Adaptive Control of Unmanned Aerial Systems

by

Zachary Thompson Dydek

B.S., California Institute of Technology (2005)
S.M., Massachusetts Institute of Technology (2007)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2010

Author . . . .
Department of Mechanical Engineering
June 30, 2010

Certified by . . . .
Anuradha M. Annaswamy
Senior Research Scientist
Thesis Supervisor

Accepted by . . . .
Professor David E. Hardt
Chairman, Committee on Graduate Students

Adaptive Control of Unmanned Aerial Systems

by

Zachary Thompson Dydek

Submitted to the Department of Mechanical Engineering
on June 30, 2010, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

Abstract

Adaptive control is considered to be one of the key enabling technologies for future high-performance, safety-critical systems such as air-breathing hypersonic vehicles. Adaptive flight control systems offer improved performance and increased robustness to uncertainties by virtue of their ability to adjust control parameters as a function of online measurements. Extensive research in the field of adaptive control theory has enabled the design, analysis, and synthesis of stable adaptive systems. We are now entering the stage in which adaptive flight control systems have reached the requisite level of maturity for application to hardware flight platforms.

Unmanned aerial systems (UAS) provide a unique opportunity for the transition of adaptive controllers from theory to practice. The small, unmanned aerial vehicles (UAVs) examined in this thesis offer a low-cost, low-risk stepping stone between simulation and application to higher-risk systems in which safety is a critical concern. Unmanned aircraft also offer several benefits over their manned counterparts including extreme persistence, maneuverability, lower weight and smaller size. Furthermore, several missions such as surveillance, exploration, search-and-track, and lifting of heavy loads are best accomplished by a UAS consisting of multiple UAVs. This thesis addresses some of the challenges involved with the application of adaptive flight control systems to UAS.

Novel adaptive control architectures are developed to overcome performance limitations of UAS, the most significant of which is a large time delay due to communication and limited onboard processing. Analytical tools that allow the calculation of a theoretically justified time delay limit are also developed. These tools in turn lead to an estimate of the time-delay margin of the closed-loop system which is an essential part of the validation and verification methodology for intelligent flight control systems. These approaches are validated numerically using a series of simulation studies. These controllers and analytical methods are then applied to the UAV, demonstrating improved performance and increased robustness to time delays. Also introduced in this thesis is a novel adaptive methodology for coordinated adaptive control of a multi-vehicle UAS. Including two distinct classes of adaptive algorithms at both the local and global levels was found to result, both in simulation and in actual flight tests, in decreased tracking error for individual vehicles, decreased errors in inter-vehicle distances, and reduced likelihood of collisions with other vehicles or obstacles in the environment.

Thesis Supervisor: Anuradha M. Annaswamy
Title: Senior Research Scientist

Acknowledgments

I am deeply indebted to my advisor Anuradha M. Annaswamy for opening her door to me when I first arrived at MIT, for always making time to discuss my research, and for her feedback and guidance throughout my nearly five years in the lab. Your commitment to rigor and excellence has contributed immeasurably to my growth as an engineer.

My sincere appreciation goes to Eugene Lavretsky, who has served as a committee member, mentor, wellspring of ideas, and friend during my time on this project. At each of your bi-yearly visits to MIT, it seems like our discussions were the genesis of a year's worth of interesting research questions and new directions.

I would like to thank Franz Hover for serving on my committee and taking an interest in my work. Meeting one-on-one with you really helped me see the big picture and shaped the direction of this thesis.

My sincere thanks also go out to Jonathan How for being on my committee and allowing the use of his laboratory for flight testing. Our collaboration has really allowed this research to take off, so to speak. I would also like to thank all the students of the Aerospace Controls Laboratory for making this work possible, particularly Josh Redding and Buddy Michini.

I would like to thank Jean-Jacques E. Slotine, who not only helped to shape portions of this research, but also taught several courses that have helped shape my interests and views on control and its applications. Our numerous enlightening conversations have been invaluable to me.

I would also like to thank several of my colleagues in particular for their discussions, advice, and friendship over the years: Jinho Jang, Yildiray Yildiz, Manohar Srikanth, Travis Gibson, and Megumi Matsutani. 고맙습니다. Teşekkür ederim. ಧನ್ಯವಾದ. Thank you. お世話になりました。 I would also like to thank labmates Jaejeen Choi, Richard James, Himani Jain, Paul Ragaller, Seunghyuck Hong, Chun Yang Ong, Hanbee Na, and Amith Somnanth.

This thesis was made possible by the financial support of The Boeing Company

Strategic University Initiative and the DOD NDSEG fellowship, which I will forever be thankful for.

I want to thank my parents, my four brothers, and sister; through your patience, love, and encouragement you've made all of this possible. As I grew up, my toys did too: from blocks, to LEGOs, to scrap wood and power tools (under supervision, of course!). I also had early access to computers and to the internet. The fact that this thesis is equal parts hardware and software really is fitting. Each of my experiences growing up and my interactions with all of you have left indelible marks on me. I love you all.

Last but not least I would like to thank my wife Vicki Dydek, without whose brilliant radiance and radiant brilliance I would not have arrived where I am today. With you at my side I know I will always be inspired to strive for ever-greater heights. This thesis exists because of you. Thank you!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Ever-increasing demands on safety and performance have necessitated the development of more and more sophisticated flight control systems. Adaptive control is considered to be one of the key enabling technologies for future high-performance, safety-critical systems such as air-breathing hypersonic vehicles. Adaptive flight control systems also offer numerous benefits over their fixed-gain counterparts, including improved performance, increased robustness to uncertainties, deceased design cycle time, as well as lower weight and lower cost. They are able to achieve these by virtue of their ability to adjust control parameters as a function of online measurements. Early attempts at adaptive flight control used controllers with unproven stability properties, sometimes with disastrous consequences; for example the fatal crash of the NASA X–15 in November, 1967 [1]. Since that time, extensive research in the field of adaptive control theory has enabled the design, analysis, and synthesis of stable adaptive systems. We are now entering the stage in which adaptive flight control systems have reached the requisite level of maturity for application to hardware flight platforms.

The aviation industry is in general slow to adopt new control algorithms, partly because of the sobering lessons learned from programs such as the X–15. Promising new technologies are often transitioned from theory to practice along a series of application platforms with increasingly stringent demands on safety. The starting point and proving ground for many potentially useful control approaches is therefore guided munitions. Successful technologies from that stage are subsequently applied to unmanned aerial vehicles, military vehicles such as fighter jets, cargo transport aircraft and finally civilian transport aircraft. In the more recent past, adaptive control has successfully navigated this path, proving itself on the laser-guided joint direct attack munition [2, 3], and has recently begun to find its way towards UAV applications such as The Boeing Company's Phantom Ray autonomous aircraft.

In addition to providing a stepping stone for the smooth transition of adaptive flight control from theory to practice, unmanned aerial systems (UAS) are themselves an important platform and an area of considerable research interest. Unmanned aircraft offer several benefits over manned systems and have an impressive track record [4]. Chief among these benefits are extreme persistence, maneuverability, ability to operate in dangerous environments without risking safety of a crew, smaller size and lower cost [5]. Furthermore, several missions such as surveillance, exploration, search-and-track, and lifting of heavy loads are best accomplished by multiple UAVs, leading to further savings in both time and money. Another important advantage to utilizing multiple vehicles is a further reduction in the risk to successful completion of a mission due to the loss of a single vehicle. When a single vehicle malfunctions, neighboring vehicles can adjust their configuration to compensate. This increased robustness can lead to a commensurate decrease in vehicle specifications and cost, further improving the argument for swarm operations.

1.2 Research objectives

The primary research objective for this thesis is to examine the efficacy of adaptive flight control for UAS, particularly in the presence of parametric uncertainties and time delays. During the course of this examination we will address the following specific problems:

- *Application of adaptive flight control to a single UAV*

  Many tasks once accomplished using a single large, expensive vehicle can be accomplished more effectively by using many small, inexpensive UAVs. Consequently, each individual UAV is more prone to uncertainties due to electrical or mechanical failures, such as actuator failures. Furthermore, many UAVs operate in dangerous or cluttered environments, increasing the risk of uncertainties due to battle damage or collisions with the environment. It is therefore desirable to design a sophisticated controller capable of accommodating these various uncertainties without sacrificing guarantees on stability. The first problem examined in this thesis is the application of an existing theory-based model reference adaptive control (MRAC) approach to the UAV flight control problem. Implementation issues such as dealing with sensor noise, parameter drift, and proper tuning of adaptive gains will be addressed.

- *Adaptive control of general systems with time delays*

  One of the largest outstanding questions that must be addressed before the widespread adoption of adaptive flight control is how to certify the performance and stability of an adaptive controller for systems with (possibly large) time delays. Time delays are an especially important issue for UAS, which often have considerable delays due to communication with ground-based operators and limited onboard computational power. Novel tools for analyzing existing adaptive control strategies in systems with time delays will be explored. Novel adaptive controllers that offer numerous benefits over existing technologies, including increased performance and analytically justified time delay margins, will also be considered.

- *Generating improved performance for adaptive control of a single UAV*

  Traditionally, the performance of adaptive control for time delay systems has been limited by a trade-off between performance and robustness to the time delay. In the MRAC structure, for example, adaptive gains must be lowered to avoid exciting high frequencies and inducing instabilities in the closed-loop system. This results in decreased tracking performance and slower convergence of parameter estimates. Using a subset of the techniques discussed for general time delay systems, the above shortcomings are alleviated by designing an adaptive controller that can generate improved performance while remaining robust to time delay and this approach is successfully demonstrated on a single UAV.

- *Adaptive control of multiple UAVs*

  Multiple UAVs executing a mission often need to act in a cooperative or coordinated manner. It is therefore essential that control strategies implemented for multi-vehicle UAS can accommodate uncertainties not only at the single-vehicle level, but also on a global level. A novel adaptive configuration control structure that addresses the problem of coordinated path planning of multiple vehicles in the presence of uncertainties will be examined.

1.3 Research approach

This thesis involves the full spectrum from conception of novel adaptive approaches, through rigorous theoretical explorations, implementation in simulation with varying levels of fidelity, and finally validation using flight tests. Furthermore, the process is iterative in the sense that lessons learned from flight tests of one particular technology are used to conceive of novel approaches to overcome performance limitations or drawbacks of a particular approach. These novel approaches are then validated numerically in the MATLAB environment using a series of simulation studies before being applied to the UAS. Flight testing is accomplished in collaboration with the Aerospace Controls Laboratory within an indoor autonomous vehicle test facility at MIT. The UAV platform used in these studies is a modified version of the four rotor helicopter ("quadrotor") UAV known as the Draganflyer V Ti. This vehicle, shown in figure 1-1, can be purchased from RC hobbyist sites.

Figure 1-1: Draganflyer V Ti four rotor helicopter UAV.

1.4 Organization

This thesis is divided into seven chapters. The contents of each chapter can be summarized as follows:

- Chapter 1, *Introduction*, motivates the research effort, introduces the objectives of the work, and details the specific approach taken. It also provides a detailed roadmap for the thesis.

- Chapter 2, *Background*, discusses the previous work in four major categories: adaptive control theory and applications to flight control, selected extensions to the MRAC approach, flight control of quadrotor UAVs, and cooperative control of multiple vehicles.

- Chapter 3, *Adaptive Control of UAVs with parametric uncertainties*, details the application of MRAC to a quadrotor UAV with the goal of increasing robustness to parametric uncertainties. Details on some of the implementation roadblocks and how they were overcome are discussed as well.

- Chapter 4, *General systems with parametric uncertainty and time delay*, describes two methods for analysis and synthesis of adaptive controllers in general systems with time delays. A describing functions approach is used to approximate an adaptive system with a linear one, enabling the use of linear tools. Additionally, a novel adaptive architecture designed to explicitly account for known time delays is presented.

- Chapter 5, *Combined / Composite Adaptive control of UAVs with parametric uncertainties*, presents the application of a subset of the techniques discussed in chapter 4 to the quadrotor UAV discussed in chapter 3.

- Chapter 6, *Adaptive configuration control of multi-vehicle UAS*, addresses the problem of adaptive outer-loop control of a UAS consisting of multiple UAVs. The approach can be combined with inner-loop approaches described in previous chapters, leading to further improvements.

- Chapter 7, *Summary and Future Work*, summarizes the main contributions of the thesis and describes some of the future applications and extensions.

Chapter 2

Background

In this chapter we discuss some of the previous work in the following four areas of research: adaptive control theory and applications to flight control, selected extensions to the MRAC approach, flight control of quadrotor UAVs, and cooperative control of multiple vehicles.

2.1 Adaptive control theory and applications to flight control

The field of adaptive control began with the motivation that a controller able to adjust its parameters online could generate improved performance over a fixed-parameter counterpart. One of the earliest examples of such a controller was the Minneapolis Honeywell MH-96 "self-adaptive" controller. This controller was flown as part of the NASA X–15 program, which recorded nearly 200 successful flights from 1959-1968. However, the MH-96 lacked an analytically-based proof of stability, which was highlighted by the fatal accident that occurred on November 15, 1967 [1, 6]. Since that time, the evolution of the field of adaptive control was directed toward the design, analysis, and synthesis of stable adaptive systems. Various adaptive control methods have been developed for controlling linear and nonlinear dynamic systems with parametric and dynamic uncertainties [7–31].

These techniques have been extended and applied to the problem of aircraft control [32–39], showing promising results in simulation. There have also recently been examples of successful adaptive control flight tests for both manned and unmanned test vehicles. In [40], a modified sequential least squares algorithm was designed to make online parameter estimates in a computationally efficient manner. This algorithm was then implemented and flight tested on a VISTA F-16 fighter aircraft equipped with a vehicle integrity monitor that would turn off the adaptive feature if it violated predetermined flight envelope or structural safety limits. Using this approach, the vehicle was flown under a number of simulated failure scenarios, including landing with a missing elevon in a strong crosswind.

An existing dynamic inversion baseline controller on the X-36 tailless fighter aircraft was augmented by an adaptive neural network reconfigurable flight control system in [41]. The X-36 is a remote-piloted aircraft with a wingspan of approximately 10.4 feet. The adaptive and baseline controllers were compared for a number of flight conditions and a number of failure scenarios, including severe failures such as locking of control surfaces at their maximum deflection. The adaptive controller was shown to provide improved handling response and resistance to departures as compared with the baseline controller.

In [42], a single hidden layer neural network adaptive controller was applied to a modified autonomous Yamaha R-Max helicopter. An adaptive anti-windup strategy known as pseudo-control hedging was used to prevent the adaptive controller from adapting to errors due to input saturation and other selected system input characteristics. The inner- and outer-loop were combined to allow for higher outer-loop bandwidth, resulting in improved path-following capabilities over a baseline approach. Parametric uncertainty due to actuator failure or damage was not considered.

Finally, [2] describes the application of an augmented model reference adaptive controller to the sensor-guided joint direct attack munition (JDAM). The baseline JDAM uses a gain-scheduled LQR-based linear controller and relies on GPS-aided inertial measurement units for accuracy. However, it was desired to add a laser-guided capability in order to track targets that were either moving or had unknown

GPS coordinates. This modification required the addition of a sensor package to the nose of the JDAM, causing significant changes to the aerodynamics. MRAC was used as a retrofit to the existing linear controller in order to accommodate the change in aerodynamics without extensive wind-tunnel testing or expensive redesign of the flight control system. The Laser JDAM is an example of a flight-proven adaptive control system that is currently used in a production model.

The strong theoretical foundation of adaptive control and promises of improved performance, robustness, and reduced design cycle time has prompted the application of several adaptive strategies to several aerial vehicles. Adaptive control has been proven to be effective at handling changes in the aerodynamics of a guided munition and is currently part of the Laser JDAM system. Adaptive approaches have also made some forays into the realm of flight control for manned and unmanned aerial vehicles, both fixed- and rotary-wing. However, the rotary-wing application in [42] does not examine uncertainties in the inner-loop, instead adapting to changes in the slower, outer-loop system. In the fixed-wing applications, actuator failures and other uncertainties were examined. However, none of these approaches address the issue of time delay, which can place severe limitations on the speed of adaptation and thus the benefit of including adaptive features. Further, almost all research to-date in the field of adaptive control pertains to the control of a single-vehicle, in particular pertaining to command following [3, 43]. Scant attention has been paid to adaptive coordinated control of multiple vehicles.

2.2 Selected extensions to the MRAC approach

In addition to these advances in the theory of stable adaptive systems, a number of modifications to the standard MRAC approach have been presented. These extensions are designed to improve performance, robustness, or to generate other desirable qualities and far too many variations exist to be covered in detail. Interested readers are referred to the recent Transactions on Automatic Control, Automatica, and the proceedings of the CDC, ACC, and GNC conferences. Rather than compiling an exhaustive list, we focus on a small number of modifications to the MRAC approach which are relevant to the exploration of adaptive time delay systems presented in chapter 4. The first of these modifications involves the combination of direct and indirect adaptive control, which is known as Combined or Composite MRAC (CMRAC). By adapting to both estimation and tracking errors, it has been observed that CMRAC systems have smoother transient performance than MRAC systems for a wide variety of problems [44–47].

Another such modification is the use of time-varying adaptive gains. In particular, a bounded-gain forgetting (BGF) law for adjusting adaptive gains based on least-squares estimation was applied to the problem of robotic manipulation [48]. This modification has the benefits of faster parameter convergence and smoother parameter estimates. While not specifically designed for systems with time delay, these two modifications share a common feature in that they both lead to smooth transients in the adaptive and estimation parameters. It could therefore be argued that the same modifications can lead to an improved time-delay margin by virtue of a reduction in the high-frequency content of the signals in the system. Therefore, we propose that these modifications to MRAC may be useful tools in designing controllers for systems with time delay as well.

One final modification to MRAC relevant to control of time delay systems is the recently developed Adaptive Posicast Controller (APC) approach detailed in [49]. The APC approach is an adaptive extension of the Smith Predictor, which uses a plant model to predict the future outputs of the plant, and then uses this prediction to cancel the effect of delay on the system. This approach has been successfully applied to the automotive problems of idle-speed control and fuel-air ratio control and, as opposed to the CMRAC and BGF modifications, explicitly accounts for known time delay in the system.

2.3 Flight control of quadrotor UAVs

The first known quadrotor flight was in the 1920's [50], but due to the difficulty of controlling four motors simultaneously with sufficient bandwidth, the project was abandoned. In 1963, the Curtiss-Wright X-19A became the first (and only) manned quadrotor to leave the ground effect. Due to a lack of stability augmentation system, stationary hover was nearly impossible [51]. In recent years, smaller, unmanned quadrotor helicopters have been an increasingly popular research platform. Their simple design and relatively low cost make them attractive candidates for swarm operations, a field of ongoing research in the UAV community. These quadrotor helicopters typically consist of two pairs of counter-rotating blades mounted on a carbon fiber frame as shown in Figure 1-1. The dynamics of quadrotor helicopters have been studied in detail by several groups [52–54]. In designing a controller for these aircraft, there are several important vehicle-specific considerations. There are numerous sources of uncertainties in the system, for example, actuator degradation, external disturbances, and potentially uncertain time delays in processing or communication. Additionally, the dynamics of quadrotors are nonlinear and multivariate. There are several effects to which a potential controller must be robust: the aerodynamics of the rotor blades (propeller and blade flapping), inertial anti-torques (asymmetric angular speed of propellers), as well as gyroscopic effects (change in orientation of the quadrotor and the plane of the propeller).

A variety of approaches to quadrotor flight control have been applied to a variety of problems. The Aerospace Controls Laboratory at MIT uses a Vicon motion capture system to enable rapid prototyping of aerobatic flight controllers for quadrotors and other aerial vehicles; robust coordination algorithms for multiple quadrotors; and vision-based sensing algorithms for indoor flight [55–57]. A linear, ground-based controller augments an onboard rate feedback controller to allow for autonomy and waypoint-following. In [58], a dynamic programming approach was used to generate a policy to allow for persistent surveillance for a fleet of vehicles with stochastic failure and fuel burn rates.

The Stanford Testbed of Autonomous Rotorcraft for Multi-Agent Control (STARMAC) quadrotor helicopter is described in [53]. The STARMAC II quadrotor helicopter features onboard measurement, communication, and processing. Along with appropriate vehicle configuration, a PID controller was designed to give robustness to nonlinear aerodynamic effects such as thrust changes due to vehicle translation, blade flapping, and vortex impingement. Flight test results show that the approach allows for autonomous tracking and regulation in indoor and outdoor settings.

In [59], a feedback linearization-based controller was augmented with a high-order sliding mode observer and applied to a simulated quadrotor system. The fourth-order sliding mode observer estimates the effects of external disturbances due to wind or noise, and allows for control with a minimal number of sensors. Simulation results show that the approach provides stability and robustness, significantly reducing chatter due to noise and other external disturbances.

Recent work at the GRASP laboratory at the University of Pennsylvania [60] has shown cooperative manipulation using several quadrotors to lift and manipulate a lightweight object with 6 degrees-of-freedom. This approach uses a Vicon motion capture system for sensing and a PID controller to generate the forces and moments necessary to achieve the desired manipulation behavior. Oscillations in the underdamped system limited the trajectory following to slow maneuvers. Sensing and actuator uncertainties were not considered.

The approach described in [61] allows for autonomous flight in unknown indoor environments. The quadrotor UAVs are equipped with onboard sensing of the vehicle state as well as laser scanners for sensing of the environment. A modified 2D SLAM algorithm that accounts for the 3D motion of the vehicle was employed to allow for mapping while flying through the unknown operating environment. The quadrotor dynamics are approximated by a double integrator and LQR control design is used to provide stable hover. The vehicle was able to successfully explore and map several indoor environments both with the guidance of a human operator and autonomously.

While most of the above approaches acknowledge the significant uncertainties present in the system and the potential for additional uncertainty due to failures, none of the approaches explicitly accounts for these uncertainties in the controller design. Typical control laws consist of linear, PID control or LQR-based design, which offer some measure of robustness to uncertainties but may not be sufficient for severe uncertainties such as those that result from actuator failure or structural damage. This thesis will examine severe failures for which a linear control design may not be adequate.

2.4 Cooperative control of multiple vehicles

Cooperative, multi-agent control is currently an area of intense research activity with notable contributions made by several groups [62–67]. We briefly discuss a few of the main approaches below.

In [62], the authors describe an agile and stable control method for distributed stochastic systems based on the theory of stochastic dynamic programming (SDP). Their rollout algorithm includes a model of the feedback mechanism, increasing performance and enabling the ability to react to possible future events. Since the standard SDP approach scales badly with the number of vehicles, approximate approaches which yield near-optimal control laws are discussed.

The approach in [63] relates the problem of formation flight to moving the vehicles to points on a virtual rigid body that can rotate, translate, expand, and contract in space. This centralized approach assumes the vehicles are point masses with fully actuated dynamics, and the goal is to minimize a set of virtual potentials, which accomplishes formation keeping and collision avoidance. These results are extended further in [64] to the case of steered particles in a plane and limited communication between vehicles.

A geodesic control law that allows for flocking and velocity alignment of several nonholonomic vehicles in 2 and 3 dimensions is proposed in [65]. This method allows for decentralized control and guarantees flocking will occur under certain assumptions on the underlying proximity graphs that determine which vehicles can communicate with one another. In [66], a routing algorithm for moving a number of fully-actuated, point-mass vehicles to target destinations in a plane is presented. These destinations are generated over time by a stochastic process. The control strategies aim to minimize the expected time that a target is unvisited by one of the vehicles. This approach makes no explicit assumptions on the communication ability of the individual vehicles.

A decentralized approach is proposed in [67] which takes ideas from both control theory and graph theory to make a formal analysis of the stability of the formation. The notion of stability is decomposed into the stability of the formation given the graph of connections between vehicles and the stability of individual vehicles given their controllers. The results in [67] only apply to vehicles with linear dynamics, but some possible extensions to vehicles with nonlinear dynamics are proposed as well.

In [68], an $\mathcal{L}_1$ adaptive output feedback controller is added to an existing outer-loop path-planning algorithm with no change to the inner-loop controller. The $\mathcal{L}_1$ adaptive augmentation takes the form of a integral disturbance rejection controller and allows for smoother command generation. The velocity profiles of multiple vehicles can be adjusted so that multiple vehicles arrive at a goal simultaneously or at a fixed arrival interval.

These results represent a wide variety of approaches in terms of the assumptions made on the ability of the individual vehicles to communicate with each other and with a centralized entity. However, there are also a number of similarities. Most of the results in [62–67] are either pure theory or simulation results using relatively simplistic models of vehicle dynamics, such as fully-actuated point masses. Most propose some cost function, based on either virtual potential fields or task completion, and seek to minimize this cost function through computation of a near-optimal control strategy either on or off-line. Additionally, the above approaches claim robustness to vehicle failures by virtue of the fact that they allow for many vehicles with no explicit leader-follower structure. That is, even if one vehicle becomes disabled, the other vehicles can carry on indifferently. None of the above discusses the case of a partial failure wherein some technical capabilities of a vehicle are diminished, but the vehicle is still flyable. In the case of these partial failures it may be possible, through adapting to the changes on-line, to complete the mission with all vehicles returning safely. In the case of partial failures the full, possibly nonlinear vehicle dynamics become important in determining how the failure affects the performance of the damaged vehicle. In turn, the degraded vehicle performance has an effect on the overall formation and the cost functions and mission objectives may have to be adjusted online. The algorithms described in [62–68] cannot fully utilize the capabilities of all vehicles when those capabilities are uncertain and dynamic.

Most of the approaches described above side-step the vehicle control problem by either assuming that the vehicle dynamics are extremely simple, or by assuming that an inner-loop controller that solves the problem exists. As UAV designs become smaller, lighter, and more agile, these assumptions no longer hold. While the approach described in [68] includes adaptation in the outer-loop control of the vehicle kinematics, this approach does not accommodate local errors in the vehicle dynamics, or global errors, such as error in the overall configuration of the vehicles. This thesis will address the problem of control of a multi-vehicle UAS in the presence of uncertainty by including adaptation in the inner- and outer- control loops and adapting to both local and global errors.

Chapter 3

Adaptive Control of UAVs with parametric uncertainties

This chapter describes the application of model reference adaptive control to a lightweight, low-cost quadrotor UAV platform. An adaptive controller was designed to augment an existing linear controller that provides autonomy and waypoint following. The design of the adaptive controller is driven by Lyapunov stability arguments and thus has a proof of stability grounded in a nonlinear framework. The approach was validated using simulations and flight tested in an indoor test facility. The adaptive controller was found to offer increased robustness to parametric uncertainties. In particular, it was found to be effective in mitigating the effects of a loss of thrust anomaly, which may occur due to component failure or physical damage. The design of the adaptive controller is presented, followed by a comparison of flight test results using the existing linear and augmented adaptive controllers.

3.1 Introduction

Quadrotor helicopters have been an increasingly popular research platform in recent years. Their simple design and relatively low cost also make them attractive candidates for swarm operations, a field of ongoing research in the UAV community. In designing a controller for these aircraft, there are several important considerations.

There are numerous sources of uncertainties in the system—actuator degradation, external disturbances, and potentially uncertain time delays in processing or communication. These problems are only amplified in the case of actuator failures, where the aircraft has lost some of its control effectiveness. Additionally, the dynamics of quadrotors are nonlinear and multivariate. There are several effects to which a potential controller must be robust: the aerodynamics of the rotor blades (propeller and blade flapping), inertial anti-torques (asymmetric angular speed of propellers), as well as gyroscopic effects (change in orientation of the quadrotor and the plane of the propeller).

The redundancy in the actuators of a quadrotor makes them robust towards a set of partial failures. Though the performance and maneuverability will most likely be reduced in the case of such a failure, it is desirable for a controller to stabilize the system and allow for reduced mode operations such as a safe return, stable hover, *etc.* Adaptive control is an attractive candidate for this aircraft because of its ability to generate high performance tracking in the presence of parametric uncertainties.

This chapter is organized as follows. Section 3.2 describes the dynamics of the quadrotor and the development of the adaptive controller. Section 3.3 shows preliminary simulation results that suggest adaptive control may be effective for this problem. Section 3.4 describes the experimental setup and existing control architecture. Some of the finer points of the implementation, including robustness modifications to deal with noisy signals and selection of the adaptation rates are described in section 3.5. Flight test results are shown in section 3.6 and conclusions are given in section 3.7.

3.2 Adaptive controller design for quadrotor helicopters

In this section we examine the dynamics of the quadrotor helicopter in order to gain the insight necessary for the adaptive control design. In particular, the reference model used by the model reference adaptive control approach is generated using linearized quadrotor dynamics and a linear baseline controller. The adaptive controller is then formulated for the problem of command tracking in the presence of parametric uncertainties in the form of actuator failures, where one or more of the four propellers loses a portion of its thrust. This type of uncertainty can be attributed to electrical component failure or physical damage.

3.2.1 Quadrotor dynamics

The dynamics of quadrotor helicopters have been studied in detail by several groups [52, 53]. A simple, rigid-body model of the quadrotor which assumes low speeds is given by:

$$
\begin{aligned}
\ddot{x} &= \left(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi\right)\frac{U_1}{m},\\
\ddot{y} &= \left(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi\right)\frac{U_1}{m},\\
\ddot{z} &= -g + \left(\cos\phi\cos\theta\right)\frac{U_1}{m},\\
\\
\ddot{\phi} &= \dot{\theta}\dot{\psi}\left(\frac{I_y - I_z}{I_x}\right) - \frac{J_R}{I_x}\dot{\theta}\Omega_R + \frac{L}{I_x}U_2,\\
\ddot{\theta} &= \dot{\phi}\dot{\psi}\left(\frac{I_z - I_x}{I_y}\right) + \frac{J_R}{I_y}\dot{\phi}\Omega_R + \frac{L}{I_y}U_3,\\
\ddot{\psi} &= \dot{\phi}\dot{\theta}\left(\frac{I_x - I_y}{I_z}\right) + \frac{1}{I_z}U_4,
\end{aligned}
\tag{3.1}
$$

where x, y, and z are the position of the center of mass in the inertial frame; ϕ, θ, and ψ are the Euler angles which describe the orientation of the body-fixed frame with respect to the inertial frame; m, I_x, I_y, and I_z are the mass and moments of inertia of the quadrotor respectively; and J_R and Ω_R are the moment of inertia and angular velocity of the propeller blades. U_1, U_2, U_3, and U_4 are the collective, roll, pitch, and yaw forces generated by the four propellers. Since the quadrotor typically operates very near the hover position, we can make small angle approximations, neglect higher order terms and let $U_1 = mg + \Delta U_1$, resulting in the linear dynamics

$$\begin{aligned}
\ddot{x} &= g\theta, \\
\ddot{y} &= -g\phi, \\
\ddot{z} &= \frac{\Delta U_1}{m}, \\
\\
\ddot{\phi} &= \frac{L}{I_x}U_2, \\
\ddot{\theta} &= \frac{L}{I_y}U_3, \\
\ddot{\psi} &= \frac{1}{I_z}U_4.
\end{aligned} \tag{3.2}$$

Although this system is quite a bit more simple than the nonlinear dynamics in (3.1), it still captures the dominant features of the quadrotor, and is accurate near the hover position. As expected, the roll, pitch and yaw inputs command moments about their respective axes and the collective input commands acceleration in the positive z-direction. Accelerations in the x- and y-directions are achieved primarily through vectoring the collective thrust. This simplified model also sheds some light on a time-scale separation present in the system. It can be seen that the dynamics of z, ϕ, θ, and ψ are double integrators, while the dynamics of x and y are quadruple integrators. The former group can be thought of as "fast" states, or as the vehicle dynamics while the latter group can be thought of as "slow" states, or the vehicle kinematics. For the purposes of control design, we can use the linear dynamics given by (3.2).

3.2.2 Problem statement

The primary function of the adaptive controller is to accommodate any uncertainties which mat arise in the dynamics (3.2). We can write the equations of motion in (3.2) along with these uncertainties as

$$\dot{x}_p = A_p x_p + B_p \Lambda u, \tag{3.3}$$

where $B_p \in \Re^{nxm}$ is constant and *known*, $A_p \in \Re^{nxn}$ is constant and *unknown*, $x_p \in \Re^n_p$, $u \in \Re^m$, $\Lambda \in \Re^{mxm}$ is an unknown diagonal positive definite constant matrix with diagonal elements $\Re \in (0, 1]$. The goal is to track a reference command $r(t)$ in the presence of the unknown A_p, and Λ. We define the system output as

$$y_p = C_p x_p. \tag{3.4}$$

In the case of the quadrotor, the output states are x, y, z, and ψ. The output tracking error is then given by

$$e_y = y_p - r. \tag{3.5}$$

Augmenting (3.3) with the integrated output tracking error

$$\dot{e}_{y_I} = e_y, \tag{3.6}$$

leads to the extended open loop dynamics

$$\dot{x}_t = A_t x_t + B_t \Lambda u + B_c r, \tag{3.7}$$

where $x_t = \begin{bmatrix} x_p^T & e_{y_I}^T \end{bmatrix}^T$ is the extended system state vector. The extended open-loop system matrices are given by

$$A_t = \begin{bmatrix} A_p & 0 \\ C_p & 0 \end{bmatrix}, \; B_t = \begin{bmatrix} B_p \\ 0 \end{bmatrix}, \; B_c = \begin{bmatrix} 0 \\ -I \end{bmatrix}, \tag{3.8}$$

and the extended system output

$$y_t = \begin{bmatrix} C_p & 0 \end{bmatrix} x_t = C_t x_t. \tag{3.9}$$

3.2.3 Reference model

A baseline controller

$$u_{bl} = K_x x_t, \tag{3.10}$$

can be designed for the system in (3.7) in the case where there is no uncertainty, that is $\Lambda = I^{nxn}$. The feedback gains K_x can be selected using LQR or classical design techniques. The reference model used by MRAC is the closed loop system given by (3.7), again in the case of no uncertainty, along with the control input in (3.10)

$$\dot{x}_m = A_t x_m + B_t u_{bl} + B_c r = A_m x_m + B_c r. \tag{3.11}$$

3.2.4 Adaptive controller

An adaptive control input is added to the baseline controller as

$$u_{ad} = \hat{K}_x^T x_t + \hat{\theta}_r^T r + \hat{\theta}_d = \hat{\theta}^T \omega, \tag{3.12}$$

where $\hat{\theta}^T = \begin{bmatrix} \hat{K}_x^T & \hat{\theta}_r^T & \hat{\theta}_d^T \end{bmatrix}$ are time-varying adaptive parameters that will be adjusted in the adaptive law given in (3.14) below and $\omega^T = \begin{bmatrix} x_t^T & r^T & 1 \end{bmatrix}$ is the regressor vector. The overall control input is thus

$$u = u_{ad} + u_{nom} = \hat{\theta}^T \omega + K_x x_t + r. \tag{3.13}$$

The canonical adaptive law is given by

$$\dot{\hat{\theta}} = -\Gamma \omega e^T P B_t, \tag{3.14}$$

where Γ is a diagonal positive definite matrix of adaptive gains, $e = x - x_m$ is the model tracking error, and P is the unique symmetric positive definite solution of the Lyapunov equation,

$$A_m^T P + P A_m = -Q, \tag{3.15}$$

where Q is also symmetric positive definite. This adaptive controller is based on nonlinear stability theory, the details of which have been touched on by many authors throughout the years [7, 8, 17, 25, 69, 70]. The augmented structure of the adaptive controller implies that in the nominal case, that is the case with no parameter uncertainty, the overall system is equivalent to the baseline control. However, when failures or other uncertainties arise, the adaptive controller works to assist the baseline controller in maintaining stability and performance.

With the Lyapunov function candidate given by

$$V = e^T P e + \operatorname{Tr}\left(\tilde{\theta}^T \Gamma^{-1} \tilde{\theta}\right), \tag{3.16}$$

where $\tilde{\theta} = \hat{\theta} - \theta$ is the parameter estimation error, it can be shown [7] that the derivative of the Lyapunov function candidate is given by

$$\dot{V} = -e^T Q e \leq 0. \tag{3.17}$$

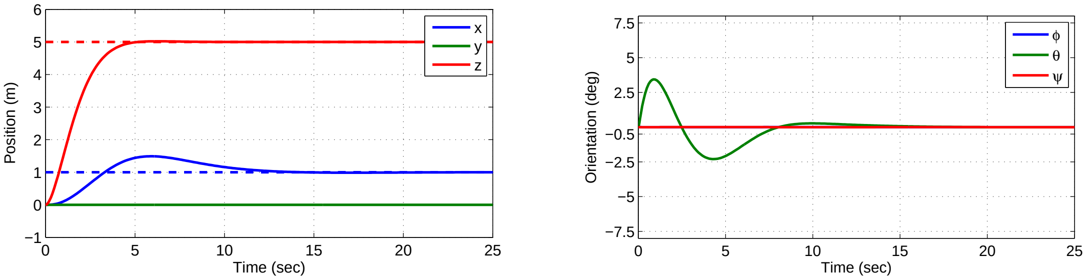
The system is globally asymptotically stable by Barbalat's lemma and the tracking error asymptotically converges to 0, that is

$$\lim_{t \to \infty} e(t) = 0. \tag{3.18}$$

3.3 Simulation results

A quadrotor simulation was used to judge the efficacy of the adaptive approach before transitioning to hardware. The simulated dynamics is chosen as the nonlinear quadrotor dynamics given in (3.1). Actuator saturation limits are also included. Sensor dynamics, sensor noise, and time delay are neglected for this preliminary simulation study. In this series of tests, the quadrotor is commanded to take off and climb to an altitude of 5 m while simultaneously translating 1 m in the x-direction. Figure 3-1 shows the position and orientation of the simulated quadrotor executing the maneuver in the nominal case, that is, in the case of no uncertainty.

We now include uncertainty in the form of the loss of control effectiveness Λ. The type of failure, the time at which the failure occurs, and the extent of the failure are unknown to the controller. For this simulation we arbitrarily select an 80% loss-of-control-effectiveness failure that is initiated in one actuator at $t = 15$ sec. The



(a) **(b)**

Figure 3-1: Simulated quadrotor (a) position and (b) orientation for the nominal case. Dashed lines represent the commanded positions.

tracking performance of the baseline controller described in (3.10) and the adaptive controller described in (3.12)-(3.15) is compared in figure 3-2. Figure 3-2(a)-(b) shows the position and orientation for the baseline controller and figure 3-2(c)-(d) shows the position and orientation for the adaptive controller in the failure case.

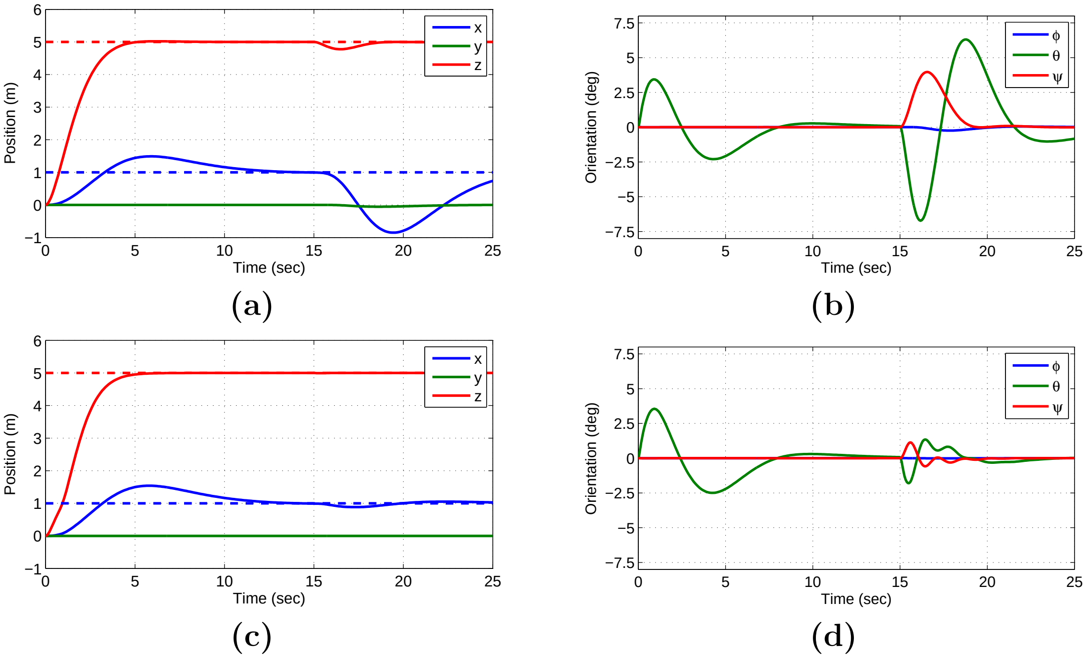


Figure 3-2: Simulated quadrotor (a) position and (b) orientation for the baseline controller and (c) position and (d) orientation for the adaptive controller. An 80% loss-of-control-effectiveness failure occurs in one actuator at $t = 15$ sec.

The adaptive controller responds quickly and effectively to the change in dynamics, keeping the roll and pitch angles less than 2∘ from the hover position. On the other hand, the baseline controller experiences excursions of over 6◦ in pitch angle and 4◦ in roll angle. These rotations vector the collective thrust, causing the vehicle to deviate nearly 2 m from the commanded position. Both controllers retain stability and the baseline controller does eventually return to the commanded position. However, the large departure from the desired position seen in figure 3-2(a) may be undesirable, especially if the quadrotor is operating in a cluttered environment with many walls, obstacles, or other UAVs.

These results suggest that the adaptive controller may be effective in mitigating the effects of partial failures such as a sudden loss of control effectiveness. The following sections detail the application of a similar adaptive control design to the hardware platform.

3.4 Experimental setup

Flight testing was done in collaboration with the Aerospace Controls Laboratory at MIT, which utilizes a UAV testbed facility known as RAVEN (Real-time indoor Autonomous Vehicle test ENvironment). RAVEN uses a Vicon motion capture system to enable rapid prototyping of aerobatic flight controllers for helicopters and aircraft; robust coordination algorithms for multiple helicopters; and vision-based sensing algorithms for indoor flight [55–57]. The Aerospace Controls Laboratory also operates several Draganflyer quadrotors autonomously using a simple, ground-based controller. The RAVEN system essentially consists of a vision-based motion capture system that provides 6DoF pose parameters at fixed frequency of 100Hz. Commands are sent to the quadrotors using a USB wireless remote-control module. The system operates on a cluster of Linux workstations, with each node assigned to the control of an individual UAV. This system is used to run the vehicle flight controller and other path planning algorithms. The adaptive controller described in section 3.2 is then implemented in C++ and run alongside the fixed-gain controller. This system allows for flight testing of the quadrotor using both baseline (fixed-gain) and adaptive controllers.

Figure 3-3: Draganflyer four rotor helicopter UAVs in the Real-time indoor Autonomous Vehicle test ENvironment (RAVEN).

3.5 Implementation issues

With the capabilities and limitations of RAVEN in mind, a number of modifications need to be made to the typical model reference adaptive approach. Due to sensor and process noise in the system, it was found that a dead-zone modification was required to avoid parameter drift. Instead of adapting to the error e, we adapt to the modified error e_{dz}, which is given by

$$e_{dz} = \begin{cases} e + dz, & \text{if} \quad e < -dz, \\ 0, & \text{if} \quad -dz < e < dz, \\ e - dz, & \text{if} \quad e > dz. \end{cases} \tag{3.19}$$

After examining several minutes worth of flight data, the size of the dead-zones dz was selected to be just larger than the maximum noise value present in the system. This approach mitigates parameter drift without having significant effect on the performance of the adaptive controller.

The adaptive law given in (3.14) was also modified to include the projection operator, which provides bounds on the adaptive parameters. The projection operator is defined column-wise for the j^{th} column as

$$\text{Proj}(\hat{\theta}_j, y_j) = \begin{cases} y_j - \frac{\nabla f(\hat{\theta}_j)\left(\nabla f(\hat{\theta}_j)\right)^T}{||\nabla f(\hat{\theta}_j)||^2} y_j f(\hat{\theta}_j), & \text{if } f(\hat{\theta}_j) > 0,\ y_j^T \nabla f(\hat{\theta}_j) > 0, \\ y_j, & \text{otherwise,} \end{cases} \tag{3.20}$$

where the convex function f is given as

$$f(\hat{\theta}_j) = \frac{||\hat{\theta}_j||^2 - \left(\theta_j^{\max}\right)^2}{\epsilon_j \left(\theta_j^{\max}\right)^2}, \tag{3.21}$$

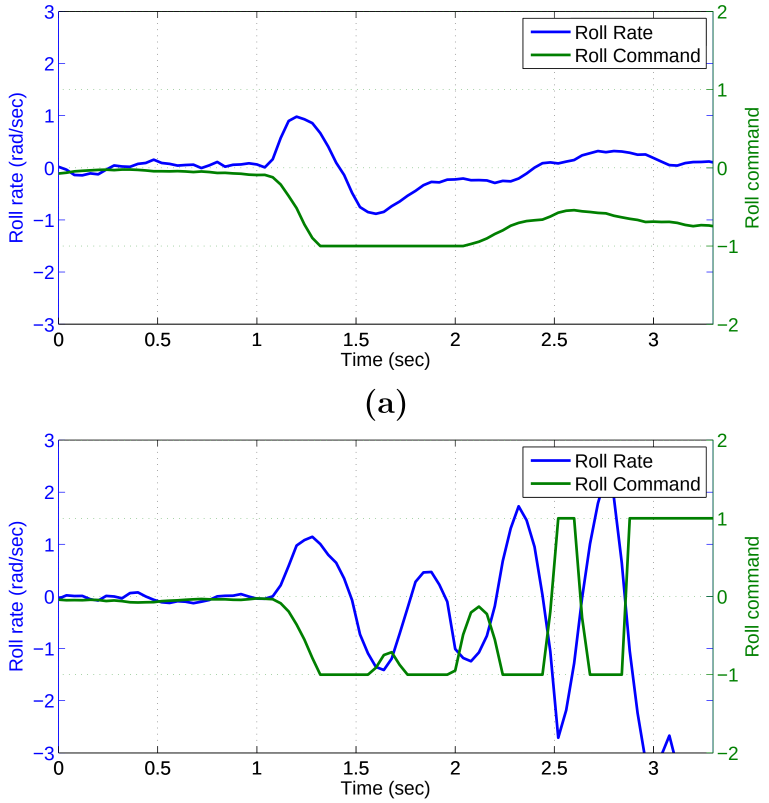
where ϵ_j is a projection tolerance. The new adaptive law is then given column-wise by

$$\dot{\hat{\theta}}_j = -\Gamma \text{Proj}(\hat{\theta}_j, (\omega e_{dz}^T P B)_j). \tag{3.22}$$

This ensures that the adaptive parameters $\hat{\theta}$ are less than some prescribed bound $\theta^{\max}$. Along with the dead-zone modification, this changes the stability result for the adaptive controller from a global to a semi-global result.

It was also found that frame misalignment could result in measured error terms being centered around a non-zero value, even in the nominal (no-failure) case. This results in the dead-zones mentioned above to be slightly off-center themselves, which can cause parameters to drift. A 10 sec period of hover is sufficient to determine the actual hover attitude, which is then used to adjust the error terms so that they are centered at 0. This is equivalent to slightly adjusting the point about which the system is linearized, and does not have any effect on the stability result.

Another important implementation issue is the selection of the adaptive gains. According to the theory described in section 3.2.4, any choice of adaptive gains results in the same stability result. In practice, however, it was found that nonlinearities, unmodeled dynamics, actuator saturation, and particularly time delay caused instabilities at higher adaptive gains such as the one shown in figure 3-4. Figure 3-4(a) shows flight-recorded data of the response of the roll rate and roll command with a 45% loss of actuator effectiveness in one motor. Figure 3-4(b) shows the same response, only with adaptive gains increased by 25%. Through some combination of actuator saturation and time delay, the signals become completely out-of-phase and the system becomes unstable, causing an eventual crash.



(a)

(b)

Figure 3-4: Flight-recorded response to a 45% loss of actuator effectiveness for the adaptive controller with (a) nominal adaptive gains, (b) adaptive gains increased 25% over nominal.

To select the adaptive gains, we employed the following empirical formula, which arises from inspection of the structure of the adaptive laws [11]

$$\Gamma = \left| \frac{\text{diag}(\vartheta)}{\tau_{min} p r_{max}} \right| + \Gamma_0, \tag{3.23}$$

where

i. $\vartheta \in \Re^n$ is vector given by the sum of the columns of $\bar{\theta}^*$ where $\bar{\theta}^*$ corresponds to the uncertainty $\Lambda = \bar{\Lambda}$ for which the plant has the most unstable eigenvalues.

The components of $\bar{\theta}^*$ are given by

$$\begin{aligned}
\bar{K}_x^* &= -K_x\bar{\Lambda}^{-1},\\
\bar{\theta}_r^* &= \bar{\Lambda}^{-1}, \qquad (3.24)\\
\bar{\theta}_d^* &= 1_{1\text{x}m}.
\end{aligned}$$

ii. τ_{min} is the smallest time constant of the reference model,

iii. p is the norm of $B^T P$,

iv. r_{max} is the maximum amplitude of the reference input signal,

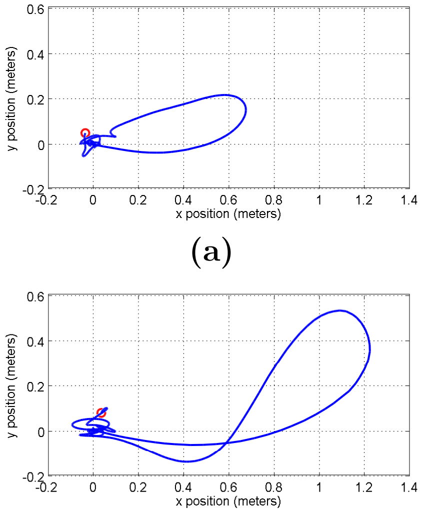
v. Γ_0 is a small positive definite diagonal matrix which ensures that Γ remains positive definite.

The resulting adaptive gains were then manually tuned to achieve desired response in each axis.

3.6 Flight test results

In the following series of flight tests, the quadrotor UAV is commanded to hover in a fixed position. A simulated loss of control effectiveness is then injected into the system and the resulting performance is compared using both the baseline (fixed-gain) controller and the augmented adaptive controller.

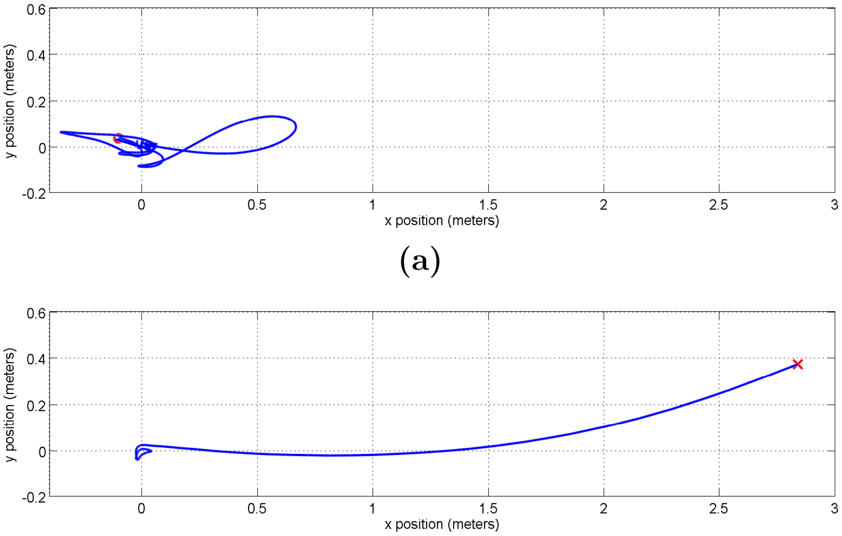
Figure 3-5 shows a birds-eye view of the trajectory traced by each controller and the deviation from the commanded hover position, here marked as (0,0). As the rotor in the positive x-direction loses 45% of its thrust, the resulting moment about the y-axis causes the quadrotor to increase its pitch angle. This has the effect of vectoring the collective thrust, causing the UAV to accelerate in the positive x-direction. This failure also results in a loss of altitude as well as a yaw moment. All four rotors must work quickly and effectively to overcome the sudden change in the dynamics due to the loss of thrust. Because the baseline controller contains integrators on the x- and y-axes, it eventually does return to the commanded position. The adaptive



(a)

(b)

Figure 3-5: Birds-eye view of the quadrotor trajectory tracking for the (a) adaptive and (b) baseline controllers following a 45% loss of thrust in one actuator. The initial position of the quadrotor is with the failed actuator pointing in the positive x-direction. The red circles denote the quadrotor position at the end of the test period.



(a)

(b)

Figure 3-6: Birds-eye view of the quadrotor trajectory tracking for the (a) adaptive and (b) baseline controllers following a 50% loss of thrust in one actuator. The initial position of the quadrotor is again with the failed actuator pointing in the positive x-direction. In the adaptive control case, the red circle denotes the quadrotor position at the end of the test period. In the baseline control case, the red × denotes the quadrotor position at the time when the vehicle kill switch was thrown to prevent collision with the wall.

controller is faster to react to the change in dynamics and returns to within 15 cm of the commanded position in 20% less time than the baseline controller. The maximal radius of departure for the adaptive controller is 46% less than that of the baseline controller. This can be a critical difference in the case of formation flying, swarm operations, or when operating in a cluttered environment.

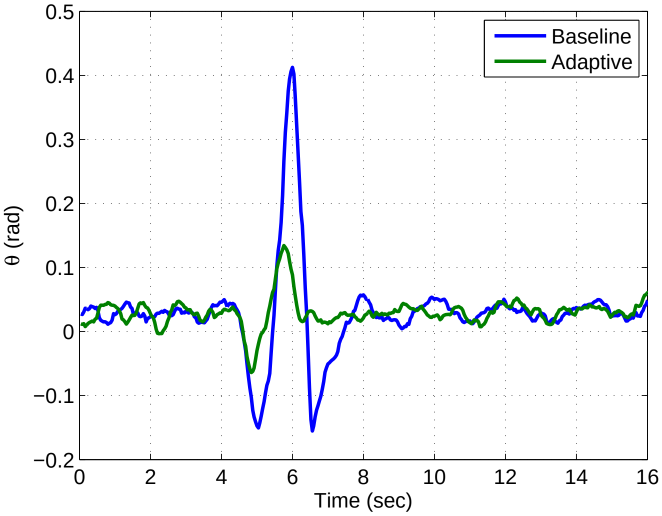


Figure 3-7: Pitch angle comparison of baseline and adaptive controllers. A 45% loss of thrust failure occurs in one propeller at $t = 4$. The adaptive controller exhibits significantly less deviation from level flight.

In figure 3-6, the same flight test is performed with a more severe, 50% loss of thrust failure. In this example, the baseline controller was not sufficient to prevent a crash. The UAV was accelerating toward the wall of the indoor testbed when the kill switch was thrown. If the quadrotor had sufficient time and space, it is possible that it may have recovered from this particular failure. However, a departure radius of over 3 meters is unacceptable for many applications.

Figure 3-7 shows the pitch angle for the maneuver shown in figure 3-5. The adaptive controller quickly corrects for the change in dynamics before the pitch angle becomes excessive, thus staying closer to the linear operating condition for which the baseline controller is designed.

However, the improved performance comes at a cost of increased control effort. In particular, if we examine the FFT of the control signals for both the baseline and

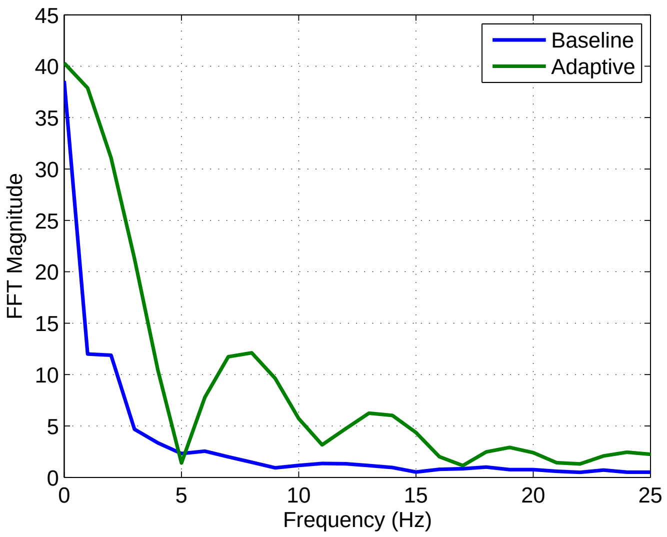


Figure 3-8: FFT of baseline and baseline+adaptive control signals during the 4 second interval after the failure occurs.

adaptive controllers during the first second after the failure occurs, we can see that the adaptive controller is using significantly more control power. Figure 3-8 shows this comparison. For this application, this level of control usage is acceptable. However, under a different set of constraints, this level of controller operation may be excessive. Excessive control effort may cause increased battery drain, leading to shorter flight times, and additional wear on the actuator, leading to premature failure.

A further set of tests involved a demonstration of a "real-world" failure. These flight tests were performed at the Boeing Vehicle Swarm Technology Laboratory in a setup similar to that of RAVEN. Instead of injecting the failure through artificially reducing the thrust of one propeller, the propeller itself was cut mid-flight. To accomplish this, a propeller was modified pre-flight by slicing both tips (approximately 25% of the propeller radius on each side) and then taping them back together using masking tape. During flight a small razor blade, mounted to a radio-controlled servo motor and attached to the quadrotor frame, can be remotely actuated into the path of the spinning blade, cutting the tape. This results in approximately a 40% loss of thrust. This experimental setup is shown in figure 3-9.

This approach allows for repeatable experiments as the blade tips can be collected and reattached to the propeller. Thus baseline and adaptive controllers can be com-

Figure 3-9: Photo of the modified propeller blade (white) with tips removed. The razor blade and servo motor can be seen attached to the quadrotor frame.

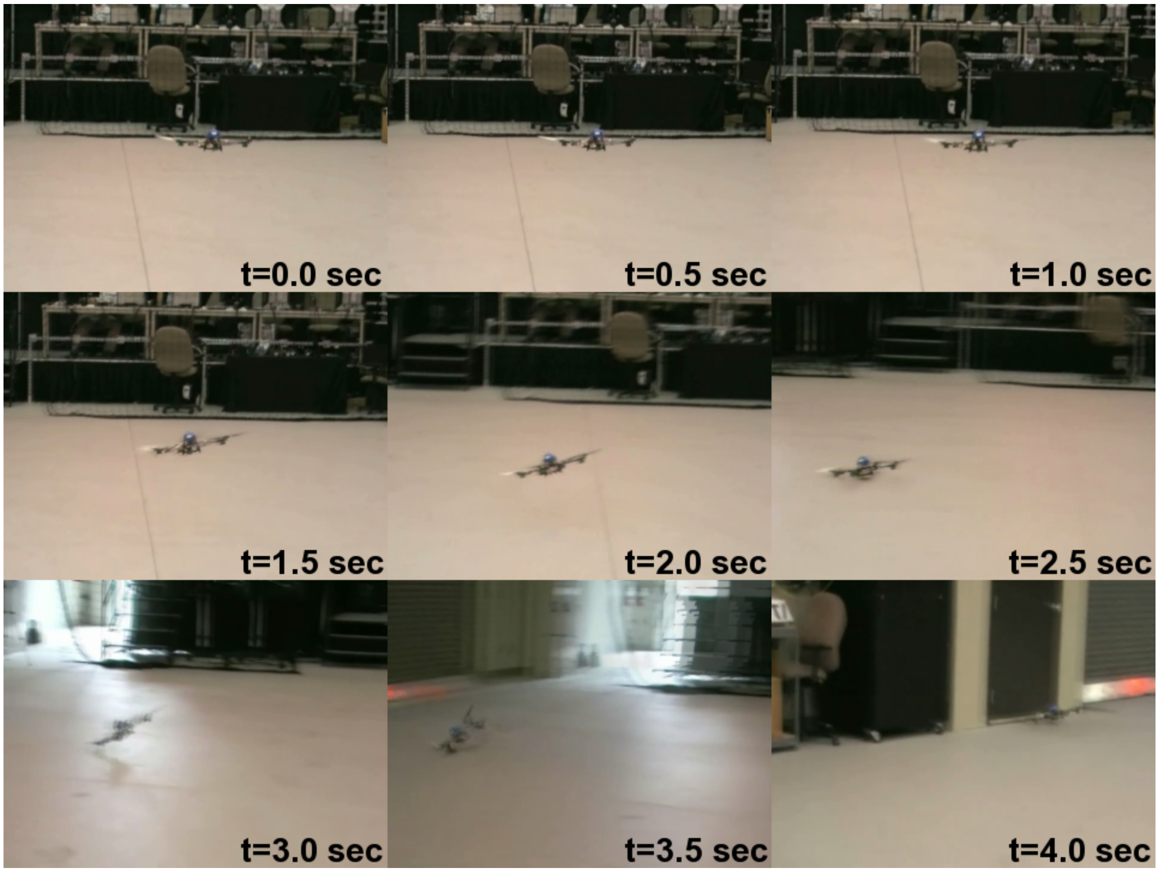


Figure 3-10: Screen captures of the blade-cutting flight test with the linear baseline controller. Time indices are given in the bottom right corner of each frame. The failure occurs between $t = 1.0$ sec and $t = 1.5$ sec.

pared easily. Videos of the flight test results have been posted online [71]. Screen captures of these videos can be seen in figures 3-10 and 3-11.

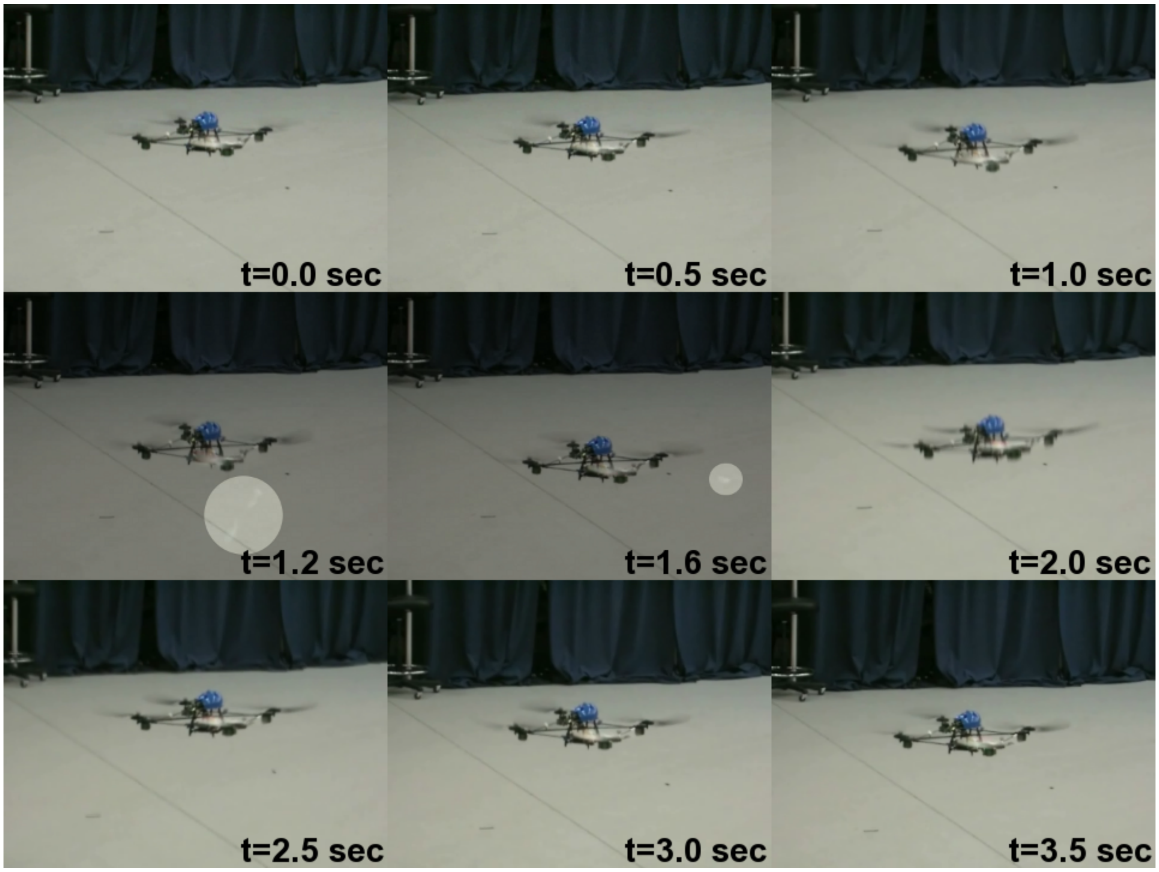


Figure 3-11: Screen captures of the blade-cutting flight test with the adaptive controller. Time indices are given in the bottom right corner of each frame. At $t = 1.2$ sec and $t = 1.6$ sec the liberated propeller tips are highlighted as they move quickly out of frame.

The goal of the maneuver shown in figures 3-10 and 3-11 is to remain in a hover position. The baseline controller is a well-designed, robust linear controller, however, it is unable to recover from the sudden change in dynamics and the vehicle crashes into the floor and a nearby wall. On the other hand, the adaptive controller quickly accounts for the loss of thrust and adjusts online, allowing for safe return to hover and safe landing. Figure 3-11 also shows the precise moments when the two propeller tips fly off screen.

3.7 Summary

In this chapter, a description of an adaptive controller based on Lyapunov stability and its application to a quadrotor UAV was presented. Flight testing was carried out in an indoor test facility using both baseline and model reference adaptive controllers. It was shown that the adaptive controller offers several benefits over the existing fixed-gain approach, particularly in the case of actuator failures. For less severe failures, the adaptive controller was faster in reacting to changes in dynamics, resulting in a decreased radius of departure. For more severe failures, the adaptive controller prevented a crash and allowed for safe operation and landing. It was found that the adaptive controller used more control effort during the period directly following the insertion of the failure. It was also found that increasing adaptive gains beyond a certain value resulted in undesirable oscillations and even instability. In the following chapters, methods for getting increased performance without increasing control power and without undesirable oscillations will be examined.

Chapter 4

General systems with parametric uncertainty and time delay

In this chapter, we temporarily turn away from particular application platforms and towards general systems. In particular, systems which have both parametric uncertainties that necessitate a nonlinear approach such as adaptive control and time delays. The adaptive system is intentionally nonlinear and therefore classical notions of gain and phase margins do not apply. Furthermore, since the verification and validation processes typically used by the aerospace community were designed around linear principles, a new set of tools must be developed to certify adaptive controllers, especially in the presence of time delays.

This chapter is divided into two sections. In the first section we examine a describing functions approach for approximating a nonlinear adaptive controller with a linear system. This allows for the use of linear tools such as gain and phase margins. The second section details the development of a novel adaptive controller designed specifically to deal with known time delay in a system. The approach has a proof of signal boundedness using a Lyapunov approach and a time delay margin that is bounded away from 0. The controller is validated using a simulation of F-16 short-period dynamics.

4.1 Describing functions approach

This section describes an alternate approach to determining time delay margin for an adaptive system whereby we approximate linear-in-state terms in MRAC by a series of step functions [72]. In this way, the nonlinear adaptive system is estimated by a closed-loop system that is linear at any given time-instant. The overall dynamics, however, remain nonlinear. The nonlinearities have merely been encoded in the form of switching. The resulting switched linear system can then be further approximated using the describing functions method [9, 48]. The resulting closed-loop system is then a linear approximation of the adaptive system. Linear tools such as the extended Nyquist criterion can then be used to examine the behavior of the approximate system.

4.1.1 Problem statement

Consider a scalar plant of the form

$$\dot{x} = a_m x + \theta\pi\left(\frac{x-C}{\sigma}\right) + u, \tag{4.1}$$

where $x \in \Re$ is the system state, $u \in \Re$ is the control input, $a_m < 0$ is the known plant, is θ is an unknown constant parameter, and $\pi(x)$ is the so-called $\pi -$ function, centered at the origin and with unity width.

$$\pi\left(\frac{x-C}{\sigma}\right) = \begin{cases} 1, & |\left(\frac{x-C}{\sigma}\right)| \leq 0.5, \\ 0, & |\left(\frac{x-C}{\sigma}\right)| > 0.5. \end{cases} \tag{4.2}$$

Since the π-function is discontinuous in x and hence not Lipschitz, the existence of a solution for the ordinary differential equation (4.1) is not guaranteed in the classical sense. Therefore we will assume that solutions of (4.1) do exist and are defined in the sense of Filipov [**?**]. We define the reference model to be

$$\dot{x}_m = a_m x + b_m r. \tag{4.3}$$

We can then generate an approximate model reference adaptive controller with the linear-in-state terms replaced by these π-functions. Let the control input be

$$u = -\hat{\theta}\pi\left(\frac{x - C}{\sigma}\right) + b_m r, \tag{4.4}$$

where the parameter estimate $\hat{\theta}$ is adjusted online according to the adaptive law

$$\hat{\theta} = \gamma \int_{t_0}^{t} \left[\pi\left(\frac{x(\tau) - C}{\sigma}\right)(x(\tau) - x_m(\tau))\right] d\tau, \tag{4.5}$$

where $\gamma > 0$ is an adaptation rate. Let the model reference tracking error $e = x - x_m$ and the parameter error be $\tilde{\theta} = \hat{\theta} - \theta$. We can then write the error dynamics for the system as

$$\dot{e} = a_m e - \tilde{\theta}\pi\left(\frac{x - C}{\sigma}\right). \tag{4.6}$$

Consider the Lyapunov function candidate

$$V = \frac{1}{2}\left(e^2 + \frac{1}{\gamma}\tilde{\theta}^2\right). \tag{4.7}$$

Substituting (4.6) and (4.5), the derivative of the Lyapunov function candidate is then

$$\dot{V} = a_m e^2 - \tilde{\theta}\pi\left(\frac{x(\tau) - C}{\sigma}\right)e + \tilde{\theta}\pi\left(\frac{x(\tau) - C}{\sigma}\right)e. \tag{4.8}$$

Therefore

$$\dot{V} = a_m e^2 \leq 0. \tag{4.9}$$

Using Barbalat's lemma one can show global, asymptotic convergence of the tracking error to 0, that is

$$\lim_{t\to\infty} e(t) = 0. \tag{4.10}$$

The controller described in (4.4)-(4.5) is thus an approximation of the MRAC approach, with a global, asymptotic stability result identical to that of MRAC. Although the treatment above only considers the single π-function case, the same proof applies to a system with arbitrarily many π-functions with widths and centers selected to

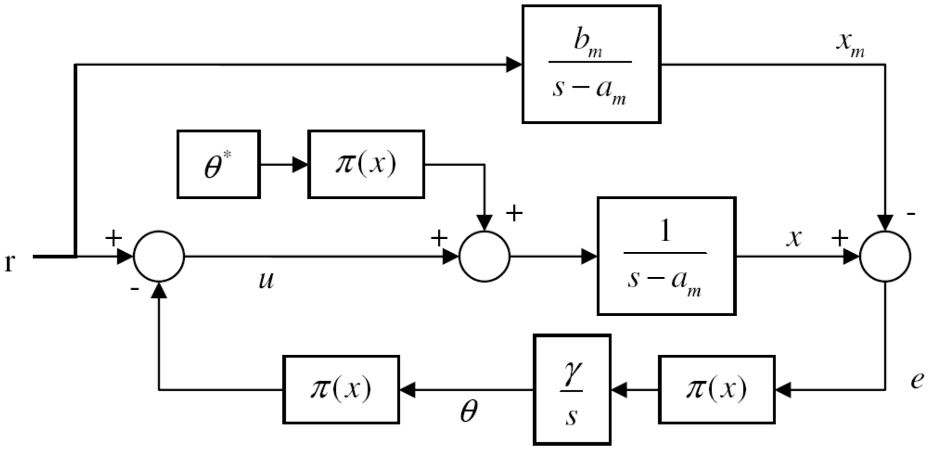


Figure 4-1: Block diagram for the single π-function scalar problem.

cover some region of interest. As the number of π-functions increases, this system becomes a step-wise approximation of MRAC. However the resulting system is still a nonlinear system and further approximations must be made in order to use classical design tools such as gain and phase margins.

4.1.2 Describing function approximation

In the following sections, we aim to use the describing functions approach to approximate the nonlinear system, thereby enabling the use of linear analysis techniques such as the extended Nyquist criterion to determine margins of the system. We only consider the single π-function case; extensions to the multiple π-function case are left as future work.

A block diagram of the closed-loop system described in (4.1)-(4.5) is shown in figure 4-1. It should be noted that the $\pi(x)$ blocks do not represent π-functions operating on the input to the block, but instead multiplication of the input with $\pi(x)$. For this example, we are considering a single π-function centered at 0, that is $C = 0$. For the purposes of the margins discussion we can ignore the input as well as the θ^* term, which appears in the block diagram as a disturbance. A block diagram containing those simplifications is shown in figure 4-2 It can be seen that all

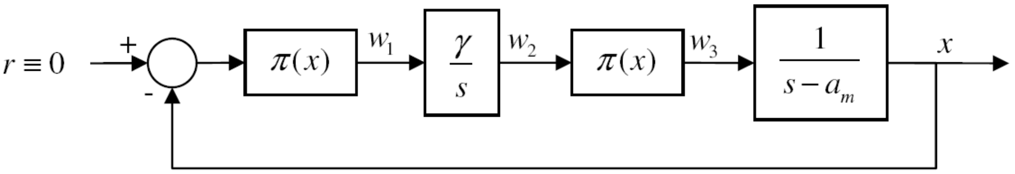


Figure 4-2: Simplified block diagram for the single π-function scalar problem.

the nonlinearity is contained within the leftmost three blocks. We can now attempt to approximate these blocks using the describing function approach. First assume that x is of the form $-A\sin(\omega t)$, and hence the input to the forward path is of the form $A\sin(\omega t)$ as shown in figure 4-3(a).

Noting that $\pi(-x) = \pi(x)$, it is clear that the π-function will be unity between the dotted lines and zero otherwise in figure 4-3(a). If $A \leq \frac{\sigma}{2}$, there is no nonlinearity and the system is stable with easily determinable margins. However, if $A > \frac{\sigma}{2}$, then w_1 will take the form shown in figure 4-3(b). The w_1 signal is then integrated, leading to the signal w_2, shown in figure 4-3(c). Finally, the signal is multiplied by the $\pi(x)$ once again, resulting in the signal w_3, shown in figure 4-3(d).

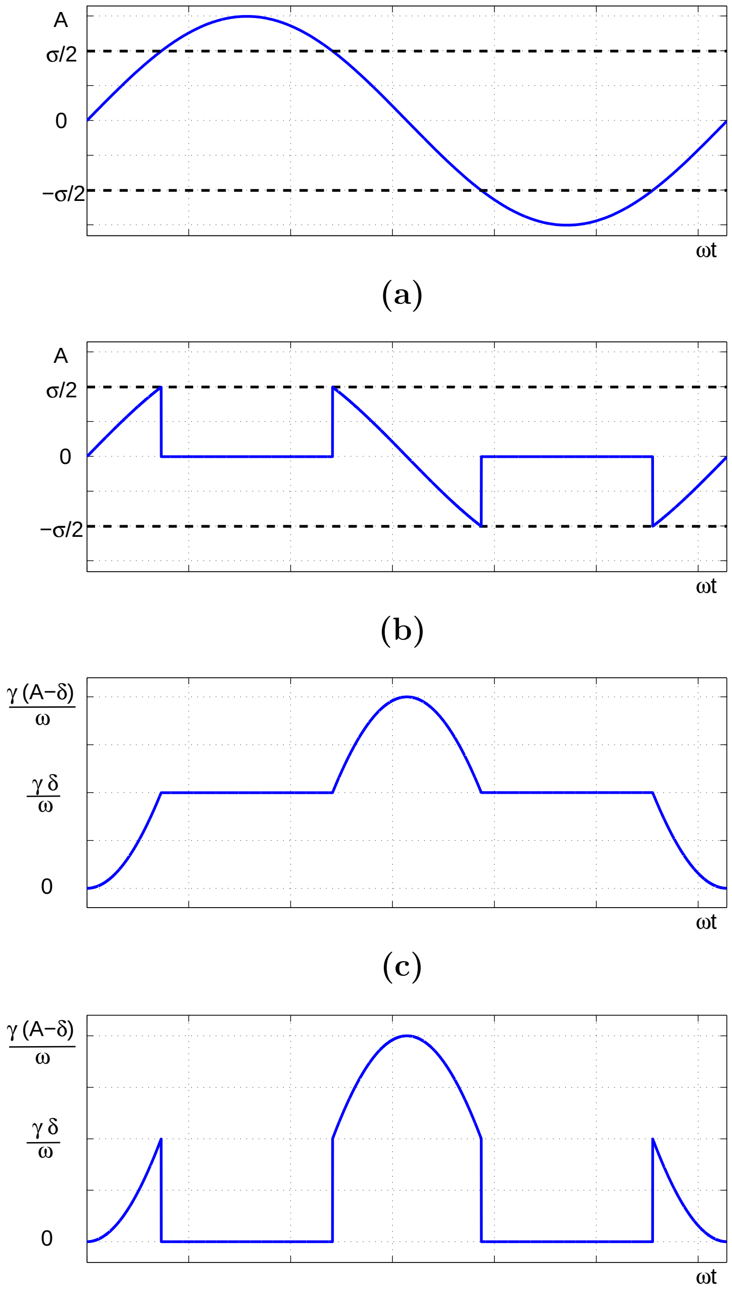
The output of the nonlinearity can be written as

$$w_3(t) = \begin{cases} -\frac{\gamma}{\omega}\left(A\cos(\omega t) - \delta\right), & \text{if} \quad \pi(2k+1) - \bar{\omega} \leq \omega t \leq \pi(2k+1) + \bar{\omega}, \\ -\frac{\gamma}{\omega}\left(A\cos(\omega t) + \delta\right), & \text{if} \quad 2\pi k - \bar{\omega} \leq \omega t \leq 2\pi k + \bar{\omega}, \\ 0, & \text{otherwise}, \end{cases} \tag{4.11}$$

where $\bar{\omega} = \sin^{-1}\left(\frac{\sigma/2}{A}\right)$ and the bias term δ is given by

$$\delta = \int_{\bar{\omega}}^{\pi/2} A\sin(\omega t)d(\omega t) = \sqrt{A^2 + \frac{\sigma^2}{4}}. \tag{4.12}$$

It should be noted that w_3 is an even function, not odd as required by the describing function approach. Therefore, we introduce a derivative and integrator pair into the block diagram as shown in figure 4-4. The signal w_4 is thus odd, and given



(a) **(b)** **(c)** **(d)**

Figure 4-3: Signals in the forward path: (a) $-x$, input to the nonlinearity, (b) w_1, input to the integrator, (c) w_2, input to the second π-function, (d) w_3, output of the nonlinearity.

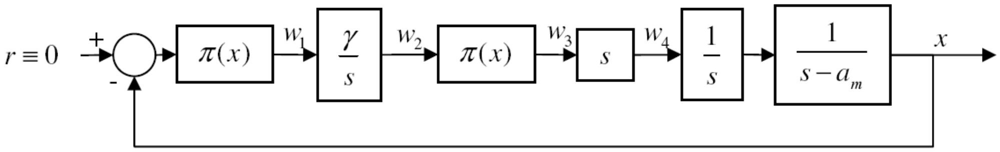


Figure 4-4: Simplified block diagram for the single π-function scalar problem, including an extra derivative / integrator pair.

by

$$w_4(t) = \begin{cases} \gamma\left(A\sin(\omega t)\right), & \pi k - \bar{\omega} < \omega t < \pi k + \bar{\omega}, \\ -\infty, & \omega t = \pi k - \bar{\omega}, \\ \infty, & \omega t = \pi k + \bar{\omega}, \\ 0, & \text{otherwise.} \end{cases} \tag{4.13}$$

We can therefore continue with the describing functions approach by calculating the first Fourier coefficient

$$\begin{aligned} b1 &= \frac{2}{\pi}\int_0^{\pi} w_4(t)\sin(\omega t)d(\omega t) \\ &= \frac{2}{\pi}\left[\int_0^{\bar{\omega}} \frac{\gamma}{\omega^2}A\sin^2(\omega t)d(\omega t) - \frac{\sigma/2}{A} + \frac{\sigma/2}{A} + \int_{\pi-\bar{\omega}}^{\pi} \frac{\gamma}{\omega^2}A\sin^2(\omega t)d(\omega t)\right] \\ &= \frac{4}{\pi}\int_0^{\bar{\omega}} \frac{\gamma}{\omega^2}A\sin^2(\omega t)d(\omega t) = \frac{2\gamma A}{\pi}\left[\bar{\omega} - \frac{\sigma/2}{A}\sqrt{1-\left(\frac{\sigma/2}{A}\right)^2}\right]. \end{aligned} \tag{4.14}$$

The describing function for the nonlinearity is thus

$$N(A) = \frac{b_1}{A} = \frac{2\gamma}{\pi}\left[\sin^{-1}\left(\frac{\sigma/2}{A}\right) - \frac{\sigma/2}{A}\sqrt{1-\left(\frac{\sigma/2}{A}\right)^2}\right]. \tag{4.15}$$

Note that for $A > \sigma/2$, $N(A)$ is real and positive for all positive γ and $N \to 0$ as $A \to \infty$. The existence and stability of any limit cycles is determined by the intersections of and $-1/N(A)$ and $G(j\omega)$ on the complex plane where

$$G(s) = \frac{1}{s\left(s - a_m\right)}. \tag{4.16}$$

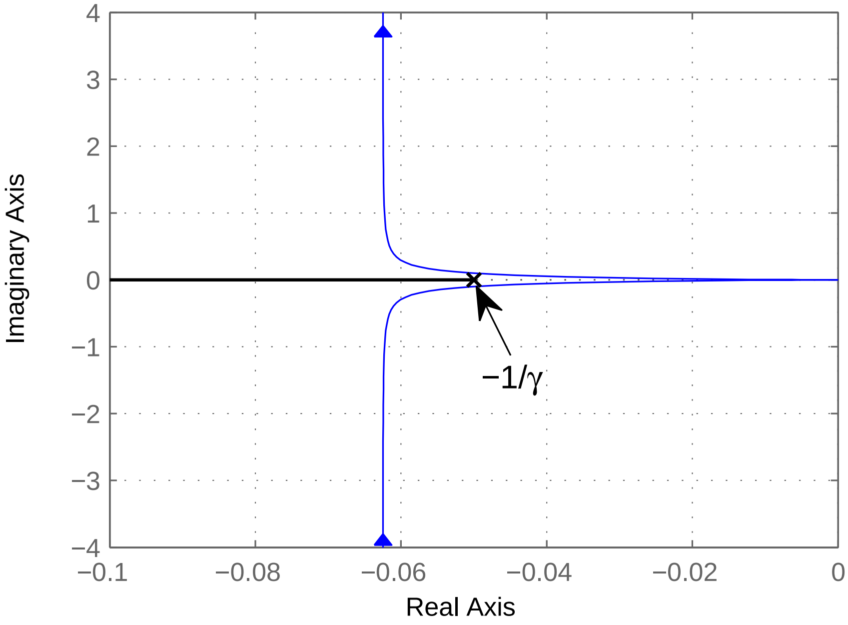


Figure 4-5: Nyquist diagram for the open-loop system $G(j\omega)$ and $-1/N(A)$ with $a_m = -4$ and $\gamma = 20$.

The Nyquist plot of $G(j\omega)$ is shown in figure 4-5.

Since the nonlinearity is only present for $A > \sigma/2$, $-1/N(A) \in (-1/\gamma, -\infty)$. The curve lies entirely on the negative real axis, the two curves never intersect and therefore there are no limit cycles present. Furthermore, $-1/N(A)$ is never encircled by $G(j\omega)$. By the Extended Nyquist Criterion the system is therefore stable everywhere and the amplitude of oscillations in the loop will always decrease. It is clear that the gain margin of the system is ∞, however, it can be seen that increasing γ can make the phase margin arbitrarily small. This result agrees with empirical results on simulation of MRAC systems that suggest high adaptation rate can significantly reduce the system's time delay margin.

4.1.3 Summary

In this section, we developed a describing function approach for approximating a switched linear system, itself an approximation of the adaptive system, with a linear closed loop system. An examination of a scalar problem with a single π-function led to a result for the margins of the approximate system. Although the concept of gain margins do not apply to the adaptive controller, it has been shown [7] that the MRAC architecture can accommodate any sign-definite loop gain, corresponding to an infinite gain margin. The phase margin is also not applicable to adaptive systems, but simulation results [73, 74] show that increasing the rate of adaptation reduces time-delay margin significantly. Both of these results are captured by the approximate linear system described.

The methods used are also applicable to the arbitrary many π-function case. This is important because as the number of π-functions covering some region of interest becomes large, the switched linear system approaches MRAC over that region. This may allow for the use of linear tools such as the extended Nyquist criterion to gain more insight about the margins of adaptive systems.

4.2 Time delay resistant adaptive control

In this section we take a different approach by using a nonlinear controller design which explicitly accounts for known time delay, and which has a proof of stability using nonlinear analysis tools. Many potential applications of adaptive control, such as adaptive flight control systems, require that the controller have high performance, stability guarantees, and robustness to time delays. These requirements typically lead to engineering trade-offs, such as a trade-off between performance and robustness. In this section, a new Time Delay Resistant (TDR) adaptive control framework is proposed using a combination of several modifications to the typical direct model reference adaptive control (MRAC) approach. The benefits of the TDR approach are explored with a simulation of the longitudinal dynamics of a fixed-wing aircraft. Comparison studies are presented for 80 ms and 250 ms delay cases.

4.2.1 Problem statement

Consider a MIMO, state variables accessible system of the form

$$\dot{x}_p(t) = A_p x_p(t) + B_p \Lambda u(t - \tau), \tag{4.17}$$

where $B_p \in \Re^{nxm}$ is constant and *known*, $A_p \in \Re^{nxn}$ is constant and *unknown*, $x_p \in \Re^n$, $u \in \Re^m$, $\Lambda \in \Re^{mxm}$ is an unknown diagonal positive definite constant matrix with diagonal elements $\Re \in (0, 1]$, and τ is a known time delay. The goal is to track a reference command $r(t)$ in the presence of the unknown A_p, Λ, and the known τ. The system output is given by

$$y(t) = C_p x_p(t), \tag{4.18}$$

and the output tracking error is given by

$$e_y(t) = y(t) - r(t - \tau). \tag{4.19}$$

Augmenting (4.17) with the integrated output tracking error

$$\dot{e}_{y_I}(t) = e_y(t) = y(t) - r(t - \tau), \tag{4.20}$$

leads to the extended open loop dynamics

$$\dot{x}(t) = Ax(t) + B\Lambda u(t - \tau) + B_c r(t - \tau), \tag{4.21}$$

where $x(t) = \begin{bmatrix} x_p^T(t) & e_{y_I}^T(t) \end{bmatrix}^T$ is the extended system state vector. The extended open-loop system matrices are given by

$$A = \begin{bmatrix} A_p & 0 \\ C_p & 0 \end{bmatrix}, \; B = \begin{bmatrix} B_p \\ 0 \end{bmatrix}, \; B_c = \begin{bmatrix} 0 \\ -I \end{bmatrix}, \tag{4.22}$$

and the extended system output

$$y(t) = \begin{bmatrix} C_p & 0 \end{bmatrix} x(t) = Cx(t). \tag{4.23}$$

We assume that there exists a constant, possibly unknown gain matrix θ_k such that $A_m = A + B\Lambda\theta_k^T$ where A_m is the known model dynamics. Then the system dynamics can be rewritten as

$$\dot{x}(t) = A_m x(t) + B\Lambda\left(u(t-\tau) + \theta_k^T x(t)\right) + B_c r(t-\tau). \tag{4.24}$$

The goal is then to design a suitable control input $u(t)$ so that the system described in (4.24) tracks the reference signal $r(t)$.

4.2.2 Modifications to MRAC

The TDR adaptive controller is comprised of several modifications to a standard MRAC approach. The overall control structure is that of a LQR baseline controller augmented by a direct adaptive posicast controller as well as an indirect adaptive controller. In both the direct and indirect adaptive parts, time-varying adaptive gains are utilized. In this section, the design of each of these modifications is described in detail.

Direct adaptive posicast controller

The adaptive posicast controller (APC) is essentially an adaptive extension of the smith predictor, an approach originated as a method to deal with systems with large delays. The APC method also brings in ideas from finite spectrum assignment [75]. The main idea in this approach is to predict the future output of the plant using a plant model, and then use this prediction to cancel the effect of the time delay on the system. It does this by adding an additional set of adaptive parameters, which in turn adds an additional term to the control law.

Through examination of the plant in (4.24), it is clear that in order to eliminate the effects of the uncertainty, one possible choice is a control input of the form

$$u(t) = -\hat{\theta}_k^T(t) x(t+\tau), \tag{4.25}$$

where $\hat{\theta}_k(t)$ are time-varying adaptive parameter estimates, and $x(t+\tau)$ is the system state positively forecasted (hence "posicast") into the future by τ seconds. This control input can be made causal by writing the future state $x(t+\tau)$ in terms of the general solution of the differential equation (4.24)

$$x(t+\tau) = e^{\theta_A \tau} x(t) + \int_{-\tau}^{0} e^{-\theta_A \eta} B\Lambda u(t+\eta) d\eta + \int_{-\tau}^{0} e^{-\theta_A \eta} B_c r(t+\eta) d\eta, \tag{4.26}$$

where $\theta_A = A_m + B\Lambda\theta_k^T$, for simplicity of notation.

We can therefore write the control input (4.25) as

$$u(t) = -\hat{\theta}_x^T(t)x(t) - \int_{-\tau}^{0} \hat{\theta}_u(t,\eta)u(t+\eta)d\eta - \int_{-\tau}^{0} \hat{\theta}_r(t,\eta)r(t+\eta)d\eta, \tag{4.27}$$

where $\theta_x = \theta_k^T e^{\theta_A \tau}$, $\theta_u(\eta) = \theta_k^T e^{-\theta_A \eta} B\Lambda$, and $\theta_r(\eta) = \theta_k^T e^{-\theta_A \eta} B_c$. In the discrete time implementation of the control law in (4.27), the integrals become sums and (4.27) can be rewritten as

$$u(t) = -\hat{\theta}^T(t)\phi(t), \tag{4.28}$$

where $\hat{\theta}(t) = \begin{bmatrix} \hat{\theta}_x^T(t) & \hat{\theta}_u(t-\tau) & ... & \hat{\theta}_u(t) & \hat{\theta}_r(t-\tau) & ... & \hat{\theta}_r(t) \end{bmatrix}$ and the regressor $\phi(t) = \begin{bmatrix} x(t) & u(t-\tau) & ... & u(t) & r(t-\tau) & ... & r(t) \end{bmatrix}$. The new regressor $\phi(t)$ therefore consists of the state $x(t)$, as well as τ/dT delayed versions of the control input $u(t)$ and the reference command $r(t)$. It is assumed that the time delay τ is a multiple of the time step dT. Note that as $dT \to 0$ the control input in (4.28) becomes a stepwise approximation of that in (4.27) .

Examining (4.25), (4.26), and (4.28), we note that the unknown parameter θ^T satisfies the following relation

$$\theta^T \phi(t) = \theta_k^T x(t+\tau). \tag{4.29}$$

Using (4.29), the plant (4.24) can be written as

$$\dot{x}(t) = A_m x(t) + B\Lambda \left(u(t-\tau) + \theta^T \phi(t-\tau) \right) + B_c r(t-\tau). \tag{4.30}$$

Substituting (4.28) into (4.30) and doing some algebraic manipulation, we can write

$$\dot{x}(t) = A_m x(t) - B\tilde{\Lambda}(t-\tau)u(t-\tau) - B\widetilde{\Lambda\theta}^T(t-\tau)\phi(t-\tau) + B_c r(t-\tau). \tag{4.31}$$

The parameters $\hat{\Lambda}(t)$ and $\widehat{\Lambda\theta}^T(t)$ are then adjusted in response to the model reference tracking error $e(t) = x(t) - x_m(t)$ where the reference model is the closed loop system given by (4.21) in feedback with a linear baseline controller in the case of no uncertainty or failures. Note the slight abuse of notation

$$\Lambda\theta^T - \underbrace{\widehat{\Lambda\theta}^T(t)}_{=\hat{\Lambda}(t)\hat{\theta}^T(t)} = \underbrace{\widetilde{\Lambda\theta}^T(t)}_{\neq\tilde{\Lambda}(t)\tilde{\theta}^T(t)} . \tag{4.32}$$

This reference model is given by

$$\dot{x}_m(t) = A_m x_m(t) + B_c r(t-\tau). \tag{4.33}$$

Note that the known time delay is included in the reference model.

Subtracting (4.33) from (4.31) yields the tracking error dynamics

$$\dot{e}(t) = A_m e(t) - B\tilde{\Lambda}(t-\tau)u(t-\tau) - B\widetilde{\Lambda\theta}^T(t-\tau)\phi(t-\tau). \tag{4.34}$$

The adaptive laws

$$\begin{aligned} \dot{\hat{\Lambda}}(t) &= \Gamma_\Lambda u(t-\tau)e^T(t)PB, \\ \dot{\widehat{\Lambda\theta}}^T(t) &= \Gamma_\theta \phi(t-\tau)e^T(t)PB, \end{aligned} \tag{4.35}$$

can be shown to lead to semi-global signal boundedness through the Lyapunov-Krasovskii functional [49]. Note that to generate the control input in (4.28), we must calculate $\hat{\theta}(t) = \hat{\Lambda}^{-1}(t)\widehat{\Lambda\theta}(t)$, for which we must assume that $\hat{\Lambda}(t)$ has an inverse. To ensure this is the case, we modify the adaptive laws as

$$\begin{aligned}\dot{\hat{\Lambda}}(t) &= \Gamma_{\Lambda}\delta_{ij}\left(u(t-\tau)e^T(t)PB\right) + f(\hat{\Lambda}(t)),\\ \dot{\widehat{\Lambda\theta}}^T(t) &= \Gamma_{\theta}\phi(t-\tau)e^T(t)PB,\end{aligned} \tag{4.36}$$

where δ_{ij} is the Kronecker delta and $f\left(\hat{\Lambda}(t)\right)$ is defined element-wise as

$$f\left(\hat{\lambda}_{ij}(t)\right) = \begin{cases}\frac{1}{\hat{\lambda}_{ij}(t)} - \frac{1}{\epsilon}, & \hat{\lambda}(t) < \epsilon,\\ 0, & \text{otherwise.}\end{cases} \tag{4.37}$$

The purpose of the function $f\left(\hat{\lambda}_{ii}(t)\right)$ is to keep the diagonal elements of $\hat{\Lambda}$ matrix bounded away from 0, ensuring that the matrix remains non-singular. This can be thought of as the converse of the projection algorithm.

Indirect adaptive controller

The so-called *CMRAC conjecture* [76], states that better transient characteristics can be obtained by using prediction errors in addition to tracking errors in the design of adaptive controllers. Thus a combined composite MRAC structure was developed by combining aspects of direct and indirect adaptive control. Numerous simulation studies have been reported over the years for various models, all of them confirming the fact that indeed CMRAC systems had transient performance better than that of direct MRAC alone [44–47]. The CMRAC conjecture, however, remains unproven.

The particular version of CMRAC used in these studies differs from some previous formulations [46, 48, 77] in that: a) it is applicable to a generic class of MIMO dynamical systems, b) online measurements of the system state derivative are not required, and c) the system is designed to augment a baseline linear controller [78].

To generate the prediction error signal required for indirect adaptation, we first rewrite the dynamics of (4.30) as

$$\begin{aligned}\dot{x}(t) + \lambda_f x(t) = \lambda_f x(t) + A_m x(t) + B_c r(t-\tau) \\ + B\Lambda\left(u(t-\tau) + \theta^T \phi(t-\tau)\right),\end{aligned} \tag{4.38}$$

and introduce a stable filter

$$G(s) = \frac{\lambda_f}{s + \lambda_f}, \tag{4.39}$$

where $\lambda_f > 0$ is the filter inverse constant. The filtered version of $x(t)$ is denoted $x_f(t)$ and is described by the dynamics

$$\dot{x}_f(t) + \lambda_f x_f(t) = \lambda_f x(t). \tag{4.40}$$

Substituting (4.40) into (4.38) and letting $z(t) = x(t) - x_f(t)$ we have

$$\dot{z}(t) + \lambda_f z(t) = A_m x(t) + B_c r(t-\tau) + B\Lambda\left(u(t-\tau) + \theta^T \phi(t-\tau)\right), \tag{4.41}$$

and, consequently

$$\begin{aligned}z(t) = z(t_0)e^{-\lambda_f t_0} + \int_{t_0}^{t} e^{-\lambda_f (t-\eta)} \left[A_m x(\eta)\right. \\ \left. + B_c r(\eta-\tau) + B\Lambda\left(u(t-\tau) + \theta^T \phi(t-\tau)\right)\right] d\eta.\end{aligned} \tag{4.42}$$

We can assume without loss of generality that the filter dynamics (4.39) and the plant dynamics (4.41) have the same initial conditions, that is, $z(t_0) = 0$ [78]. Noting that the integral in (4.42) represents the application of the filter (4.39), we can rewrite (4.42) as

$$x(t) - x_f(t) = A_m \frac{x_f(t)}{\lambda_f} - B_c \frac{r_f(t-\tau)}{\lambda_f} + B\Lambda \frac{u_f(t-\tau)}{\lambda_f} + B\Lambda\theta^T \frac{\phi_f(t-\tau)}{\lambda_f}, \tag{4.43}$$

where $r_f(t)$, $u_f(t)$, and $\phi_f(t)$ are filtered versions of $r(t)$, $u(t)$, and $\phi(t)$, respectively. Assuming that B is full rank, we can write the Moore-Penrose pseudoinverse [79] as

$B^+ = \left(B^T B\right)^{-1} B^T$ and $\left(B^T B\right)$ is invertible. We can therefore rearrange (4.43) as

$$\begin{aligned} B^+ \left[\lambda_f(x(t) - x_f(t)) \right. &\left. - A_m x_f(t) - B_c r_f(t-\tau)\right] \\ &= \Lambda u_f(t-\tau) + \Lambda \theta^T \phi_f(t-\tau). \end{aligned} \tag{4.44}$$

We now denote the left-hand side of (4.44) as $Y(t) \in \Re^m$ and note that the value of $Y(t)$ can be computed online at every time instant t using the current state $x(t)$, the filtered state $x_f(t)$ and a stored value of the command input $r(t-\tau)$. The right-hand side of (4.44) contains the unknown parameters Λ and θ, which can be estimated using the predictor model

$$\hat{Y}(t) = \hat{\Lambda}(t-\tau) u_f(t-\tau) + \widehat{\Lambda\theta^T}(t-\tau)\phi_f(t-\tau). \tag{4.45}$$

The predictor error $e_Y(t) = \hat{Y}(t) - Y(t)$ can thus be written as

$$e_Y(t) = \tilde{\Lambda}(t-\tau) u_f(t-\tau) + \widetilde{\Lambda\theta^T}(t-\tau)\phi_f(t-\tau). \tag{4.46}$$

Including the indirect adaptation is accomplished by simply adding indirect adaptive terms to the direct adaptive laws (4.36)

$$\begin{aligned} \dot{\hat{\Lambda}}(t) &= \Gamma_\Lambda \delta_{ij} \left(u(t-\tau) e^T(t) PB - u_f(t-\tau)\gamma_c e_Y^T(t)\right) + f(\hat{\Lambda}(t)), \\ \dot{\widehat{\Lambda\theta^T}}(t) &= \Gamma_\theta \left(\phi(t-\tau) e^T(t) PB - \phi_f(t-\tau)\gamma_c e_Y^T(t)\right), \end{aligned} \tag{4.47}$$

where $\gamma_c > 0$ is a gain on the indirect part. We can write $\dot{\tilde{\Theta}}^T(t) = \begin{bmatrix} \dot{\tilde{\Lambda}}(t) & \dot{\widetilde{\Lambda\theta^T}}(t) \end{bmatrix}$ and also note that the adaptive law can be broken into direct and indirect parts as

$$\dot{\tilde{\Theta}}(t) = \dot{\tilde{\Theta}}_d(t) + \dot{\tilde{\Theta}}_i(t) + F\left(\hat{\Lambda}(t)\right), \tag{4.48}$$

where $\dot{\tilde{\Theta}}_d(t)$ consists of the term containing the tracking error e and $\dot{\tilde{\Theta}}_i(t)$ consists of the term containing the prediction error e_Y and $F\left(\hat{\Lambda}(t)\right)^T = \begin{bmatrix} f\left(\hat{\Lambda}(t)\right) & 0 \end{bmatrix}$. It is clear that in the case where $\gamma_c = 0$, the $\dot{\tilde{\Theta}}_i(t)$ term vanishes and the above adaptive control laws reduce to those of (4.36).

Variable adaptive gains

Allowing time-varying adaptive gains allows for the faster convergence of parameter estimates as well as improved noise sensitivity. In particular, the bounded-gain-forgetting (BGF) gain adjustment allows for fast convergence and smooth parameter estimates [44]. The rules for adjusting the adaptive gains are chosen as

$$\dot{\Gamma}^{-1}(t) = -\rho(t)\Gamma^{-1}(t) + \omega_f(t-\tau)\gamma_c\omega_f^T(t-\tau), \tag{4.49}$$

where $\Gamma(t) = \begin{bmatrix} \Gamma_\Lambda(t) & 0 \\ 0 & \Gamma_\theta(t) \end{bmatrix}$, $\omega_f^T(t) = \begin{bmatrix} u_f^T(t) & \phi_f^T(t) \end{bmatrix}$, and $\rho(t)$ is a positive scalar forgetting factor given by

$$\rho(t) = \rho_0\left(1 - \frac{||\Gamma^T(t)\Gamma(t)||}{\gamma_0^2}\right), \tag{4.50}$$

where ρ_0 is the positive constant maximum forgetting rate and γ_0 is the positive constant bound on the adaptive gain matrix magnitude. The inverse gain adjustment law in (4.49) has several important properties. First, it ensures that $\Gamma(t)$ remains positive definite. Second, the data forgetting feature allows for parameter tracking, and is only activated when ω_f is persistently exciting [48].

In implementation, we avoid taking computationally expensive inverses by using the equivalent gain adjustment law

$$\dot{\Gamma}(t) = \rho(t)\Gamma(t) - \Gamma(t)\omega_f(t-\tau)\gamma_c\omega_f^T(t-\tau)\Gamma(t). \tag{4.51}$$

Time delay resistant adaptive control

We can now combine the modifications discussed previously into one coherent structure, the time delay resistant (TDR) adaptive controller. The control law is of the form (4.25) and the parameters $\hat{\Theta}$ are adjusted according to the adaptive laws

$$\dot{\hat{\Theta}}(t) = \underbrace{\Gamma(t)\Delta_{ij}\omega(t-\tau)e^T(t)PB}_{\dot{\hat{\Theta}}_d(t)} - \underbrace{\Gamma(t)\Delta_{ij}\omega_f(t-\tau)\gamma_c e_Y^T(t)}_{\dot{\hat{\Theta}}_i(t)} + F(\hat{\Lambda}(t)), \tag{4.52}$$

where $\Delta_{ij}^T = \begin{bmatrix}\delta_{ij} & I\end{bmatrix}$ and $\omega^T(t) = \begin{bmatrix}-u^T(t) & \phi^T(t)\end{bmatrix}$. The adaptive gains are adjusted according to (4.51). Using this nomenclature, the prediction error can be written compactly as

$$e_Y(t) = \tilde{\Theta}(t-\tau)\omega_f(t-\tau), \tag{4.53}$$

and the error dynamics in (4.34) can be written as

$$\dot{e}(t) = A_m e(t) - B\tilde{\Theta}^T(t)\omega(t-\tau) + B\left[\tilde{\Theta}^T(t) - \tilde{\Theta}^T(t-\tau)\right]\omega(t-\tau). \tag{4.54}$$

Using the Leibniz-Newton formula, we have:

$$\begin{aligned}\tilde{\Theta}^T(t) - \tilde{\Theta}^T(t-\tau) &= \int_{-\tau}^{0} \dot{\tilde{\Theta}}^T(t+\nu)d\nu \\ &= \left(\int_{-\tau}^{0} \left[\Gamma(t+\nu)\Delta_{ij}\left(\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right.\right. \\ &- \left.\left.\left.\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu)\right) + F\left(\hat{\Lambda}(t+\nu)\right)\right] d\nu\right)^T.\end{aligned} \tag{4.55}$$

Using (4.55), we can write the error dynamics (4.54) as

$$\begin{aligned}\dot{e}(t) &= A_m e(t) - B\tilde{\Theta}^T(t)\omega(t-\tau) \\ &+ B\left(\int_{-\tau}^{0} \left[\Gamma(t+\nu)\Delta_{ij}\left(\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right.\right. \\ &- \left.\left.\left.\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu)\right) + F\left(\hat{\Lambda}(t+\nu)\right)\right] d\nu\right)^T \omega(t-\tau).\end{aligned} \tag{4.56}$$

Equations (4.25) and (4.50)-(4.52) fully describe the TDR adaptive controller.

Theorem 1. *For the closed-loop system given by the dynamics* (4.24) *and the adaptive controller given in* (4.25) *and* (4.50)-(4.52)*, given initial conditions on adaptive parameters* $\Theta(0)$ *and regressor vector* $\omega(\xi)$ *where* $\xi \in [-\tau,\ 0]$*, there exists a* $\tau^* > 0$

such that for all $\tau \in [0,\ \tau^*]$, the signals in the closed-loop system are uniformly ultimately bounded. Furthermore, both errors $e(t)$ and $e_Y(t)$ tend to 0 asymptotically as $t \to \infty$.

Proof. In the delay-free case, that is $\tau \equiv 0$, a Lyapunov function that proves global asymptotic stability of the system is given by

$$V_1(e, \tilde{\Theta}) = e^T(t)Pe(t) + \mathrm{Tr}\left(\tilde{\Theta}^T(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right), \tag{4.57}$$

where P is the unique symmetric positive definite solution of the Lyapunov equation,

$$A_m^T P + PA_m = -Q, \tag{4.58}$$

where Q is also symmetric positive definite. Based on the Lyapunov function given in (4.57), we now introduce the Lyapunov-Krasovskii functional

$$V(e, \tilde{\Theta}, \dot{\tilde{\Theta}}) = V_1(e, \tilde{\Theta}) + V_2(\dot{\tilde{\Theta}}_d) + V_3(\dot{\tilde{\Theta}}_i), \tag{4.59}$$

where the additional terms are given by:

$$\begin{aligned} V_2(\dot{\tilde{\Theta}}_d) &= 2\mathrm{Tr}\left(\int_{-\tau}^{0}\left(\int_{t+\nu}^{t} \dot{\tilde{\Theta}}_d(\xi)\dot{\tilde{\Theta}}_d(\xi)^T d\xi\right) d\nu\right), \\ V_3(\dot{\tilde{\Theta}}_i) &= \mathrm{Tr}\left(\int_{-\tau}^{0}\left(\int_{t+\nu}^{t} \dot{\tilde{\Theta}}_i(\xi)\dot{\tilde{\Theta}}_i(\xi)^T d\xi\right) d\nu\right). \end{aligned} \tag{4.60}$$

Using (4.58) and (4.54), the derivative of V_1 can be computed as

$$\begin{aligned} \dot{V}_1(t) = &-e^T(t)Qe(t) - 2e^T(t)PB\tilde{\Theta}(t)\omega(t-\tau) \\ &+ 2e^T(t)PB\left(\int_{-\tau}^{0} \Gamma(t+\nu)\Delta_{ij}\left[\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right. \\ &- \left.\left.\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu) + F\left(\hat{\Lambda}(t+\nu)\right)\right] d\nu\right)^T \omega(t-\tau) \\ &+ 2\mathrm{Tr}\left(\tilde{\Theta}^T(t)\Gamma^{-1}(t)\dot{\tilde{\Theta}}(t)\right) + \mathrm{Tr}\left(\tilde{\Theta}^T(t)\dot{\Gamma}^{-1}(t)\tilde{\Theta}(t)\right). \end{aligned} \tag{4.61}$$

We now note that

$$\mathrm{Tr}\left(\tilde{\Theta}^T(t)\Gamma^{-1}(t)\dot{\tilde{\Theta}}(t)\right) \leq \mathrm{Tr}\left(\tilde{\Theta}^T(t)\Gamma^{-1}(t)\dot{\tilde{\Theta}}_d(t)\right) + \mathrm{Tr}\left(\tilde{\Theta}^T(t)\Gamma^{-1}(t)\dot{\tilde{\Theta}}_i(t)\right), \quad (4.62)$$

since by the definition of the function $f\left(\hat{\Lambda}(t)\right)$ given in (4.37),

$$\mathrm{Tr}\left(\tilde{\Lambda}^T(t)\Gamma_\Lambda^{-1}(t)F(\hat{\Lambda})(t)\right) = \sum_i^n \frac{\tilde{\lambda}_{ii}(t)}{\gamma_{\Lambda_i}(t)} f\left(\hat{\lambda}_{ii}(t)\right) \geq 0, \quad (4.63)$$

for $\lambda_{ii} > \epsilon$. By making ϵ arbitrarily small, we can ensure the above relation holds. In reality, applying this approach with very small ϵ may cause numerical problems, which necessitates the use of a stronger assumption on Λ, such as $\lambda_{ii} \in (0.1 \quad 1]$.

Using (4.49), (4.52), and (4.62), the derivative of the Lyapunov function in (4.61) becomes

$$\begin{aligned}\dot{V}_1(t) \leq\ & -e^T(t)Qe(t) \\ & + 2e^T(t)PB\left(\int_{-\tau}^0 \Gamma(t+\nu)\Delta_{ij}\left[\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right. \\ & \left.\left. -\ \omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu) + F\left(\hat{\Lambda}(t+\nu)\right)\right] d\nu\right)^T \omega(t-\tau) \\ & - 2\mathrm{Tr}\left(\tilde{\Theta}^T(t)\omega_f(t-\tau)\gamma_c e_Y^T(t)\right) - \mathrm{Tr}\left(\tilde{\Theta}^T(t)\rho(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right) \\ & + \mathrm{Tr}\left(\tilde{\Theta}^T(t)\omega_f(t-\tau)\gamma_c\omega_f^T(t-\tau)\tilde{\Theta}(t)\right).\end{aligned} \quad (4.64)$$

Using (4.53), this becomes

$$\begin{aligned}\dot{V}_1(t) \leq\ & -e^T(t)Qe(t) - \mathrm{Tr}\left(e_Y\gamma_c e_Y^T(t)\right) - \mathrm{Tr}\left(\tilde{\Theta}^T(t)\rho(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right) \\ & + 2e^T(t)PB\left(\int_{-\tau}^0 \Gamma(t+\nu)\Delta_{ij}\left[\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right. \\ & \left.\left. -\ \omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu) + F\left(\hat{\Lambda}(t+\nu)\right)\right] d\nu\right)^T \omega(t-\tau).\end{aligned} \quad (4.65)$$

Using the trace identity, $a^Tb = \mathrm{Tr}\left(ba^T\right)$ for codimensional vectors a and b, we can write (4.65) as

$$\begin{aligned}\dot{V}_1(t) \leq & -e^T(t)Qe(t) - e_Y^T\gamma_c e_Y(t) - \mathrm{Tr}\left(\tilde{\Theta}^T(t)\rho(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right) \\ & + 2\mathrm{Tr}\left(\omega(t-\tau)e^T(t)PB\left(\int_{-\tau}^{0}\Gamma(t+\nu)\left[\omega(t-\tau+\nu)e^T(t+\nu)PB\right.\right.\right. \\ & \left.\left.\left. -\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu) + F\left(\hat{\Lambda}(t+\nu)\right)\right]d\nu\right)^T\right).\end{aligned} \tag{4.66}$$

The derivative of V_2 can be calculated as

$$\begin{aligned}\dot{V}_2 = & 2\tau\mathrm{Tr}\left(\Gamma(t)\omega(t-\tau)e^T(t)PB\left(\Gamma(t)\omega(t-\tau)e^T(t)PB\right)^T\right) \\ & - 2\mathrm{Tr}\left(\int_{-\tau}^{0}\Gamma(t+\nu)\omega(t-\tau+\nu)e^T(t+\nu)PB\right. \\ & \left.\left(\Gamma(t+\nu)\omega(t-\tau+\nu)e^T(t+\nu)PB\right)^T d\nu\right),\end{aligned} \tag{4.67}$$

and the derivative of V_3 can be calculated as

$$\begin{aligned}\dot{V}_3 = & \tau\mathrm{Tr}\left(\Gamma(t)\omega_f(t-\tau)\gamma_c e_Y^T(t)\left(\Gamma(t)\omega_f(t-\tau)\gamma_c e_Y^T(t)\right)^T\right) \\ & - \mathrm{Tr}\left(\int_{-\tau}^{0}\Gamma(t+\nu)\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu)\right. \\ & \left.\left(\Gamma(t+\nu)\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu)\right)^T d\nu\right).\end{aligned} \tag{4.68}$$

We can now write the derivative of the Lyapunov-Krasovskii functional V as

$$\begin{aligned}\dot{V} \leq & - e^T(t)Qe(t) - e_Y^T(t)\gamma_c e_Y(t) - \mathrm{Tr}\left(\tilde{\Theta}^T(t)\rho(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right) \\ & + \mathrm{Tr}\left(\int_{-\tau}^{0}\left[2\bar{a}^T\bar{b} + \bar{a}^T\Gamma^T(t)\Gamma(t)\bar{a} - \bar{b}^T\bar{b}\right]d\nu\right) \\ & + \mathrm{Tr}\left(\int_{-\tau}^{0}\left[2\bar{a}^T\bar{d} + \bar{a}^T\Gamma^T(t)\Gamma(t)\bar{a} - \bar{d}^T\bar{d}\right]d\nu\right) - \mathrm{Tr}\left(\int_{-\tau}^{0}\bar{b}^T\bar{b}d\nu\right) \\ & + \mathrm{Tr}\left(\tau\bar{c}^T\Gamma^T(t)\Gamma(t)\bar{c}\right) + 2\mathrm{Tr}\left(e^T(t)PB\omega(t-\tau)\int_{-\tau}^{0}F\left(\hat{\Lambda}(t+\nu)\right)d\nu\right),\end{aligned} \tag{4.69}$$

where

$$\begin{aligned}\bar{a}^T &= \omega(t-\tau)e^T(t)PB, & \bar{b}^T &= \Gamma(t+\nu)\omega(t-\tau+\nu)e^T(t+\nu)PB, \\ \bar{c}^T &= -\omega_f(t-\tau)\gamma_c e_Y^T(t), & \bar{d}^T &= -\Gamma(t+\nu)\omega_f(t-\tau+\nu)\gamma_c e_Y^T(t+\nu).\end{aligned} \tag{4.70}$$

If we require $\hat{\Lambda} > \epsilon \quad \forall t \in (-\tau, 0)$, the final term in (4.69) vanishes. After completing the square and removing some negative semidefinite terms, we have

$$\begin{aligned}
\dot{V} \leq & - e^T(t)Qe(t) - e_Y^T(t)\gamma_c e_Y(t) \\
& + 2\tau \mathrm{Tr}\left(\bar{a}^T\left(\Gamma^T(t)\Gamma(t) + I\right)\bar{a}\right) + \tau \mathrm{Tr}\left(\bar{c}^T\Gamma^T(t)\Gamma(t)\bar{c}\right) \\
\leq & - e^T(t)Qe(t) - e_Y^T(t)\gamma_c e_Y(t) - \mathrm{Tr}\left(\tilde{\Theta}^T(t)\rho(t)\Gamma^{-1}(t)\tilde{\Theta}(t)\right) \\
& + 2\tau \mathrm{Tr}\left(\omega(t-\tau)e^T(t)PB\left(\Gamma^T(t)\Gamma(t) + I\right)B^T Pe(t)\omega^T(t-\tau)\right) \\
& + \tau \mathrm{Tr}\left(\omega_f(t-\tau)\gamma_c e_Y^T(t)\Gamma^T(t)\Gamma(t)e_Y(t)\gamma_c\omega_f^T(t-\tau)\right).
\end{aligned} \tag{4.71}$$

Using the trace identity $\mathrm{Tr}\,(abc) = \mathrm{Tr}\,(bca)$, and after some algebraic manipulation (4.71) becomes

$$\begin{aligned}
\dot{V} \leq & - e^T(t)Qe(t) - e_Y^T(t)\gamma_c e_Y(t) \\
& + 2\tau e^T(t)PB\left(\Gamma^T(t)\Gamma(t) + I\right)B^T Pe(t)\omega^T(t-\tau)\omega(t-\tau) \\
& + \tau\gamma_c e_Y^T(t)\Gamma^T(t)\Gamma(t)e_Y(t)\gamma_c\omega_f^T(t-\tau)\omega_f(t-\tau).
\end{aligned} \tag{4.72}$$

After taking some norms and applying the triangle inequality,

$$\begin{aligned}
\dot{V} \leq & - e^T(t)Qe(t) - e_Y^T(t)\gamma_c e_Y(t) \\
& + 2\tau\left(||\Gamma^T(t)\Gamma(t)|| + ||I||\right)||\omega(t-\tau)||^2 e^T(t)PBB^T Pe(t) \\
& + \tau\gamma_c^2||\Gamma^T(t)\Gamma(t)||||\omega_f(t-\tau)||^2 e_Y^T(t)e_Y(t).
\end{aligned} \tag{4.73}$$

Finally, combining the quadratic terms in e and e_Y,

$$\begin{aligned}
\dot{V} \leq & - e^T(t)\left[Q - 2\tau\left(\gamma_0^2 + 1\right)||\omega(t-\tau)||^2 PBB^T P\right]e(t) \\
& - e_Y^T(t)\left[\gamma_c - \tau\gamma_c^2\gamma_0^2||\omega_f(t-\tau)||^2\right]e_Y(t).
\end{aligned} \tag{4.74}$$

Therefore, there are 2 conditions that must hold for $\dot{V}$ to be negative definite:

i. $Q - 2\tau\left(\gamma_0^2 + 1\right)||\omega(t-\tau)||^2 PBB^T P > 0$,

ii. $\gamma_c - \tau\gamma_0^2\gamma_c^2||\omega_f(t-\tau)||^2 > 0$.

From here the proof follows that of previous work [49, 80]. The rest of the proof involves putting bounds on $||\omega||$ and $||\omega_f||$. Given a bound on $||\omega||$ and $||\omega_f||$, it is clear that the above two relations will hold for all $\tau < \tau^*$ where τ^* is a time-delay margin, and is bounded away from zero by virtue of the fact that Q is positive definite and γ_c is positive, thus proving theorem 1.

$\square$

4.2.3 Simulation results

In this section, we compare the performance of an LQR baseline to a series of six adaptive controllers which consist of various combinations of the modifications discussed in 4.2.2, including the TDR adaptive controller itself. The performance of the various controllers will be examined using a simulation model of the F-16 short period longitudinal dynamics. Neglecting the effects of gravity and thrust, the short period dynamics are

$$\begin{bmatrix} \dot{\alpha} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \frac{Z_\alpha}{V} & 1 + \frac{Z_q}{V} \\ M_\alpha & M_q \end{bmatrix} \begin{bmatrix} \alpha \\ q \end{bmatrix} + \begin{bmatrix} \frac{Z_{\delta_e}}{V} \\ M_{\delta_e} \end{bmatrix} \delta_e, \tag{4.75}$$

where α is the aircraft angle of attack, q is the pitch rate, δ_e is the elevator deflection, and V is the trimmed (constant) air speed [81]. The partial derivatives of the aerodynamic vertical force Z with respect to α, q, and δ_e, are given by Z_α, Z_q, and Z_{δ_e} respectively. Similarly, the partial derivatives of the pitching moment M with respect to α, q, and δ_e, are given by M_α, M_q, and M_{δ_e} respectively. The numerical values for these aerodynamic derivatives can be found in [81], Example 5.5-3, Table 3.4.-3. These particular values correspond to an F-16 aircraft trimmed at sea level, with $V_T = 502$ ft/sec, $\bar{Q} = 300$ lb/ft^2, and $CG = 0.35\bar{c}$.

With α in units of radians, q in radians/second, and δ_e in degrees, the resulting open-loop system matrices are given by

$$A = \begin{bmatrix} -1.0189 & .9051 \\ 0.8223 & -1.0774 \end{bmatrix}, \quad B = \begin{bmatrix} -0.0022 \\ -0.1756 \end{bmatrix}. \tag{4.76}$$

Matched uncertainty in the form of a linear-in-state uncertainty $K_{xp,pert}^T x$, and a constant loss of control effectiveness $\Lambda > 0$ were included in the system. Numerical values for these uncertainties are given by

$$K_{xp,pert}^T = \begin{bmatrix} -4.6839 & -9.8197 \end{bmatrix}, \quad \Lambda = 0.5. \tag{4.77}$$

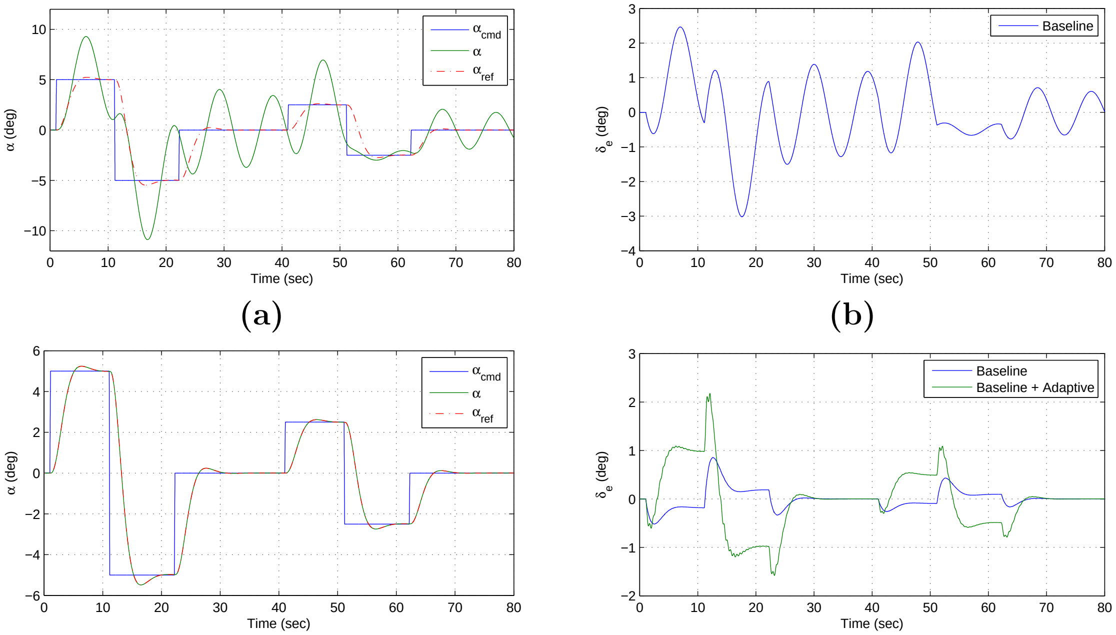
The values of the uncertainties given in (4.77) are equivalent to a 50% increase in M_α, an 80% decrease in M_q, and a 50% decrease in the elevator effectiveness M_{δ_e}. Thus, the vehicle has become 50% more statically unstable, lost 80% of its pitch damping ability, and the aircraft controllability has decreased by 50%. Time delays tested ranged from 0-250 ms. The extent of the uncertainties and the size of time delays tested are quite drastic and perhaps even unrealistic. These choices were motivated by the intent to demonstrate the effectiveness of the proposed TDR design, and to highlight the differences between the various modifications to the MRAC approach.

The full aircraft dynamics, along with the uncertainties described above, now take the form of (4.17), where $x_p = \begin{bmatrix} \alpha & q \end{bmatrix}^T$ is the state vector, $\Lambda > 0$ is the uncertainty in the elevator effectiveness. The controlled output state y was chosen to be the angle of attack α. Thus, the control goal becomes tracking of a bounded, time-varying angle of attack command $r = \alpha_{cmd}$ in the presence of the uncertainties and the time delay τ. To ensure steady state tracking performance, integrated α tracking error dynamics,

$$\dot{e}_{y_I}(t) = \alpha(t) - \alpha_{cmd}(t - \tau), \tag{4.78}$$

are included as well. The resulting extended open-loop dynamics then takes the form of (4.24).

The reference matrix A_m was defined as in (3.10)-(3.11) with the baseline feedback gains selected using the LQR method [81], with $Q_{lqr} = \text{diag}\left(\begin{bmatrix} 0 & 0 & 100 \end{bmatrix}\right)$ and $R_{lqr} = 1$. This resulted in LQR feedback gains $K_x = \begin{bmatrix} 10.8786 & 6.0589 & 10 \end{bmatrix}$. The reference model angle-of-attack is given by α_{ref}. A direct MRAC controller was then designed for the system described in section 4.2.1 and adaptive gains were selected to give a good balance of tracking performance and control effort.



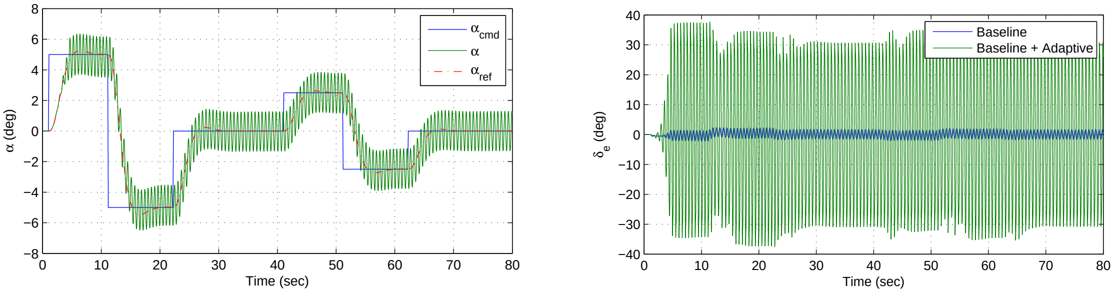
(a) **(b)**

(c) **(d)**

Figure 4-6: LQR (a) α command following and (b) corresponding elevator deflection, as well as MRAC (c) α command following and (d) corresponding elevator deflection for the failure case with $\tau = 0$ ms.

The first simulation study consists of the closed-loop system with the two types of uncertainties mentioned previously, but with no time delay, that is $\tau \equiv 0$. Figure 4-6(c)-(d) shows that the MRAC approach solves the tracking problem, while it can be seen in figure 4-6(a)-(b) that the LQR approach is unable to accommodate the uncertainties and has very poor tracking performance. This result is unsurprising given that adaptive control has been shown to be very effective in dealing with parametric uncertainty and MRAC has a global asymptotic proof of stability for the simulated system.

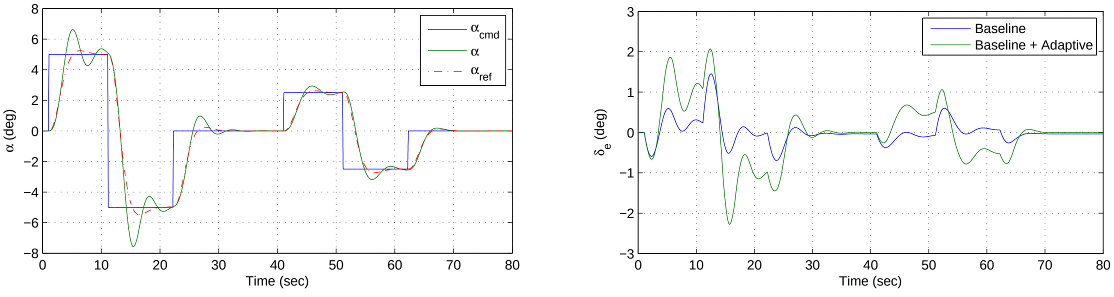
However, when even a small amount of time delay (40 ms, in this case) is added to the system, the performance of the well-tuned adaptive controller deteriorates dramatically. In Figure 4-7 (a), the MRAC approach displays undesirable oscillations. Figure 4-7 (b) shows that the elevator deflection is unreasonably large. One possible solution to this problem is to reduce the adaptive gains. For this motivational example, a single scaling factor across all adaptive gains is sufficient. Reducing the



(a) **(b)**

Figure 4-7: MRAC (a) α command following and (b) corresponding elevator deflection for the failure case with $\tau = 40$ ms.

gains by a factor of 100 eliminates the undesirable oscillations. However, it can be seen by comparing figure 4-6(c) to figure 4-8(a) that reducing adaptive gains has a detrimental effect on transient performance.



(a) **(b)**

Figure 4-8: Low-gain MRAC (a) α command following and (b) corresponding elevator deflection for the failure case with $\tau = 40$ ms.

It is therefore desirable to develop a modified MRAC approach that allows high performance and high robustness to time delays simultaneously. In the following simulation studies, we examine the performance of every reasonable combination of the MRAC modifications described in section 4.2.2 for the $\tau = 80$ ms (nominal time delay value) and the $\tau = 250$ ms (large time delay value) case. The features of the seven controllers examined are summarized in table 4.1. The table culminates with the TDR adaptive controller, which is the combination of all the modifications.

The performance of the various controllers depends heavily on the selection of the free parameters, particularly the choices of the adaptive gains. To ensure a fair

	LQR baseline	Direct Adaptation	Posicast Control	Indirect Adaptation	Variable Ad. Gains
Baseline	×				
MRAC	×	×			
APC	×	×	×		
CMRAC	×	×		×	
BGF	×	×		×	×
CAPC	×	×	×	×	
TDR	×	×	×	×	×

Table 4.1: Features of the baseline controller, model reference adaptive controller (MRAC), adaptive posicast controller (APC), combined / composite MRAC (CMRAC), CMRAC with bounded-gain-forgetting adaptive gains (BGF), combined / composite adaptive posicast controller (CAPC), and time delay resistant adaptive controller (TDR).

comparison, all free parameters were selected to minimize the cost function

$$J = \frac{1}{3}\left(\bar{e}^T\bar{e} + \bar{\delta}_e^T\bar{\delta}_e + \dot{\bar{\delta}}_e^T\dot{\bar{\delta}}_e\right), \tag{4.79}$$

where $\bar{e}$, $\bar{\delta}_e$, and $\dot{\bar{\delta}}_e$ are non-dimensionalized tracking error, control usage, and control rate usage, respectively. These signals are non-dimensionalized as

$$\bar{e} = \frac{||e||}{||e_{\mathrm{LQR}^*}||}, \quad \bar{\delta}_e = \frac{||\delta_e||}{||\delta_{e_{\mathrm{LQR}^*}}||}, \quad \dot{\bar{\delta}}_e = \frac{||\dot{\delta}_e||}{||\dot{\delta}_{e_{\mathrm{LQR}^*}}||}, \tag{4.80}$$

where e_{LQR^*}, $\delta_{e_{\mathrm{LQR}^*}}$, and $\dot{\delta}_{e_{\mathrm{LQR}^*}}$ are the tracking error, control usage, and control rate usage of an LQR baseline controller with perfect knowledge of the uncertainty. This fictional "LQR*" controller is the best possible without explicitly accounting for the delay in the system. This controller performs equally well regardless of the severity of the failure and is of course infeasible in practice as the uncertainties encountered are not known *a priori.*

80 ms time delay case

The first case under consideration is that of an 80 ms time delay. This is a fairly reasonable estimate for the actual delay in the loop for an aircraft such as the F-16. Seven controllers were designed using the modifications described in table 4.1 and free parameters were selected to minimize the cost function given in (4.79). The non-dimensionalized errors, control usage, and control rate usage of the various controllers can be found in table 4.2 and the cost associated with each controller is shown in figure 4-9. A comparison of the tracking performance and control usage for the controllers is shown in figure 4-10.

	Baseline	MRAC	APC	CMRAC	BGF	CAPC	TDR	LQR*
$\bar{e}$	45.6	5.44	2.56	1.62	1.47	1.06	**0.921**	1.00
$\bar{\delta}_e$	2.14	1.04	1.02	1.01	1.01	1.00	**0.976**	1.00
$\bar{\dot{\delta}}_e$	2.32	1.05	1.44	1.25	1.34	1.21	**0.976**	1.00

Table 4.2: Comparison of tracking error and control effort for the $\tau = 80$ ms case.

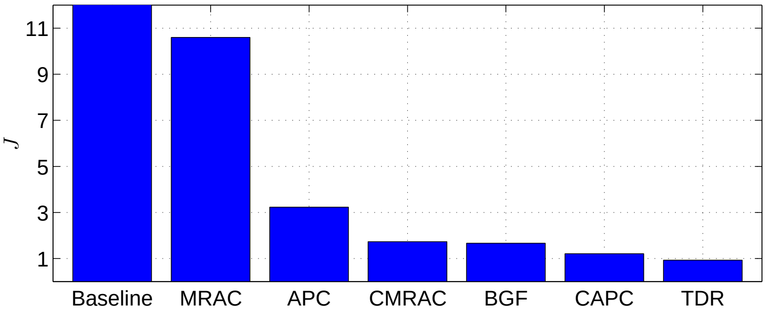
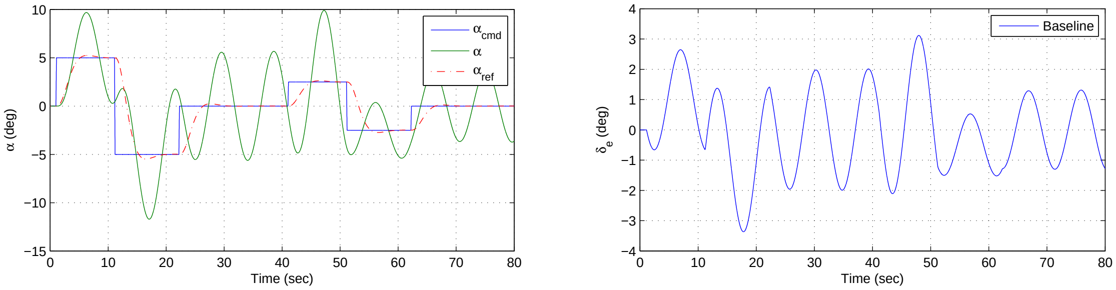
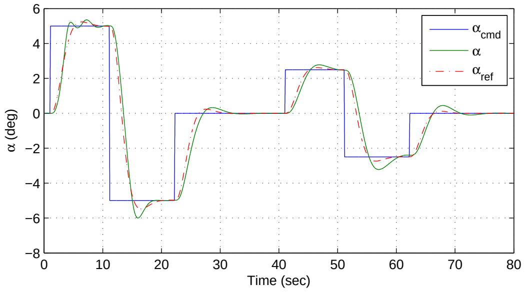


Figure 4-9: Comparison of the cost associated with each of the seven controllers for the $\tau = 80$ ms case. Note the y-axis is cropped, the baseline controller cost is well off the chart at about 700.

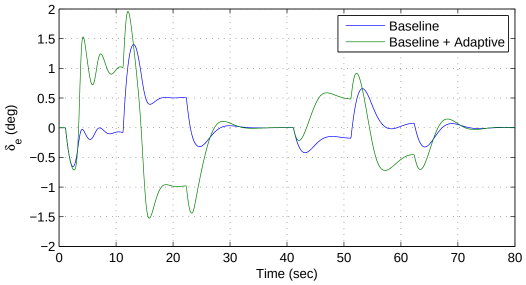


(a) **(b)**



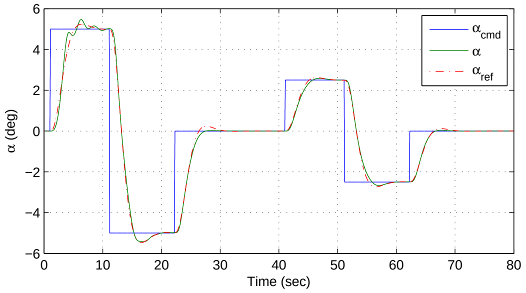
α_{cmd}
α
α_{ref}
α (deg)
Time (sec)

(c)



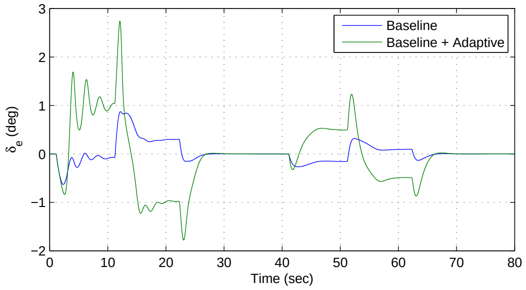
Baseline
Baseline + Adaptive
δ_e (deg)
Time (sec)

(d)



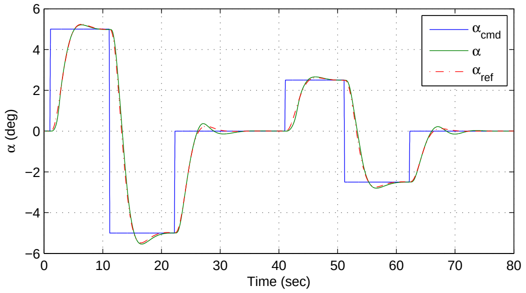
α_{cmd}
α
α_{ref}
α (deg)
Time (sec)

(e)



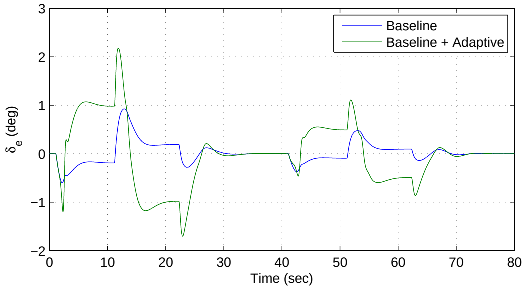
Baseline
Baseline + Adaptive
δ_e (deg)
Time (sec)

(f)



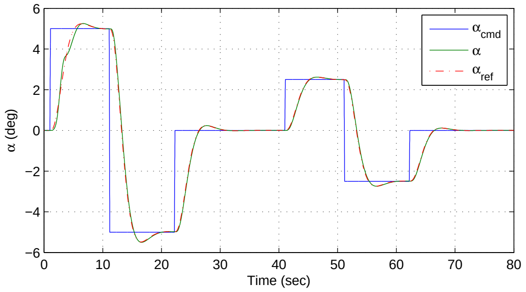
α_{cmd}
α
α_{ref}
α (deg)
Time (sec)

(g)



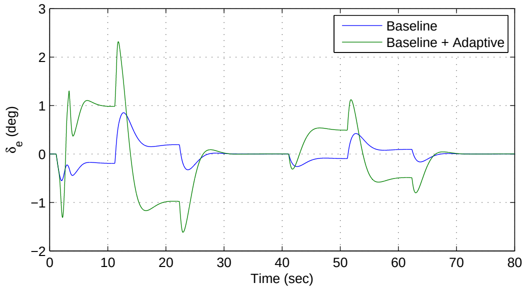
Baseline
Baseline + Adaptive
δ_e (deg)
Time (sec)

(h)



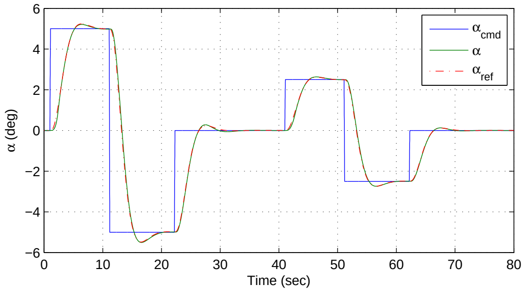
α_{cmd}
α
α_{ref}
α (deg)
Time (sec)

(i)



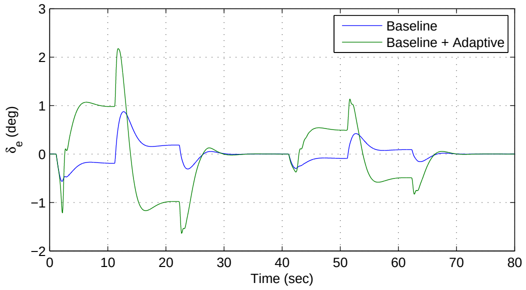
Baseline
Baseline + Adaptive
δ_e (deg)
Time (sec)

(j)



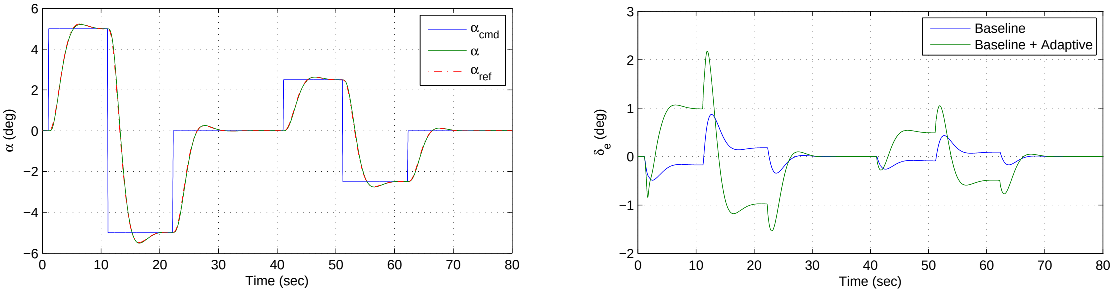
α_{cmd}
α
α_{ref}
α (deg)
Time (sec)

(k)



Baseline
Baseline + Adaptive
δ_e (deg)
Time (sec)

(l)



(m) **(n)**

Figure 4-10: Comparison of α command following and corresponding elevator deflection, for (a)-(b) Baseline control, (c)-(d) MRAC, (e)-(f) APC, (g)-(h) CMRAC, (i)-(j) BGF, (k)-(l) CAPC, (m)-(n) TDR adaptive control with $\tau = 80$ ms.

The results in table 4.2, figure 4-9 and figure 4-10 show that as each of the additional features is added to the Baseline controller, performance increases and cost decreases. The TDR adaptive controller has the best overall performance, even improving on the LQR* controller in terms of both tracking error and control usage. In the absence of any optimal control design techniques for systems with time delay, the LQR* controller is for the plant given by (4.75) without the time delay. Since there is indeed delay in the system, the LQR* is no longer optimal. The TDR controller explicitly accounts for the known time delay in the system, allowing it to achieve better performance than that of the LQR* controller.

250 ms time delay case

The second case under consideration is that of a 250 ms time delay. This amount of delay is possibly unreasonable for the F-16 case, as military specifications place tight restrictions on the allowable time delay in the loop. Nevertheless, it was selected in order to demonstrate the effectiveness of the proposed modifications for general systems with large time delays. The free parameters of the six adaptive controllers were redesigned for the larger time delay, again minimizing the cost function in (4.79). For the most part, this redesign consisted of reducing the adaptive gains to avoid exciting high-frequencies in the system. The adaptive gains for the 250 ms case are 5-10 times smaller than those of the 80 ms case. Non-dimensionalized errors, control usage, and control rate usage for the 250ms case can be found in table 4.3 and the cost associated with each of the seven controllers is shown in figure 4-11. Tracking performance and control usage for the controllers is shown in figure 4-12.

	Baseline	MRAC	APC	CMRAC	BGF	CAPC	TDR	LQR*
$\bar{e}$	198	2.75	2.04	1.69	1.43	1.47	1.12	**1.00**
$\bar{\delta}_e$	31.6	1.05	1.12	1.06	1.07	1.03	**1.00**	**1.00**
$\bar{\dot{\delta}}_e$	28.8	1.11	1.22	1.30	1.18	1.22	1.08	**1.00**

Table 4.3: Comparison of tracking error and control effort for the $\tau = 250$ ms case.

The addition of features to MRAC improves performance for the 250 ms case as well. In the 250 ms case, the relative difference between the various modifications seems to be diminished with the exception of the baseline controller,which is unstable for the 250 ms case.

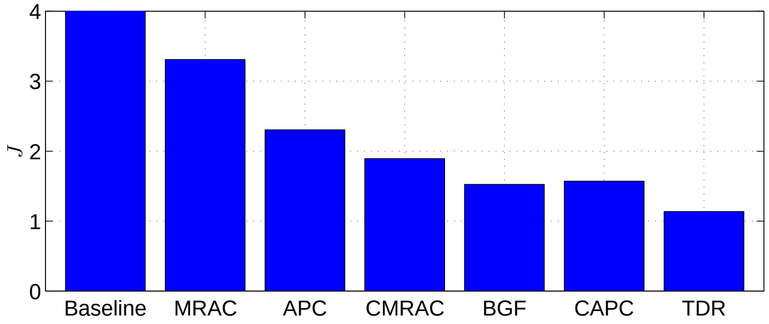
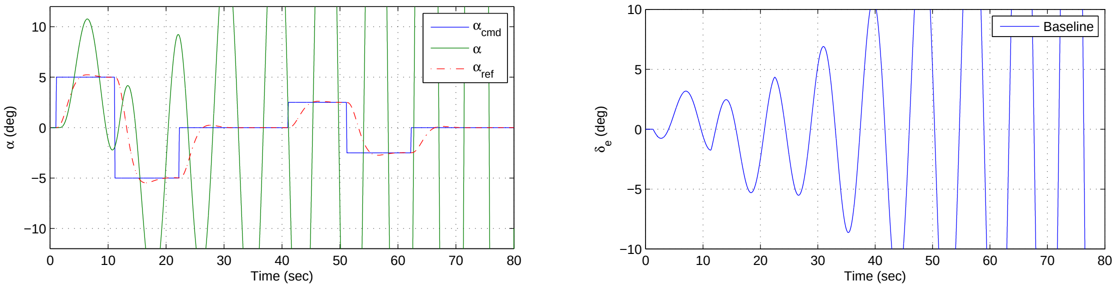
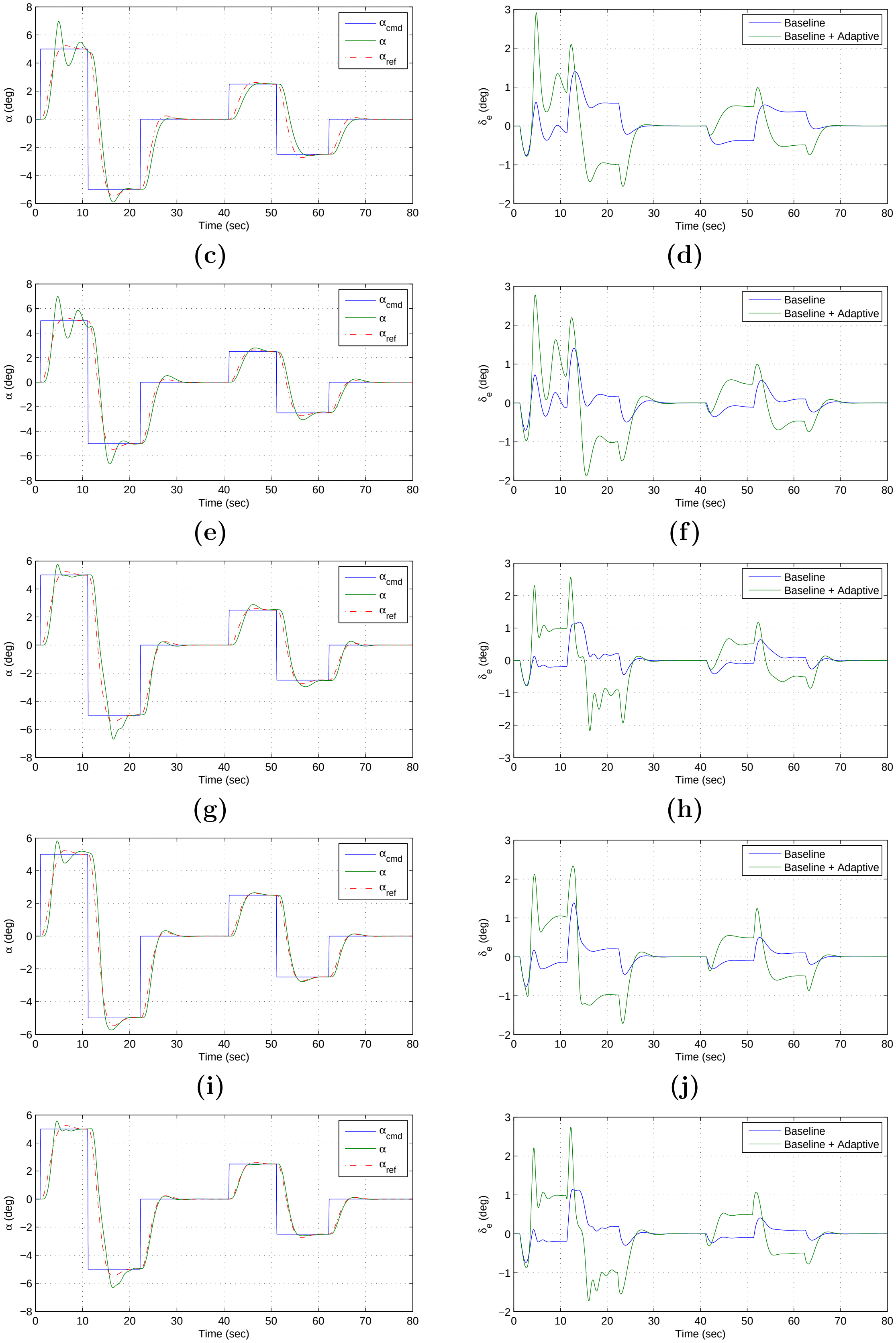


Figure 4-11: Comparison of the cost associated with each of the seven controllers for the $\tau = 250$ ms case. Note that the y-axis is cropped, the cost of the Baseline controller is well of the chart at about 14000 due to the instability.



(a) **(b)**


α (deg)
Time (sec)
α_{cmd}
α
α_{ref}
δ_e (deg)
Baseline
Baseline + Adaptive
0
10
20
30
40
50
60
70
80

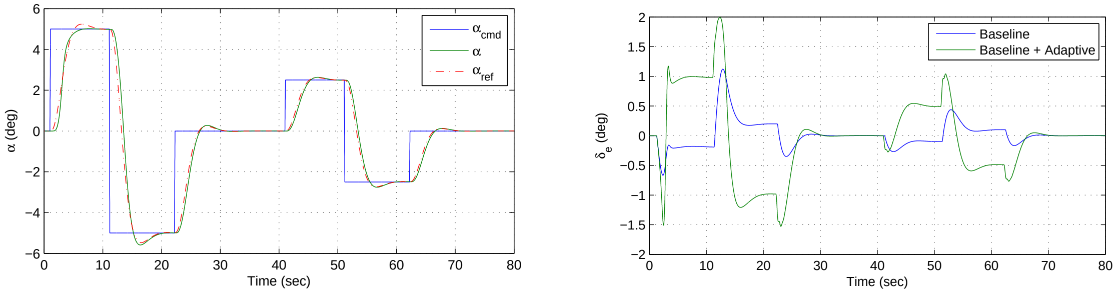
(c) (d)

(e) (f)

(g) (h)

(i) (j)

(k) (l)



(m) **(n)**

Figure 4-12: Comparison of α command following and corresponding elevator deflection, for (a)-(b) Baseline control, (c)-(d) MRAC, (e)-(f) APC, (g)-(h) CMRAC, (i)-(j) BGF, (k)-(l) CAPC, and (m)-(n) TDR adaptive control with $\tau = 250$ ms.

It should be noted that this improved performance does come at a cost in terms of computational efficiency. Table 4.4 shows how the number of differential equations for each of the seven controllers considered scales with system state n, number of control inputs m, and the number of posicast parameters $p = \tau/dT$.

	# of diff. eqs.
Baseline	$O(n)$
MRAC	$O(nm)$
APC	$O(nm + mp)$
CMRAC	$O(nm + m^2)$
BGF	$O(nm + m^2 + n^2)$
CAPC	$O(nm + m^2 + mp)$
TDR	$O(nm + m^2 + mp + n^2 + p^2)$

Table 4.4: Number of differential equations required for simulation of each of the seven controllers.

It may be possible to make optimizations to improve how well the various controllers scales, but adding additional features will always increase complexity and CPU time. In particular, the BGF modification adds terms of $O(n^2)$ and $O(p^2)$ which may be large for systems with many states, large delays, or fast sampling rates. Of the modifications to MRAC presented, the CMRAC approach is notable in that it offers a large increase in performance at very little additional computational cost.

4.2.4 Theory-driven design of TDR

Up until this point we have only discussed the selection of free parameters such as adaptive gains so as to minimize some cost function. The cost function given in (4.79) focuses predominately on performance. In this section, we utilize (4.74) to design the TDR adaptive controller with robustness to time delay in mind. In particular, we make some reasonable assumptions on the bound on ω and use the expressions in (4.72) to estimate the time delay margin τ^*. We then use the simulation model to experimentally calculate the true time delay margin by increasing the time delay in the system until instability is reached.

The regressor vector ω contains the state x, control inputs u, reference commands r, and τ/dT delayed versions of the control inputs and the reference commands. If we assume the aircraft remains in some region (the "flight envelope"), the state x is bounded. The control inputs are bounded due to actuator saturation and the reference commands are assumed to be bounded. Therefore ω can be assumed to be bounded using suitable values for the above components. It should be noted that since there are no guarantees on transient performance of the system, it is possible that the vehicle will move outside the pre-determined bounds on the flight envelope. Therefore this approach is only an approximate time delay margin.

Given a bound on $||\omega||$ and $||\omega_f||$ it is clear that through selection of the various free parameters in (4.72), the TDR adaptive controller can be designed to yield any approximate time delay margin. To examine the usefulness of this tool, a series of 8 sets of parameter choices were chosen to generate 8 TDR controller candidates with approximate time delay margins of τ_{design} = 5, 10, 20, 30, 40, 60, 80, and 100 ms. These controller candidates were then simulated using the F-16 simulations described above, for the $\tau = 80$ ms case. Each controller candidate successfully performed the maneuver and it was verified that the bounds on ω were satisfied at all times. Each of the candidates were then pushed harder by increasing the true time delay in the system with no redesign of the controller to determine the critical time delay τ_{crit} for which the system becomes unstable.

A new cost function which includes the approximate time delay margin is given by

$$J_2 = \frac{1}{3}\left(\bar{e}^T\bar{e} + \bar{\delta}_e^T\bar{\delta}_e + \dot{\bar{\delta}}_e^T\dot{\bar{\delta}}_e\right) + \frac{1}{\bar{\tau}^2}, \tag{4.81}$$

where $\bar{\tau}$ is the approximate time delay margin as calculated above divided by some normalizing factor τ_{goal}. Through adjustment of τ_{goal}, one can make the trade-offs between performance and robustness to time delays explicit.

Figure 4-13 shows the 8 controller candidates, plotted against the cost function J_2. These points represent the best possible selection of free parameters given the τ_{design} of each controller candidate. For small values of $\bar{\tau}$, the robustness term dominates J_2. As the control designs become more and more conservative, the decrease in performance begins to dominate the cost function. The resulting cost function allows for selection of one of the candidate controllers based on the trade-off between performance and robustness. In the example shown in figure 4-13, controller candidate 4 minimizes the cost function and is therefore the appropriate choice.

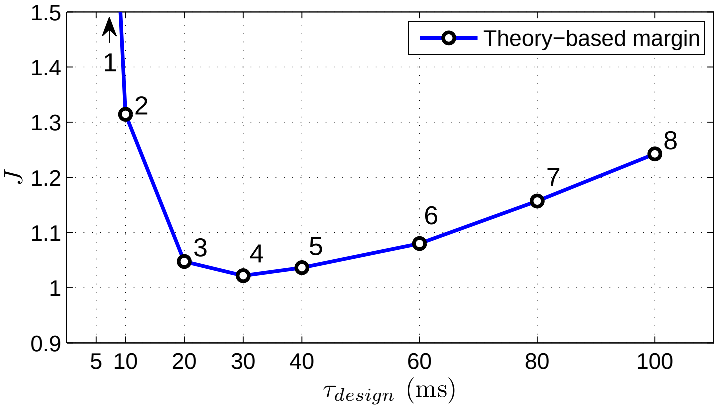


Figure 4-13: Cost associated with each of the 8 controller candidates plotted against τ_{design} with values of 5 ms to 100 ms. Cost is calculated using simulation with 80 ms time delay in the loop.

However, this tool is only useful if the approximate time delay margin accurately predicts the trend of the true time delay margin. Figure 4-14 shows a comparison of the theory-based margin τ_{design} and the simulation-based margin τ_{crit} for the controller candidates 1-8. The theory-based margins are smaller than their simulation-based counterparts, which is expected as the theory is conservative. It should also be noted that the simulation-based margin represents an upper bound on the true time delay margin since another set of initial conditions, reference commands, and uncertainties present may result in a smaller critical time delay.

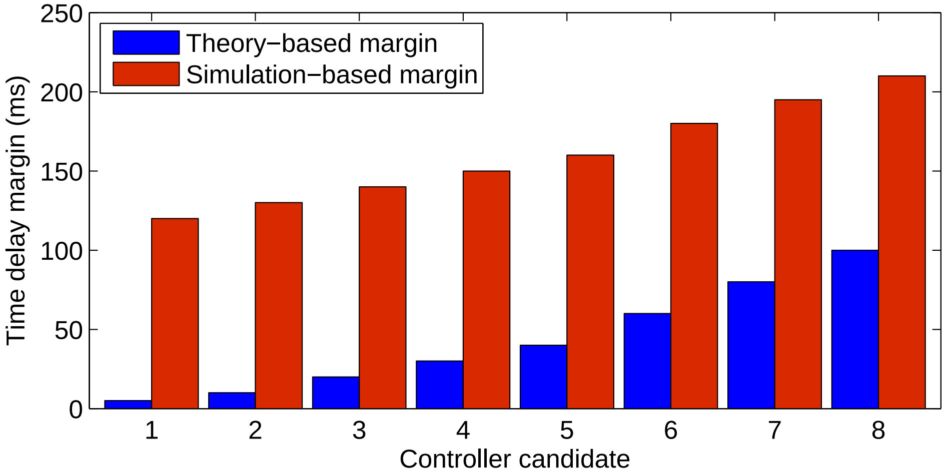


Figure 4-14: Comparison of theory-based margins τ_{design} and simulation-based margins τ_{crit} for controller candidates 1-8.

Figure 4-14 shows that the controller candidates 1-8 designed with increasing robustness to time delay as predicted by the theory are indeed increasingly robust to delays in the simulation.

4.2.5 Extensions to noisy systems and applications of the TDR approach

The simulation and theory described above neglects the contributions of sensor and process noise in the system. There are several methods for accommodating this noise. If the noise is uniform, bounded, white noise, a dead-zone approach similar to that described in section 3.5 can be employed, with similar implications on the theory. There have also been a number of papers on stochastic adaptive control [82–84], in which the Lyapunov functions described above are replaced by expected values of Lyapunov functions. For systems with high frequency noise that may excite unmodeled dynamics, a low-pass filter can be included in the system. A variety of filters exist for this purpose. Figure 4-15 shows the gain of several 4^{th} order low-pass filters. The inclusion of these filters introduces an additional phase lag, which has a detrimental effect on the time delay margins of the system. Figure 4-16 shows the group delay associated with the each of the 4^{th} order low-pass filters shown in figure 4-16. Since the group delay associated with each of these filters depends on the frequency of the input signals, we select τ_f as the supremum of the group delay shown in figure 4-16. For example, a 4th order Butterworth filter with a corner frequency of 20 Hz will introduce an additional delay $\tau_f = 50$ ms. The proof of stability for the TDR adaptive controller still holds if $\tau + \tau_f < \tau^*$. Thus, the free parameters of the adaptive system may need to be redesigned to accommodate the additional delay due to the filter.

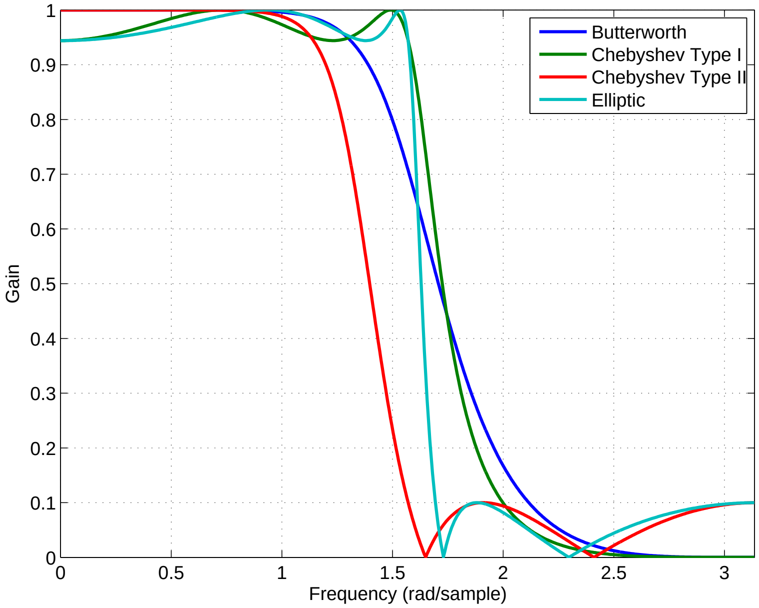


Figure 4-15: Amplitude of 4^{th} order Butterworth, Chebyshev Type I, Chebyshev Type II, and Elliptic low-pass filters.

The simulation example discussed in this chapter and the hardware test platforms in subsequent chapters are both flight platforms. However, the TDR adaptive approach is general in nature and can be applied to other systems of the form (4.24). There are many applications of this form that are subject to parametric uncertain-

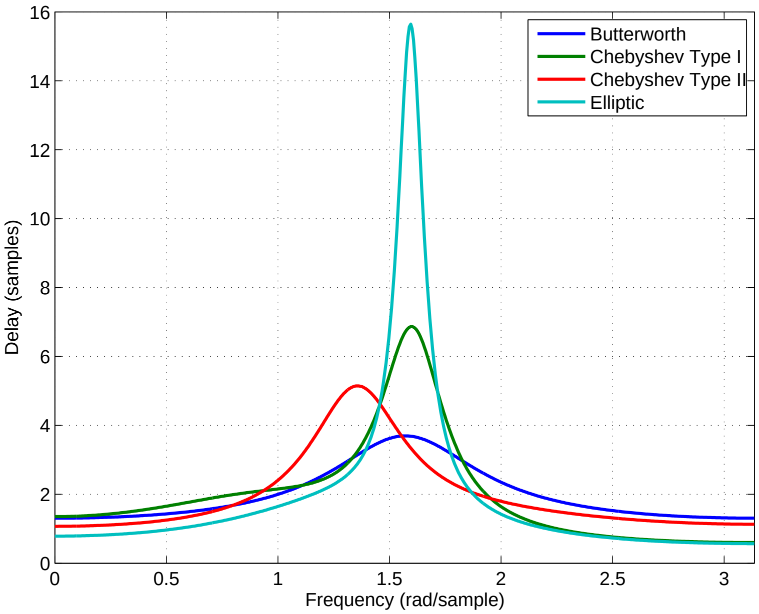


Figure 4-16: Group delay of 4^{th} order Butterworth, Chebyshev Type I, Chebyshev Type II, and Elliptic low-pass filters.

ties and potentially large time delays. Some examples include underwater robotics, automotive systems, teleoperated robotics, and space applications.

4.2.6 Summary

Several modifications to the typical MRAC approach were examined with application to systems with time delay. The modifications presented were either designed specifically to counter the effect of time delays, or had the effect of smoothing the adaptive or estimated parameters. The proposed TDR approach integrates all of these modifications into a coherent control structure which offers increased performance in simulation of an F-16 short period dynamics model. Typically in control systems there is a design trade-off between tracking performance and control power used. However, by utilizing these modifications to MRAC, it is possible to both increase tracking performance and decrease control effort as compared with MRAC. Furthermore, the TDR approach also has a proof of stability and an analytically justifiable time delay margin, which is bounded away from 0. It was shown that by making some assumptions on the system, an approximate time delay margin can be used to understand the robustness of the adaptive system and make informed decisions about the free parameters.

Chapter 5

Combined / composite adaptive control of UAVs with parametric uncertainties

In this chapter we build on the results in chapter 3 by transitioning a subset of the technologies examined in section 4.2 to flight test. In particular, the CMRAC approach showed several benefits in simulation, namely, smoother parameter estimates, improved parameter convergence, higher adaptive gains, and increased performance. Furthermore, CMRAC accomplished this while adding minimal computational cost, which is an important consideration for UAVs with embedded processors or limited onboard computation. The aim of the following study is to reproduce the benefits of CMRAC observed in simulation on a hardware system with a time delay in the loop. Robustness to losses of control effectiveness in regulation and tracking tasks will be examined.

This chapter is organized as follows. Section 5.1 describes the design of CMRAC for the quadrotor UAV. Section 5.2 discusses some of the unique features of the CMRAC approach with examples from flight recorded data. In section 5.3, we validate the CMRAC design through flight testing of a quadrotor UAV. Regulation and tracking problems are examined. Section 5.4 summarizes the results and gives some conclusions.

5.1 CMRAC design

The direct adaptive controller described in section 3.2 uses the error between the plant state, given by (3.7), and the reference model, given by (3.11) to adjust its parameters. To add the indirect adaptive components, we must start by generating a suitable prediction error $e_Y(t) = \hat{Y}(t) - Y(t)$. Following the procedure outlined in 4.2.2, we generate the signal $Y(t)$ as

$$Y(t) = \left(B_t^T B_t\right)^{-1} B_t^T \left(\lambda_f(x_t(t) - x_{t_f}(t)) - A_m x_{t_f}(t) - B_c r_f(t-\tau)\right). \tag{5.1}$$

Note that the value of $Y(t)$ can be computed at any time t using the state $x_t(t)$, the filtered state $x_{t_f}(t)$, and the reference signal delayed by τ seconds $r(t-\tau)$. The bilinear predictor model is given by

$$\hat{Y}(t) = \hat{\Lambda}(t)\left(u_f(t-\tau) + \hat{\theta}^T(t)\omega_f(t)\right). \tag{5.2}$$

Also note that the time-shifted signals in equations (5.1) and (5.2) require that the time delay τ be known. For the quadrotor system, this time delay has been measured to be 40 ms.

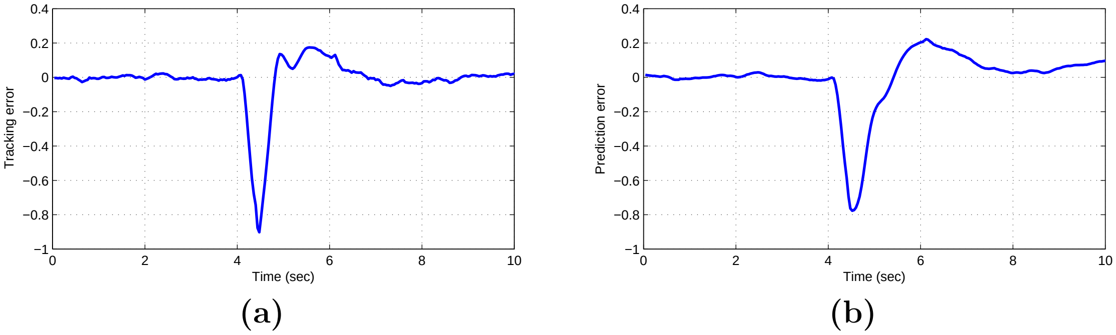
The indirect adaptive part is then added to the adaptive laws in (3.14), resulting in the CMRAC laws

$$\begin{aligned} \dot{\hat{\theta}}(t) &= \Gamma_\theta\left(\omega(t)e^T(t)PB_t - \omega_f(t)\gamma_c e_Y^T(t)\right), \\ \dot{\hat{\Lambda}}^T(t) &= -\Gamma_\Lambda\left(u_f(t) + \hat{\theta}^T(t)\omega_f(t)\right)\gamma_c e_Y^T(t). \end{aligned} \tag{5.3}$$

Note that the parameters are now adjusted according to the tracking error e, as in the MRAC case, as well as the prediction error e_Y. Additionally, the parameters $\hat{\Lambda}$ provide an estimate of the vehicle health. It can be seen that if the gain on the indirect part γ_c is set to 0, the above equations reduce to those of (3.14).

5.2 Properties of the CMRAC approach

Before comparing performance of the baseline, MRAC, and CMRAC approaches, it is useful to note some of the unique properties of the CMRAC system. First, it uses two sources of error information. Figure 5-1 shows flight-recorded error signals for the CMRAC system undergoing a sudden loss of thrust. A peak can be observed in both signals after the loss of thrust is initiated at $t = 4$ sec. Both error signals are then driven to 0 as the quadrotor stabilizes and returns to its hover position.

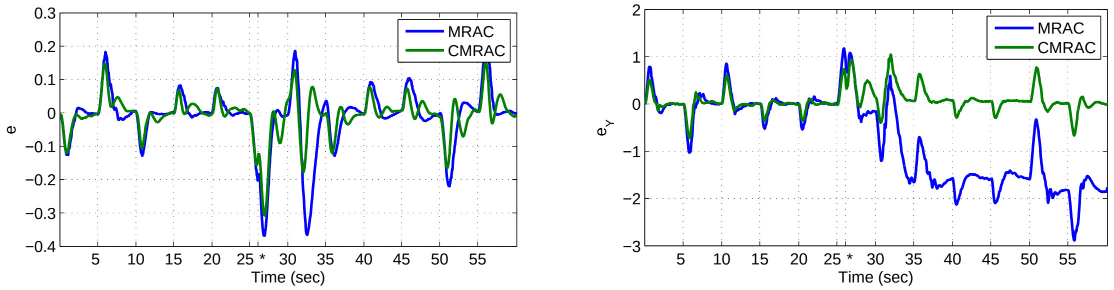


(a) **(b)**

Figure 5-1: The two sources of error information: (a) tracking error and (b) prediction error with CMRAC in the loop. A loss-of-thrust failure occurs at $t = 4$ sec.

Although a similar trend is captured in the two sources of error signals, there is unique information in the prediction error that is not contained in the tracking error signal. An illustrative example of this is shown in figure 5-2. This figure shows both tracking and prediction error signals for the MRAC and CMRAC approaches. The small spikes at intervals of 5 sec are caused by a series of step commands. The larger spikes around $t = 26$ sec are caused by a loss of collective thrust failure. Both the MRAC and CMRAC approaches drive tracking error back to the pre-failure values, as expected. However, MRAC is not able to drive prediction error back to 0 after the failure.

This implies that CMRAC has moved its parameters to different location than MRAC, one that simultaneously drives both tracking error and prediction error to 0. This is possible because the mapping from adaptive parameters to control inputs is a many-to-few mapping. For the quadrotor problem, there are 84 parameters and



(a) **(b)**

Figure 5-2: Comparison of MRAC and CMRAC (a) tracking errors and (b) prediction errors during a series of step commands. A loss-of-thrust failure occurs at approximately $t = 26$ seconds.

only 4 control inputs. It is therefore possible to achieve similar command-following performance with a variety of parameter estimates.

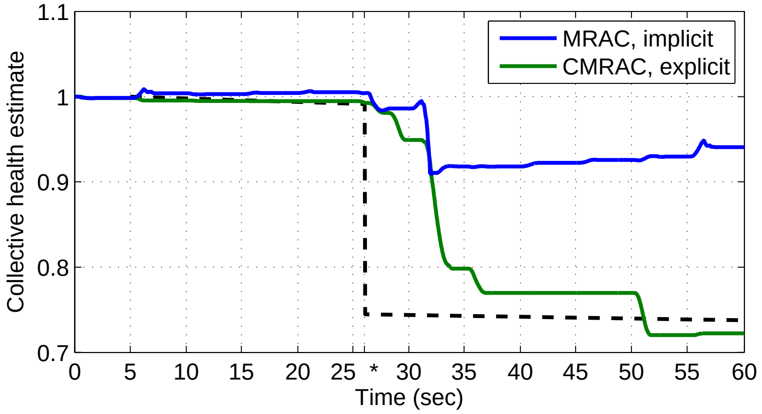
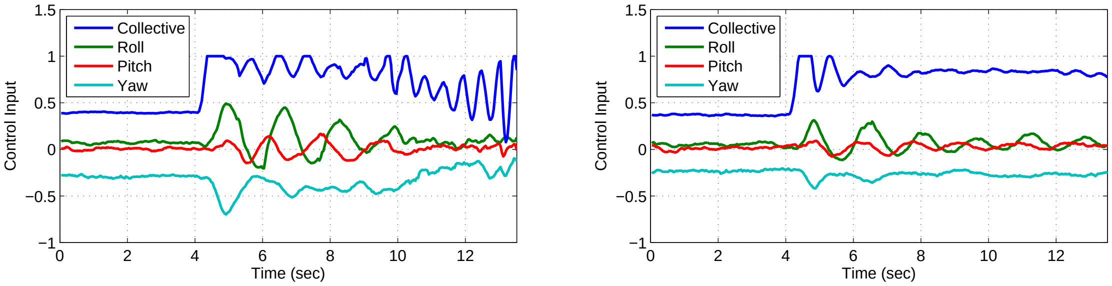


Figure 5-3: Collective health estimates for MRAC and CMRAC. The dashed line denotes the true collective health Λ_{11}.

Although MRAC does not provide an explicit estimate of the actuator effectiveness Λ, the value can be implicitly calculated from the parameter estimates $\hat{\theta}$. Figure 5-3 shows a comparison of the MRAC and CMRAC estimates for the collective health $\hat{\Lambda}_{11}$. The dashed line shows the true value of Λ_{11}, which is unknown to the controllers. Note the 25% reduction at approximately $t = 26$ sec, corresponding to the loss of collective thrust. Also note the slow decline of Λ_{11} both before and after the failure due to battery discharge. The CMRAC approach more accurately approximates the extent of the loss of control effectiveness.

Another feature of CMRAC observed in simulation of time delay systems was the ability to increase adaptive gains higher than those of MRAC without exciting high frequencies. Figure 5-4 shows flight recorded control input data from the MRAC and CMRAC approaches for a 25% collective failure with the same adaptive gains. Adaptive gains on z and $\dot{z}$ for both controllers were selected as $\gamma_z = 3$ and $\gamma_{\dot{z}} = 37.5$.



(a) **(b)**

Figure 5-4: Control inputs for (a) MRAC and (b) CMRAC with the same adaptive gains. A 25% loss of collective control effectiveness occurs at $t = 4$ sec.

Both MRAC and CMRAC increase their collective control inputs appropriately after the loss of thrust failure is injected at $t = 4$ sec. However, the MRAC approach exhibits undesirable oscillations which are not damped, eventually leading to instability.

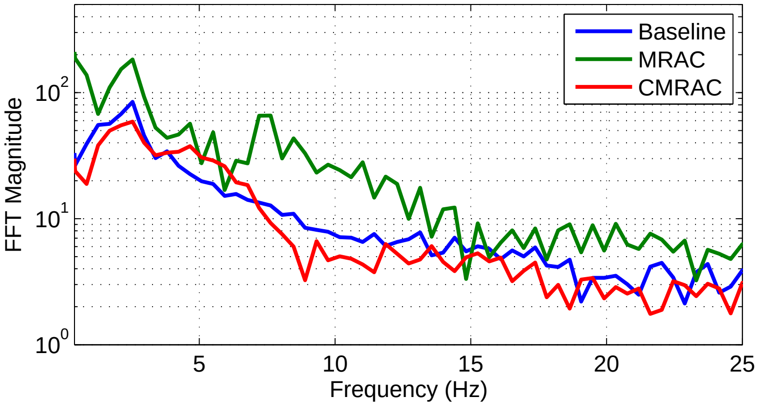


Figure 5-5: Fourier transform of the control inputs for the baseline, MRAC, and CMRAC approaches after a 25% loss of thrust.

Figure 5-5 shows the Fourier transform of the control inputs after the 25% loss of control effectiveness discussed previously. It can be seen that CMRAC offers a 70% power reduction over MRAC at 10 Hz, and has a control power usage similar to that of the baseline controller. This reduction of high frequency content in the control inputs can be attributed to the low-pass filtering effect of the indirect adaptive part of CMRAC. This effect can be clearly seen by rewriting the indirect part of the adaptive law as

$$\dot{\tilde{\Theta}}(t) = -\left(\Gamma\omega_2(t)\gamma_c\omega_2^T(t)\right)\tilde{\Theta}, \tag{5.4}$$

where the parameter error $\tilde{\Theta}^T(t) = \begin{bmatrix}\tilde{\Lambda}(t) & \Lambda\tilde{\theta}^T(t)\end{bmatrix}$, the adaptive gains $\Gamma = \begin{bmatrix}\Gamma_\Lambda & 0 \\ 0 & \Gamma_\theta\end{bmatrix}$. The indirect adaptive laws therefore take the form of a time-varying low-pass filter. Overall, it was observed that the CMRAC system was able to be tuned to 25-50% higher adaptive gains without exciting high frequencies.

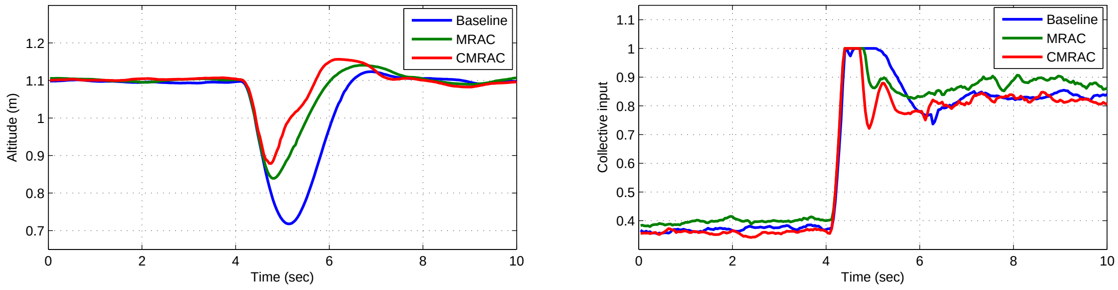
5.3 Flight test results

Flight tests were again conducted in the RAVEN testbed as described in section 3.4 using the baseline controller, MRAC design, and CMRAC design. The CMRAC controller was implemented in C and run alongside the baseline controller. Using this setup, it is possible to compare baseline, MRAC, and CMRAC. Adaptive gains were selected for the MRAC and the CMRAC approaches individually to give maximum performance without exciting high frequencies in the loop, as in the case of figure 5-4(a). The total delay in the loop, including computation and communication delays, is 40ms.

Two test scenarios were considered: a regulation problem and a tracking problem. The regulation problem involves maintaining hover at a desired altitude while the tracking problem involves the execution of a maneuver consisting of a series of altitude step commands. During each test, an actuator anomaly is injected into the system in the form of a loss of collective thrust, that is, a simultaneous loss of thrust in all four propellers. This is representative of a sudden change in mass, or alternatively a battery failure or electrical fault. The extent of the simulated failure corresponds to a 25% reduction in thrust. More severe failures would more effectively highlight the advantages of the MRAC and CMRAC approaches. However, thrust loss of more than 25% is not possible with this vehicle because it does not have the payload capacity to maintain hover with less than 75% of nominal thrust.

5.3.1 Regulation problem

Figure 5-6 shows flight recorded data of altitude regulation performance in the presence of a loss of collective thrust and 40 ms time delay in the loop. The three controllers being compared are the baseline controller, the MRAC approach, and the CMRAC approach. The reference command is to hover at an altitude of 1.1 m. A 25% loss of collective thrust occurs at $t = 4$ sec, unknown to the controllers.



(a) **(b)**

Figure 5-6: Altitude regulation performance in the presence of 40ms time delay and a sudden 25% loss of thrust at $t = 4$ sec. (a) Altitude, (b) collective control input.

It can be seen in figure 5-6(a) that the CMRAC approach recovers most quickly from the failure, followed by the MRAC approach and finally by the LQR controller. The norm of the tracking errors for the MRAC approach is 21% less than that of the baseline controller, and the norm of the tracking errors for the CMRAC approach is 29% less than that of the baseline. Figure 5-6(b) shows that all three controllers hit the actuator limits, but CMRAC is first to come off the limit, followed closely by MRAC, then later by the baseline controller. Because the baseline controller is slower at responding to the failure, it has to work significantly harder to reverse its downwards velocity and return to hover. Consequently, the baseline controller spends about twice as long at the actuator limits than the MRAC and CMRAC approaches. It should be noted that with all 4 motors spinning at their maximum angular velocity, the onboard rate damping signals and other stability augmentation loops are ineffective. Therefore, it is desirable to avoid saturating the actuators for extended periods of time.

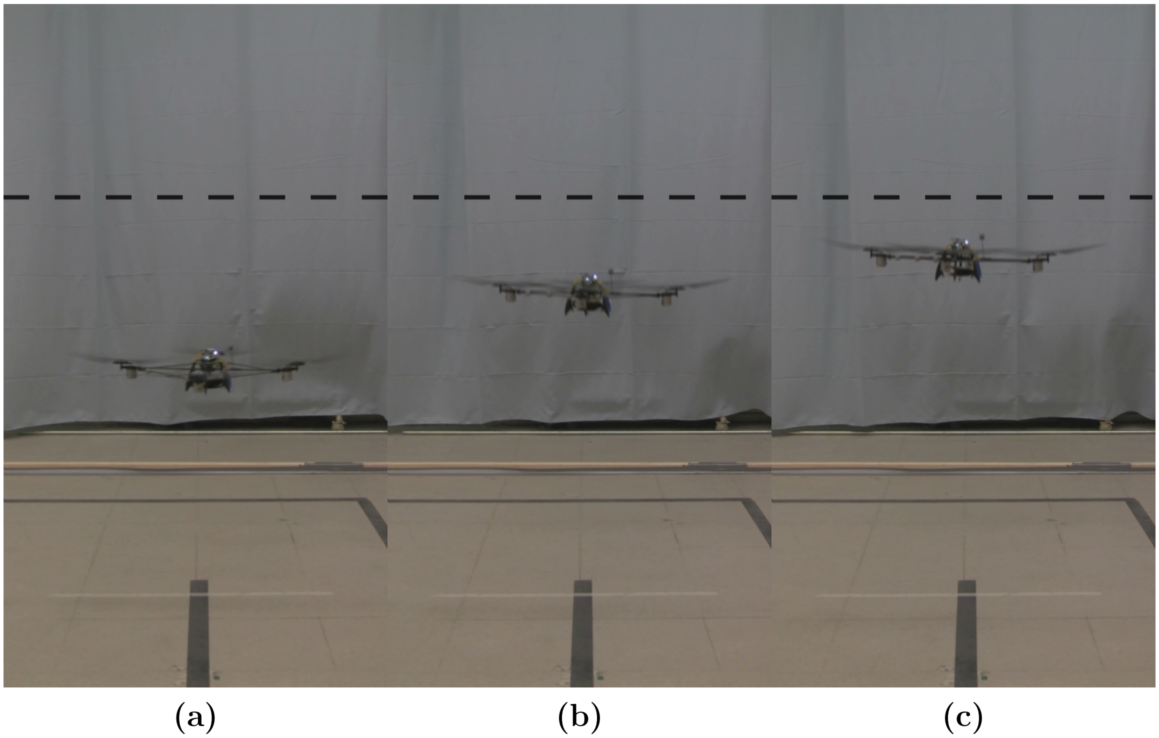

(a) **(b)** **(c)**

Figure 5-7: Snapshot of altitude regulation performance for (a) baseline control, (b) MRAC, and (c) CMRAC in the presence of 40ms time delay and a sudden 25% loss of thrust. The instant shown corresponds to the time at which the baseline controller reaches its minimum altitude. The dashed line represents the commanded altitude.

Figure 5-7 shows a screen capture of flight test videos [85] with side-by-side comparisons of the three controllers. The 3 flight tests depicted in figure 5-7 (a),(b), and (c) are synchronized so that the failure occurs in all three vehicles at the same instant in time. The moment shown in figure 5-7 represents the moment in which the baseline controller has reached its minimum altitude, or maximum departure from the commanded position. The MRAC and CMRAC vehicles have already begun climbing back to the commanded altitude, denoted by the dashed line. The baseline controller exhibits an altitude drop of approximately 38 cm, while the MRAC and CMRAC approaches drop in altitude by 27 cm, and 22 cm, respectively. When operating in cluttered or confined environments or flying in formation with other vehicles, these 10-20 cm can be the difference between colliding with an obstacle or another vehicle and safely avoiding any collisions.

5.3.2 Tracking problem

In the following set of flight tests the goal is command tracking, in this case to track a series of aggressive altitude step commands. During the course of the maneuver, a sudden loss of thrust in all four propellers occurs and the vehicles must recover and continue to track the commands. This approach provides insight about tracking performance and learning ability of the baseline, MRAC and CMRAC approaches.

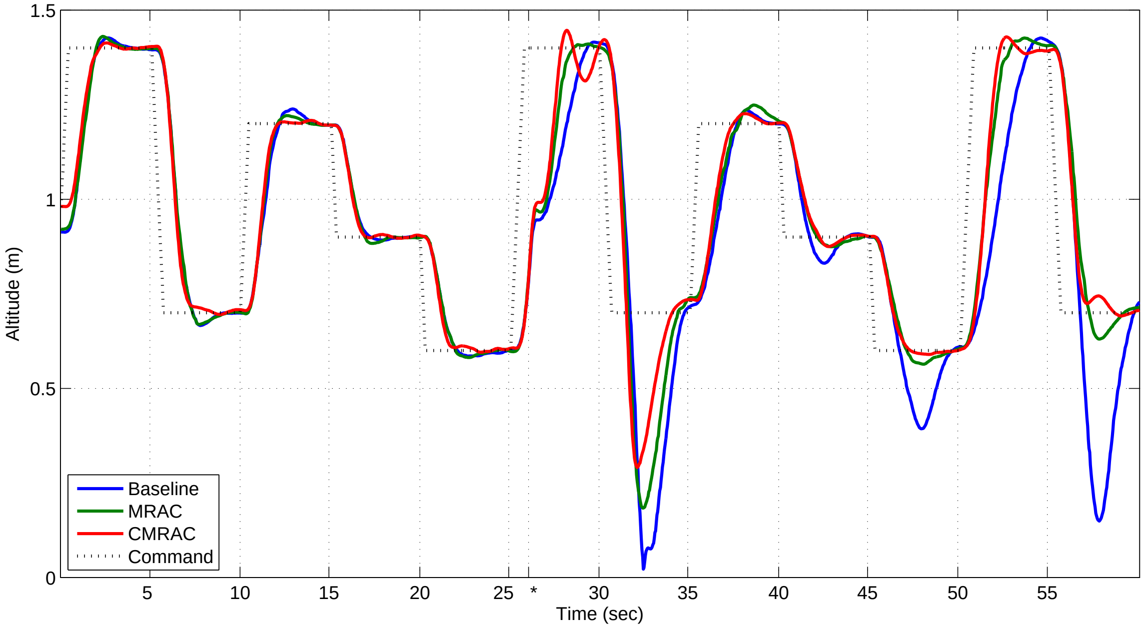


Figure 5-8: Altitude tracking performance of the baseline, MRAC, and CMRAC controllers. Loss of thrust uncertainty occurs at approximately $t = 26$ sec (denoted by the asterisk).

Figure 5-8 shows flight recorded altitude tracking data for the baseline, MRAC, and CMRAC approaches. The failure occurs at around $t = 26$ sec during an aggressive upwards step command. MRAC and CMRAC quickly accommodate the failure and continue to climb to the desired altitude. Even the baseline controller, which contains integral action on the altitude state, is able to complete the climb eventually.

However, the difference in performance of the three approaches is made clearer in the subsequent downwards step command. The spike in the baseline controller altitude at around $t = 33$ sec corresponds to the vehicle bouncing off the ground. Due to plastic feet on the bottom of the vehicle, it was not permanently damaged and was able to complete the maneuver. Both MRAC and CMRAC experience significant overshoot as well, with CMRAC being the lesser of the two. This suggests that neither adaptive controller had sufficiently learned the extent of the parameter uncertainty during the climb after the failure is inserted. This is unsurprising given that the constant altitude command is not persistently exciting and parameter convergence is therefore not guaranteed [7].

	Baseline	MRAC	CMRAC
Before Failure	63.03	62.47	**56.21**
Transient	82.68	68.27	**59.52**
After Failure	123.4	83.92	**71.96**

Table 5.1: Comparison of tracking error for the tracking problem with a 25% loss of collective control effectiveness.

To further analyze the wealth of information contained within the flight data shown in figure 5-8, we break up the time history into three flight conditions: before failure (t =0-25 sec), transient (t =25-35 sec), and after failure (t =35-60 sec). Table 5.1 shows the tracking performance of the three controllers in each of these three sections. CMRAC displays improved performance over all three flight conditions. The improvement in tracking performance even before the failure occurs can be attributed to the fact that the reference model is, in reality, slightly faster than the baseline system. The performance increases of CMRAC during the transient and after the failure can be attributed to the improved learning of the CMRAC approach, as well as the 25%-50% higher adaptive gains that are possible with CMRAC.

		Baseline	MRAC	CMRAC
Before Failure	Upwards	2.02 sec	1.86 sec	**1.82 sec**
	Downwards	1.95 sec	1.91 sec	**1.80 sec**
Transient	Upwards	4.11 sec	3.27 sec	**2.89 sec**
	Downwards	1.75 sec	1.64 sec	**1.56 sec**
After Failure	Upwards	3.19 sec	2.55 sec	**2.28 sec**
	Downwards	**1.71 sec**	2.03 sec	1.99 sec

Table 5.2: Comparison of rise time for the tracking problem with a 25% loss of collective control effectiveness.

Rise time and overshoot are shown in tables 5.2 and 5.3 respectively. In these tables, the flight is further broken down into upward steps and downward steps to highlight some of the differences between the three approaches. Table 5.2 shows that CMRAC generally has the lowest rise times. The only exception being the baseline controller during downward steps after the failure occurs. Note that the baseline controller also exhibits unacceptable overshoot for the downward steps, as seen in table 5.3.

In addition to examining the magnitude of these rise times, it is also informational to examine the difference between upward and downward steps for the various controllers. Before the failure occurs, all three controllers have similar rise times and "fall times." This is the expected result for a linear system. However, after the failure occurs, a disparity develops between the rise and fall times. For the baseline controller, this disparity is 1.48 sec in the after failure flight condition. For the MRAC controller it is 0.52 sec, and for CMRAC, 0.29 sec. CMRAC is able to learn the true values of the parameters better than MRAC, and it is therefore able to render the performance of the closed-loop adaptive system closer to that of the reference model (for which no distinction exists between upwards and downwards steps). The baseline controller, which has no learning capability outside of the integrator in the altitude loop, exhibits the largest disparity between rise times and fall times.

Table 5.3 shows the overshoot of the various controllers broken down in a similar fashion. CMRAC has the lowest overshoot with the exception of the climb portion of the transient flight condition, which is caused by a small oscillation at around $t = 28$ sec. Again, when comparing the disparity between upward overshoot and downward

		Baseline	MRAC	CMRAC
Before Failure	Upwards	7.78%	4.26%	**1.17%**
	Downwards	4.13%	5.39%	**1.07%**
Transient	Upwards	1.93%	**1.32%**	5.82%
	Downwards	96.9%	73.8%	**58.5%**
After Failure	Upwards	5.19%	6.55%	**4.46%**
	Downwards	56.9%	10.2%	**4.36%**

Table 5.3: Comparison of overshoot for the tracking problem with a 25% loss of collective control effectiveness.

overshoot ("undershoot") that exists in the after failure section, CMRAC appears to behave more closely to the linear reference system than MRAC and the baseline controller. Since the baseline controller has very limited learning, it continues to exhibit very large undershoots, particularly noticeable around $t = 57$ sec on figure 5-8. Of the three controllers compared, the CMRAC approach displays improved tracking error, decreased rise times, and decreased overshoot over all three flight conditions.

5.4 Summary

In this chapter, the application of CMRAC to a quadrotor UAV with time delay in the loop was presented. It was shown that, similar to the simulation results of previous chapters, the CMRAC approach delivered smoother parameter estimates, allowed for higher adaptive gains, and increased performance over that of MRAC alone. It was shown that CMRAC was more effective than MRAC in learning the true value of uncertain parameters in the system, offering numerous benefits in terms of tracking performance.

Chapter 6

Adaptive configuration control of multi-vehicle UAS

Several missions such as surveillance, exploration, search-and-track, and lifting of heavy loads are best accomplished by multiple UAVs. Another important advantage to utilizing multiple vehicles is a reduction in the risk to successful completion of a mission due to the loss of a single vehicle. When a single vehicle malfunctions, neighboring vehicles can adjust their configuration to compensate. This increased robustness can lead to a commensurate decrease in vehicle specifications and cost, further improving the argument for swarm operations. The methodology discussed in previous chapters is concerned with the stability and performance of a single plant. In this chapter, we address the question of how one can ensure that vehicles, with each vehicle prescribed a specific position (and possibly orientation) maintain those positions (and orientations) with respect to other vehicles in the presence of parametric uncertainty and time delay.

Malfunction in a vehicle represents a wide range of scenarios. And many of these scenarios include instances where the vehicle failure is partial. That is, a vehicle fails or is damaged, but is still able to contribute to the mission in some reduced capacity and possibly return and land safely for repairs. This chapter discusses the development of an adaptive configuration control approach that accommodates partial failures and carries out the requisite mission with multiple UAVs satisfactorily.

Distinguishing features of this architecture are that it fully retains an advanced inner-loop control structure and performs coordinated control actions using information from distributed sources, using local and global performance measures. By including adaptation at both the vehicle level and the collective level, we ensure that any uncertainties have minimal effect on vehicle performance and significantly enhance the ability of the collective group to accomplish the overall mission goals.

This chapter is organized as follows. Section 6.1 describes the class of multi-UAV problems being addressed in this chapter. In section 6.2 describes the novel adaptive configuration controller developed to solve the class of multi-UAV missions in general and for the specific problem of planar, circular motion of a fleet of UAVs. Section 6.3 describes a series of simulation results for the multi-UAV problem that motivate the application of the adaptive configuration control approach to the hardware UAS. In section 6.4, we validate the adaptive configuration control using flight tests of a UAS consisting of several quadrotor UAVs. Section 6.5 summarizes the results and gives some conclusions.

6.1 Problem statement

As the number of vehicles participating in a multi-UAV mission increases, the complexity of coordinated motion through space increases as well. The overall multi-UAV control objectives can be broken down into four nested levels of increasing order of scope that can be described thusly: 1) control of the individual vehicle dynamics, or inner-loop control, which is typically performed by an onboard controller or autopilot; 2) control of the individual vehicle kinematics, commonly referred to as outer-loop control and often accomplished by a ground-based path-planner that provides waypoints for the vehicle; 3) coordinated control of the configuration of the collective of vehicles, enabling specific behaviors on the global level such as formation flight; and finally 4) mission management, focusing on questions of refueling, allocating resources, and how to best achieve overall mission objectives. These four levels of control are summarized in figure 6-1.

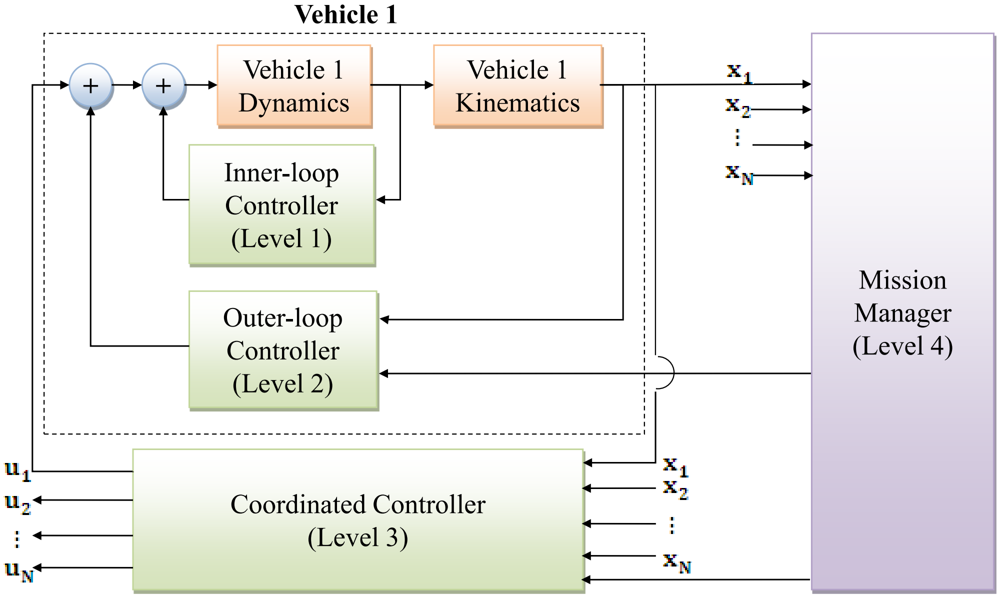


Figure 6-1: Schematic diagram of several nested control loops in a multi-vehicle UAS. Vehicle 1 (shown within in the dotted line) consists of an inner-loop controller, which stabilizes the vehicle dynamics, and an outer-loop controller which allows the vehicle kinematics to follow waypoint commands. Vehicles 2 through N are omitted for clarity.

The inner-loop adaptive controller developed in previous chapters addresses level 1, while the outer-loop controller discussed in this chapter focuses on levels 2 and 3. The specific objectives are to carry out coordinated control for a UAS consisting of a fleet of several UAVs so as to maintain a desired configuration in the presence of individual vehicle uncertainties in level 1. We will address this problem using an adaptive approach which suitably utilizes local and global information to carry out trajectory control, path planning, and configuration control.

We assume there are N vehicles flying in formation, and each vehicle has an inner loop controller that stabilizes the vehicle dynamics. This inner loop controller can be adaptive, such as the adaptive inner-loop controllers discussed in chapters 3 and 5, or linear, such as the baseline approaches also discussed in those chapters. The main task of the inner loop controller is provide stabilization and reject disturbances. The vehicles are also assumed to have some sort of outer-loop controller that allows for tracking of waypoints in $\Re^n$. The waypoints commanded for each vehicle are then $X_{i,cmd} \in \Re^n$ for $i = 1 \ldots N$. Waypoints may consist of positions and attitude, also know as the vehicle "pose". These waypoints are generated by the outer-loop and coordinated controller (levels 2 and 3 in figure 6-1). The primary control goals for the formation-flying problem, in order of priority, are to reduce the likelihood of collisions, complete the mission successfully, reduce inter-vehicle distance errors, and reduce individual vehicle tracking errors. The problem at hand is thus generation of waypoint commands for all vehicles which accomplish the aforementioned goals in the presence of uncertainties such as actuator failures or battle damage.

6.2 Adaptive configuration control

In this section we develop the adaptive configuration control approach to suitably generate waypoint commands for all vehicles in the UAS. Analogous to the design approach for adaptive control of a single vehicle, using MRAC for example, we first generate a suitable error signal. We then determine a parametrization of the uncertainty and its effect on the system. Next, we determine an algorithm for the adjustment of control parameters using online measurements of the error signals. We begin by presenting the algorithm for the general problem involving formation flight of multiple UAVs. We then discuss the specific problem of planar, circular formation flight of multiple quadrotor UAVs circling a central point. We then follow up the specific case with a simulation study and flight tests in the RAVEN testbed.

6.2.1 General approach

We first address the general problem of waypoint generation for multiple vehicles executing a coordinated maneuver through space. Consider a waypoint generation law of the form

$$\begin{aligned} X_{i,cmd}(t) &= f\left(e_{form}\right) + X_{i,des}\left(\eta(t)\right), \\ \dot{\eta}(t) &= 1 - g(e_{track}), \end{aligned} \tag{6.1}$$

where the function $f\left(e_{form}\right)$ acts as a position bias term and the function $g(e_{track}) \in [0,\ 1]$ determines the velocity profile of the vehicle along some prescribed trajectory $X_{i,des}(t)$ through adjustment of the temporal coordinate $\eta(t)$. The two distinct errors e_{form} and e_{track} encapsulate the effect of uncertainties on both path planning and formation keeping. The former is global in nature, and addresses the effects of vehicular uncertainty on distances between neighboring vehicles, while the latter concerns the effects of failures on the individual vehicle and is therefore more local in character.

To generate the global formation-keeping error term e_{form}, we first must encode the desired formation shape and size in the form of desired relative distances between vehicle i and vehicle $i+1$ as a function of time. Let

$$\begin{aligned} X_{i,form}(t) &= X_{i,des}(\eta(t)) - X_{i+1,des}(\eta(t)), \quad \text{for } i = 1...N-1, \\ X_{N,form}(t) &= X_{N,des}(\eta(t)) - X_{1,des}(\eta(t)). \end{aligned} \tag{6.2}$$

The formation error contribution from each vehicle $e_{i,form}$ can then be written as

$$\begin{aligned} e_{i,form}(t) &= (X_i(t) - X_{i+1}(t)) - X_{i,form}(t), \quad \text{for } i = 1...N-1, \\ e_{N,form}(t) &= (X_N(t) - X_1(t)) - X_{N,form}(t), \end{aligned} \tag{6.3}$$

where $X_i(t)$ is the pose of the i^{th} vehicle. To develop the function $f\left(e_{form}\right)$, we first calculate a position bias term X_{bias} that minimizes the largest $e_{i,form}$ over all vehicles, that is

$$X_{bias} = \underset{X}{\operatorname{argmin}} \left(\max_i (e_{i,form} - X) \right). \tag{6.4}$$

The position adjustment function $f(e_{form})$ is then given by

$$f(e_{form}) = \begin{cases} X_{bias}, & \text{for } ||X_{bias}|| \leq X_{max}, \\ \frac{X_{max}}{||X_{bias}||} X_{bias} & \text{otherwise}, \end{cases} \tag{6.5}$$

where X_{max} is a positive constant bound on the region of interest within the controlled space. This function has two important properties. First, the function is bounded by X_{max}. Second, within the region of interest this adjustment law minimizes the maximum inter-vehicle distance error. Assuming all vehicles move in all directions equally quickly, this adjustment also guarantees that all vehicles will return to formation in minimum time.

We now develop the local error term e_{track}, which can be viewed as the tracking error of the vehicle kinematics, and the function $g(e_{track})$, which adjusts the vehicle's velocity profiles. It can be seen from the waypoint law (6.1) that if $g(e_{track}) = 0$, the temporal scaling factor $\dot{\eta}(t) = 1$, meaning that the commanded trajectory $X_{i,des}$ is followed at nominal speed. However, as $g(e_{track}) \to 1$, $\dot{\eta}(t)$ goes to 0, meaning the commanded velocities go to 0 as well. In this way, the velocity of the vehicles is decreased when the tracking performance becomes poor. We can write the local error term e_{track} for each vehicle as

$$e_{i,track}(t) = X_i(t) - X_{i,cmd}(t), \quad \text{for } i = 1...N, \tag{6.6}$$

where $X_{i,cmd}(t)$ is the commanded waypoint at time t. One possible choice of the velocity adjustment function $g(e_{track})$ is

$$g(e_{track}) = \frac{1}{2}\left[\tanh\left(2\alpha\left(\sum_i^N e_{i,track}^T Q_i e_{i,track} - \beta\right)\right) + 1\right], \tag{6.7}$$

where α, β, and Q_i are free parameters. This is a tunable sigmoid function with values bounded between 0 and 1, as shown in figure 6-2. The parameter β determines the

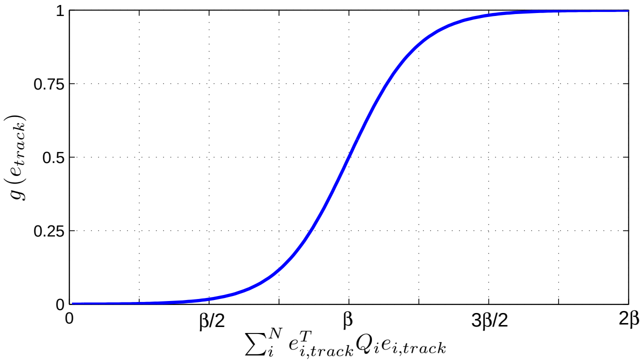


Figure 6-2: Tunable sigmoid function with parameter β determining the 0.5 crossing and α determining the slope at the 0.5 crossing.

distinction between acceptable and unacceptable tracking errors. The parameter α determines the slope at $g(\beta)$. For large α, $g(e_{track})$ becomes closer to a binary-valued function. Smaller values of α result in a smoother adjustment of the vehicle velocity profiles. The parameter Q_i weights individual error states on a per vehicle basis.

It should be noted that both $f(e_{form})$ and $g(e_{track})$ are bounded functions, and therefore the waypoints generated by (6.1) are bounded as well. If the inner-loop controller stabilizes the vehicle dynamics, the overall closed-loop system is then guaranteed to be stable.

6.2.2 Specific approach

The previous discussions represent a very general multi-vehicle task. In this section we present one potential example of the adaptive configuration control methodology. The following example involves planar, collective motion of a UAS consisting of multiple quadrotor UAVs. The goal is for a formation of vehicles to circle a central pole while keeping a constant heading angle as shown in figure 6-3. For quadrotor helicopters, this maneuver involves constant adjustment of the thrust of each of the four propellers to achieve the necessary pitch and roll moments. We only consider the case of constant radius r and constant height h, hence the quadrotor waypoint commands are of the form

$$\begin{bmatrix} x_{i,cmd} \\ y_{i,cmd} \\ z_{i,cmd} \\ \psi_{i,cmd} \end{bmatrix} = \begin{bmatrix} r\cos\theta_{i,cmd} \\ r\sin\theta_{i,cmd} \\ h \\ 0 \end{bmatrix}, \quad \text{for } i = 1...N. \tag{6.8}$$

Therefore the commanded space is $X_{i,cmd} \equiv \theta_{i,cmd}$ in this case.

The desired trajectory for this maneuver is given by

$$\theta_{i,des}(t) = \omega t + \theta_{i,0}, \tag{6.9}$$

where $\theta_{i,0} = i\frac{2\pi}{N}$. This corresponds to circular motion with constant angular velocity ω and an equal spacing between vehicles. The formation error is then a measure of

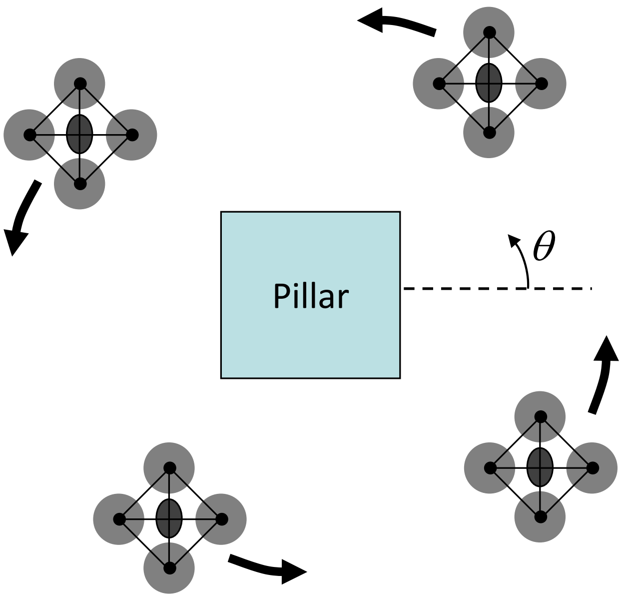


Figure 6-3: Planar circular formation of four quadrotor UAVs. The direction of motion is counter-clockwise when viewed from above.

the differences in phase angles of the vehicles and is given by

$$\begin{aligned} e_{i,form}(t) &= (\theta_i(t) - \theta_{i+1}(t)) + \frac{2\pi}{N}, \quad \text{for } i = 1...N-1, \\ e_{N,form}(t) &= (\theta_N(t) - \theta_1(t)) + \frac{2\pi}{N}. \end{aligned} \tag{6.10}$$

The tracking error for the i^{th} vehicle is given by

$$e_{i,track} = \theta_i - \theta_{i,cmd}, \tag{6.11}$$

and the waypoints $\theta_{i,cmd}$ are generated according to the waypoint generation law

$$\begin{aligned} X_{i,cmd}(t) &= f\left(e_{form}\right) + \theta_{i,des}(\eta(t)), \\ \dot{\eta}(t) &= 1 - g(e_{track}), \end{aligned} \tag{6.12}$$

where the function $f\left(e_{form}\right)$ is given by

$$f(e_{form}) = \begin{cases} \theta_{bias}, & \text{for } ||\theta_{bias}|| \leq \theta_{max}, \\ \frac{||\theta_{max}||}{||\theta_{bias}||}\theta_{bias}, & \text{otherwise,} \end{cases} \tag{6.13}$$

where $\theta_{max} = \pi$, since $\theta \in [-\pi,\ \pi]$, and θ_{bias} is given by

$$\theta_{bias} = \underset{\theta}{\operatorname{argmax}} \left(\max_i (e_{i,form} - \theta) \right). \tag{6.14}$$

The function $g\left(e_{track}\right)$ is given by

$$g(e_{track}) = \frac{1}{2}\left[\tanh\left(2\alpha(\sum_i^N e_{i,track}^T Q_i e_{i,track} - \beta)\right) + 1\right], \tag{6.15}$$

with Q_i selected as unity for all i, meaning all vehicles are given equal weight in determining whether or not tracking errors are acceptable.

6.3 Simulation results

We now discuss a series of simulation results using the linear quadrotor model given by (3.2), the inner-loop MRAC approach described in section 3.2 and the adaptive configuration controller described in section 6.2.2. The decision to use the linearized quadrotor model as opposed to a fully nonlinear approach was made to decrease computational complexity. In this section we use a large number of these simulations, which each contain multiple vehicles and obstacles, to describe the behavior of the UAS over a range of potential failure scenarios.

As a yardstick to which we can compare the adaptive outer-loop approach, we generate a baseline outer-loop controller by taking $f(e_{form}) \equiv 0$ and $g(e_{track}) \equiv 0$. Waypoints are therefore generated according to $\theta_{i,des}(t)$, with no modification. Since the inner-loop controller can be either baseline or adaptive as well, there are four possible controller combinations. Each of these combinations will be simulated for the task described in 6.2.2, with a 75% loss of thrust failure inserted into the left-most propeller of one of the vehicles at $\theta_{fail} = -2.06$ rad, shown in figure 6-4 as the red-tinted propeller.

Tables 6.1 and 6.2 show tracking errors and formation error for the four possible controller combinations discussed. All values are normalized by those of the baseline

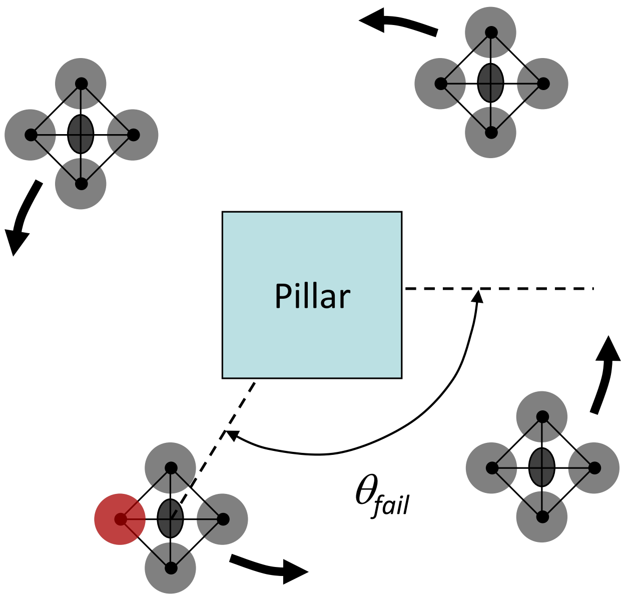


Figure 6-4: Planar circular formation of four quadrotor UAVs. A loss of thrust failure occurs in the left-most propeller at θ_{fail}.

inner-loop/baseline outer-loop case. Adaptation in the inner-loop appears to decrease tracking error, while both inner- and outer-loop adaptation decrease the formation error. This is expected since the primary function of the inner-loop adaptive controller is to reduce model reference error in the presence of failures. Since the model dynamics track the waypoint commands closely, the this has the effect of reducing e_{track} as well.

		Outer loop	
		Baseline	Adaptive
Inner loop	Baseline	1.00	0.99
	Adaptive	**0.72**	0.73

Table 6.1: Tracking error for the simulated four-UAV circular motion problem. A 75% loss of thrust failure is inserted into the left-most propeller at $\theta_{fail} = -2.06$ rad. Adaptation in the inner-loop reduces tracking error in this case.

Besides reducing tracking and formation errors, another important goal of including adaptation in the outer-loop is to reduce the likelihood of collisions with other vehicles and with obstacles in the environment. In the next series of tests, we execute the same circular motion task, this time initiating the failure at values of θ_{fail} ranging from $-\pi$ to π. Since vehicles are commanded to remain at a constant heading, the

		Outer loop	
		Baseline	Adaptive
Inner loop	Baseline	1.00	0.55
	Adaptive	0.48	**0.45**

Table 6.2: Formation error for the simulated four-UAV circular motion problem. A 75% loss of thrust failure is inserted into the left-most propeller at $\theta_{fail} = -2.06$ rad. Adaptation in both the inner- and outer-loop reduces formation error.

failed actuator is always in the negative x-direction (the left-most propeller, when viewed from above). Thus, different values of θ_{fail} will generate different behaviors, and the potential for different collisions (or combinations of collisions) with the pillar, the trailing, leading, or opposite quadrotors. Collision dynamics are not included, and vehicles involved in collisions are not removed from the remainder of the simulation. It is therefore possible for one vehicle experience several collisions over the course of one test.

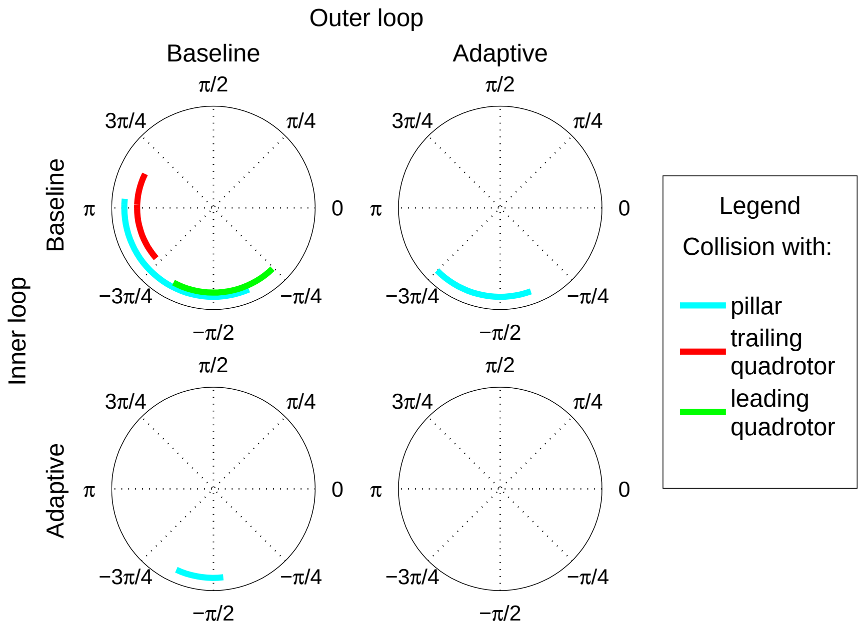


Figure 6-5: Dependence of collisions on θ_{fail} for the 75% loss of controller effectiveness case. Regions of θ_{fail} for which collisions occur are denoted by the colored arcs as described in the legend at right.

Using a sufficiently large number of simulations, we explore the all possible angles at which the uncertainty is initiated, $\theta_{fail} \in$. Figure 6-5 shows the dependance of collisions on θ_{fail}. Colored arcs here represent the region in which a particular collision occurs, and the arc lengths correspond to the size of each region. The baseline inner-loop/baseline outer-loop approach has the largest region in which collisions occur, 46% of the entire space. This implies that were an actuator failure to occur at a random time instant while this task is taking place, there is a 46% chance that a collision between vehicles or with obstacles would occur. Including adaptation in the inner- or outer-loop reduces the size of this region, and including adaptation in both loops erases it completely.

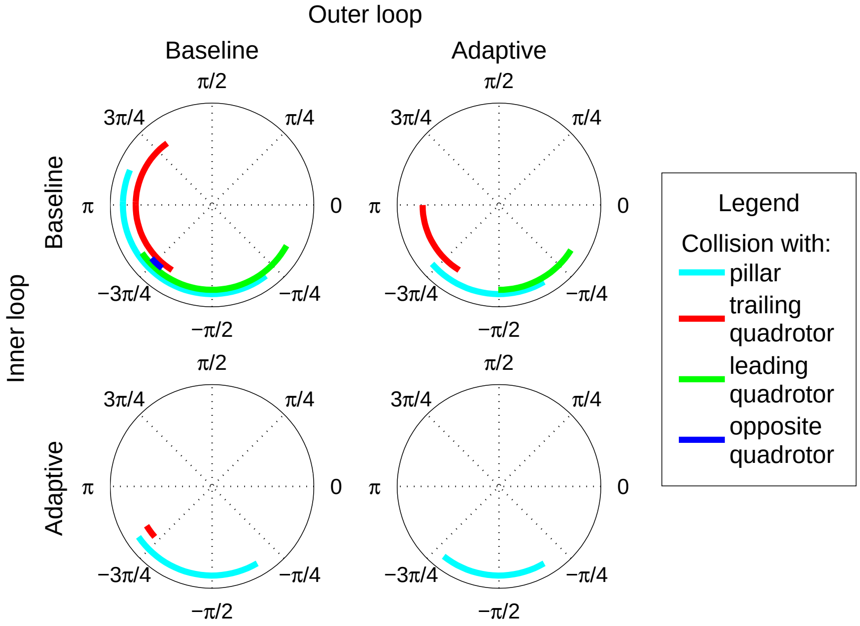


Figure 6-6: Dependence of collisions on θ_{fail} for the 80% loss of controller effectiveness case. Regions of θ_{fail} for which collisions occur are denoted by the colored arcs as described in the legend at right.

With a more severe 80% loss of control effectiveness, we can observe the same trend in figure 6-6. Because of the more severe failure, the vehicle experiences larger departures from the commanded positions and the likelihood of collisions increases for each of the four controller combinations. For the baseline inner-loop/baseline outer-loop combination, the region is 58% of the total space. There is also small region around $-3\pi/4$ in which the failed vehicle collides with every obstacle in the space. For the adaptive inner-loop/adaptive outer-loop approach, collisions with all other vehicles are eliminated, and the likelihood of collision with the pillar is 20%.

These simulation results show that including adaptation in the outer-loop has the potential for increasing performance in terms of tracking error and formation error, and also decreasing the likelihood of collisions.

6.4 Flight test results

We now aim to reproduce the simulation results from the previous section on a flight test platform with a hardware UAS that includes all nonlinearities as well as a time delay of 40 ms in the loop. In these tests, three vehicles were used to execute the mission described in section 6.2.2. A loss of collective control effectiveness of one vehicle's propellers will then be injected during the course of the task.

Tables 6.3 and 6.4 show tracking error and formation error results from the flight recorded data. Again, all values are normalized by those of the baseline inner-loop/baseline outer-loop case. As in the simulations, including adaptation in the inner-loop significantly reduces tracking error. In the flight tests it appears that outer-loop adaptation also decreases tracking error somewhat, with the adaptive inner-loop/adaptive outer-loop approach having 37% less tracking error than the baseline inner-loop/baseline outer-loop approach.

		Outer loop	
		Baseline	Adaptive
Inner loop	Baseline	1.00	0.82
	Adaptive	0.69	**0.63**

Table 6.3: Tracking error for the three UAV circular motion flight test. A 50% loss of thrust failure is inserted into the left-most propeller at $\theta_{fail} = -0.79$ rad. Adaptation in both in inner- and outer-loops decreased the tracking error in this case.

The formation error flight test results line up similarly to those of the simulation, with the adaptive inner-loop/adaptive outer-loop approach reducing formation error by 62%.

Flight test videos for the baseline inner-loop/baseline outer-loop and adaptive inner-loop/adaptive outer-loop cases have been posted online [86]. Screen captures

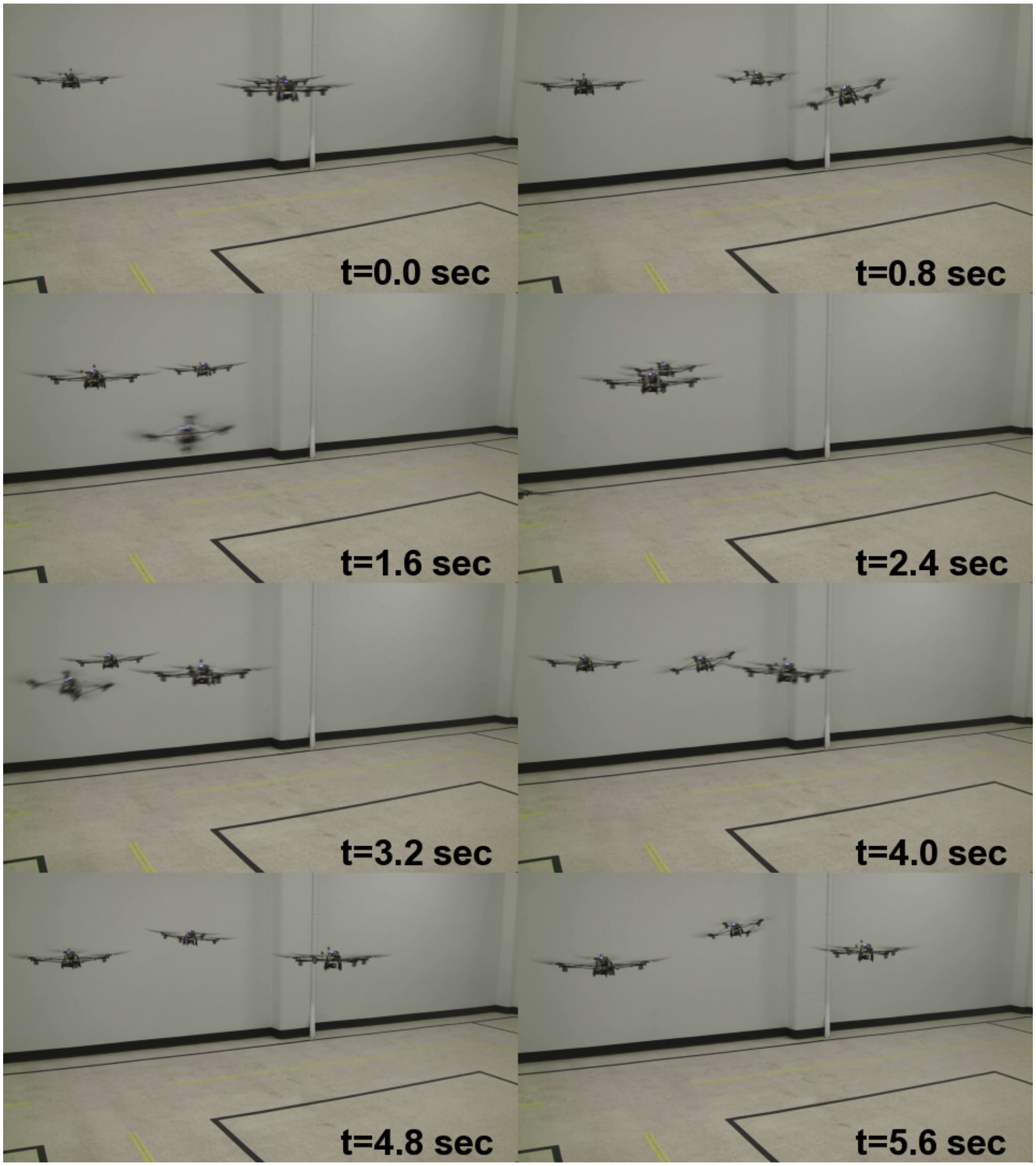


Figure 6-7: Screen captures from flight test video of the three quadrotor circular motion problem with baseline inner- and outer-loops. Time indices are given in the bottom right corner of each frame.

		Outer loop	
		Baseline	Adaptive
Inner loop	Baseline	1.00	0.49
	Adaptive	0.56	**0.38**

Table 6.4: Formation error for the three UAV circular motion flight test. A 50% loss of thrust failure is inserted into the left-most propeller at $\theta_{fail} = -0.79$ rad. Adaptation in both in inner- and outer-loops decreased the formation error as well.

of baseline case are shown in figure 6-7. A 50% loss of controller effectiveness in the left-most propeller at $\theta_{fail} = -0.79$ rad, just before $t = 0.8$ sec. This causes the vehicle to pitch down, and begin accelerating in the negative x-direction (from right to left across the frame). The failed quadrotor passes directly underneath the trailing quadrotor, avoiding one potential collision. At $t = 2.4$ sec, the failed vehicle can be seen almost out of the frame, but the integrators in the baseline controller are beginning to recover stable flight. At $t = 3.2$, the failed vehicle attempts to return to its commanded waypoint, which has now moved around the circle. To do this it moves towards oncoming traffic, suffering a minor collision with the leading quadrotor.

Figure 6-8 shows the same test scenario using adaptive inner- and outer-loops. Again, the failure occurs just before $t = 0.8$ sec, and causes the quadrotor to pitch down. However, the adaptive inner-loop more quickly accounts for the loss of thrust and adjusts to compensate. Furthermore, the velocity profiles of the commanded waypoints are slowed, and the position adjustment actually forces the other vehicles to take evasive action. By $t = 1.6$ sec, the failed vehicle is returning to its commanded waypoint, while the other vehicles wait for the tracking errors to decrease to an acceptable value. At $t = 2.4$ sec, normal operation has resumed with no collisions or near misses.

Figure 6-9 shows a birds-eye view of the trajectories of the three quadrotors for the flight tests shown in figures 6-7 and 6-8, as well as the baseline inner-loop/adaptive outer-loop and adaptive inner-loop/baseline outer-loop cases. For clarity, each vehicle is assigned a color: blue corresponding to the failed vehicle, red for the trailing vehicle, and green for the leading vehicle. Each vehicle traces out its trajectory in the corresponding color. Figure 6-9(a) shows the trajectories in the baseline inner-

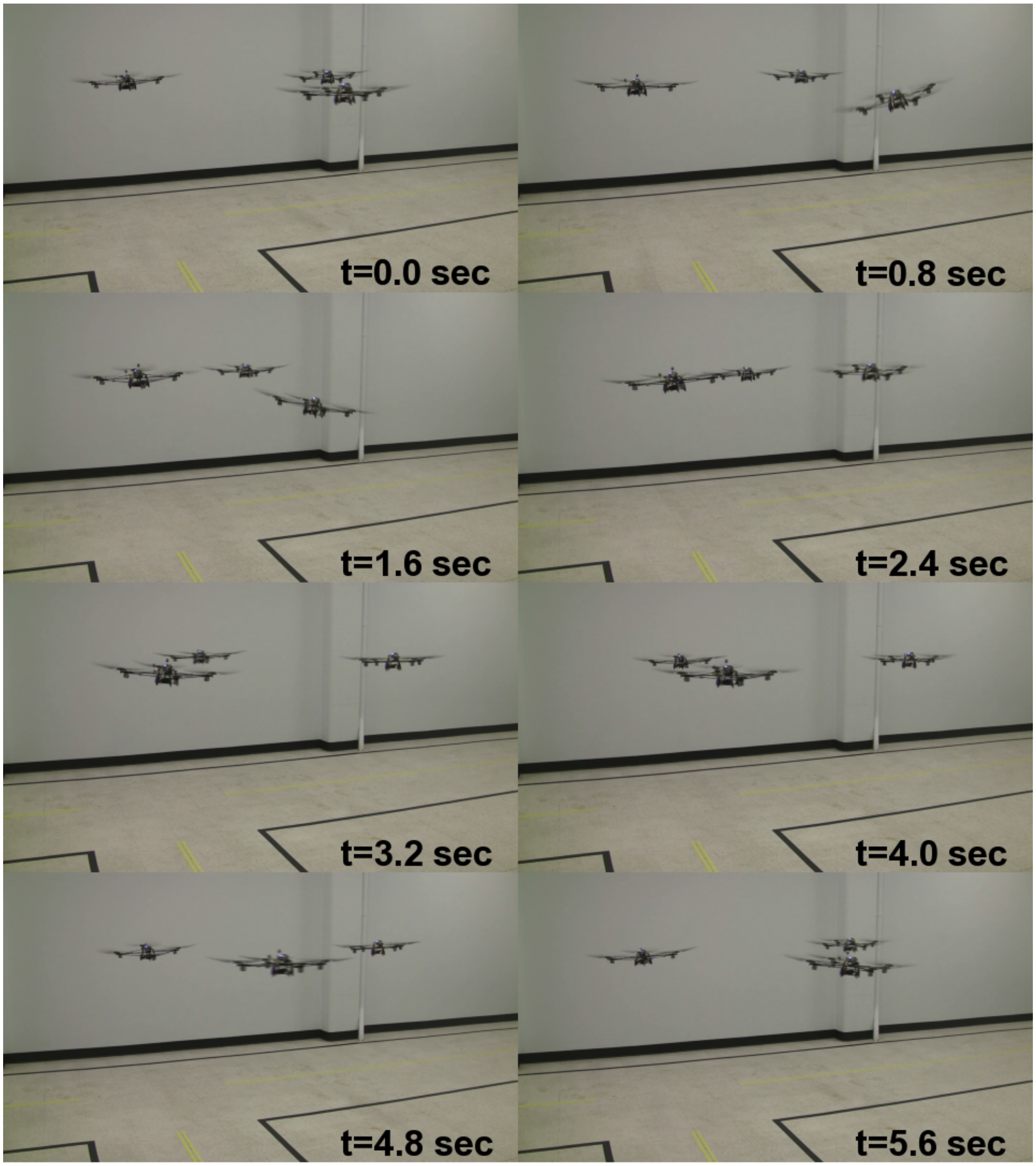
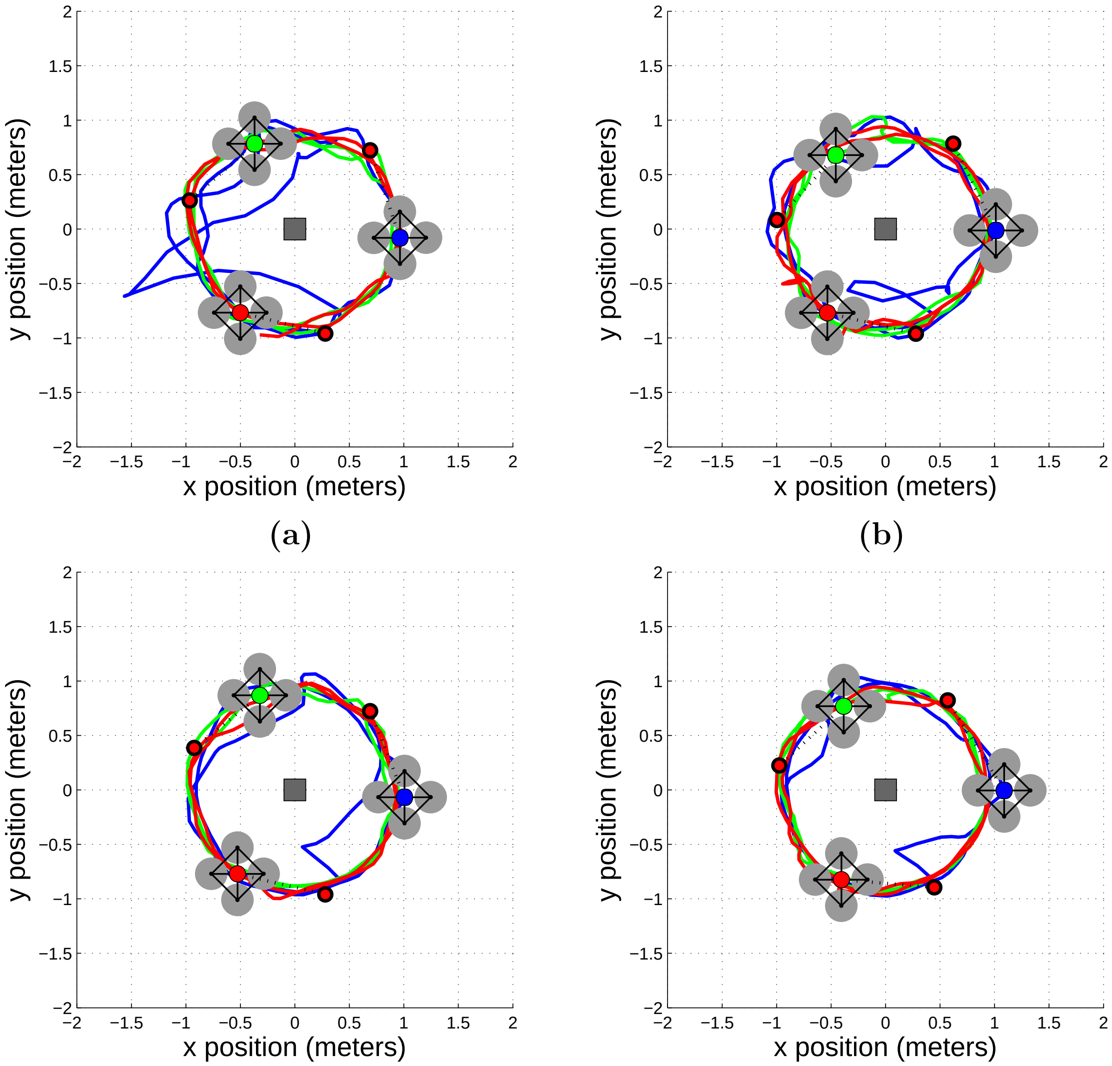


Figure 6-8: Screen captures from flight test video of the three quadrotor circular motion problem with adaptive inner- and outer-loops. Time indices are given in the bottom right corner of each frame.

loop/baseline outer-loop case, with the blue trajectory corresponding to the failed quadrotor. Comparing figure 6-9(a) with figure 6-9(b) highlights the effect of including adaptation in the outer-loop. Since neither of these two cases include adaptation



(a) **(b)**

(c) **(d)**

Figure 6-9: Birds-eye projection of flight test data for the (a) baseline inner-loop/baseline outer-loop (b) baseline inner-loop/adaptive outer-loop, (c) adaptive inner-loop/baseline outer-loop, and (d) adaptive inner-loop/adaptive outer-loop test cases.

in the inner-loop, the loss of control effectiveness causes large excursions in the failed vehicle's position. In the case of the baseline outer-loop controller, the trailing quadrotor continues to move in the counter-clockwise direction around the circle, causing it to move its propeller wash directly above the failed quadrotor. As the failed vehicle moves into the propeller wash, not only does it lose altitude, but it also experiences a negative pitch moment, causing it to move further in the negative x-direction. As the failed quadrotor exits the wash, it experiences a positive pitch moment which actually helps it recover; note the sharp point in the trajectory at the left of figure 6-9(a). Also note that the path taken by the failed quadrotor to return to its commanded waypoint takes it in a clockwise direction, against the flow of oncoming traffic. On the other hand, in the adaptive outer-loop case, the trailing vehicle slows down and even moves temporarily in the opposite direction (clockwise). Thus, the failed quadrotor avoids any negative effect of the propeller wash of the trailing quadrotor. Although the tracking performance is still poor, no collisions between vehicles occur.

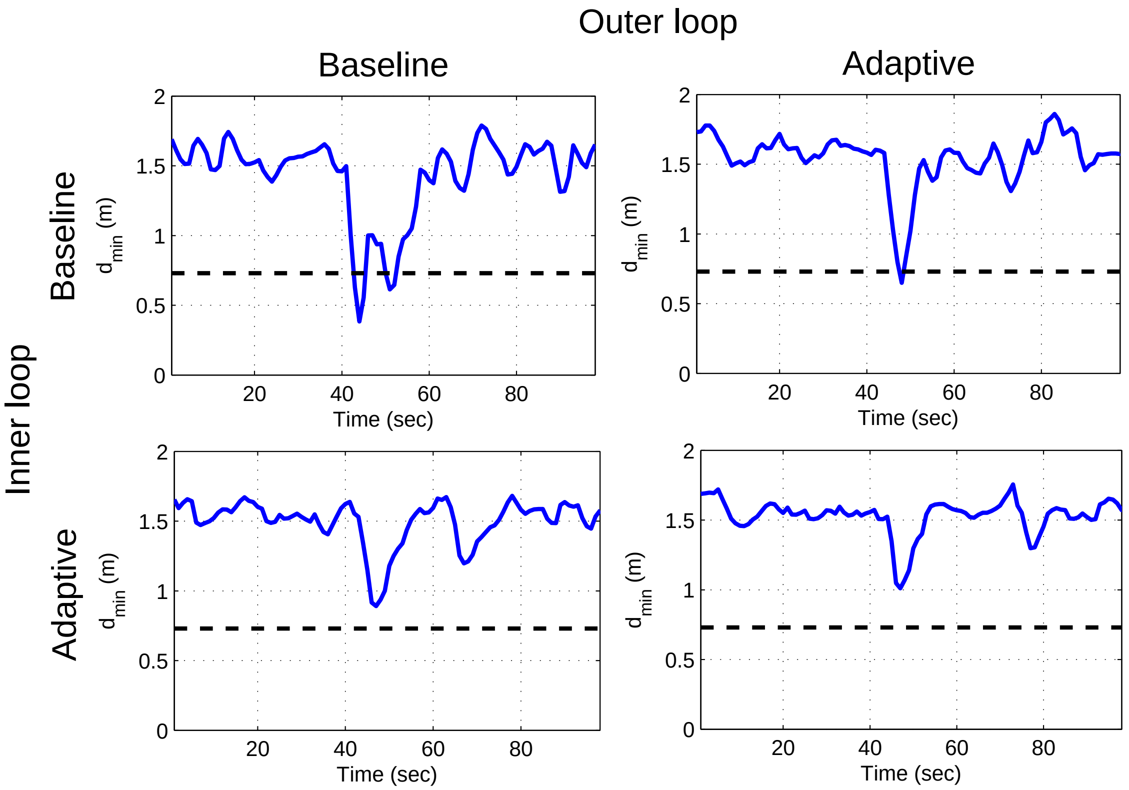


Figure 6-10: Flight recorded data of the minimum distance between vehicles. The dashed line corresponds to the maximum diameter of the quadrotors, *i.e.* the distance below which collisions may occur.

Figures 6-9(c) and (d) show that including adaptation in the inner-loop dramatically decreases the departure radius, the maximal distance from the commanded waypoint. Note that in the adaptive inner-loop/baseline outer-loop case, shown in figure 6-9(c), the failed vehicle takes a spiraling arc outward, back to its commanded position. The reason for this is that the commanded waypoint continues to move counter-clockwise around the circle. In the adaptive outer-loop case, shown in figure 6-9(d), the commanded waypoint stops until the failed quadrotor can return, allowing it to take the short path almost directly in the positive x-direction to return to formation.

The preceding flight tests results show that including adaptation in either the inner- or outer-loop reduces the likelihood of collisions in this specific example. To make a more general conclusion, we can examine the minimum distance between vehicles d_{min} over the course of the mission, shown in figure 6-10. The dashed line represents the diameter of the quadrotor at its largest. Thus, values of d_{min} less than the dotted line represent possible collisions. There are two valleys in the baseline inner-loop/baseline outer-loop that pass below this line. The first valley corresponds to the failed quadrotor passing directly underneath the trailing quadrotor. No collision occurs here because the quadrotors are thinner in the z-direction than in the x- and y-directions. The second valley corresponds to the collision that occurred between the failed quadrotor and the leading quadrotor. In the baseline inner-loop/adaptive outer-loop case, the minimum distance momentarily passes below the dashed line, but again no collisions occur. It is clear that the adaptive inner-loop/adaptive outer-loop does the best job of maintaining inter-vehicle distances, and has the largest minimum value of d_{min}. For scenarios with less spacing between vehicles, higher angular velocities, or more severe uncertainties, it is therefore expected that the adaptive inner-loop/adaptive outer-loop approach will have the least likelihood of collisions.

6.5 Summary

An adaptive configuration controller for multi-vehicle UAS was presented. The controller uses local and global error information to adjust its waypoint generation law. The two primary features of this adjustment are a position bias term, which allows vehicles to return to formation quickly, and a velocity shaping term, which slows down the commanded trajectory as tracking performance degrades. The approach described is independent of the inner-loop controller design, and can be combined with the adaptive inner-loop controllers discussed in previous chapters.

A specific example involving quadrotor UAV planar circular motion about a central point in the presence of unknown actuator failures was presented. Simulation results showed that including adaptation in the outer-loop decreases formation error and decreases the likelihood of collisions. Follow-up studies on a multi-vehicle hardware UAS system consisting of several quadrotor UAVs with time delay in the loop was also presented, similarly showing that inclusion of adaptation in the outer-loop allows for lower tracking error, lower formation-keeping error, and higher minimum inter-vehicle distances, reducing the likelihood of collisions between vehicles.

Chapter 7

Summary and future work

7.1 Summary

Adaptive control is considered to be one of the key enabling technologies for future high-performance, safety-critical systems such as air-breathing hypersonic vehicles. Adaptive flight control systems offer improved performance and increased robustness to uncertainties by virtue of their ability to adjust control parameters as a function of online measurements. Extensive research in the field of adaptive control theory has laid the foundation for application of adaptive flight control to hardware platforms. Unmanned aerial systems (UAS) provide a unique opportunity for the transition of adaptive controllers from theory to practice. Motivated by performance goals of extreme persistence, maneuverability, ability to operate in dangerous environments without risking safety of the crew, stealth, and lower cost, several missions such as surveillance, exploration, search-and-track, and lifting of heavy loads are best accomplished by a UAS consisting of multiple unmanned aerial vehicles (UAVs). In this thesis we have addressed some of the challenges involved with the design and implementation of an adaptive flight control systems for UAS.

We began in chapter 3 by presenting an adaptive controller based on Lyapunov stability and validated the design using simulation and flight test of a quadrotor UAV. Flight testing was carried out in an indoor test facility using both baseline and model reference adaptive controllers (MRAC). It was shown that the adaptive controller offers several benefits over the existing fixed-gain approach, particularly in the case of actuator failures. For less severe failures, the adaptive controller was faster in reacting to the change in dynamics, resulting in a decreased radius of departure. For more severe failures, the adaptive controller prevented a crash and allowed for safe operation and landing. Methods for accommodating potential implementation roadblocks such as sensor noise, parameter drift, and adaptive gain selection were discussed.

In chapter 4 we discussed several modifications to the typical MRAC approach with application to general time delay systems. The modifications presented were either designed specifically to counter the effect time delays, or had the effect of smoothing the adaptive or estimated parameters. A novel time delay resistant (TDR) adaptive control approach that integrates all of the modifications into a coherent control structure was presented. The TDR adaptive controller demonstrated increased performance in simulation of an F-16 short period dynamics model with time delay in the loop. Typically in control systems there is a design trade-off between tracking performance and control power used. However, by utilizing these modifications to MRAC, it is possible to both increase tracking performance and decrease control effort as compared with MRAC alone. Furthermore, the TDR approach has a proof of signal boundedness and an analytically justifiable time delay margin that is bounded away from 0. It was shown that by making some assumptions on the system, this time delay margin can be used to understand the robustness of the adaptive system and make informed selections of adaptive gains and other free parameters.

It was found that one modification in particular, the combined / composite model reference adaptive control (CMRAC) approach offered an impressive increase in performance with little additional computational cost. Therefore, in chapter 5 a CMRAC design was generated for the quadrotor system and the approach was validated with flight tests of the quadrotor UAV in the presence of actuator uncertainties and a 40 ms time delay in the loop. All of the benefits displayed in simulation were reproduced in the flight tests. In particular, the CMRAC approach demonstrated improved regulation and tracking performance. Additionally, it was found that the CMRAC approach allows for higher adaptive gains than MRAC alone without exciting high frequencies. Including indirect adaptation to MRAC also generated smoother parameter estimates and demonstrated improved parameter convergence. It was also shown that CMRAC was more effective than MRAC in learning the true value of uncertain parameters in the system, offering numerous benefits in terms of tracking performance.

Finally, in chapter 6 a novel adaptive configuration controller for multi-vehicle UAS was presented. The controller uses local and global error information to adjust its waypoint generation law. The two primary features of this adjustment are a position bias term, which allows vehicles to return to formation quickly, and a velocity shaping term, which slows down the commanded trajectory as tracking performance degrades. The approach described is independent of the inner-loop controller design, and can be combined with the adaptive inner-loop controllers discussed in previous chapters. A specific example involving planar, circular motion of a formation of quadrotor UAVs about a central point in the presence of unknown actuator failures was presented. Simulation results showed that including adaptation in the outer-loop decreases formation error and decreases the likelihood of collisions. Follow-up flight testing on a multi-vehicle UAS consisting of several quadrotor UAVs with time delay in the loop were also presented, similarly showing that inclusion of adaptation in the outer-loop allows for reduced tracking error, reduced formation-keeping error, and increased minimum inter-vehicle distances, reducing the likelihood of collisions between vehicles.

7.2 Future work

In this section we introduce possible future extensions and applications of the research presented in this thesis. The main areas of potential future work are as follows: improvements to the adaptive algorithms used for flight tests, applications to new vehicles, further explorations and extensions for adaptive time delay systems, additional analysis of CMRAC and applications to mission management, and adaptive configuration control for vehicles with limited communication.

One potentially useful addition to the adaptive controller presented in chapter 3 is a magnitude saturation accommodating feature in the adaptive laws [10, 42]. Actuator saturation is a nonlinearity, but the nature of the nonlinearity is well-understood and typically known *a priori*. Therefore it is possible to remove error due to actuator saturation from the error signal used to train the adaptive controller. This would allow for sustained operation at or near the saturation limits with no adverse effects on the adaptation process. Additionally, more of the features presented in section 4.2 could be transitioned to the quadrotor problem. The adaptive posicast feature, in particular, could be added with relatively little computational overhead.

The adaptive controller discussed in chapter 3, either with or without these additional features, is quite general in nature and could be applied to a variety of UAVs. One example of a potential application platform is an agile fixed-wing aircraft such as the modified Twinstar II model airplane known as the "Extreme Star" [87], shown in figure 7-1. This aircraft features 16 control inputs and a suite of sensors, including GPS, accelerometers, gyroscopes, and a magnetometer. The vehicle is equipped with an Intel Atom processor for running flight control code and is capable of wireless communication with other vehicles or with a ground-based centralized control. The large number of redundant actuators allows for testing of various failure scenarios such as missing ailerons or flaps. This platform is more challenging than that of the quadrotor due to its faster dynamics, onboard sensing and control, and limited communication capabilities.

The time delay results in chapter 4 could also be extended in several ways. The describing functions approach for approximating the closed-loop adaptive system could be extended to the multiple π-function case, and to more general, higher-order plants. These extensions will bring the describing functions approximation closer to the MRAC structures typically seen in practice, increasing the usefulness of the tool. Extensions to the TDR approach include examining additional methods for dealing with time delays such as wave variables, as well as an extending the theoretical results to a more general regressor vector, such as one composed of a series of radial basis functions (RBFs). RBFs have proven useful in parameterizing some nonlinear

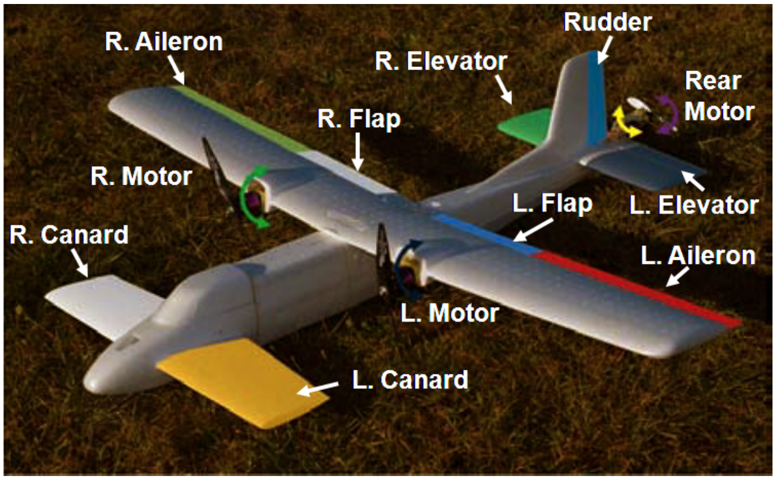


Figure 7-1: Extreme Star model aircraft. Control inputs consist of 9 control surfaces (labeled), variable thrust of the three motors, and 4 thrust vectoring inputs: one for each of the wing-mounted motors and two for the rear motor.

aerodynamic phenomenon such as the pitch break and rolling shock phenomena.

Future directions involving the CMRAC approach include a closer examination as to how exactly the controller is able to adapt and learn so quickly and effectively. It is also worth exploring how the prediction error differs from the tracking error used in an MRAC approach. A theoretical justification for why CMRAC systems exhibit better transient performance and smoother parameter estimates also remains an open problem. Another possible future direction for CMRAC is the combination with a health-aware mission manager [58]. The online health estimates provided by the CMRAC approach could be passed to the mission manager, allowing it to decide which vehicles can perform certain tasks or which vehicles need to come in for repairs.

There are many potential extensions to the adaptive configuration control approach presented in chapter 6, including extensions to more and different types of vehicles, more complex 3D trajectories, obstacle-filled environments, and more aggressive maneuvers. The approach presented could also be extended to address problems of task allocation and stochastic environments [62]. The results presented in chapter 6 represent a proof-of-concept and make a compelling case for the inclusion of adaptation in the outer-loop. However, a more systematic analysis of the stability of the overall system should be explored. Another important direction of future work is the extension of the approach to the case of limited intra-UAV communication. The approach described in chapter 6 is based on an all-to-all communication scheme, which is not always possible in practice due to limited onboard communication range, for example. Extending the adaptive configuration controller to the case of limited communication case would increase the power and applicability of the approach.

The work presented in this thesis represents one step along the path of transitioning adaptive flight control from theory to practice. However, the path is long. There are many obstacles and roadblocks; many open problems and opportunities. The future work discussed above represents a few of the next steps along this path. As these topics are investigated, new avenues will undoubtedly open up for future scientists, researchers, and engineers. Subsequent research will enable the widespread application of adaptive control to manned and unmanned aerial vehicles, ushering in a new era of stability, reliability, and performance.

Bibliography

[1] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control and the NASA X-15-3 flight revisited," *IEEE Control Systems Magazine*, vol. 1066, no. 033X/10, 2010.

[2] M. Sharma, E. Lavretsky, and K. A. Wise, "Application and flight testing of an adaptive autopilot on precision guided munitions," in *Proc. of the AIAA Guidance Navigation and Control Conference*, Keystone, CO, August 2006.

[3] K. Wise, E. Lavretsky, J. Zimmerman, J. Francis Jr, D. Dixon, and B. Whitehead, "Adaptive flight control of a sensor guided munition," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, San Francisco, CA, August 2005.

[4] Office of the Under Secretary of Defense For Acquisition, Technology, and Logistics, "Defense science board study on unmanned aerial vehicles and uninhabited combat aerial vehicles," Washington, D.C. 20301-3140, Tech. Rep., February 2004.

[5] United States Air Force, "Unmanned aircraft systems flight plan 2009-2047," United States Air Force, Washington, D.C., Tech. Rep., May 2009.

[6] D. R. Jenkins, "Hypersonics before the shuttle: A concise history of the X-15 research airplane," NASA, Special Publication SP-2000-4518, June 2000, Monographs in Aerospace History: Number 18. Available at ntrs.nasa.gov.

[7] K. S. Narendra and A. M. Annaswamy, *Stable Adaptive Systems.* Englewood Cliffs, NJ: Prentice-Hall, 1989.

[8] P. A. Ioannou and J. Sun, *Robust Adaptive Control.* Upper Saddle River, NJ: Prentice-Hall, 1996.

[9] H. K. Khalil and J. W. Grizzle, *Nonlinear Systems.* Upper Saddle River, NJ: Prentice-Hall, 1996.

[10] S. P. Karason and A. M. Annaswamy, "Adaptive control in the presence of input constraints," *IEEE Transactions on Automatic Control*, vol. 39, no. 11, pp. 2325–2330, November 1994.

[11] Z. T. Dydek, H. Jain, J. Jang, A. M. Annaswamy, and E. Lavretsky, "Theoretically verifiable stability margins for an adaptive controller," in *Proc. AIAA*

Conference on Guidance, Navigation, and Control, Keystone, CO, August 2006, AIAA-2006-6416.

[12] J. Jang, A. M. Annaswamy, and E. Lavretsky, "Towards verifiable adaptive flight control in the presence of actuator anomalies," in *Proc. Conference on Decision and Control*, San Diego, CA, December 2006, pp. 3300–3305.

[13] G. Tao, *Adaptive Control Design and Analysis.* Hoboken, NJ: John Wiley & Sons, 2003.

[14] M. Krstic, P. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design.* New York, NY: John Wiley & Sons, 1995.

[15] R. Isermann, D. Matko, and K. Lachmann, *Adaptive Control Systems.* Upper Saddle River, NJ: Prentice-Hall, 1992.

[16] P. A. Ioannou and K. Tsakalis, "A robust direct adaptive controller," *IEEE Transactions on Automatic Control*, vol. 31, no. 11, pp. 1033–1043, November 1986.

[17] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness.* Englewood Cliffs, NJ: Prentice-Hall, 1989.

[18] J. J. E. Slotine and J. A. Coetsee, "Adaptive sliding controller synthesis for nonlinear systems," *International Journal of Control*, vol. 43, no. 6, pp. 1631–1651, 1986.

[19] B. Egardt and D. Whitacre, *Stability of Adaptive Controllers.* Secaucus, NJ: Springer-Verlag, 1979.

[20] A. Morse, "Global stability of parameter-adaptive control systems," *IEEE Transactions on Automatic Control*, vol. 25, no. 3, pp. 433–439, 1980.

[21] B. Anderson, T. Brinsmead, F. De Bruyne, J. Hespanha, D. Liberzon, and A. Morse, "Multiple model adaptive control. Part 1: Finite controller coverings," *International Journal of Robust and Nonlinear Control*, vol. 10, no. 11-12, pp. 909–929, 2000.

[22] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control.* Englewood Cliffs, NJ: Prentice-Hall, 1984.

[23] I. D. Landau, R. Lozano, and M. M'Saad, *Adaptive Control.* London, England: Springer, 1998.

[24] G. Feng and R. Lozano, *Adaptive Control Systems.* Burlington, MA: Newnes, 1999.

[25] K. J. Åström and B. Wittenmark, *Adaptive Control.* Boston, MA: Addison-Wesley Longman Publishing Co., Inc., 1995.

[26] E. Lavretsky and N. Hovakimyan, "Stable adaptation in the presence of actuator constraints with flight control applications," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, p. 337, 2007.

[27] J. Jang, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of time-varying systems with gain-scheduling," in *Proc. of the American Control Conference*, Seattle, Washington, June 2008.

[28] G. Tao and P. V. Kokotovic, "Adaptive control of plants with unknown hystereses," *IEEE Transactions on Automatic Control*, vol. 40, no. 2, pp. 200–212, February 1995.

[29] R. T. Anderson, G. Chowdhary, and E. N. Johnson, "Comparison of RBF and SHL neural network based adaptive control," *Journal of Intelligent and Robotic Systems*, vol. 54, no. 1-3, pp. 183–199, March 2009.

[30] N. Nguyen, K. Krishnakumar, J. Kaneshige, and P. Nespeca, "Flight dynamics and hybrid adaptive control of damaged aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 6, pp. 1837–1838, November-December 2008.

[31] Y. Shin, A. J. Calise, and M. D. Johnson, "Adaptive control of advanced fighter aircraft in nonlinear flight regimes," *Journal of Guidance, Control, and Dynamics*, vol. 31, no. 5, pp. 1464–1477, September-October 2008.

[32] J. Bošković and R. Mehra, "Multiple-model adaptive flight control scheme for accommodation of actuator failures," *Journal of Guidance, Control, and Dynamics*, vol. 25, no. 4, pp. 712–724, 2002.

[33] J. Bošković, L. Chen, and R. Mehra, "Adaptive control design for nonaffine models arising in flight control," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 2, pp. 209–217, 2004.

[34] B. S. Kim and A. J. Calise, "Nonlinear flight control using neural networks," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 1, pp. 26–33, 1997.

[35] A. J. Calise and R. T. Rysdyk, "Nonlinear adaptive flight control using neural networks," *IEEE Control Systems Magazine*, vol. 18, no. 6, pp. 14–25, 1998.

[36] M. Pachter, P. Chandler, and M. Mears, "Reconfigurable tracking control with saturation," *Journal of Guidance, Control, and Dynamics*, vol. 18, no. 5, pp. 1016–1022, 1995.

[37] K. P. Groves, D. O. Sigthorsson, A. Serraniz, S. Yurkovich, M. A. Bolender, and D. B. Doman, "Reference command tracking for a linearized model of an air-breathing hypersonic vehicle," *Proc. of the AIAA Guidance, Navigation, and Control Conference*, August 2005, AIAA-2005-6144.

[38] T. Gibson, L. G. Crespo, and A. M. Annaswamy, "Adaptive control of hypersonic vehicles in the presence of modeling uncertainties (I)," in *Proc. of the American Control Conference*, St. Louis, Missouri, June 2009.

[39] S. Ferrari and R. F. Stengel, "Online adaptive critic flight control," *Journal of Guidance, Control, and Dynamics*, vol. 27, pp. 777–786, 2004.

[40] D. Ward, J. Monaco, and M. Bodson, "Development and flight testing of a parameter identification algorithm for reconfigurable control," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 6, pp. 948–956, 1998.

[41] J. Brinker and K. Wise, "Flight testing of reconfigurable control law on the X-36 tailless aircraft," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 5, pp. 903–909, 2001.

[42] E. Johnson and S. Kannan, "Adaptive trajectory control for autonomous helicopters," *Journal of Guidance Control and Dynamics*, vol. 28, no. 3, pp. 524–538, 2005.

[43] R. Rysdyk and A. Calise, "Robust nonlinear adaptive flight control for consistent handling qualities," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 6, pp. 896–910, 2005.

[44] J. Slotine and W. Li, "Composite adaptive control of robot manipulators." *Automatica*, vol. 25, no. 4, pp. 509–519, 1989.

[45] M. Duarte-Mermoud, J. Rioseco, and R. Gonzalez, "Control of longitudinal movement of a plane using combined model reference adaptive control," *Aircraft Engineering and Aerospace Technology: An International Journal*, vol. 77, no. 3, pp. 199–213, 2005.

[46] M. Duarte and K. Narendra, "Combined direct and indirect approach to adaptive control," *IEEE Transactions on Automatic Control*, vol. 34, no. 10, pp. 1071–1075, 1989.

[47] E. Lavretsky, "Combined / composite model reference adaptive control," in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, Chicago, Illinois, 2009.

[48] J. Slotine and W. Li, *Applied Nonlinear Control.* Englewood Cliffs, NJ: Prentice-Hall, 1991.

[49] Y. Yildiz, A. Annaswamy, I. Kolmanovsky, and D. Yanakiev, "Adaptive posicast controller for time-delay systems with relative degree $n^* \leq 2$," *Automatica*, vol. 46, pp. 279–289, 2010.

[50] J. Leishman, *Principles of Helicopter Aerodynamics.* Cambridge, United Kingdom: Cambridge University Press, 2006.

[51] S. B. Anderson, “Historical overview of V/STOL aircraft technology,” NASA, Ames Research Center, Moffett Field, CA, Technical Memorandum D-1060, June 1981. Available at ntrs.nasa.gov.

[52] S. Bouabdallah, P. Murrieri, and R. Siegwart, “Design and control of an indoor micro quadrotor,” in *IEEE International Conference on Robotics and Automation*, vol. 5, 2004, pp. 4393–4398.

[53] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin, “Quadrotor helicopter flight dynamics and control: Theory and experiment,” in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, Hilton Head, SC, 2007.

[54] P. McKerrow, “Modelling the Draganflyer four-rotor helicopter,” in *IEEE International Conference on Robotics and Automation*, New Orleans, LA, April-May 2004, pp. 3596–3601.

[55] G. Tournier, M. Valenti, J. How, and E. Feron, “Estimation and control of a quadrotor vehicle using monocular vision and moirè patterns,” in *Proc. of the AIAA Guidance, Navigation and Control Conference*, Keystone, CO, 2006, pp. 21–24.

[56] M. Valenti, B. Bethke, G. Fiore, J. How, and E. Feron, “Indoor multi-vehicle flight testbed for fault detection, isolation, and recovery,” in *Proc. of the AIAA Guidance, Navigation, and Control Conference*, Keystone, CO, 2006.

[57] J. How, B. Bethke, F. A., D. Dale, and J. Vian, “Real-time indoor autonomous vehicle test environment,” *IEEE Control Systems Magazine*, vol. 28, no. 2, pp. 51–64, April 2008.

[58] B. Bethke, J. P. How, and J. Vian, “Group health management of UAV teams with applications to persistent surveillance,” in *Proc. of the American Control Conference*, 2008, pp. 3145–3150.

[59] A. Benallegue, A. Mokhtari, and L. Fridman, “High-order sliding-mode observer for a quadrotor UAV,” *International Journal of Robust and Nonlinear Control*, vol. 18, no. 4, pp. 427–440, 2008.

[60] N. Michael, J. Fink, and V. Kumar, “Cooperative manipulation and transportation with aerial robots,” in *Proc. of the Robotics: Science and Systems*, Seattle, WA, June 2009.

[61] A. Bachrach, R. He, and N. Roy, “Autonomous flight in unknown indoor environments,” *International Journal of Micro Air Vehicles*, vol. 1, no. 4, pp. 217–228, 2009.

[62] D. Castañón and J. Wohletz, “Model predictive control for stochastic resource allocation,” *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1739–1750, 2009.

[63] P. Ogren, E. Fiorelli, and N. Leonard, “Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment,” *IEEE Transactions on Automatic Control*, vol. 49, no. 8, pp. 1292–1302, 2004.

[64] R. Sepulchre, D. Paley, and N. Leonard, “Stabilization of planar collective motion with limited communication,” *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 706–719, 2008.

[65] N. Moshtagh and A. Jadbabaie, “Distributed geodesic control laws for flocking of nonholonomic agents,” *IEEE Transactions on Automatic Control*, vol. 52, no. 4, pp. 681–686, 2007.

[66] A. Arsie, K. Savla, and E. Frazzoli, “Efficient routing algorithms for multiple vehicles with no explicit communications,” *IEEE Transactions on Automatic Control*, vol. 54, no. 10, pp. 2302–2317, 2009.

[67] J. Fax, R. Murray, N. Syst, and C. Woodland Hills, “Information flow and cooperative control of vehicle formations,” *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1465–1476, 2004.

[68] I. Kaminer, O. Yakimenko, V. Dobrokhodov, A. Pascoal, N. Hovakimyan, C. Cao, A. Young, and V. Patel, “Coordinated path following for time-critical missions of multiple UAVs via L1 adaptive output feedback controllers,” in *Proc. of the AIAA Guidance, Navigation and Control Conference*, Hilton Head, SC, 2007.

[69] I. D. Landau, *Adaptive Control: The Model Reference Approach.* New York, NY: Marcel Dekker, 1979.

[70] K. Tsakalis and P. Ioannou, *Linear Time-Varying Systems: Control and Adaptation.* Upper Saddle River, NJ: Prentice-Hall, 1993.

[71] Active Adaptive Control Laboratory, “Adaptive Control of a Quadrotor UAV,” Retrieved May 27th, 2010, from http://www.youtube.com/watch?v=FhgMy4ss0bw.

[72] E. Lavretsky, A. M. Annaswamy, Z. T. Dydek, and W. Vega-Brown, “On the computation of stability margins for adaptive controllers using linear system tools,” in *Proc. of the AIAA Guidance, Navigation and Control Conference*, Chicago, Illinois, August 2009.

[73] Z. T. Dydek, A. M. Annaswamy, J.-J. E. Slotine, and E. Lavretsky, “Time delay resistant adaptive control of mini-UAVs,” in *Proc. of the IFAC Workshop on Time Delay Systems*, Prague, Czech Republic, 2010.

[74] ——, “High performance adaptive control in the presence of time delays,” in *Proc. of the American Control Conference*, Baltimore, MD, 2010.

[75] A. Manitius and A. Olbrot, "Finite spectrum assignment problem for systems with delays," *IEEE Transactions on Automatic Control*, vol. 24, no. 4, pp. 541–552, 1979.

[76] M. Duarte and K. Narendra, "Error models with parameter constraints," *International Journal of Control*, vol. 64, no. 6, pp. 1089–1111, 1996.

[77] J. Nakanishi, J. Farrell, and S. Schaal, "Composite adaptive control with locally weighted statistical learning," *Neural Networks*, vol. 18, no. 1, pp. 71–90, 2005.

[78] E. Lavretsky, "Combined / composite model reference adaptive control," *IEEE Transactions on Automatic Control*, vol. 54, no. 11, pp. 2692–2697, 2009.

[79] A. Ben-Israel and T. Greville, *Generalized inverses: Theory and applications.* New York, NY: Springer-Verlag, 2003.

[80] S. Niculescu and A. Annaswamy, "An adaptive Smith-controller for time-delay systems with relative degree $n^* \leq 2$," *Systems and control letters*, vol. 49, no. 5, pp. 347–358, 2003.

[81] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation.* Hoboken, NJ: John Wiley and Sons, 1992.

[82] K. strm, "Theory and applications of adaptive control–a survey," *Automatica*, vol. 19, no. 5, pp. 471 – 486, 1983. [Online]. Available: http://www.sciencedirect.com/science/article/B6V21-47WTDRM-M/2/0fd4a465e17d0d9cc927ecb6b0646d91

[83] G. Goodwin, K. Sin, and K. Saluja, "Stochastic adaptive control and prediction–the general delay-colored noise case," *Automatic Control, IEEE Transactions on*, vol. 25, no. 5, pp. 946 – 950, oct 1980.

[84] K. Sin and G. Goodwin, "Stochastic adaptive control using a modified least squares algorithm," *Automatica*, vol. 18, no. 3, pp. 315–321, 1982.

[85] Active Adaptive Control Laboratory, "Altitude regulation for quadrotor UAVs," Retrieved May 27th, 2010, from http://www.youtube.com/watch?v=Tt239r5MquI.

[86] ——, "Adaptive configuration control - formation flight," Retrieved May 27th, 2010, from http://www.youtube.com/watch?v=tdZAIt7OrPs.

[87] R. Wiedenmann, "13 auf einen streich," *Elektro modellflug-praxis*, January 2010.