho}$. Specifically, the input matrix B takes the form $B = \varphi_B(x)^T \left(\hat{\rho} + \tilde{\rho}\right)$ where $\varphi_B(x) \in \mathbb{R}^{q \times m}$ is called the *input matrix regressor.*

Assumption 7 and Assumption 8 restrict the class of uncertainties to ones that are linear in the unknown parameters but (possibly) nonlinear regressors. These are standard assumptions within the adaptive control/MPC community and one that many physical systems satisfy.

So far throughout this thesis, the premise for reducing model uncertainty has been primarily anecdotal: reducing model uncertainty reduces conservativeness and performance improves as a result. We are now at the point where this insight can be established mathematically. Using Assumption 7 and Assumption 8, the boundary layer controller from Chapter 3, again ignoring input matrix uncertainty for clarity, becomes

$$u = \left(\varphi_B(x)^T \hat{\rho}\right)^{-1} \left[-\varphi_F(x)^T \hat{\theta} - y_p^{(r)} - K(x) \mathrm{sat}\left({}^{s}/_{\Phi}\right)\right], \tag{5.4}$$

where

$$K(x) = |\varphi_F(x)^T \tilde{\theta}| + D + \eta - \dot{\Phi}, \tag{5.5}$$

and

$$\dot{\Phi} = -\alpha\Phi + |\varphi_F(x^*)^T\tilde{\theta}| + D + \eta. \tag{5.6}$$

The goal of updating the prediction model is to obtain a more accurate estimate of the parameter vector $\hat{\theta}$ (i.e., $\tilde{\theta} \rightarrow 0$). Let K_u and Φ_u denote the robust gain and boundary layer thickness with the updated parameter vector. In the limit as $\tilde{\theta} \rightarrow 0$ then

$$K_\mathrm{u}(x) = D + \eta - \dot{\Phi}_\mathrm{u}, \tag{5.7}$$

and

$$\dot{\Phi}_\mathrm{u} = -\alpha\Phi_\mathrm{u} + D + \eta. \tag{5.8}$$

It is clear that $K_\mathrm{u} \leq K$ and $\Phi_\mathrm{u} \leq \Phi$. Thus, by reducing model uncertainty, *less* control effort and *smaller* boundary layer thickness (and hence tracking error from Theorem 1) are achieved. The remainder of this chapter will develop the methodology of obtaining $\hat{\theta}$ and how it can be integrated into DTMPC without losing stability and recursive feasibility guarantees.

5.3 Set Membership Identification

Techniques to update model parameters can be broadly classified as being adaptive- or estimation-based. Adaptive control entails updating model parameters based on instantaneous tracking error with an adaptation law derived using Lyapunov stability arguments. More traditional estimation, such as recursive least squares, Kalman filtering, smoothing, rely on prediction error in the input-output data to update model parameters. While the implementation of these approaches can be quite different, they both obtain a point estimate of the true parameter value. In doing so, however, the parameter error may not monotonically decrease since state observations are corrupted by noise and/or external disturbances. As a result, nothing can be inferred about the current/future parameter error from the current parameter value. The inability to predict the evolution of the parameter error rules out these techniques as possible means for updating the prediction model within DTMPC.

Set membership identification (SMID) takes a different approach: it eliminates parameter values that are not consistent with the dynamics, disturbance model, and observations. By eliminating values that are not physically realizable, an estimate of the true value can be obtained. Furthermore, and arguably more importantly, the parameter error can be shown to decrease monotonically; the defining feature that enables combining SMID and DTMPC while maintaining recursive feasibility.

In addition to Assumption 4 (disturbance bound), Assumption 7, and Assumption 8, SMID requires the following assumptions.

Assumption 9. The true parameter vectors θ^* and ρ^* belongs to a *known* closed, convex set Θ (i.e., $\theta^*, \ \rho^* \in \Theta$).

Assumption 9 is not restrictive in practice as many robust control design techniques (including boundary layer control) require the model error bound to be known.

Assumption 10. The control law u must be a certainty equivalence controller. Specifically, u must stabilize the system $\forall \ \theta, \ \rho \in \Theta$.

Assumption 10 can, in general, be hard to satisfy even for linear systems. However, as was proved in Chapter 3, boundary layer control not only stabilizes (5.1) (under assumptions about stable zeros) but generates a robust control invariant tube for both bounded model error and bounded disturbances.

The idea behind SMID is simple: since the dynamics given by (5.3) can be rewritten as

$$d = y^{(r)} - \varphi_F(x)^T\theta - \varphi_B(x)^T\rho u, \tag{5.9}$$

and since $d \in \mathbb{D}$, a new set Ξ can be constructed that contains θ and ρ. Let t_k denote a particular sample time that a state measurement is acquired. Then, the set Ξ takes the form

$$\Xi = \left\{\theta \in \mathbb{R}^p, \ \rho \in \mathbb{R}^{q\times m} : y^{(r)}(t_k) - \varphi_F(x(t_k))^T\theta - \varphi_B(x(t_k))^T\rho u(t_k) \in \mathbb{D}\right\}. \tag{5.10}$$

Fundamentally, the set given by (5.10) is constructed by finding θ and ρ such that the dynamics in (5.3) belongs to the disturbance set $\mathbb{D}$. Since $\mathbb{D} = \{d : |d| \leq D\}$ via Assumption 4, then (5.10) can be rewritten as

$$\Xi = \left\{\theta \in \mathbb{R}^p,\ \rho \in \mathbb{R}^{q\times m} : \left|y^{(r)}(t_k) - \varphi_F(x(t_k))^T\theta - \varphi_B(x(t_k))^T\rho u(t_k)\right| \leq D\right\}, \tag{5.11}$$

which can easily be constructed by solving a linear program for the upper and lower bound on each element of θ and ρ. The uncertainty set Θ at solve iteration j is found by performing the set following intersection

$$\Theta^j = \Xi \cap \Theta^{j-1}. \tag{5.12}$$

This allows the establishment of the following lemma.

Lemma 1. *Using the set membership identification procedure from (5.11) and (5.12), the parameter set monotonically decreases since*

$$\Theta^j \subseteq \Theta^{j-1},\ \forall j > 0 \tag{5.13}$$

Proof. Follows from (5.12). Let $\theta \in \Theta^j \implies \theta \in \Xi \cap \Theta^{j-1}$. For this implication to hold, $\theta \in \Theta^{j-1}$. Hence $\Theta^j \subseteq \Theta^{j-1}$. □

Before presenting the general algorithm, consider the following simple example of SMID.

Example 2. Let $\dot{y} = ay + u + d$ where all variables are scalars, $|d| \leq D$, and a is an unknown parameter that belongs to the known set $\Theta = \{a : \underline{a} \leq a \leq \bar{a}\}$. A new parameter set Θ^j can then be constructed by first solving two linear programs to obtain new bounds on a

$$\begin{aligned} s_1 = \ & \underset{a}{\text{argmin}}\ (-a) \\ & \text{s.t. } |\dot{y}(t_k) - ay(t_k) - u(t_k)| \leq D,\ \forall k \end{aligned} \tag{5.14}$$

and

$$\begin{aligned} s_2 = \ & \underset{a}{\text{argmin}}\ a \\ & \text{s.t. } |\dot{y}(t_k) - ay(t_k) - u(t_k)| \leq D,\ \forall k. \end{aligned} \tag{5.15}$$

Since maximizing/minimizing with respect to a is only guaranteed to produce an extrema, the max and min of each solution s_1 and s_2 must be taken

$$\bar{a} = \max\left(s_1, s_2\right), \quad \underline{a} = \min\left(s_1, s_2\right). \tag{5.16}$$

The new set Ξ is given by

$$\Xi = \{a : \underline{a} \leq a \leq \bar{a}\}. \tag{5.17}$$

Finally, the new parameter set Θ^j is found be taking the intersection of Ξ and Θ^{j-1},

$$\bar{a}^j = \max\left(\bar{a}^{j-1}, \bar{a}\right), \quad \underline{a}^j = \min\left(\underline{a}^{j-1}, \underline{a}\right), \tag{5.18}$$

where $\Theta^j = \{a : \underline{a}^j \leq a \leq \bar{a}^j\}$. This procedure is then continued until $\Theta = \{a\}$ (i.e., $\bar{a}^j \rightarrow \underline{a}^j$ for some j) or some pre-defined convergence threshold is reached. □

The procedure presented in Example 2 can be generalized to systems with N unknown parameters, as summarized in Algorithm 1.

5.4 Discussion

Before proceeding to the main result of this chapter, it is instructive to discuss the computational complexity and robustness of SMID. From Algorithm 1, two linear programs must be solved at each time step for every element in the model uncertainty set Θ. The number of uncertain parameters typically will not govern the complexity of Algorithm 1. Rather, the complexity will be dictated by the number of constraints within the linear program, a consequence of many systems having access to high-rate sensor information. Instead of letting the number of constraints grow indefinitely, redundant constraints can be checked for and eliminated by solving another linear program [105]; reducing computational complexity for computationally-constrained systems. Nevertheless, the computational complexity of SMID is low compared to other techniques since linear programming is considered one of the easiest optimization problems to solve with many efficient solvers readily available.

Unlike other adaptation/estimation techniques, SMID *explicitly* accounts for the distur-

Algorithm 1 Set Membership Identification

```
procedure SMID($y^{(r)}(t_k)$, $\varphi_F(x(t_k))$, $u(t_k)$, $\underline{\theta}_i^{j-1}$, $\bar{\theta}_i^{j-1}$, $D$)
    $y^{(r)}_{hist}$.append($y^{(r)}(t_k)$)
    $\varphi_{hist}$.append($\varphi_F(x(t_k))$)
    $u_{hist}$.append($u(t_k)$)
    $L \leftarrow y^{(r)}_{hist}$.length()
    $s_1 = \underset{\theta}{\text{argmin}}\ (-\theta_i)$
        s.t. $\left| y^{(r)}_{hist}(k) - \varphi_{hist}(k)^T\theta - u_{hist}(k) \right| \leq D,\ k = 0 : L$
    $s_2 = \underset{\theta}{\text{argmin}}\ \theta_i$
        s.t. $\left| y^{(r)}_{hist}(k) - \varphi_{hist}(k)^T\theta - u_{hist}(k) \right| \leq D,\ k = 0 : L$
    $sol^+ = \max(s_1, s_2)$
    $sol^- = \min(s_1, s_2)$
    $\bar{\theta}_i^j = \max(sol^+, \bar{\theta}_i^{j-1})$
    $\underline{\theta}_i^j = \min(sol^-, \underline{\theta}_i^{j-1})$
    Return $\underline{\theta}_i^j$, $\bar{\theta}_i^j$
```

bance when constructing a new parameter uncertainty set. This characteristic, in conjunction with the set intersection operation, is what guarantees the monotonic decrease of the parameter error. However, instead of requiring parameter excitation to guarantee convergence, as in adaptive/estimation techniques, the disturbance must now be exciting to achieve convergence (i.e., $|d| = D$ for some time interval $[t,\ t+\tau]$). Hence, the disturbance bound must be tight in order for SMID to obtain the true parameter values. While parameter convergence is of course ideal, even a mild reduction in model uncertainty can drastically improve performance (as will be shown in Chapter 6).

The observant reader will notice that the SMID formulation presented above requires the rate of the state vector. This is not always easily obtainable for real systems that have measurement noise. However, since SMID can be performed outside of the control loop, non-causal filtering techniques, such as smoothing, can be applied to significantly reduce noise. Causal filters can also be used so long as the filter is applied to each term in (5.11), not just the rate vector. The measurement confidence value can then be added to the disturbance bound to ensure correct parameter convergence. More precisely, if the noise/confidence level of the filtered version of the states is denoted as $\mathcal{N}$, then (5.11) can be re-written as

$$\Xi = \left\{\theta \in \mathbb{R}^p,\ \rho \in \mathbb{R}^{q\times m} : \left|\hat{y}^{(r)}(t_k) - \hat{\varphi}_F(x(t_k))^T\theta - \hat{\varphi}_B(x(t_k))^T\rho\hat{u}(t_k)\right| \leq D + \mathcal{N}\right\}, \quad (5.19)$$

where $\hat{\cdot}$ denotes the filtered value. Note that the filtered version of the *regression* vector must be used, as opposed to the filtered state, to ensure the filtering process does not effect the inequality in (5.19). As stated above, the tighter the disturbance and noise/confidence bound the better SMID will perform. The effect of noise on SMID will be explored in Chapter 6.

5.5 Main Result

The main technical result of this chapter is given by the following theorem.

Theorem 3 (SMID Recursive Feasibility). *First assume that an optimal solution x^* exists to the DTMPC optimization given by Problem 5 at $t = 0$ but with the modified boundary layer dynamics*

$$\dot{\Phi} = -\alpha\Phi + \left|\varphi_F(x^*)^T\tilde{\theta}^j\right| + D + \eta, \quad (5.20)$$

where

$$\tilde{\theta}_i^j = \sup_{\theta_i} \Theta^j - \inf_{\theta_i} \Theta^j \quad (5.21)$$

is the parameter bound given by the parameter set Θ^j of the j^{th} iteration of SMID for parameter θ_i. Then DTMPC remains is feasible for $t \geq 0$ and for each SMID iteration $j > 0$.

Proof. Recall from Chapter 3, Theorem 1 established that the boundary layer controller induces a robust invariant tube so $|\tilde{x}| \leq \Omega$ with

$$\dot{\Omega} = A_c\Omega + B_c\Phi \quad (5.22)$$

and A_c, B_c are the state and input matrix in controllable canonical form. Let Φ^j and Ω^j be the boundary layer thickness and tube geometry for the parameter set Θ^j. From Lemma 1,

$$\Theta^j \subseteq \Theta^{j-1} \implies |\tilde{\theta}^j| \leq |\tilde{\theta}^{j-1}|, \tag{5.23}$$

and since $\Phi^j(0) = \Phi^{j-1}(0)$, then $\Phi^j(t) \leq \Phi^{j-1}(t)$ $\forall t > 0$ from (5.20). Using an identical argument for Ω^j and Ω^{j-1}, we can conclude that

$$|\tilde{x}| \leq \Omega^j \leq \Omega^{j-1}, \tag{5.24}$$

and hence the error bound $|\tilde{x}|$ *decreases* as the SMID iteration number j increases. Since an optimal solution existed with the larger error bound, a solution is guaranteed to exist with the small error bound. Hence, DTMPC with SMID is recursively feasible. □

Theorem 3 proves that combining SMID with DTMPC (or any other recursively feasible MPC algorithm) does not impact recursive feasibility. This is a direct result of the uncertainty set monotonically decreasing as the iteration number of SMID increases. This is a very important result since monotonicity, and hence recursive feasibility, cannot be achieved when other adaptive/estimation techniques are used. While these other methods will reduce the *instantaneous* tracking/prediction error, very little can be guaranteed in terms of convergence when disturbances/noise are present. Specifically, the parameter uncertainty set could grow at any point along the prediction horizon, likely leading to constraint violation. The monotonicity of SMID is able to ensure DTMPC remains feasible while improving performance as the uncertainty is reduced.

5.6 Adaptive Dynamic Tube MPC

The formal combination of SMID and DTMPC, known as Adaptive Dynamics Tube MPC (ADTMPC), is given by Problem 8 for a known input matrix.

Problem 8 – Adaptive Dynamic Tube MPC

$$
\begin{aligned}
u^*(t), x^*(t), \alpha^*(t) = \underset{\check{u}(t),\check{v}(t),\check{x}(t),\check{\alpha}(t)}{\text{argmin}} \quad & J = h_f(\check{x}(t_f)) + \int_{t_0}^{t_f} \ell(\check{x}(t), \check{u}(t), \check{\alpha}(t), \check{v}(t))dt \\
\text{subject to} \quad & \dot{\check{x}}(t) = \varphi_F(\check{x}(t))^T\hat{\theta}^j + \varphi_B(\check{x}(t))^T\rho\check{u}(t),\ \dot{\check{\alpha}}(t) = \check{v}(t), \\
& \dot{\Phi}(t) = -\check{\alpha}(t)\Phi(t) + |\varphi_F(\check{x}(t))^T\tilde{\theta}^j| + D + \eta, \\
& \dot{\Omega}(t) = A_c\Omega(t) + B_c\Phi(t), \quad \Omega(t_0) = |\tilde{x}(t_0)|, \\
& \check{x}(t_0) = x_0^*, \quad \check{\alpha}(t_0) = \alpha_0^* \quad \Phi(t_0) = \Phi_0, \\
& \check{x}(t) \in \bar{\mathbb{X}}, \quad \check{u}(t) \in \bar{\mathbb{U}}, \quad \check{\alpha}(t) \in \mathbb{A}, \quad \check{v}(t) \in \mathbb{V}, \\
& \check{x}\,(t_f) \in \bar{\mathbb{X}}_f, \quad \check{\alpha}(t_f) \in \mathbb{A}_f, \\
\\
\text{where} \quad & \underline{\theta}^j,\ \bar{\theta}^j = \text{SMID}(\dot{x}(t_k), \varphi_F(x(t_f)), u(t_k), \underline{\theta}^{j-1}, \bar{\theta}^{j-1}, D), \\
& \hat{\theta}^j = \frac{1}{2}\left(\underline{\theta}^j + \bar{\theta}^j\right),\ \tilde{\theta}^j = \bar{\theta}^j - \underline{\theta}^j.
\end{aligned}
$$

Again note that SMID does not have to be run at the same rate as the optimization or the ancillary controller. It can be done in parallel at its own rate; the model in the optimization and controller are then updated when a new parameter set becomes available. As was proved in Theorem 3, Problem 8 is recursively feasible.

5.7 Summary

This chapter presented the Adaptive Dynamic Tube MPC (ADTMPC) algorithm that addresses a key limitation of existing nonlinear tube MPC algorithms: the inability to update the prediction model while still maintaining recursive feasibility. It was shown that set membership identification (SMID) guarantees the uncertainty in model parameters monotonically decreases even when disturbances/measurement noise are present (assuming a bound is known). Guaranteed monotonicity is unobtainable with other adaptive/estimation schemes which prevents establishing recursive feasibility. By combining DTMPC and SMID, ADMPT can: 1) simultaneously optimize a trajectory and tube geometry; 2) leverage state-dependent uncertainty to construct less conservative trajectories; and 3) further reduce conservativeness by using SMID to update model parameters. All three features are unique to ADTMPC and are not obtainable by current nonlinear tube MPC algorithms.

Chapter 6

Adaptive Dynamic Tube MPC: Applications and Analysis

6.1 Overview

This chapter presents simulation results that show how Adaptive Dynamic Tube MPC (ADTMPC) can improve upon the performance of DTMPC by robustly reducing model parameter uncertainty online. ADTMPC is applied to two domains, both with nonlinear dynamics and state/performance constraints. The first domain is a double integrator system with nonlinear drag performing obstacle avoidance. Results will show that ADTMPC is able to achieve a higher speed around obstacles, a direct consequence of obtaining accurate parameter estimates even when disturbances/sensor noise are present. The second domain is a high-performance aircraft whose lateral dynamics are susceptible to oscillations; a phenomenon known as wing rock. Results will show that ADTMPC obtains significant performance enhancement, as measured by lower bandwidth control, by obtaining accurate estimates of multiple aerodynamic parameters.

6.2 Collision Avoidance

6.2.1 Model

As in Section 4.2, a double integrator model with nonlinear drag is used to represent a mechanical system performing collision avoidance. The dynamics, controller, robust gain, and boundary layer dynamics were presented in Section 4.2 but are presented again for completeness. With $r = [r_x \; r_y \; r_z]^T \in \mathbb{R}^3$ denoting the inertial position of the system, the dynamics are

$$\ddot{r} = -C_d \left\|\dot{r}\right\| \dot{r} + g + u + d, \tag{4.1}$$

where $g \in \mathbb{R}^3$ is the gravity vector, $C_d \in \Theta = \left\{C_d \in \mathbb{R} : \underline{C}_d \leq C_d \leq \bar{C}_d\right\}$ is an unknown drag coefficient, and d is a disturbance such that $|d| \leq D$. The control law is

$$u = \ddot{r}^* + \hat{C}_d \left\|\dot{r}\right\| \dot{r} - \lambda\dot{\tilde{r}} - K\text{sat}\left({}^{s}/_{\Phi}\right), \tag{4.2}$$

where $\hat{C}_d := \frac{1}{2}(\bar{C}_d + \underline{C}_d)$ is the best estimate of the drag coefficient, $s = \dot{\tilde{r}} + \lambda\tilde{r}$ is the sliding variable, and

$$K = \tilde{C}_d \left[\left\|\dot{r}\right\| |\dot{r}| - \left\|\dot{r}^*\right\| |\dot{r}^*|\right] + \alpha\Phi, \tag{4.3}$$

$$\dot{\Phi} = -\alpha\Phi + \tilde{C}_d \left\|\dot{r}^*\right\| |\dot{r}^*| + D + \eta, \tag{4.4}$$

are the robust gain and boundary layer dynamics with $\tilde{C}_d := \bar{C}_d - \underline{C}_d$ being the parameter error, respectively.

6.2.2 Set Membership Identification

The goal of set membership identification (SMID) is to reduce parametric uncertainty by eliminating parameter values that are not consistent with the state observations. For the model in (4.1), this corresponds to eliminating uncertainty in the drag coefficient C_d. As discussed in Chapter 5, the primary benefit of eliminating model uncertainty is to reduce conservativeness in the trajectory optimization. For this system, speed around obstacles is a

Algorithm 2 Drag Coefficient Set Membership Identification

```
procedure DRAG_SMID($\dot{r}(t_k)$, $\ddot{r}_x(t_k)$, $u_x(t_k)$, $\underline{C}_d^{j-1}$, $\bar{C}_d^{j-1}$, $D$, $\mathcal{N}$)
    $\dot{r}_{hist}$.append($\dot{r}(t_k)$)
    $\ddot{r}_{x,hist}$.append($\ddot{r}_x(t_k)$)
    $u_{x,hist}$.append($u_x(t_k)$)
    $L \leftarrow x_{hist}$.length()
    $s_1 = \underset{C_d}{\text{argmin}}\ (-C_d)$
        s.t. $|\ddot{r}_{x,hist}(k) - C_d\|\dot{r}_{hist}(k)\|\dot{r}_{x,hist}(k) - u_{x,hist}(k)| \leq D + \mathcal{N},\ \ k = 0 : L$
    $s_2 = \underset{C_d}{\text{argmin}}\ C_d$
        s.t. $|\ddot{r}_{x,hist}(k) - C_d\|\dot{r}_{hist}(k)\|\dot{r}_{x,hist}(k) - u_{x,hist}(k)| \leq D + \mathcal{N},\ \ k = 0 : L$
    $sol^+ = \max(s_1, s_2)$
    $sol^- = \min(s_1, s_2)$
    $\bar{C}_d^j = \max(sol^+, \bar{C}_d^{j-1})$
    $\underline{C}_d^j = \min(sol^-, \underline{C}_d^{j-1})$
    Return $\underline{C}_d^j$, $\bar{C}_d^j$
```

direct indicator for conservativeness. Since $C_d \in \mathbb{R}$, then any acceleration component can be used in SMID. Algorithm 2 summarizes the SMID algorithm using the $\ddot{r}_x$ acceleration.

6.2.3 Collision Avoidance ADTMPC

Again let H, p_c, and r_o denote the shape, location, and size of an obstacle. The minimum time and control effort ADTMPC optimization with collision avoidance for system (4.1) is presented in Problem 9.

Problem 9 – Collision Avoidance ADTMPC

$$
\begin{aligned}
u^*(t),\ r^*(t) = \underset{\check{u}(t),\check{r}(t),t_f}{\operatorname{argmin}} \quad & J = \int_{t_0}^{t_f} \check{u}(t)^T Q \check{u}(t) dt \\
\text{subject to} \quad & \ddot{\check{r}}(t) = -\hat{C}_d^j \left\| \dot{\check{r}}(t) \right\| \dot{\check{r}}(t) + g + \check{u}(t), \\
& \dot{\Phi}(t) = -\alpha \Phi(t) + \tilde{C}_d^j \left\| \dot{\check{r}}(t) \right\| \left| \dot{\check{r}}(t) \right| + D + \eta, \\
& \dot{\Omega}(t) = A_c \Omega(t) + B_c \Phi(t), \quad \Omega(t_0) = |\tilde{r}(t_0)| \\
& \check{r}\,(t_0) = r_0^*, \quad \Phi\,(t_0) = \Phi_0, \quad \check{r}\,(t_f) = r_f^*, \\
& \|H_i \check{r}(t) - p_{c,i}\| \geq r_{o,i} + \|H_i \tilde{r}(t)\|, \ i = 1, \cdots, N_o, \\
& |\dot{\check{r}}(t)| \leq \dot{r}_m - |\dot{\tilde{r}}(t)|, \quad \|\check{u}(t)\| \leq u_m - \bar{u}_{fb} \\
\\
\text{where} \quad & \underline{C}_d^j,\ \bar{C}_d^j = \text{DRAG\_SMID}(\dot{r}(t_k), \ddot{r}_x(t_k), u_x(t_k), \underline{C}_d^{j-1}, \bar{C}_d^{j-1}, D, \mathcal{N}), \\
& \hat{C}_d^j = \frac{1}{2}(\underline{C}_d^j + \bar{C}_d^j),\ \tilde{C}_d^j = \bar{C}_d^j - \underline{C}_d^j,
\end{aligned}
$$

$\| \cdot \|$ is the L^2 norm, $| \cdot |$ is the element-wise absolute value, $\dot{r}_m$ is the peak desired speed, and N_o is the number of obstacles.

6.2.4 Simulation Environment

ADTMPC was tested in simulation to demonstrate its ability to construct less conservative trajectories as the prediction model is update with SMID. As in Chapter 4, obstacles were placed non-uniformly so the speed along the trajectory had to be modulated based on the level of parametric uncertainty. Tests were conducted with both perfect and noisy acceleration information provided to Algorithm 2. The acceleration and regression vector measurements were not filtered to test SMID's performance with real-time noisy data; noise was sampled from a uniform distribution with bound $\mathcal{N}$. Since the obstacle locations were known *a priori*, a new trajectory was generated only when SMID produced an updated estimate of the parameter bounds. The resulting trajectories were tracked by the controller given by (4.2) that had to compensate for both the model error and a sinusoidal disturbance, shown in Fig. 6-1. Sequential convex programming was again employed where the optimization was

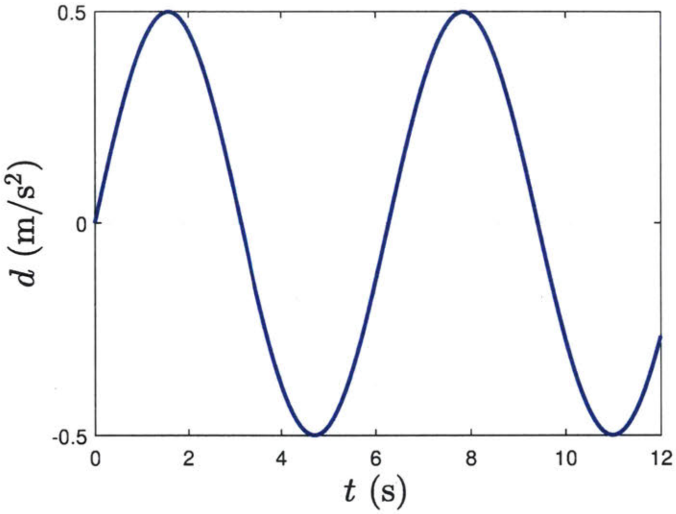


Figure 6-1: Sinusoidal disturbance applied to the double integrator, nonlinear drag system performing obstacle avoidance.

Table 6.1: ADTMPC Collision Avoidance Simulation Parameters.

Param.	**Value**	**Param.**	**Value**
r_0	$[0\ 0\ 1]^T$ m	r_f	$[0\ 25\ 1]^T$ m
$\dot{r}_0$	$[0\ 1\ 0]^T$ m/s	$\dot{r}_f$	$[0\ 1\ 0]^T$m/s
λ	$[2\ 2\ 2]^T$ rad/s	g	$[0\ 0\ \text{-}9.8]^T$m/s^2
u_m	5 m/s^2	$\dot{r}_m$	2 m/s
α	2 rad/s	D	0.5 m/s^2
$\underline{C}_d^0$	0 kg/m	$\bar{C}_d^0$	0.5 kg/m
C_d^*	0.15 kg/m	η	0.1
N_o	5 rad/s^2	Q	$2I_3$
$\mathcal{N}$	0, 0.25, 0.5, 1, 2.5 m/s^2	-	-

initialized with a naïve straight-line solution and solved using YALMIP [99] and MOSEK [100] in MATLAB. In order to minimize the final time, the optimization was solved for a relaxed final time t_f which was then iteratively reduced until a solution no longer existed. The simulation parameters are summarized in Table 6.1.

6.2.5 Results and Analysis

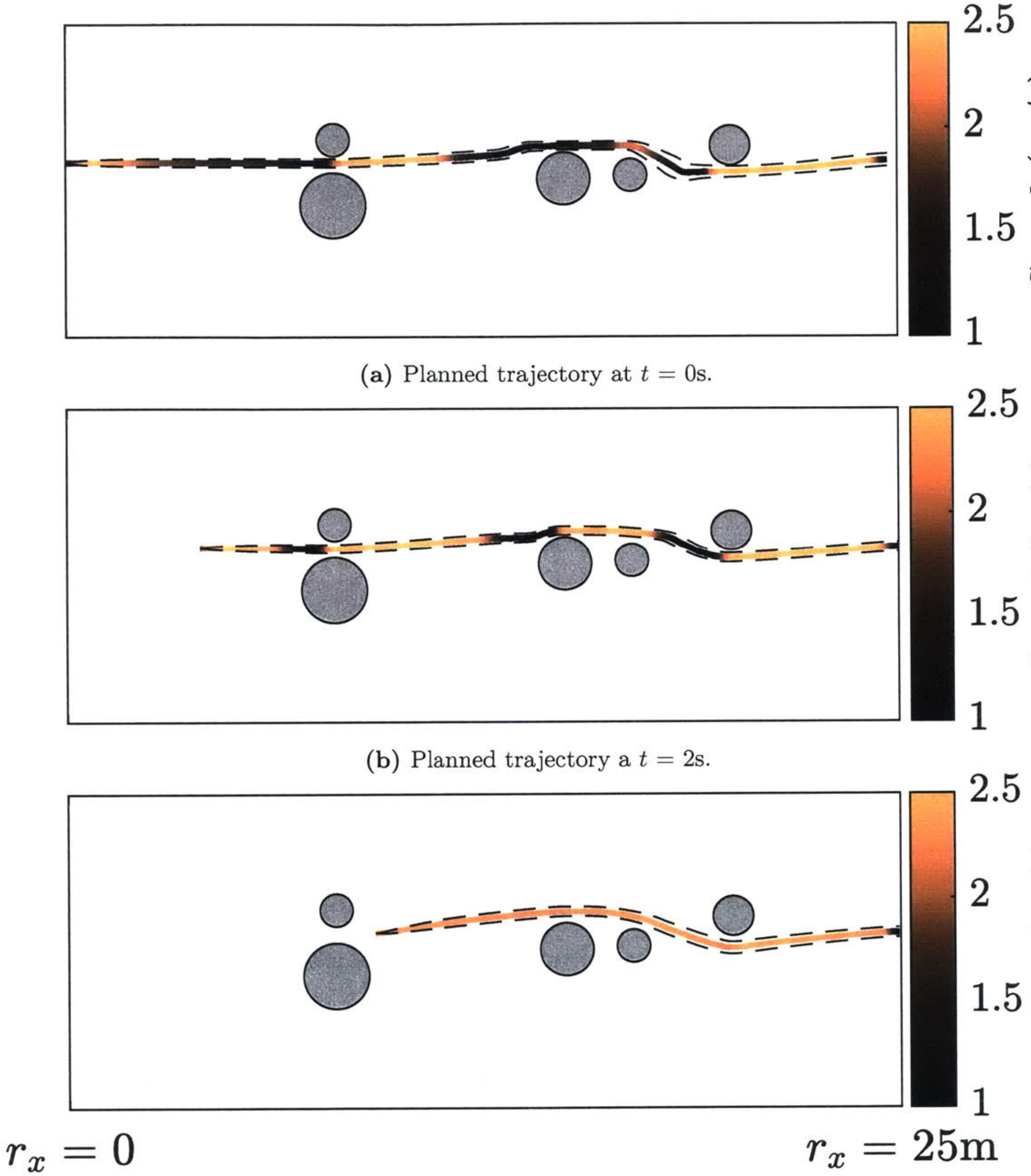
ADTMPC No Noise

The goal of ADTMPC is to reduce model uncertainty through SMID and produce less conservative trajectories. Fig. 6-2a shows the initial planned trajectory with the original drag coefficient uncertainty bounds. The speed along the trajectory can be visualized by the color map, where darker/lighter color corresponds slower/faster speed. Since the model uncertainty scales with speed (as seen in (4.4)), the speed along the trajectory is lower in the vicinity of obstacles. As the system progresses along the desired trajectory, ADTMPC begins to obtain state measurements that are used to update the parameter bounds. Fig. 6-2b shows a new planned trajectory when a better estimate of the parameter bounds is available, which is less conservative around the second set of obstacles as seen by the faster speed. Fig. 6-2c shows the resulting trajectory with the final parameter bounds from SMID. Since the model uncertainty is essentially eliminated, the optimizer is free to generate a trajectory at the maximum allowable speed, as desired.

The trajectories shown in Fig. 6-2 clearly demonstrate the benefits of reducing model uncertainty and the effectiveness of ADTMPC. Fig. 6-3 shows the evolution of the parameter bounds as the system progresses along the trajectories generated by ADTMPC (with no noise in the acceleration measurement). The upper (blue) and lower (red) bounds converge to the true value (black) at approximately $t = 5s$, which corresponds to the disturbance profile reaching its lower bound (from Fig. 6-1). Hence, parameter convergence is contingent on the excitation of the disturbance. While convergence is of course ideal, Fig. 6-3 does show that the parameter bounds can be significantly reduced even before the disturbance reaches its bounds. The ability to still converge to the true parameter value even when an unknown disturbance is present is a remarkable characteristic that other adaptive control or estimation schemes do not possess; further substantiating the use of SMID within ADTMPC.

ADTMPC With Noise

It is critical to test the performance of any adaptation/estimation scheme with noise-corrupted measurements to ensure adequate performance can still be achieved and stability maintained.



(a) Planned trajectory at t = 0s.

(b) Planned trajectory a t = 2s.

(c) Planned trajectory at t = 5s.

Figure 6-2: Receding horizon implementation of ADTMPC with no measurement noise. Obstacles (dark-grey) are known and the speed along the trajectory is given by the color map where darker/lighter corresponds to low/high speed. (a): First planned trajectory has low/high speed when the trajectory is closer/farther from obstacles. (b): Second planned trajectory is less conservative (higher speed) with SMID updating model uncertainty bounds. (c): Final trajectory utilizes maximum allowable speed indicating SMID eliminated all model uncertainty.

Uniformly sampled noise with bound $\mathcal{N}$ was added to the acceleration measurement to test the performance and robustness of ADTMPC. Fig. 6-4a shows the true (blue) and corrupted (red)

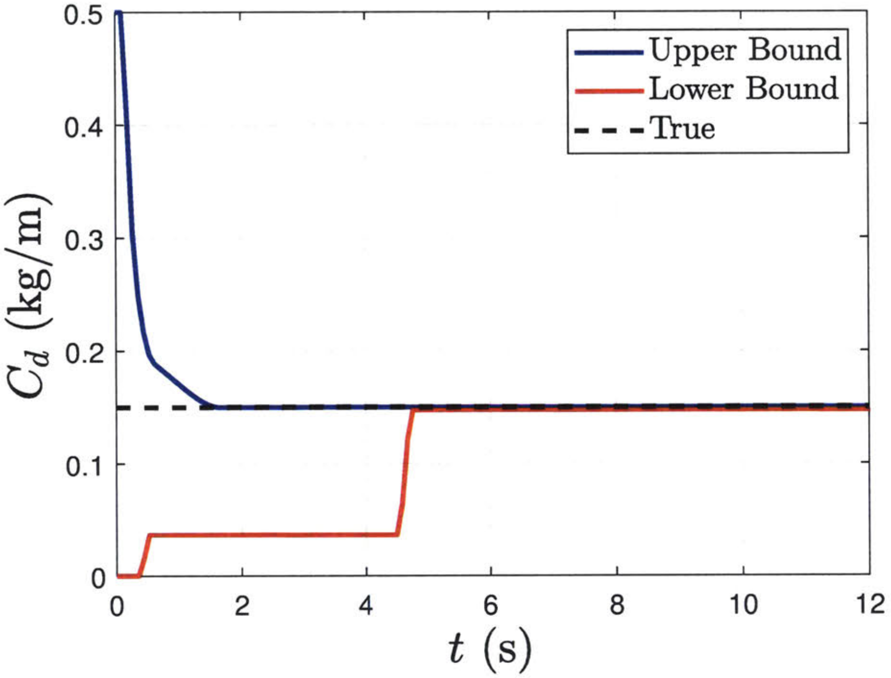
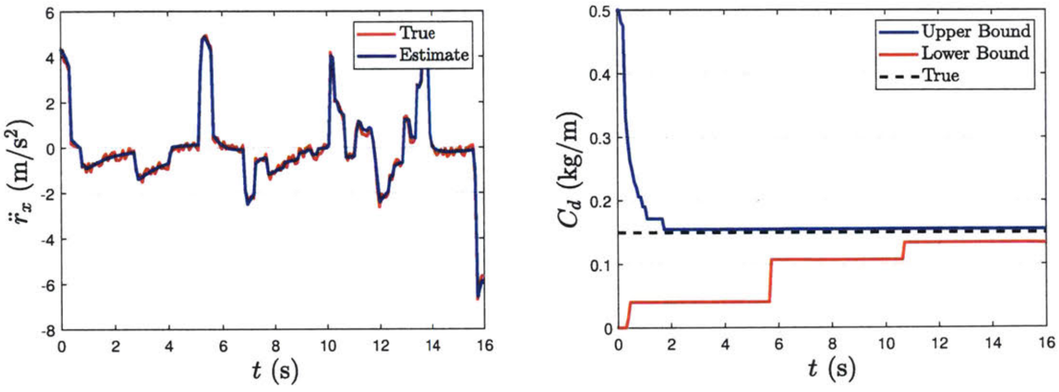


Figure 6-3: Drag coefficient estimate with set membership identification. The upper (red) and lower (blue) bound both converge to the true value (black). Convergence occurs after the disturbance excites its upper and lower bound.

acceleration measurements used by ADTMPC as the system tracks the initial trajectory in Fig. 6-2a for $\mathcal{N} = D/2$. Fig. 6-4 shows the upper (blue) and lower (red) parameter bounds and the true (black) parameter value if the initial trajectory is tracked for its entirety. Although the bounds do not converge to the true parameter value, they are reduced considerably: the estimated value is within 3.63% of the true value with an uncertainty of ± 0.01 kg/m (a 96% reduction). Fig. 6-5, Fig. 6-6, and Fig. 6-7 show the acceleration measurements and parameter bounds for noise levels[1] of $\mathcal{N} = D,\ 2D,\ 5D$, respectively. It is clear that as the noise level increases the parameter bounds loosen, indicating more uncertainty in the parameter estimate; an effect likely caused by the disturbance signal being dominated by the noise in the acceleration measurements. Nonetheless, the percent reduction in the uncertainty is still substantial: 82.4% reduction for $\mathcal{N} = 5D$, the highest noise level tested. Further, the error in the parameter estimate is still at most 3.63%. Table 6.2 summarizes the estimated parameter value, estimation error, uncertainty level, and percent uncertainty reduction for

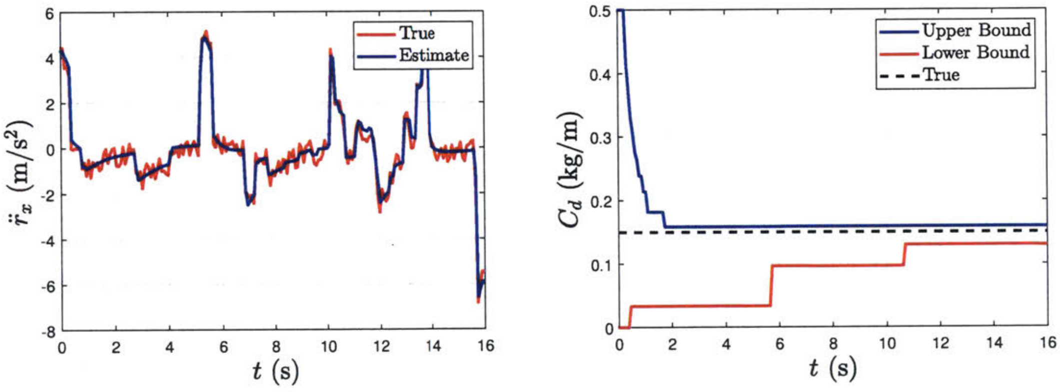
[1] The same random seed was used for each $\mathcal{N}$ tested.



(a) Acceleration $\ddot{r}_x$.

(b) Drag coefficient C_d.

Figure 6-4: Acceleration and drag coefficient estimate for $\mathcal{N} = D/2$. (a): True (blue) and noise-corrupted (red) acceleration. (b): The drag coefficient upper (blue) and lower (red) bounds still nearly converge to the true value (black) with a 96.0% reduction in uncertainty.



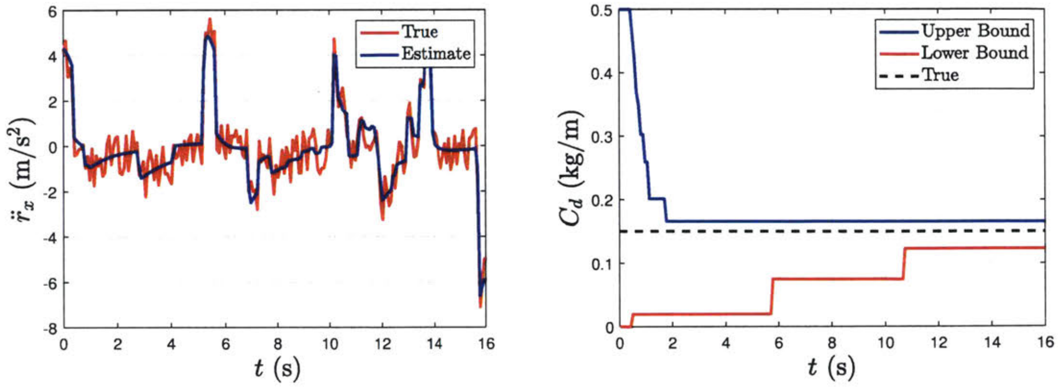
(a) Acceleration $\ddot{r}_x$.

(b) Drag coefficient C_d.

Figure 6-5: Acceleration and drag coefficient estimate for $\mathcal{N} = D$. (a): True (blue) and noise-corrupted (red) acceleration. (b): The drag coefficient upper (blue) and lower (red) bounds are not as tight as the low-noise cases but still see a significant 94.4% reduction in uncertainty.

each noise level. These results again demonstrate the ability of ADTMPC (via SMID) to obtain accurate parameter estimates even when noise/disturbances are present.

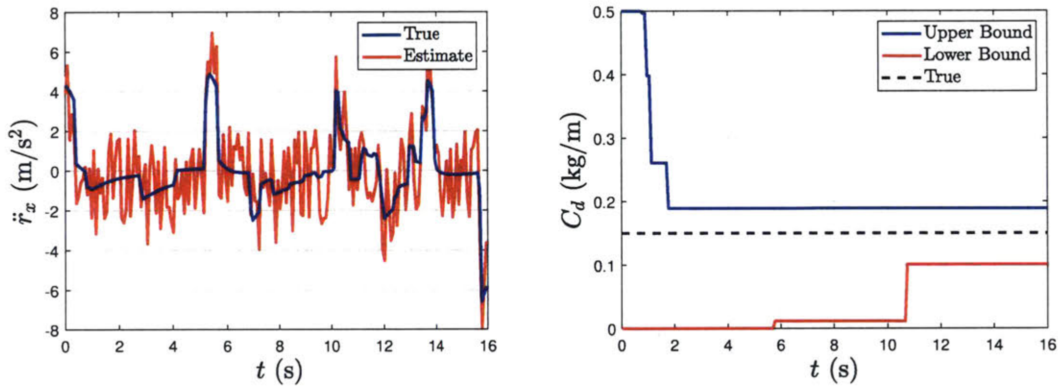
Fig. 6-8 shows the trajectories generated by ADTMPC at different time instances as the model uncertainty bounds are updated with noisy acceleration measurements with $\mathcal{N} = D$. The initial trajectory, shown in Fig. 6-8a is of course unchanged: slower speeds around obstacles to reduce model uncertainty. Fig. 6-8b shows the resulting trajectory with the new uncertainty bounds at $t = 2s$. While there are still instances where the speed is low, the



(a) Acceleration $\ddot{r}_x$.

(b) Drag coefficient C_d.

Figure 6-6: Acceleration and drag coefficient estimate for $\mathcal{N} = 2D$. (a): True (blue) and noise-corrupted (red) acceleration with. (b): The drag coefficient upper (blue) and lower (red) bounds are not as tight as the low-noise cases but still see a significant 91.2% reduction in uncertainty.



(a) Acceleration $\ddot{r}_x$.

(b) Drag coefficient C_d.

Figure 6-7: Acceleration and drag coefficient estimate for $\mathcal{N} = 5D$. (a): True (blue) and noise-corrupted (red) acceleration. (b): The drag coefficient upper (blue) and lower (red) bounds are not as tight as the low-noise cases but still see a significant 82.4% reduction in uncertainty.

Table 6.2: Estimated Drag Coefficient and Uncertainty Level for Different Noise Levels.

$\mathcal{N}$ (m/s^2)	$\hat{C}_d$ (kg/m)	**% Error**	**Unc. Level** (kg/m)	**% Reduc.**
0	0.15	0.0	±0	100
$D/2$	0.1445	3.63	±0.01	96.0
D	0.1445	3.63	±0.014	94.4
$2D$	0.1447	3.51	±0.022	91.2
$5D$	0.1454	3.1	±0.044	82.4

reduced uncertainty bounds allow the optimizer to generate a faster speed around the second set of obstacles, indicated by the lighter color. Fig. 6-8c shows that the model uncertainty bounds at $t = 6s$ are reduced enough to result in a speed near the maximum allowable for the remainder of the trajectory. Hence, significantly improved performance, as measured by speed when in close proximity to obstacles, is achieved, even though the parameter bounds do not converge to the true parameter value.

6.3 Wing Rock

6.3.1 Model

The nonlinear latitude dynamics of a high-performance aircraft that exhibit high-frequency oscillations, a phenomenon known as wing rock, were presented in Section 4.3. For completeness, the dynamics, controller, robust gain, and boundary layer dynamics are presented again here. With ϕ and $\dot{\phi}$ denoting the roll angle and roll rate, the dynamics are

$$\ddot{\phi} = a_1\phi + a_2\dot{\phi} + a_3|\phi|\dot{\phi} + a_4|\dot{\phi}|\dot{\phi} + u + d, \tag{4.5}$$

where d an unknown but bounded (i.e., $|d| \leq D$) disturbance and $a_i \in \Theta = \{a_i \ : \ \underline{a}_i \leq a_i \leq \bar{a}_i\}$ for $i = 1, \dots, 4$ are unknown aerodynamic coefficients. The boundary layer controller is given by

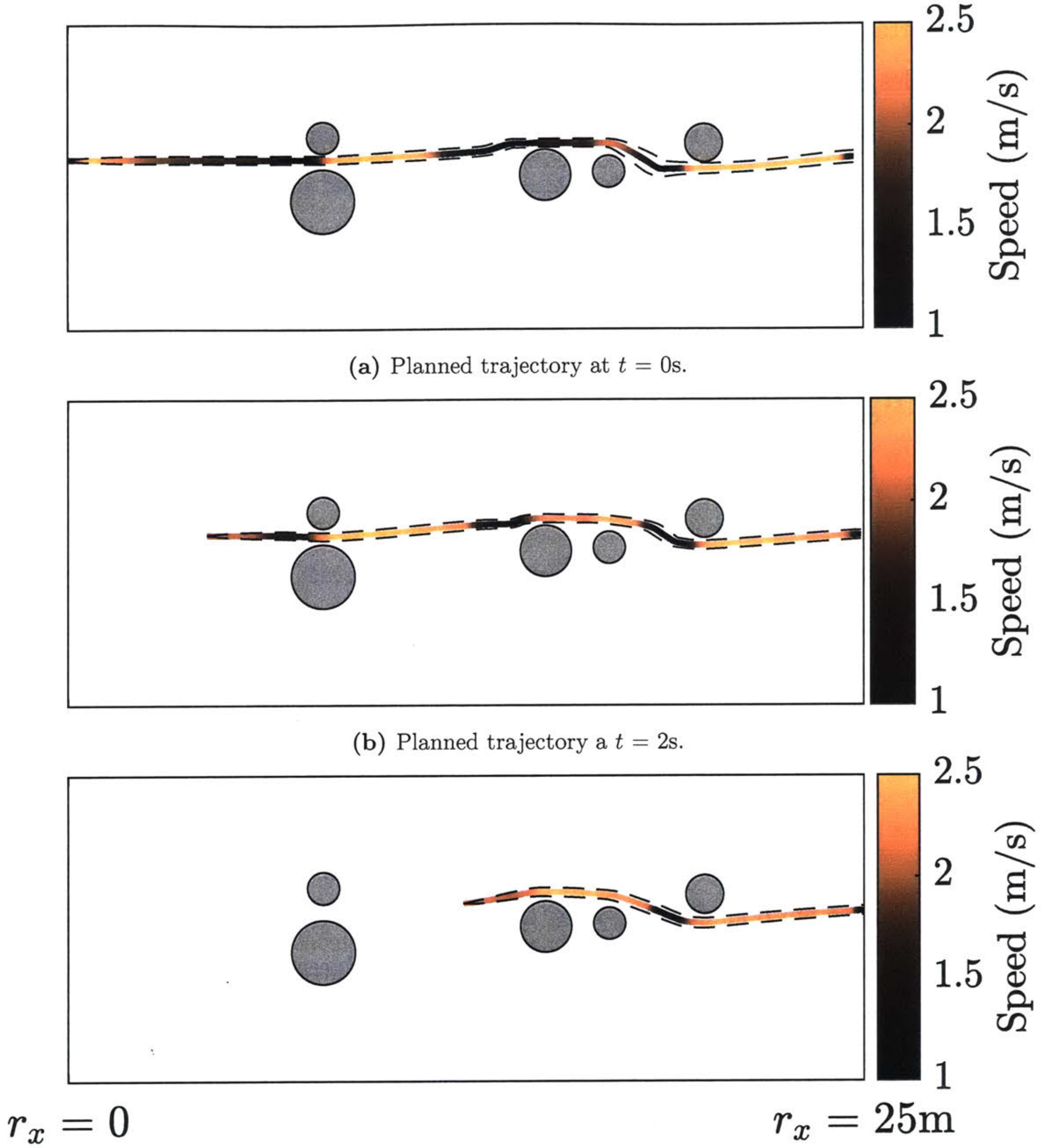
$$u = \ddot{\phi}^* - \hat{a}_1\phi - \hat{a}_2\dot{\phi} - \hat{a}_3|\phi|\dot{\phi} - \hat{a}_4|\dot{\phi}|\dot{\phi} - \lambda\dot{\tilde{\phi}} - K\mathrm{sat}\left({}^{s}/_{\Phi}\right) \tag{4.6}$$

where $\hat{a}_i := \frac{1}{2}\left(\bar{a}_i + \underline{a}_i\right)$ is the best estimate of parameter a_i, $s = \dot{\tilde{\phi}} + \lambda\tilde{\phi}$ is the sliding variable, and

$$K = \left|\tilde{a}_1\phi^* + \tilde{a}_2\dot{\phi}^* + \tilde{a}_3|\phi^*|\dot{\phi}^* + \tilde{a}_4|\dot{\phi}^*|\dot{\phi}^*\right| + D + \eta - \Phi, \tag{4.7}$$

$$\dot{\Phi} = -\alpha\Phi + \left|\tilde{a}_1\phi^* + \tilde{a}_2\dot{\phi}^* + \tilde{a}_3|\phi^*|\dot{\phi}^* + \tilde{a}_4|\dot{\phi}^*|\dot{\phi}^*\right| + D + \eta, \tag{4.8}$$

are the robust gain and boundary layer dynamics with $\tilde{a}_i := \bar{a}_i - \underline{a}_i$ being the parameter error.



(a) Planned trajectory at t = 0s.

(b) Planned trajectory a t = 2s.

(c) Planned trajectory at t = 6s.

Figure 6-8: Receding horizon implementation of ADTMPC with acceleration measurement noise. Obstacles (dark-grey) are known and the speed along the trajectory is given by the color map where darker/lighter corresponds to low/high speed. (a): First planned trajectory has low/high speed when the trajectory is closer/farther from obstacles. (b): Second planned trajectory is less conservative (higher speed) with SMID updating model uncertainty bounds. (c): Final trajectory where the maximum allowable speed it utilized almost for its entirety, indicating some model uncertainty remains but is far less than in the original trajectory design.

Algorithm 3 Wing Rock Coefficient Set Membership Identification

```
procedure WR_SMID($\phi(t_k)$, $\dot{\phi}(t_k)$, $\ddot{\phi}(t_k)$, $u(t_k)$, $\underline{a}_i^{j-1}$, $\bar{a}_i^{j-1}$, $D$, $\mathcal{N}$)
    $\phi_{hist}$.append($\phi(t_k)$)
    $\dot{\phi}_{hist}$.append($\dot{\phi}(t_k)$)
    $\ddot{\phi}_{hist}$.append($\ddot{\phi}(t_k)$)
    $u_{hist}$.append($u(t_k)$)
    $L \leftarrow \phi_{hist}$.length()
    $s_1 = \underset{a}{\text{argmin}}\ (-a_i)$
          s.t. $\left|\ddot{\phi}_{hist}(k) - \sum_{\ell=1}^{4} a_\ell \varphi_\ell(\phi_{hist}(k), \dot{\phi}_{hist}(k)) - u_{hist}(k)\right| \leq D + \mathcal{N},\ k = 0 : L$
    $s_2 = \underset{a}{\text{argmin}}\ a_i$
          s.t. $\left|\ddot{\phi}_{hist}(k) - \sum_{\ell=1}^{4} a_\ell \varphi_\ell(\phi_{hist}(k), \dot{\phi}_{hist}(k)) - u_{hist}(k)\right| \leq D + \mathcal{N},\ k = 0 : L$
    $sol^+ = \max(s_1, s_2)$
    $sol^- = \min(s_1, s_2)$
    $\bar{a}_i^j = \max(sol^+, \bar{a}_i^{j-1})$
    $\underline{a}_i^j = \min(sol^-, \underline{a}_i^{j-1})$
    Return $\underline{a}_i^j$, $\bar{a}_i^j$
```

6.3.2 Set Membership Identification

The wing rock phenomenon has received considerable attention from the adaptive and nonlinear control communities because of the high degree of uncertainty in the aerodynamic coefficients and the nonlinearities in the dynamics. While numerous adaptive control techniques have been applied to (4.5) [103, 104, 106], few have successfully identified the parameters when unmodeled disturbances are present. Further, these adaptive techniques are unable to guarantee monotonicity in the parameter error. ADTMPC via SMID, on the other hand, as shown in the previous section, can successfully identify parameters even when disturbances are present. Further, wing rock is a good domain to apply ADTMPC because of the number of unknown parameters and the nonlinearities in the dynamics. Algorithm 3 summarizes the SMID algorithm used to identify the wing rock aerodynamic coefficients.

6.3.3 Wing Rock ADTMPC

The minimum control bandwidth and tracking error threshold Adaptive Dynamic Tube MPC formulation for the wing rock domain is given by Problem 10.

Problem 10 – Wing Rock ADTMPC

$$
\begin{aligned}
\alpha^*(t) = \quad & \underset{\check{v}(t),\check{\alpha}(t)}{\text{argmin}} \quad J = \int_{t_0}^{t_f} \tilde{\alpha}(t)^2 dt \\
& \text{subject to} \quad \dot{\Phi}(t) = -\check{\alpha}(t)\Phi(t) + F(\phi^*(t), \dot{\phi}^*(t)) + D + \eta, \\
& \qquad \dot{\check{\alpha}}(t) = \check{v}(t), \quad \dot{\Omega}(t) = -\lambda\Omega(t) + \Phi(t), \\
& \qquad \Phi\left(t_0\right) = \Phi_0, \quad \check{\alpha}\left(t_0\right) = \alpha_0, \quad \Omega(t_0) = |\tilde{\phi}(t_0)|, \\
& \qquad \Omega(t) \leq \tilde{\phi}_m \quad |\check{v}(t)| \leq v_m, \quad 0 < \underline{\alpha} \leq \check{\alpha}(t) \leq \bar{\alpha},
\end{aligned}
$$

$$
\begin{aligned}
\text{where} \quad & F(\phi^*(t), \dot{\phi}^*(t)) = \left|\tilde{a}_1^j\phi^*(t) + \tilde{a}_2^j\dot{\phi}^*(t) + \tilde{a}_3^j|\phi^*(t)|\dot{\phi}^*(t) + \tilde{a}_4^j|\dot{\phi}^*(t)|\dot{\phi}^*(t)\right|, \\
& \tilde{\alpha}(t) = \check{\alpha}(t) - \underline{\alpha},
\end{aligned}
$$

$$
\begin{aligned}
\text{and} \quad & \underline{a}_i^j,\ \bar{a}_i^j = \text{WR\_SMID}\left(\phi(t_k), \phi(\dot{t_k}), \phi(\ddot{t_k}), u(t_k), \underline{a}_i^{j-1}, \bar{a}_i^{j-1}, D, \mathcal{N}\right), \\
& \tilde{a}_i^j = \bar{a}_i^j - \underline{a}_i^j,\ i = 1, \cdots, 4
\end{aligned}
$$

6.3.4 Simulation Environment

Simulations were again conducted in MATLAB using YALMIP [99] and MOSEK [100]. The same aerodynamic coefficients used in Section 4.3 are also used here and can be found in Table 4.3. The simulation parameters are summarized in Table 6.3, with $\mathcal{N} = 0$ being the only addition. SMID (given by Algorithm 3) was performed online as the system tracked the desired roll angle while subject to a sinusoidal disturbance, shown in Fig. 6-9.

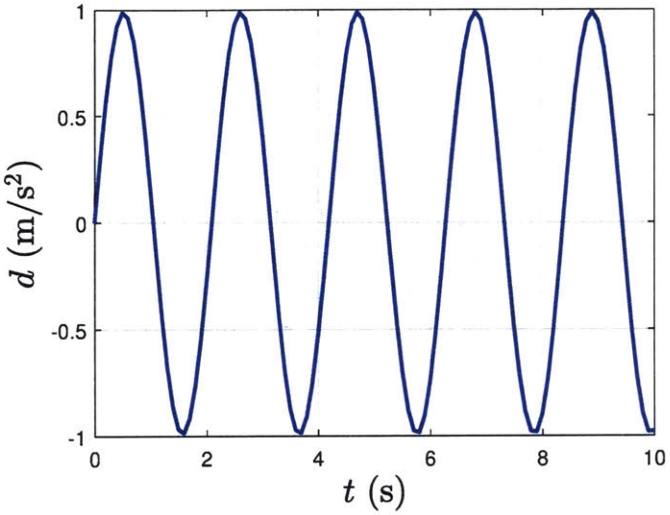


Figure 6-9: Sinusoidal disturbance applied to the nonlinear longitudinal dynamics of a high-performance aircraft that experiences wing rock.

Table 4.3: Wing Rock Aerodynamic Coefficients.

$\boldsymbol{a_1}$ (s^{-2})	$\boldsymbol{a_2}$ (s^{-1})	$\boldsymbol{a_3}$ (s^{-1})	$\boldsymbol{a_4}$ (-)
-32.7 ± 2.5	1.43 ± 0.5	-5.48 ± 1.2	0.1 ± 0.02

Table 6.3: ADTMPC Wing Rock Simulation Parameters.

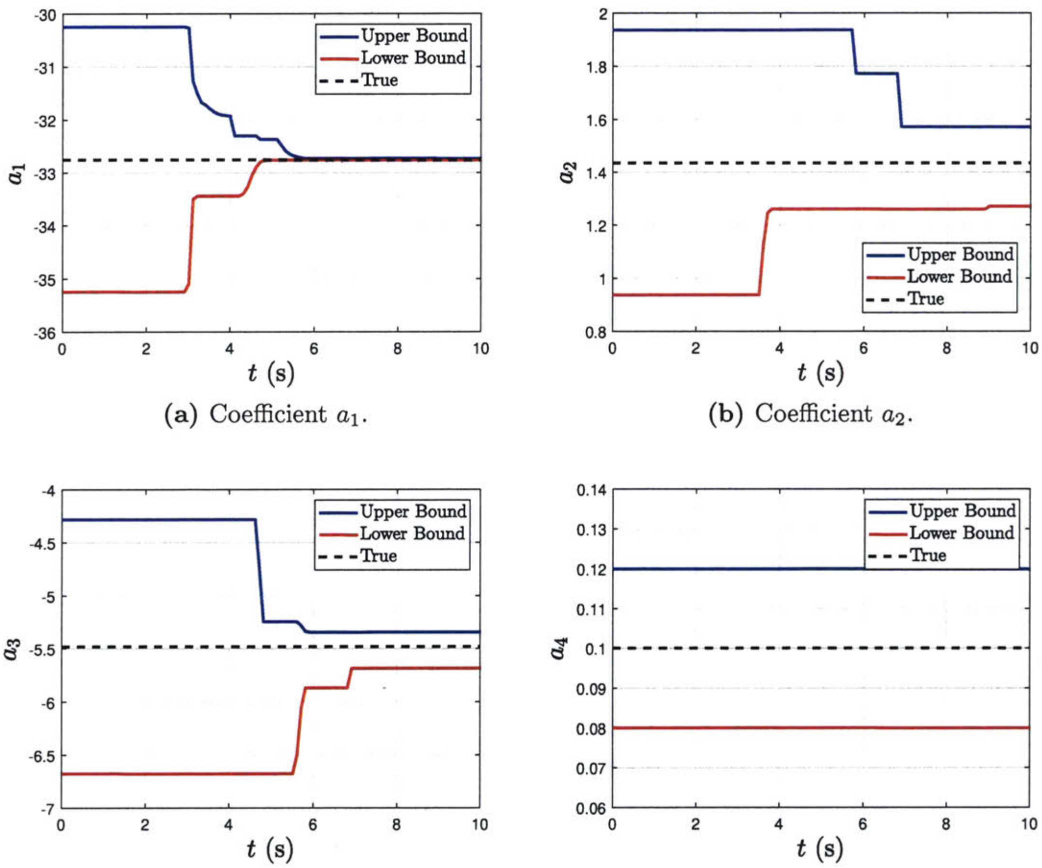
Param.	**Value**	**Param.**	**Value**
$\underline{\alpha}$	4 rad/s	$\bar{\alpha}$	10 rad/s
$\tilde{\phi}_m$	10 deg	v_m	5 rad/s^2
λ	2 rad/s	D	57 deg/s^2
$\Phi(t_0)$	15 deg/s	$\alpha(t_0)$	4 rad/s
$\Omega(0)$	0 deg	η	5.7 deg/s^2
$\mathcal{N}$	0 deg/s^2	-	-

6.3.5 Results and Analysis

ADTMPC is used to track the desired roll angle and roll rate (previously shown in Fig. 4-6) while estimating the unknown aerodynamic coefficients. Fig. 6-10 shows the upper (blue) and lower (red) bounds for each coefficient a_i, the true value (black) is shown for reference. The first three coefficients (Fig. 6-10a-Fig. 6-10b) see reduction in their uncertainty. However, the fourth coefficient (Fig. 6-10d) sees no reduction. This indicates that the model uncertainty, for this desired trajectory, is primarily dominated by the first three coefficients. Table 6.4

Table 6.4: Wing Rock Parameter Uncertainty Reduction with SMID.

Coefficient	Original Uncertainty	Final Uncertainty	Percent Reduction
a_1	±2.5	±0.0325	98.7
a_2	±0.5	±0.302	39.6
a_3	±1.2	±0.344	71.3
a_4	±0.02	±0.02	0.0



(a) Coefficient a_1.

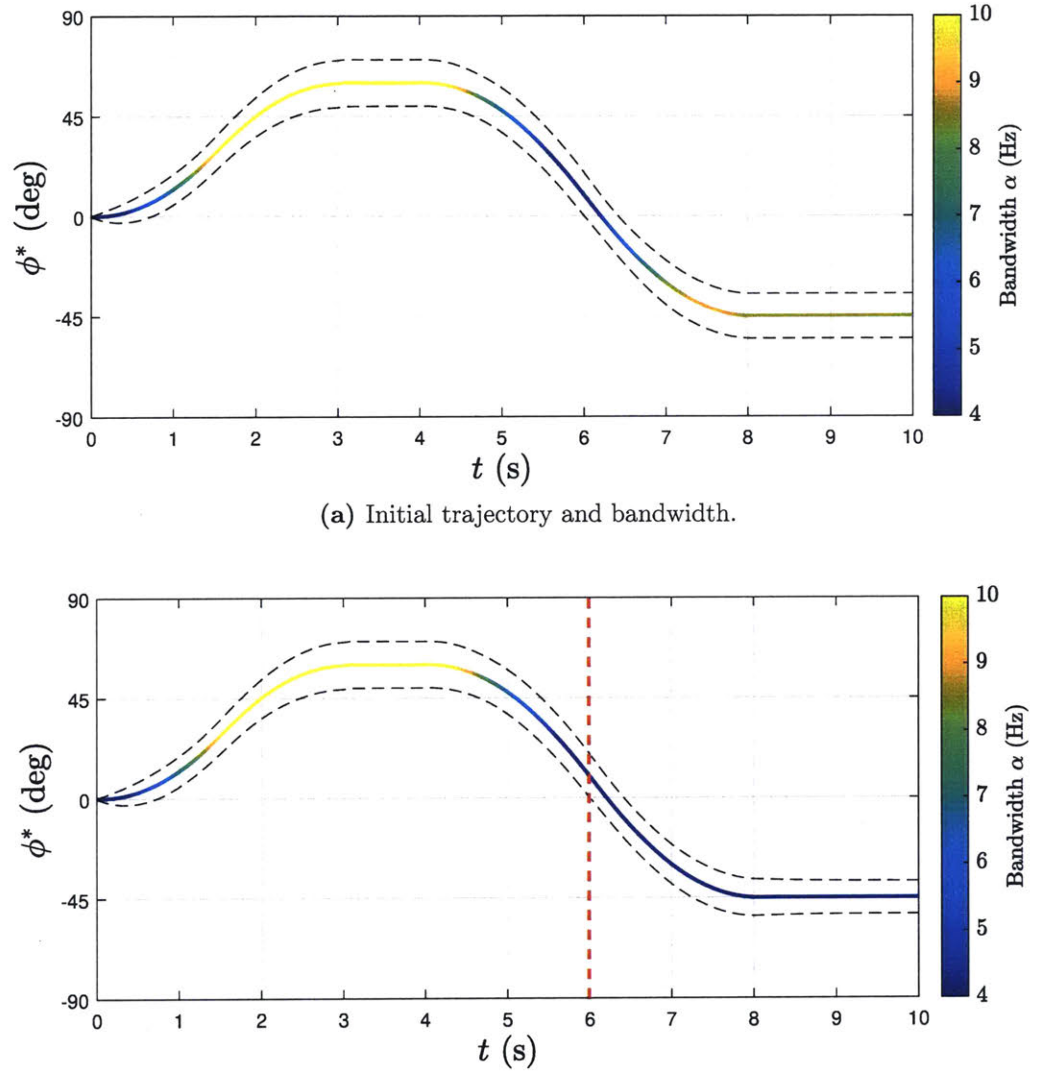
(b) Coefficient a_2.

(c) Coefficient a_3.

(d) Coefficient a_4.

Figure 6-10: Wing rock coefficient upper/lower (blue/red) bounds and true value (black). The first three coefficient (a)-(c) show reduction in their uncertainty as both the upper and lower bounds approach the true value. The fourth coefficient (d) shows no reduction, indicating it has little affect on the model uncertainty.

summarizes the performance of ADTMPC in terms of percent reduction in the uncertainty of each parameter. Every coefficient, besides a_4, sees a reduction of at least 39.6%, further demonstrating ADTMPC's ability to accurately identify multiple parameters even when an unknown disturbance is present.



(a) Initial trajectory and bandwidth.

(b) Modified trajectory and bandwidth at $t = 6$s.

Figure 6-11: Initial trajectory (multi-color) and bandwidth (color map) from DTMPC and modified solution from ADTMPC. (a): DTMPC uses high-bandwidth (light color) to compensate for model uncertainty at large roll angles. (b): ADTMPC uses significantly less bandwidth (darker color) using updated model parameters at $t = 6$s (red line) indicating the reduction of model uncertainty.

Fig. 6-11 shows the level of improvement obtained by ADTMPC over its static model counterpart DTMPC. The initial trajectory (multi-color) and tube geometry (black) are shown in Fig. 6-11a. The temporal variation of the bandwidth is given by the color map. High-bandwidth control (lighter color) is used for large roll angles and low-bandwidth control (darker color) is used for small roll angles; indicating the model uncertainty is predominately from the aerodynamic coefficient on the roll angle term. Fig. 6-11b shows the modified trajectory if ADTMPC were to re-optimize the control bandwidth at $t = 6$s (indicated by the vertical red line). With the updated model parameters, the bandwidth in the later part of the trajectory is reduced by a factor of 2.12 while still satisfying the tracking error constraint. Therefore, ADTMPC is able to achieve significantly better performance, as measured by lower control bandwidth while meeting other objectives, by robustly reducing model uncertainty through SMID.

6.4 Summary

This chapter presented simulation results and analysis of ADTMPC's performance in two domains: 1) a double integrator system with nonlinear drag performing collision avoidance; and 2) the nonlinear latitude dynamics of a high-performance aircraft. The collision avoidance results demonstrated that ADTMPC can successfully estimate an unknown drag coefficient, leading to faster, less conservative trajectories. It was also shown that ADTMPC is able to handle measurement noise so long as a noise bound is known. Results of the nonlinear latitude dynamics of an aircraft exhibiting wing rock showed that ADTMPC is able to achieve the same level of performance (in terms of tracking error) as DTMPC but with significantly less control bandwidth. This domain also showed ADTMPC can accurately estimate multiple parameters. The observed performance enhancement in both domains is a direct consequence of ADTMPC being able to robustly update the prediction model online, a feature that is unobtainable in current nonlinear tube MPC frameworks.

Chapter 7

Conclusion and Future Work

7.1 Summary of Contributions

This thesis presented a tube Model Predictive Control framework for nonlinear systems that have uncertain dynamics and are subject to unknown external disturbances. The key contributions of the developed framework are: 1) simultaneously optimizing the tube geometry to satisfy changing constraints and objectives; 2) leverage state-dependent uncertainty to reduce conservativeness in the trajectory design; and 3) estimate model parameters to reduce model uncertainty while maintaining stability and recursive feasibility.

Chapter 3 presented the Dynamic Tube MPC (DTMPC) algorithm that addresses a number of the limitations of state-of-the-art nonlinear tube MPC algorithms. First, the open-loop MPC optimization is augmented with the tube geometry dynamics enabling the trajectory and tube to be optimized simultaneously. Second, DTMPC is able to utilize state-dependent uncertainty to reduce conservativeness and improve optimization feasibility. And third, the tube geometry and error dynamics can be combined to further reduce conservativeness. All three properties are a result of using boundary layer sliding control.

Chapter 4 presented simulation results and analysis of DTMPC's performance in two domains: 1) a double integrator system with nonlinear drag performing collision avoidance; and 2) the nonlinear latitude dynamics of a high-performance aircraft. The collision avoidance results separately demonstrated that DTMPC can optimize the tube geometry (via control bandwidth) to satisfy changing constraints and leverage state-dependent uncertainty to construct less conservative trajectories. Results of the nonlinear latitude dynamics of an aircraft susceptible to wing rock showed that DTMPC is able to optimize the control bandwidth to directly compensate for model uncertainty while satisfying other performance specifications (e.g., maximum allowable tracking error). All results are a consequence of the unique features of DTMPC and are not obtainable with existing nonlinear tube MPC frameworks.

Chapter 5 presented the Adaptive Dynamic Tube MPC (ADTMPC) algorithm that addresses a key limitation of existing nonlinear tube MPC algorithms: the inability to update the prediction model while still maintaining recursive feasibility. It was shown that set membership identification (SMID) guarantees the uncertainty in model parameters monotonically decreases even when disturbances/measurement noise are present (assuming a bound is known). Guaranteed monotonicity is unobtainable with other adaptive/estimation schemes which prevents establishing recursive feasibility. By combining DTMPC and SMID, ADTMPC can: 1) simultaneously optimize a trajectory and tube geometry; 2) leverage state-dependent uncertainty to construct less conservative trajectories; and 3) further reduce conservativeness by using SMID to update model parameters. All three features are unique to ADTMPC and are not obtainable by current nonlinear tube MPC algorithms.

Finally, Chapter 6 presented simulation results and analysis of ADTMPC's performance in two domains: 1) a double integrator system with nonlinear drag performing collision avoidance; and 2) the nonlinear latitude dynamics of a high-performance aircraft. The collision avoidance results demonstrated that ADTMPC can successfully estimate an unknown drag coefficient, leading to faster, less conservative trajectories. It was also shown that ADTMPC is able to handle measurement noise so long as a noise bound is known. Results of the nonlinear latitude dynamics of an aircraft exhibiting wing rock showed that ADTMPC is able to achieve the same level of performance (in terms of tracking error) as DTMPC but with significantly less control bandwidth. This domain also showed ADTMPC can accurately estimate multiple parameters. The observed performance enhancement in both domains is a direct consequence of ADTMPC being able to robustly update the prediction model online, a feature that is unobtainable in current state-of-the-art nonlinear tube MPC frameworks.

7.2 Future Work

The simulation results presented in this thesis demonstrated the benefits DTMPC and ADTMPC. However, experiments on real hardware would further substantiate the usefulness of DTMPC/ADTMPC. In particular, using DTMPC/ADTMPC on a multi-rotor performing obstacle avoidance with on-board sensing and computing would really showcase the benefits of the developed frameworks. The biggest challenge of applying DTMPC/ADTMPC to this hardware domain is transforming sensor information about the environment into a form suitable for optimization. This has been investigated by the author [107] but in the context of using motion primitives instead of optimization for collision avoidance. If a suitable transformation is either too computationally intensive or unobtainable, combining DTMPC/ADTMPC with motion primitives would be required. Combining optimization and motion primitives could be a very fruitful area of research.

Boundary layer control enabled the open-loop trajectory and tube geometry to be simultaneously optimized while leveraging the structure of the uncertainty to reduce conservativeness As mentioned in Chapter 3, boundary layer control can only be used on systems that are input-state/input-output linearizable or systems that can put (either explicitly or approximately) into cascaded form. While many physical systems fall into one of these categories, there exist real-world systems that require a different nonlinear control strategy. For instance, nonholonomic systems (e.g., a differential drive robot) do not satisfy the integrability condition for input-state linearization. Furthermore, these systems tend to possess singularities in input-output linearization controllers. Systems that do not satisfy the controllability condition also pose significant challenges for feedback linearization/boundary layer controllers. One example of such a system is the longitudinal dynamics of an aircraft with nonlinear aerodynamics. A more general control strategy like Control Contraction Metrics [108, 109] could expand the ideas presented in this thesis to a broader class of systems.

Many of the theoretical results for MPC and its alternatives are derived assuming full state feedback. In real-world scenarios, however, some states are not directly measurable so an estimator/observer is required. Further, noisy measurements and/or model error lead to imperfect state estimates. By ignoring the state estimate uncertainty, state constraints can no longer be guaranteed satisfied. Mayne et al. [11, 110] addressed this issue by relying on the separation principle for linear systems to develop the robust output feedback MPC framework. However, the complexity of control and observer design for nonlinear systems make a nonlinear extension very difficult. Developing a systematic framework for controller and observer design for nonlinear systems that satisfy the separation principle is a research direction that can lead to significant contributions in the field of controls and MPC.

Bibliography

[1] T. M. Howard, C. J. Green, and A. Kelly, "Receding horizon model-predictive control for mobile robot navigation of intricate paths," in *Field and Service Robotics.* Springer, 2010, pp. 69–78.

[2] G. Oriolo, G. Ulivi, and M. Vendittelli, "Real-time map building and navigation for autonomous robots in unknown environments," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 28, no. 3, pp. 316–333, 1998.

[3] D. P. Scharf, M. W. Regehr, G. M. Vaughan, J. Benito, H. Ansari, M. Aung, A. Johnson, J. Casoliva, S. Mohan, D. Dueri *et al.*, "Adapt demonstrations of onboard large-divert guidance with a vtvl rocket," in *2014 IEEE Aerospace Conference.* IEEE, 2014, pp. 1–18.

[4] T. E. Moore, "Space shuttle entry terminal area energy management," 1991.

[5] M. A. Henson and D. E. Seborg, *Nonlinear process control.* Prentice Hall PTR Upper Saddle River, New Jersey, 1997.

[6] T. E. Vollmann, *Manufacturing planning and control for supply chain management*, 2005.

[7] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[8] D. P. Bertsekas and I. B. Rhodes, "On the minimax reachability of target sets and target tubes," *Automatica*, vol. 7, no. 2, pp. 233–247, 1971.

[9] D. P. Bertsekas and S. Shreve, *Stochastic optimal control: the discrete-time case*, 2004.

[10] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.

[11] D. Q. Mayne, S. Raković, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems," *Automatica*, vol. 42, no. 7, pp. 1217–1222, 2006.

[12] D. Q. Mayne, E. C. Kerrigan, E. Van Wyk, and P. Falugi, "Tube-based robust nonlinear model predictive control," *International Journal of Robust and Nonlinear Control*, vol. 21, no. 11, pp. 1341–1353, 2011.

[13] S. V. Rakovic, B. Kouvaritakis, M. Cannon, C. Panos, and R. Findeisen, "Parameterized tube model predictive control," *IEEE Transactions on Automatic Control*, vol. 57, no. 11, pp. 2746–2761, 2012.

[14] S. V. Raković, B. Kouvaritakis, R. Findeisen, and M. Cannon, "Homothetic tube model predictive control," *Automatica*, vol. 48, no. 8, pp. 1631–1638, 2012.

[15] S. V. Raković, W. S. Levine, and B. Açıkmeşe, "Elastic tube model predictive control," in *American Control Conference (ACC), 2016*. IEEE, 2016, pp. 3594–3599.

[16] B. Lopez, J.-J. Slotine, and J. How, "Dynamic tube mpc for nonlinear systems," *ACC submission*, 2019.

[17] V. Adetola, D. DeHaan, and M. Guay, "Adaptive model predictive control for constrained nonlinear systems," *Systems & Control Letters*, vol. 58, no. 5, pp. 320–326, 2009.

[18] D. DeHaan and M. Guay, "Adaptive robust mpc: A minimally-conservative approach," in *American Control Conference, 2007. ACC'07*. IEEE, 2007, pp. 3937–3942.

[19] D. Q. Mayne and H. Michalska, "Adaptive receding horizon control for constrained nonlinear systems," in *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*. IEEE, 1993, pp. 1286–1291.

[20] D. DeHaan, V. Adetola, and M. Guay, "Adaptive robust mpc: An eye towards computational simplicity," *IFAC Proceedings Volumes*, vol. 40, no. 12, pp. 228–233, 2007.

[21] V. Adetola and M. Guay, "Robust adaptive mpc for constrained uncertain nonlinear systems," *International Journal of Adaptive Control and Signal Processing*, vol. 25, no. 2, pp. 155–167, 2011.

[22] J. S. Shamma and M. Athans, "Gain scheduling: Potential hazards and possible remedies," *IEEE Control Systems*, vol. 12, no. 3, pp. 101–107, 1992.

[23] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36,

no. 10, pp. 1401–1425, 2000.

[24] R. Tedrake, I. R. Manchester, M. Tobenkin, and J. W. Roberts, "Lqr-trees: Feedback motion planning via sums-of-squares verification," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 1038–1052, 2010.

[25] G. C. Calafiore and L. Fagiano, "Robust model predictive control via scenario optimization," *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 219–224, 2013.

[26] G. Garimella, M. Sheckells, J. Moore, and M. Kobilarov, "Robust obstacle avoidance using tube nmpc."

[27] S. Lall, J. E. Marsden, and S. Glavaški, "A subspace approach to balanced truncation for model reduction of nonlinear control systems," *International journal of robust and nonlinear control*, vol. 12, no. 6, pp. 519–535, 2002.

[28] A. J. Krener and K. Ide, "Measures of unobservability," in *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on.* IEEE, 2009, pp. 6401–6406.

[29] B. T. Hinson and K. A. Morgansen, "Observability optimization for the nonholonomic integrator," in *American Control Conference (ACC), 2013.* IEEE, 2013, pp. 4257–4262.

[30] K. Hausman, J. Preiss, G. S. Sukhatme, and S. Weiss, "Observability-aware trajectory optimization for self-calibration with application to uavs," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1770–1777, 2017.

[31] J. A. Preiss, K. Hausman, G. S. Sukhatme, and S. Weiss, "Trajectory optimization for self-calibration and navigation," *Robotics: Science and Systems XIII*, 2017.

[32] R. L. Kosut, M. K. Lau, and S. P. Boyd, "Set-membership identification of systems with parametric and nonparametric uncertainty," *IEEE Transactions on Automatic Control*, vol. 37, no. 7, pp. 929–941, 1992.

[33] M. Canale, L. Fagiano, and M. Milanese, "Set membership approximation theory for fast implementation of model predictive control laws," *Automatica*, vol. 45, no. 1, pp. 45–54, 2009.

[34] M. Tanaskovic, L. Fagiano, R. Smith, and M. Morari, "Adaptive receding horizon control for constrained mimo systems," *Automatica*, vol. 50, no. 12, pp. 3019–3029,

2014.

[35] M. Lorenzen, F. Allgöwer, and M. Cannon, "Adaptive model predictive control with robust constraint satisfaction," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 3313–3318, 2017.

[36] M. G. Safonov and T.-C. Tsao, "The unfalsified control concept and learning," in *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 3. IEEE, 1994, pp. 2819–2824.

[37] R. L. Kosut, "Uncertainty model unfalsification: A system identification paradigm compatible with robust control design," in *Proceedings of 1995 34th IEEE Conference on Decision and Control*, vol. 4. IEEE, 1995, pp. 3492–3497.

[38] W. Langson, I. Chryssochoos, S. Raković, and D. Q. Mayne, "Robust model predictive control using tubes," *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.

[39] M. Rubagotti, D. M. Raimondo, A. Ferrara, and L. Magni, "Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 556–570, 2011.

[40] D. Althoff, M. Althoff, and S. Scherer, "Online safety verification of trajectories for unmanned flight with offline computed robust invariant sets," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 3470–3477.

[41] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *The International Journal of Robotics Research*, vol. 36, no. 8, pp. 947–982, 2017.

[42] S. Singh, A. Majumdar, J.-J. Slotine, and M. Pavone, "Robust online motion planning via contraction theory and convex optimization," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5883–5890.

[43] B. T. Lopez, J.-J. Slotine, and J. P. How, "Robust planning via sliding control for high-speed navigation," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2018.

[44] K. J. Åström and B. Wittenmark, *Adaptive control*. Courier Corporation, 2013.

[45] J.-J. E. Slotine, W. Li *et al.*, *Applied nonlinear control*. prentice-Hall Englewood Cliffs,

NJ, 1991, vol. 199, no. 1.

[46] M. Shouche, H. Genceli, V. Premkiran, and M. Nikolaou, "Simultaneous constrained model predictive control and identification of darx processes," *Automatica*, vol. 34, no. 12, pp. 1521–1530, 1998.

[47] H. Genceli and M. Nikolaou, "New approach to constrained predictive control with simultaneous model identification," *AIChE journal*, vol. 42, no. 10, pp. 2857–2868, 1996.

[48] B. Hernandez and P. Trodden, "Persistently exciting tube mpc," in *American Control Conference (ACC), 2016*. IEEE, 2016, pp. 948–953.

[49] A. H. González, A. Ferramosca, G. A. Bustos, J. L. Marchetti, M. Fiacchini, and D. Odloak, "Model predictive control suitable for closed-loop re-identification," *Systems & Control Letters*, vol. 69, pp. 23–33, 2014.

[50] A. Anderson, A. González, A. Ferramosca, A. D'Jorge, and E. Kofman, "Robust mpc suitable for closed-loop re-identification, based on probabilistic invariant sets," *Systems & Control Letters*, vol. 118, pp. 84–93, 2018.

[51] G. Marafioti, R. R. Bitmead, and M. Hovd, "Persistently exciting model predictive control," *International Journal of Adaptive Control and Signal Processing*, vol. 28, no. 6, pp. 536–552, 2014.

[52] C. Rohrs, L. Valavani, M. Athans, and G. Stein, "Robustness of continuous-time adaptive control algorithms in the presence of unmodeled dynamics," *IEEE Transactions on Automatic Control*, vol. 30, no. 9, pp. 881–889, 1985.

[53] M. Bujarbaruah, X. Zhang, U. Rosolia, and F. Borrelli, "Adaptive mpc for iterative tasks," in *2018 IEEE Conference on Decision and Control (CDC)*. IEEE, 2018, pp. 6322–6327.

[54] E. F. Camacho and C. B. Alba, *Model predictive control*. Springer Science & Business Media, 2013.

[55] J. B. Rawlings and D. Q. Mayne, "Model predictive control: Theory and design," 2009.

[56] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in identification and control*. Springer, 1999, pp. 207–226.

[57] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Systems*, vol. 36, no. 6, pp. 30–44, 2016.

[58] D. P. Bertsekas, "Dynamic programming and suboptimal control: A survey from adp to mpc," *European Journal of Control*, vol. 11, no. 4-5, pp. 310–334, 2005.

[59] D. Mayne, "Robust and stochastic model predictive control: Are we going in the right direction?" *Annual Reviews in Control*, vol. 41, pp. 184–192, 2016.

[60] P. O. Scokaert and D. Mayne, "Min-max feedback model predictive control for constrained linear systems," *IEEE Transactions on Automatic control*, vol. 43, no. 8, pp. 1136–1142, 1998.

[61] A. Bemporad, F. Borrelli, and M. Morari, "Min-max control of constrained uncertain discrete-time linear systems," *IEEE Transactions on automatic control*, vol. 48, no. 9, pp. 1600–1606, 2003.

[62] J. Löfberg, *Minimax approaches to robust model predictive control.* Linköping University Electronic Press, 2003, vol. 812.

[63] D. M. Raimondo, D. Limon, M. Lazar, L. Magni, and E. F. ndez Camacho, "Min-max model predictive control of nonlinear systems: A unifying overview on stability," *European Journal of Control*, vol. 15, no. 1, pp. 5–21, 2009.

[64] J. A. Rossiter, B. Kouvaritakis, and M. Rice, "A numerically robust state-space approach to stable-predictive control strategies," *Automatica*, vol. 34, no. 1, pp. 65–73, 1998.

[65] D. Q. Mayne and E. C. Kerrigan, "Tube-based robust nonlinear model predictive control," in *Proc. of the 7th IFAC Symposium on Nonlinear Control Systems*, 2007, pp. 110–115.

[66] M. Althoff, O. Stursberg, and M. Buss, "Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on.* IEEE, 2008, pp. 4042–4048.

[67] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[68] V. Utkin, "Variable structure systems with sliding modes," *IEEE Transactions on Automatic control*, vol. 22, no. 2, pp. 212–222, 1977.

[69] K. R. Muske, H. Ashrafiuon, and M. Nikkhah, "A predictive and sliding mode cascade controller," in *American Control Conference, 2007. ACC'07.* IEEE, 2007, pp. 4540–4545.

[70] M. Spasic, M. Hovd, D. Mitic, and D. Antic, "Tube model predictive control with an auxiliary sliding mode controller," 2016.

[71] S. E. Benattia, S. Tebbani, and D. Dumur, "Hierarchical control strategy based on robust mpc and integral sliding mode-application to a continuous photobioreactor," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 212–217, 2015.

[72] G. P. Incremona, A. Ferrara, and L. Magni, "Hierarchical model predictive/sliding mode control of nonlinear constrained uncertain systems," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 102–109, 2015.

[73] D. Mitić, M. Spasić, M. Hovd, and D. Antić, "An approach to design of sliding mode based generalized predictive control," in *Applied Computational Intelligence and Informatics (SACI), 2013 IEEE 8th International Symposium on.* IEEE, 2013, pp. 347–351.

[74] A. Chakrabarty, V. C. Dinh, G. T. Buzzard, S. H. Zak, and A. E. Rundell, "Robust explicit nonlinear model predictive control with integral sliding mode." in *ACC*, 2014, pp. 2851–2856.

[75] I. R. Manchester and J.-J. E. Slotine, "Control contraction metrics and universal stabilizability," *IFAC Proceedings Volumes*, vol. 47, no. 3, pp. 8223–8228, 2014.

[76] A. G. Richards, "Robust constrained model predictive control," Ph.D. dissertation, Massachusetts Institute of Technology, 2005.

[77] A. Richards and J. How, "Robust stable model predictive control with constraint tightening," in *American Control Conference, 2006.* IEEE, 2006, pp. 6–pp.

[78] L. Chisci, J. A. Rossiter, and G. Zappa, "Systems with persistent disturbances: predictive control with restricted constraints," *Automatica*, vol. 37, no. 7, pp. 1019–1028, 2001.

[79] D. Angeli *et al.*, "A lyapunov approach to incremental stability properties," *IEEE Transactions on Automatic Control*, vol. 47, no. 3, pp. 410–421, 2002.

[80] J. Köhler, M. A. Müller, and F. Allgöwer, "A novel constraint tightening approach for nonlinear robust model predictive control," in *American Control Conference (ACC)*, 2018.

[81] D. Bernardini and A. Bemporad, "Scenario-based model predictive control of stochastic constrained linear systems," in *Decision and Control, 2009 held jointly with the 2009*

28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on. IEEE, 2009, pp. 6333–6338.

[82] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems.* Courier Corporation, 2012.

[83] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 182–201, 2014.

[84] J. Hernandez, K. Tsotsos, and S. Soatto, "Observability, identifiability and sensitivity of vision-aided inertial navigation," in *International Conference on Robotics and Automation (ICRA).* IEEE, 2015, pp. 2319–2325.

[85] M. Rafieisakhaei, S. Chakravorty, and P. Kumar, "On the use of the observability gramian for partially observed robotic path planning problems," *arXiv preprint arXiv:1801.09877*, 2018.

[86] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *International Conference on Robotics and Automation (ICRA), pages=723–730, year=2011, organization=IEEE.*

[87] M. W. Achtelik, S. Lynen, S. Weiss, M. Chli, and R. Siegwart, "Motion-and uncertainty-aware path planning for micro aerial vehicles," *Journal of Field Robotics*, vol. 31, no. 4, pp. 676–698, 2014.

[88] V. Indelman, L. Carlone, and F. Dellaert, "Planning in the continuous domain: A generalized belief space approach for autonomous navigation in unknown environments," *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 849–882, 2015.

[89] G. Costante, C. Forster, J. Delmerico, P. Valigi, and D. Scaramuzza, "Perception-aware path planning," *arXiv preprint arXiv:1605.04151*, 2016.

[90] J.-J. E. Slotine, "Sliding controller design for non-linear systems," *International Journal of control*, vol. 40, no. 2, pp. 421–434, 1984.

[91] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach.* Malaysia; Pearson Education Limited,, 2016.

[92] A. Isidori, *Nonlinear control systems.* Springer Science & Business Media, 2013.

[93] H. Sussmann and P. Kokotovic, "The peaking phenomenon and the global stabilization

of nonlinear systems," *IEEE Transactions on automatic control*, vol. 36, no. 4, pp. 424–440, 1991.

[94] M. Krstic, I. Kanellakopoulos, P. V. Kokotovic *et al.*, *Nonlinear and adaptive control design*, vol. 222.

[95] S. M. LaValle, *Planning algorithms.* Cambridge university press, 2006.

[96] J. Deng, V. Becerra, and R. Stobart, "Input constraints handling in an mpc/feedback linearization scheme," *International Journal of Applied Mathematics and Computer Science*, vol. 19, no. 2, pp. 219–232, 2009.

[97] D. Simon, J. Löfberg, and T. Glad, "Nonlinear model predictive control using feedback linearization and local inner convex constraint approximations," in *2013 European Control Conference (ECC).* IEEE, 2013, pp. 2056–2061.

[98] Y. Mao, M. Szmuk, and B. Açıkmeşe, "Successive convexification of non-convex optimal control problems and its convergence properties," in *Decision and Control (CDC), 2016 IEEE 55th Conference on.* IEEE, 2016, pp. 3636–3641.

[99] J. Löfberg, "Yalmip : A toolbox for modeling and optimization in matlab," in *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[100] "The mosek optimization software," *Online at http://www. mosek. com*, vol. 54, no. 2-1, p. 5, 2010.

[101] L. Nguyen, L. Yip, and J. Chambers, "Self-induced wing rock of slender delta wings," in *7th Atmospheric Flight Mechanics Conference*, 1981, p. 1883.

[102] C.-H. Hsu and C. E. Lan, "Theory of wing rock," *Journal of Aircraft*, vol. 22, no. 10, pp. 920–924, 1985.

[103] M. M. Monahemi and M. Krstic, "Control of wing rock motion using adaptive feedback linearization," *Journal of guidance, control, and dynamics*, vol. 19, no. 4, pp. 905–912, 1996.

[104] C.-M. Lin and C.-F. Hsu, "Recurrent neural network adaptive control of wing-rock motion," *Journal of guidance, control, and dynamics*, vol. 25, no. 6, pp. 1163–1165, 2002.

[105] T. H. Mattheiss, "An algorithm for determining irrelevant constraints and all vertices in systems of linear inequalities," *Operations Research*, vol. 21, no. 1, pp. 247–260, 1973.

[106] G. Chowdhary, M. Mühlegg, J. P. How, and F. Holzapfel, "Concurrent learning adaptive model predictive control," in *Advances in Aerospace Guidance, Navigation and Control*. Springer, 2013, pp. 29–47.

[107] B. T. Lopez and J. P. How, "Aggressive 3-d collision avoidance for high-speed navigation," in *International Conference on Robotics and Automation (ICRA)*. IEEE, 2017.

[108] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for non-linear systems," *Automatica*, vol. 34, no. 6, pp. 683–696, 1998.

[109] I. R. Manchester and J.-J. E. Slotine, "Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 3046–3053, 2017.

[110] D. Q. Mayne, S. Raković, R. Findeisen, and F. Allgöwer, "Robust output feedback model predictive control of constrained linear systems: Time varying case," *Automatica*, vol. 45, no. 9, pp. 2082–2087, 2009.