

4-22-2021

Adaptive and Neural Network-Based Aircraft Tracking Control with Synthetic Jet Actuators

Joshua Teramae
Embry-Riddle Aeronautical University

Follow this and additional works at: <https://commons.erau.edu/edt>



Part of the [Navigation, Guidance, Control and Dynamics Commons](#)

Scholarly Commons Citation

Teramae, Joshua, "Adaptive and Neural Network-Based Aircraft Tracking Control with Synthetic Jet Actuators" (2021). *PhD Dissertations and Master's Theses*. 581.
<https://commons.erau.edu/edt/581>

This Thesis - Open Access is brought to you for free and open access by Scholarly Commons. It has been accepted for inclusion in PhD Dissertations and Master's Theses by an authorized administrator of Scholarly Commons. For more information, please contact commons@erau.edu.

ADAPTIVE AND NEURAL NETWORK-BASED AIRCRAFT
TRACKING CONTROL WITH SYNTHETIC JET ACTUATORS

BY
JOSHUA TERAMAE

A Thesis

Submitted to the Department of Physical Sciences
and the Committee on Graduate Studies
In partial fulfillment of the requirements
for the degree of
Master in Science in Engineering Physics

04/2021

Embry-Riddle Aeronautical University
Daytona Beach, Florida

© Copyright by Joshua Teramae 2021
All Rights Reserved

ADAPTIVE AND NEURAL NETWORK-BASED AIRCRAFT TRACKING
CONTROL USING SYNTHETIC JET ACTUATORS

by

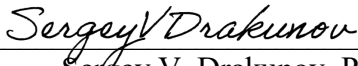
Joshua Teramae

This thesis was prepared under the direction of the candidate's Thesis Committee Chair, Dr. William MacKunis, Associate Professor, Daytona Beach Campus, and Thesis Committee Members Dr. Sergey V. Drakunov, Professor, Daytona Beach Campus, and Dr. Muhammad Omer Farooq, Associate Professor, Daytona Beach Campus, and has been approved by the Thesis Committee. It was submitted to the Department of Physical Sciences in partial fulfillment of the requirements for the degree of Master of Science in Engineering Physics.

Thesis Review Committee:




William MacKunis, Ph.D.
Committee Chair



Sergey V. Drakunov, Ph.D.
Committee Member



John M. Hughes, Ph.D.
Interim Chair,
Department of Physical Sciences

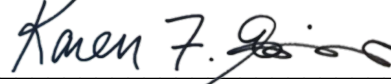


Christopher D. Grant, Ph.D.
Associate Provost of Academic Support

Muhammad
Omer Farooq

Digitally signed by Muhammad
Omer Farooq
Date: 2021.04.27 16:32:05 -04'00'

Muhammad Omer Farooq, Ph.D.
Committee Member



Karen F. Gaines, Ph.D.
Dean, College of Arts and Sciences

Date

Abstract

Wing-embedded synthetic jet actuators (SJA) can be used to achieve maneuvering control in aircraft by delivering controllable airflow perturbations near the wing surface. Trajectory tracking control design for aircraft equipped with SJA is particularly challenging, since the controlling actuator itself has an uncertain dynamic model. These challenges necessitate advanced nonlinear control design methods to achieve desirable performance for SJA-based aircraft (e.g., micro air vehicles (MAVs)). In this research, adaptive and neural-network based control methods are investigated, which are specifically designed to compensate for the SJA dynamic model uncertainty and unpredictable operating conditions characteristic of real-world MAV applications. The control design methods discussed in this Defense are rigorously developed to achieve a prescribed level of trajectory tracking control performance, and numerical simulation results are presented to demonstrate the performance of the controllers in the presence of adversarial operating conditions.

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. William MacKunis, because without his assistance and dedicated involvement throughout my research, this paper would have never been accomplished. Thank you very much for your continuous support and patience throughout my academic career and pushing me to bring my work to a higher level.

And of course none of this could have happened without the encouragement and love from my friends and family - Rachel, Mom, Dad, Zach, and Ollie.

Contents

Abstract	iv
Acknowledgments	v
1 Introduction	1
1.1 Active Flow Control	1
1.2 Challenges Using Synthetic Jet Actuators	4
1.3 Neural Networks in Adaptive Control Systems	4
1.4 Benefits of Adaptive and Neural Network-based Control	7
2 Mathematical Background	8
2.1 Stability Definitions	8
2.2 Lyapunov Stability Analysis	9
2.2.1 Definitions for Lyapunov Stability Analysis	10
2.2.2 Lyapunov’s Second Theorem (The Direct Method of Lyapunov)	10
2.3 Barbalat’s Lemma	11
2.4 Online Parameter Estimation	12
3 Controller Design with Exact Model Knowledge	13
3.1 Dynamic Model and Properties	13
3.2 Control Objective and Open Loop Dynamics	14
3.3 Stability Analysis	16
3.4 Numerical Simulation Results	16

4	Adaptive Control	23
4.1	Dynamic Model and Properties	23
4.2	Control Objective and Open Loop Error System	23
4.3	Control Design and Closed-loop Error System	25
4.4	Stability Analysis	26
4.5	Numerical Simulation Results	28
5	Neural Network Based Adaptive Control	34
5.1	Dynamic Model and Properties	34
5.2	Control Objective and Open Loop Error System	35
5.3	Control Design and Closed-loop Error System	35
5.3.1	NN Properties	35
5.4	Stability Analysis	37
5.5	Simulation Results	39
6	Conclusion	48

List of Figures

1.1	A type classification of flow control actuators [2]	2
1.2	Schematic layout of a SJA [23]	3
1.3	Diagram of a neural network	6
3.1	Closed-loop regulation of the steady state error for 20 runs	18
3.2	Closed-loop regulation of the virtual deflection angle for 20 runs . . .	19
3.3	Control voltage signals commanded for the six SJA arrays	20
3.4	Model reference (blue) and actual state (red) during closed-loop exact- model knowledge control	21
3.5	State error during closed-loop exact-model knowledge control	22
4.1	Closed-loop regulation of the steady state error for 20 runs	29
4.2	Closed-loop regulation of the virtual deflection angle for 20 runs . . .	30
4.3	Control voltage signals commanded for the six SJA arrays	31
4.4	Model reference (blue) and actual state (red) with adaptive control .	32
4.5	State error with adaptive control	33
5.1	Closed-loop regulation of the steady state error for 20 runs	40
5.2	Closed-loop regulation of the virtual deflection angle for 20 runs . . .	41
5.3	Control voltage signals commanded for the six SJA arrays	42
5.4	State error during neural network based adaptive control	44
5.5	Comparison of mean squared error for x_1	45
5.6	Comparison of mean squared error for x_2	46
5.7	Comparison of mean squared error for x_3	47

Chapter 1

Introduction

1.1 Active Flow Control

Active Flow Control (AFC) is the on-demand addition of energy into a boundary layer for maintaining, recovering, or improving vehicle aerodynamic performance. Aerodynamic applications include flow control for airplanes, ground vehicles, and wind turbine blades. Use of active flow control can enhance maneuverability of aircraft operating at high angles of attack (delaying stall), improve aerodynamic performance by reducing drag, and enable roll-control authority without the use of heavy mechanical control surfaces (e.g., ailerons and elevators) [5]. Active flow control is an important trend in the development of future aircraft. In 2020, DARPA awarded contracts to develop an X-Plane program based on active flow control [3].

There are various types of actuators used in flow control applications; a classification of AFC types is shown in Figure 1.1. The most common type is the fluidic actuator, which uses fluid injection or suction. One example of a fluidic actuator is the fluidic oscillator, a type of *nonzero* mass flux actuator. Fluidic oscillators create an unsteady oscillating jet with a frequency that depends primarily on the internal fluid dynamics of the oscillator. Fluidic oscillators are attracting increased interest for use as flow control actuators because they have no moving parts, yet they offer high control authority and oscillation over a wide range of operating frequencies [25]. Fluidic oscillators have been proven to yield significant aerodynamic improvements

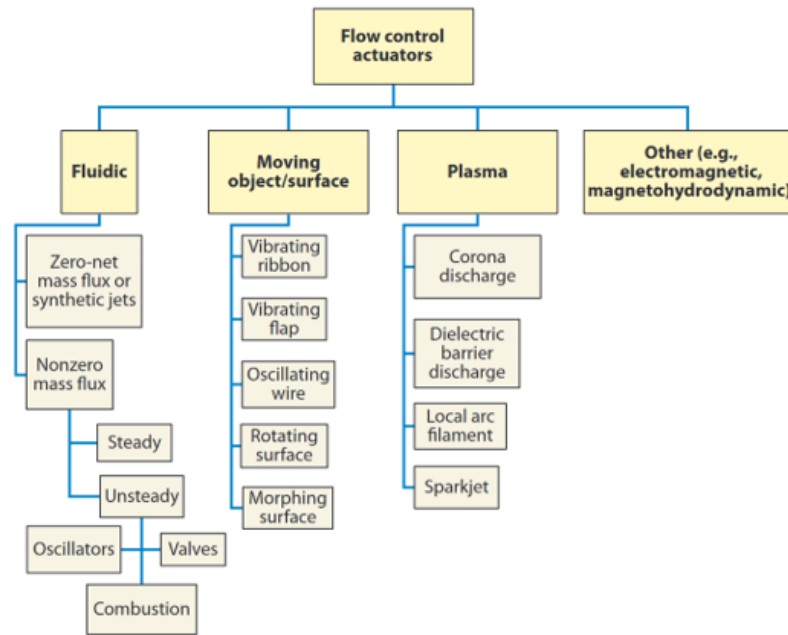


Figure 1.1: A type classification of flow control actuators [2]

by preventing flow separation over a highly deflected flap [1, 22].

Pulsed-plasma jets have also been studied in the use of flight control [16]. A pulsed-plasma jet (sometimes called a "spark jet") is a high-speed synthetic jet that is generated by striking an electrical discharge in a small cavity. The gas in the cavity pressurizes owing to the heating and escapes through a small orifice. This high-pressure gas issues through the orifice and forms the pulsed-plasma jet. The velocity of the plasma jet is related to the cavity pressure, which in turn depends on the rate of deposition of energy. Inside the cavity, the compression process that drives the jet is followed by a rarefaction wave that draws fluid back into the cavity and, hence, recharges it for the next pulse. Because the actuator operates without an external gas supply, the pulsed-plasma jet is a type of synthetic jet, which has *zero net* mass flux across the orifice [27].

The focus of this thesis will be on another type of a fluidic actuator called the synthetic jet actuator (SJA), another type of zero net mass flux actuator; and as the name suggests, this actuator does not use an external fluid source. Figure 1.2

illustrates the components of a single SJA. An array of these actuators can be embedded in the surface of an airfoil (i.e., aircraft wing). Each actuator forms a train of vortices at the orifice of a small cavity created by a vibrating diaphragm controlled by a voltage signal. The jet vortices interact with the cross-flow over the airfoil to deflect the airflow in a controllable way, and this boundary-layer flow control can be used to improve aerodynamic efficiency or increase aircraft trajectory tracking control authority at high angles of attack. Additionally, SJAs are light-weight, low cost, amenable to various sizes, and less mechanically complex than conventional control surfaces [2].

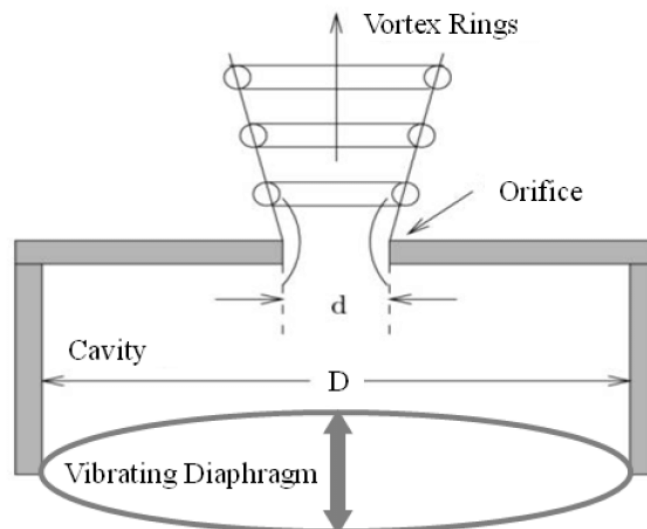


Figure 1.2: Schematic layout of a SJA [23]

Applications of SJAs include trajectory tracking control, limit cycle oscillation suppression, and boundary-layer flow control. Studies have shown that they can achieve control effectiveness at high angles of attack, as demonstrated in high fidelity CFD studies [6]. SJAs have also been shown to provide flow separation control in airfoils [7, 15, 26]. Moreover, SJAs can expand the operational range of effectiveness of trajectory tracking control systems when used in conjunction with mechanical deflection control surfaces [4, 7, 15, 28].

1.2 Challenges Using Synthetic Jet Actuators

One challenge in developing control laws using SJAs is compensating for the nonlinear characteristics of the actuator dynamics. The input-output relationship of SJA is nonlinear and contains parametric uncertainty. The actuator non-linearity function can be denoted by $N()$ and depends on the SJA input voltage in the sense that

$$u(t) = N(v(t)) = N(A_{pp}^2)$$

where $v(t) = A_{pp}^2$, with A_{pp} being the peak-to-peak amplitude of the input voltage signal applied to the SJA piezoelectric diaphragm, which generates the air flow; and $u(t)$ is the equivalent *virtual deflection angle* applied to the airfoil (i.e., the equivalent deflection angle of a mechanical control flap).

Empirical evidence has shown that the SJA's nonlinear characteristic $N()$ depends significantly on uncertain values of constant aerodynamic parameters, which results in parametric uncertainty in the SJA model. As shown in [4], the parameterized model for the j^{th} SJA in the i^{th} array is given by:

$$u_{ij}(t) = \theta_2^* i j - \frac{\theta_{1ij}^*}{v_i(t)}, \quad \text{for } i = 1, 2, \dots, m$$

The parameters θ_1^* and θ_2^* represent uncertain aerodynamic parameters for which the control scheme must compensate.

1.3 Neural Networks in Adaptive Control Systems

One method to compensate for uncertain or unmodeled dynamics is through the use of intelligent control systems. Intelligent control refers to a class of control techniques that employ various computing approaches such as neural networks (NN), Bayesian probability, fuzzy logic, machine learning, reinforcement learning, evolutionary computation, and genetic algorithms.

The application of neural network-based control algorithms gained popularity around the 1980s and early 1990s. Their popularity died down briefly in the late

90s; but they have seen a recent resurgence as a result of advancements in computing capabilities and microelectronics. Today, NNs are a popular technology used in many applications, such as classification algorithms and pattern recognition.

A NN is a machine learning algorithm originally motivated by the goal to develop algorithms that can mimic the functions of the human brain. Experimental evidence in neuroscience suggests that learning algorithms that process the senses stem from a single algorithm. These experiments show, for example, that if the optic nerve is rerouted to the auditory cortex, that this portion of the brain can learn how to process images [8]. This learning capability of biological neural networks can be mimicked using artificial NNs by virtue of the *universal approximation property* of NNs, and this property can be leveraged to develop NN-based control systems with unknown or ill-defined dynamic models [14].

A NN is a function that is trained by using measurable input-output data. The trained NN can be used to predict the outputs of new input values. A typical multilayer NN with an input layer, an output layer, and one hidden layer is shown in Figure 1.3. This NN has two layers of adjustable weights. This type of NN is a static feedforward network, which is the earliest and simplest type of artificial NN. Given trained matrices of ideal NN weights W and V , the feedforward mapping is mathematically expressed as

$$y = W^T \sigma(V^T x)$$

where $\sigma(\cdot)$ denotes a nonlinear activation function.

NN activation functions are defined via mathematical input-output relations. Commonly used activation functions include the sigmoid function, which can be expressed as

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

the hyperbolic tangent, which is defined as

$$\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

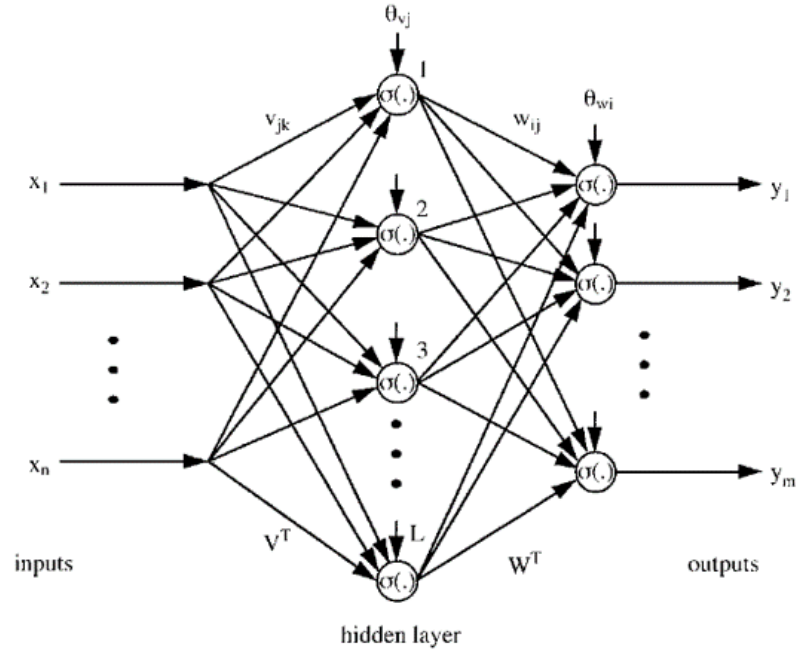


Figure 1.3: Diagram of a neural network

and the rectifier, which is given by

$$\sigma(z) = \max(0, z)$$

It has been shown that a multilayer feedforward network is capable of approximating any general mathematical function $f(x)$ to within a prescribed degree of accuracy, such that

$$f(x) = W^T \sigma(V^T x) + \varepsilon$$

for a given number L of hidden layers, where ε denotes the functional reconstruction error. Based on the universal approximation property of NNs, the functional reconstruction error can be upper bounded by a known positive constant ε_N in the sense that

$$|\varepsilon| < \varepsilon_N$$

The NN approximate $\hat{f}(x)$ of $f(x)$ can be expressed using standard “hat” notation as

$$\hat{f}(x) = \hat{W}^T \sigma(\hat{V}^T x)$$

where $\hat{W}(t)$ and $\hat{V}(t)$ are adaptive estimates of the constant ideal NN weights W and V that are generated online using the subsequently described weight tuning algorithms [10].

1.4 Benefits of Adaptive and Neural Network-based Control

There have been numerous recent results by Ramos-Pedroza et al. [17, 18, 19, 20, 21], which have shown promising results for robust nonlinear control methods to compensate for the model uncertainty in SJAs. Robust control methods (e.g., standard sliding mode control methods) can often require high-gain feedback, which can lead to an increased control energy requirement. The adaptive and NN-based control methods investigated in this thesis are motivated, in part, by the desire to reduce the need for high-gain feedback, and hence, the control energy requirement. Philosophically, the adaptive control methods achieve this reduction by adapting in real-time to parametric uncertainty, as opposed to compensating for the maximum possible deviation (or upper bound) in the model uncertainty. A comparative numerical study showing the improvement achieved using adaptive and NN-based control over robust control (referred to as exact model knowledge (EMK) control in this work) is one of the primary contributions in this thesis.

The control methods presented in this thesis are based on the assumption that the complete state is available for feedback. This is a mild assumption since it has been shown in recent research [9, 11, 12, 17, 18, 19, 20, 21] that nonlinear estimators can be utilized to achieve finite-time state estimation in the presence of model uncertainty and SJA parameter deviations. The derivation of robust nonlinear estimators (or observers) is beyond the scope of the work in this thesis, and details of an estimator design are omitted here to avoid distraction from the main results of this research.

Chapter 2

Mathematical Background

This section summarizes the fundamental concepts of Lyapunov stability theory for nonautonomous systems. The definitions and concepts that will be used in the paper are presented here with no proofs. A detailed treatment of the analysis of nonlinear systems can be found in [13] and in [24] for adaptive control.

2.1 Stability Definitions

Consider the nonautonomous dynamic system

$$\dot{x} = f(x, t) \quad x(t_0) = x_0 \quad x \in \mathbb{R}^n \quad (2.1)$$

with the assumption that $f(x, t)$ satisfies the conditions for uniqueness and existence: it is piecewise continuous in t , uniformly continuous in t , and Lipschitz continuous with respect to x . The point $x_e \in \mathbb{R}^n$ is an equilibrium point of 2.1 if $f(x_e, t) \equiv 0$. That equilibrium point is said to be:

Uniformly stable if for any given $\varepsilon > 0$, $\exists \delta(\varepsilon) > 0$ such that

$$\|x(t_0)\| < \delta(\varepsilon) \quad \Rightarrow \quad \|x(t)\| < \varepsilon, \forall t > 0.$$

Uniformly asymptotically stable if it is uniformly stable and if $\exists \delta_1 > 0$ such that

$$\|x(t_0)\| < \delta_1 \quad \Rightarrow \quad \|x(t)\| \rightarrow 0 \quad \text{as } t \rightarrow \infty.$$

Globally uniformly asymptotically stable if it is uniformly asymptotically stable $\forall x(t_0)$.

The term uniformly means that the stability result is not dependent on time; the term globally means that the stability result is not dependent on initial conditions.

2.2 Lyapunov Stability Analysis

The fundamental idea behind the direct method of Lyapunov is based on the stability theorem of Lagrange-Dirichlet. That is, when the potential energy function is at a minimum or maximum, the system is in a state of equilibrium. When it is at a minimum, the equilibrium will be stable with respect to small disturbances. Instead of using a potential energy function, however, Lyapunov's method proves the stability of nonlinear differential equations using a generalized notion of energy functions.

The function $V(x)$ is referred to as a Lyapunov candidate function (or simply a Lyapunov function), and it represents a contribution to the total energy by the states of the system. The time derivative of a Lyapunov function along trajectories of the system in (2.1) can be expressed as:

$$\dot{V}(x, t) = \sum_{i=1}^n \frac{\partial V}{\partial x_i} f_i(x, t) + \frac{\partial V}{\partial t} = \frac{\partial V}{\partial x} f(x, t) + \frac{\partial V}{\partial t} \quad (2.2)$$

The stability properties of the nonlinear, nonautonomous system in (2.1) can be analyzed via the Lyapunov derivative in Equation (2.2) by using a Lyapunov-based stability analysis.

In the following section, the definitions for positive definiteness, positive semi-definiteness, decreasence, and radial unboundedness of functions will be given. The

definitions will then be utilized to provide an overview of Lyapunov stability analysis for nonautonomous systems.

2.2.1 Definitions for Lyapunov Stability Analysis

Let a function $W : \mathcal{D} \times \mathbb{R}^+ \rightarrow \mathbb{R}$ be continuously differentiable with respect to all its arguments, where $\mathcal{D} \subset \mathbb{R}^n$. Then, $W(x, t)$ is said to be:

positive semidefinite if

$$W(0, t) = 0, \forall t \in \mathbb{R}^+, \text{ and}$$

$$W(x, t) \geq 0, \forall x \in \mathcal{D} - \{0\}, \forall t \in \mathbb{R}^+$$

positive definite if

$$W(0, t) = 0, \forall t \in \mathbb{R}^+, \text{ and}$$

$$W(x, t) > 0, \forall x \in \mathcal{D} - \{0\}, \forall t \in \mathbb{R}^+$$

decreasing if

$$W(0, t) = 0, \forall t \in \mathbb{R}^+, \text{ and}$$

$$W(x, t) < V(x), \forall x \in \mathcal{D} - \{0\}, \forall t \in \mathbb{R}^+$$

where $V(x) : \mathcal{D} \rightarrow \mathbb{R}$ is an autonomous positive definite function

radially unbounded if

$$W(x, t) \rightarrow \infty \text{ as } \|x\| \rightarrow \infty \text{ uniformly in } t.$$

2.2.2 Lyapunov's Second Theorem (The Direct Method of Lyapunov)

Let $V(x, t)$ be a non-negative function with derivative $\dot{V}(x, t)$ along the trajectories of the system in (2.1). Then, an equilibrium point is:

1. **Stable** if

$V(x, t)$ is positive definite, and

$\dot{V}(x, t)$ is negative semidefinite.

2. Uniformly stable if

$V(x, t)$ is positive definite and decrescent, and

$\dot{V}(x, t)$ is negative semidefinite.

3. Globally uniformly stable if

$V(x, t)$ is positive definite, decrescent, and radially unbounded, and

$\dot{V}(x, t)$ is negative semidefinite.

4. Globally uniformly asymptotically stable if

$V(x, t)$ is positive definite, decrescent, and radially unbounded, and

$\frac{\partial V}{\partial t} + \frac{\partial V}{\partial x} f(x, t) \leq -W_3(x)$, where $W_3(x)$ is positive definite.

It is important to note that these conditions are sufficient conditions, not necessary. This means that if the conditions are not met, one cannot conclude that the system is unstable. Chetaev's theorem can be used to prove instability for a system.

2.3 Barbalat's Lemma

In many cases, a Lyapunov function for a given dynamic system may not provide sufficient information to prove asymptotic stability. It can be very difficult to find a Lyapunov function with a negative definite derivative. In cases like these, Barbalat's Lemma can be used to prove asymptotic stability. Barbalat's Lemma states:

Let $f(t) : \mathbb{R} \rightarrow \mathbb{R}$ be a uniformly continuous function (i.e., $\dot{f}(t) \in L_\infty$) on $[0, \infty)$. Suppose that $\lim_{t \rightarrow \infty} \int_0^t f(\tau) d\tau$ exists and is finite. Then $\lim_{t \rightarrow \infty} f(t) = 0$.

2.4 Online Parameter Estimation

The model structure of the SJA dynamics is known, however the values of the model parameters often deviate from nominal, so they cannot be determined a priori. Input and output data can be processed online to estimate these parameters. Methods of parameter estimation, listed below, can fall in to two categories:

1. Phenomenological Models (physics-based models)
 - Equation error method
 - Output error method
 - Filter error method
2. Black-Box Models
 - Artificial neural networks
 - Fuzzy logic
 - Neuro-fuzzy
 - Support vector machine
 - Machine learning based methods

In this thesis, adaptive and neural network-based control methods (i.e., black-box models) are utilized to achieve asymptotic trajectory tracking control for aircraft in the presence of SJA model uncertainty.

Chapter 3

Controller Design with Exact Model Knowledge

In this chapter, a controller will be designed for a SJA-based UAV model by assuming that exact model knowledge (EMK) is available. Note that the control design in this chapter is being presented for the purposes of comparison only. In real-world control applications, EMK is not a realistic assumption; so the example in this chapter is included to motivate the adaptive and NN-based control designs presented in Chapters 4 and 5.

3.1 Dynamic Model and Properties

We consider a standard quasi-linear dynamic model of an aircraft equipped with SJA arrays, which can be expressed as

$$\dot{x} = Ax + Bu + f(x, t) = Ax + \sum_{i=1}^m b_i u_i + f(x, t) \quad (3.1)$$

where $A \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^{n \times m}$ denote known, constant state and input matrices, respectively; and $f(x, t) \in \mathbb{R}^n$ denotes a known nonlinear function. The control input (SJA array virtual deflection angle) $u(t) \in \mathbb{R}^m$ is a nonlinear function of the SJA array input voltages $v(t) \in \mathbb{R}^m$, which is explicitly given by [4]

$$u_i(t) = \theta_{2i}^* - \frac{\theta_{1i}^*}{v_i(t)}, i = 1, \dots, m \quad (3.2)$$

where $\theta_{1i}^*, \theta_{2i}^* \in \mathbb{R}$, for $i = 1, \dots, m$, denote known aerodynamic parameters, which are empirically determined a priori. Note, again, that the reason A , B , $f(x, t)$, and $\theta_{1i}^*, \theta_{2i}^*$ are being unrealistically treated as known quantities is that the controller in this chapter is based on the EMK assumption.

To address the SJA nonlinearity, the SJA array input voltages will be designed using the inverse function operation $NI(u_i(t))$ on the nonlinear model $u_i(t) = N(v_i(t))$, which can be expressed as

$$v_i(t) = NI(u_i(t)) = \frac{\theta_{1i}^*}{\theta_{2i}^* - u_{di}(t)} \quad (3.3)$$

where $u_{di}(t) \in \mathbb{R}$, for $i = 1, \dots, m$, denote auxiliary control signals. By substituting the inverse function in (3.3) into the SJA plant model described in (3.2), the aircraft dynamic model in (3.1) can be rewritten in control-affine form in terms of the auxiliary control signals $u_{di}(t)$.

3.2 Control Objective and Open Loop Dynamics

The objective is to design $v(t)$ such that $x(t)$ follows a desired model reference state $x_m(t) \in \mathbb{R}^n$. The model reference state is generated via the state space model

$$\dot{x}_m = A_m x_m + B_m \delta(t) \quad (3.4)$$

where $A_m \in \mathbb{R}^{n \times n}$ and $B \in \mathbb{R}^n$ denote model reference state and input matrices, respectively; and $\delta(t) \in \mathbb{R}$ is a reference input (e.g., a pilot joystick command).

To quantify the control objective, the tracking error $e(t) \in \mathbb{R}^n$ is defined as

$$e = x - x_m \quad (3.5)$$

The objective can be mathematically stated as $e(t) \rightarrow 0$.

The open loop dynamics can be obtained by taking the time derivative of $e(t)$ and using (3.1) - (3.4) as follows:

$$\begin{aligned}\dot{e} &= \dot{x} - \dot{x}_m \\ \dot{e} &= Ax + Bu + f(x, t) - A_m x_m - B_m \delta(t) \\ \dot{e} &= Ax + \sum_{i=1}^m b_i \left(\theta_{2i}^* - \frac{\theta_{1i}^*}{v_i(t)} \right) + f(x, t) - A_m x_m - B_m \delta(t) \\ \dot{e} &= Ax + \sum_{i=1}^m b_i \left(\theta_{2i}^* - \theta_{1i}^* \frac{\theta_{2i}^* - u_{di}(t)}{\theta_{1i}^*} \right) + f(x, t) - A_m x_m - B_m \delta(t)\end{aligned}$$

where EMK of the aerodynamic parameters θ_{1i}^* and θ_{2i}^* was assumed. The resulting expression can be simplified as

$$\dot{e} = Ax - A_m x_m + f(x, t) - B_m \delta(t) + Bu_d$$

By defining auxiliary terms as $\tilde{N}(t) \triangleq Ax - A_m x_m + f(x, t)$ and $N_d(t) \triangleq B_m \delta(t)$, the open loop dynamics can be expressed as

$$\dot{e} = \tilde{N} + N_d + Bu_d \quad (3.6)$$

For this preliminary EMK scenario, $\tilde{N}(t)$, $N_d(t)$ and B are known precisely; and the auxiliary control input is designed as

$$u_d = B^\# (-\tilde{N} - N_d - f(x, t) - ke) \quad (3.7)$$

where the superscript $\#$ denotes the pseudoinverse of a nonsquare matrix. After substituting (3.7) into (3.6), the closed loop dynamics can be obtained as

$$\dot{e} = -ke \quad (3.8)$$

3.3 Stability Analysis

Theorem 1: The control design in (3.2), (3.3), and (3.7) achieves global asymptotic tracking of the model reference state in the sense that

$$e(t) \rightarrow 0 \quad \text{as} \quad t \rightarrow \infty \quad (3.9)$$

for all $x(t_0) \in \mathbb{R}^n$.

Proof: Consider a positive definite function defined as

$$V = \frac{1}{2}e^T e \quad (3.10)$$

After taking the time derivative of (3.10) along trajectories of the closed-loop system in (3.8), $\dot{V}(t)$ can be expressed as

$$\dot{V} = -k\|e\|. \quad (3.11)$$

Based on (3.10) and (3.11), $V(t)$ is positive definite, and $\dot{V}(t)$ is negative definite. Thus, Lyapunov's Second Theorem can be used to conclude that $\|e(t)\| \rightarrow 0$ as $t \rightarrow \infty$.

3.4 Numerical Simulation Results

A numerical simulation was created to test the performance of the control design. The simulation is based on the dynamic model where $n = 3$ and $m = 6$ (i.e., 3-DOF flight control using 6 SJA arrays). The state vector contains the roll, pitch, and yaw rates, and the tracking error vector can be expressed as

$$e(t) = \begin{bmatrix} e_1(t) & e_2(t) & e_3(t) \end{bmatrix}^T$$

The state and input matrices, A and B , and reference state and reference input matrices, A_m and B_m , are defined based on the Barron Associates nonlinear tailless aircraft model (BANTAM) (for further details of the simulation model, see (Deb et

al., 2007)). The 3-DOF linearized model for the BANTAM was obtained analytically during trim conditions, where $M = 0.455$ is the Mach number, $\alpha = 2.7$ [deg] is angle of attack, and $\beta_s = 0$ [deg] denotes the side slip angle.

The reference state and input matrices used in the simulation are explicitly defined as

$$A_m = \begin{bmatrix} -61.1446 & 0 & -7.5238 \\ 0 & -174.3473 & 0 \\ -7.1579 & 0 & -1.4007 \end{bmatrix} \quad (3.12)$$

$$B_m = \begin{bmatrix} -1.7517 & 0 & 0.3096 \end{bmatrix}^T \quad (3.13)$$

As a proof of concept, the model reference state $x_m(t)$ in the simulation represents the desired external body axis motion that is generated in response to a reference command of

$$\delta(t) = \sin(t)$$

The matrices A and B were obtained analytically from the dimensional aerodynamic coefficients of the BANTAM [4]. For an aircraft equipped with six arrays of SJAs, these matrices are given by

$$A = \begin{bmatrix} -61.1273 & 0 & -7.6409 \\ 0 & -174.3472 & 0 \\ -7.2692 & 0 & -0.4543 \end{bmatrix} \quad (3.14)$$

$$B = \begin{bmatrix} -0.2292 & 0.2292 & -0.2292 & 0.2292 & -0.0306 & 0.0306 \\ 0.0599 & 0.0599 & 0.0804 & 0.0804 & -0.0256 & -0.0256 \\ -0.0084 & 0.0084 & -0.0535 & 0.0535 & 0.1177 & -0.1177 \end{bmatrix} \quad (3.15)$$

The results of 20 Monte Carlo-type simulations are shown in Figures 3.1 to 3.3. The results were obtained using the control gains selected as $k = \text{diag}\{15, 15, 15\}$. Each set of axes shows the control performance for 20 different scenarios, where each

plot shows the closed-loop response in the presence of 20 different sets of off-nominal values for the actual (plant) SJA parameters θ_{1i}^* , θ_{2i}^* , and B for $i = 1, \dots, 6$. The 20 sets of parameter values were randomly generated, which resulted in deviations of the actual parameters by up to 20% off nominal.

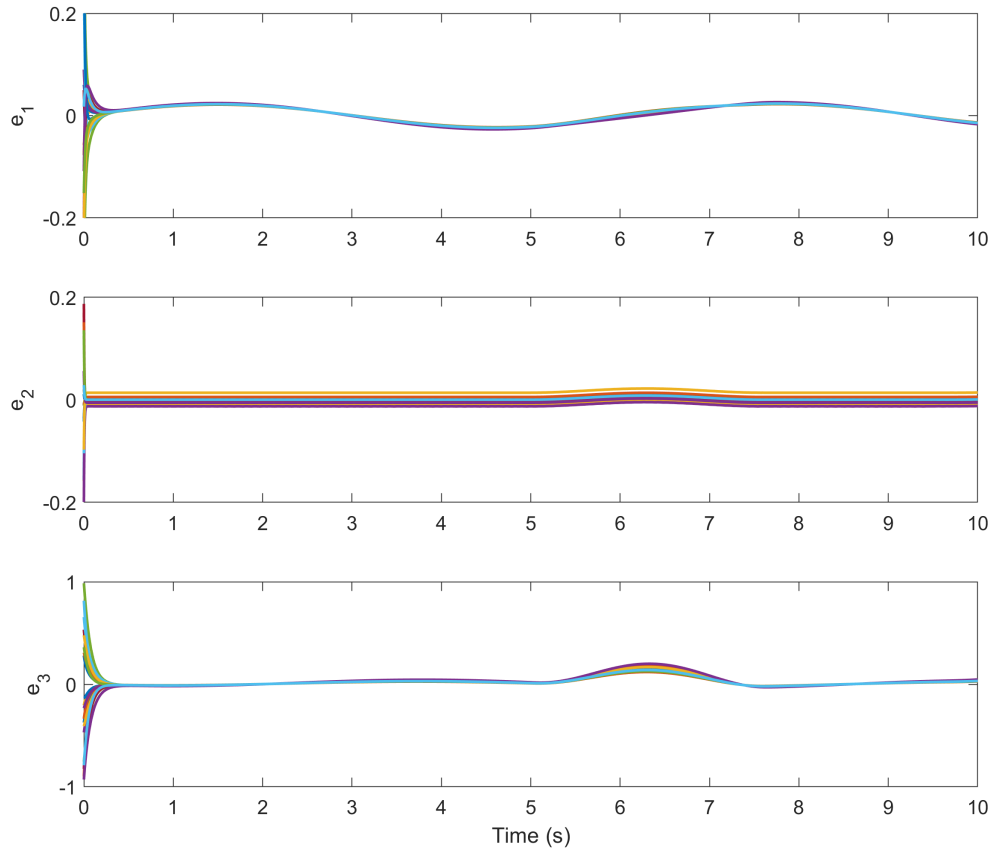


Figure 3.1: Closed-loop regulation of the steady state error for 20 runs

Figure 3.1 shows the closed-loop tracking error response and demonstrates rapid convergence of the tracking error to zero in all 20 cases. Figure 3.2 shows the virtual surface deflection control commands during closed-loop operation, and Figure 3.3 shows the SJA voltage control inputs commanded during closed-loop operation. The results demonstrate that the closed-loop system shows some steady-state error in all 20 cases, especially for the state $x_2(t)$. The subsequent adaptive and NN-based results

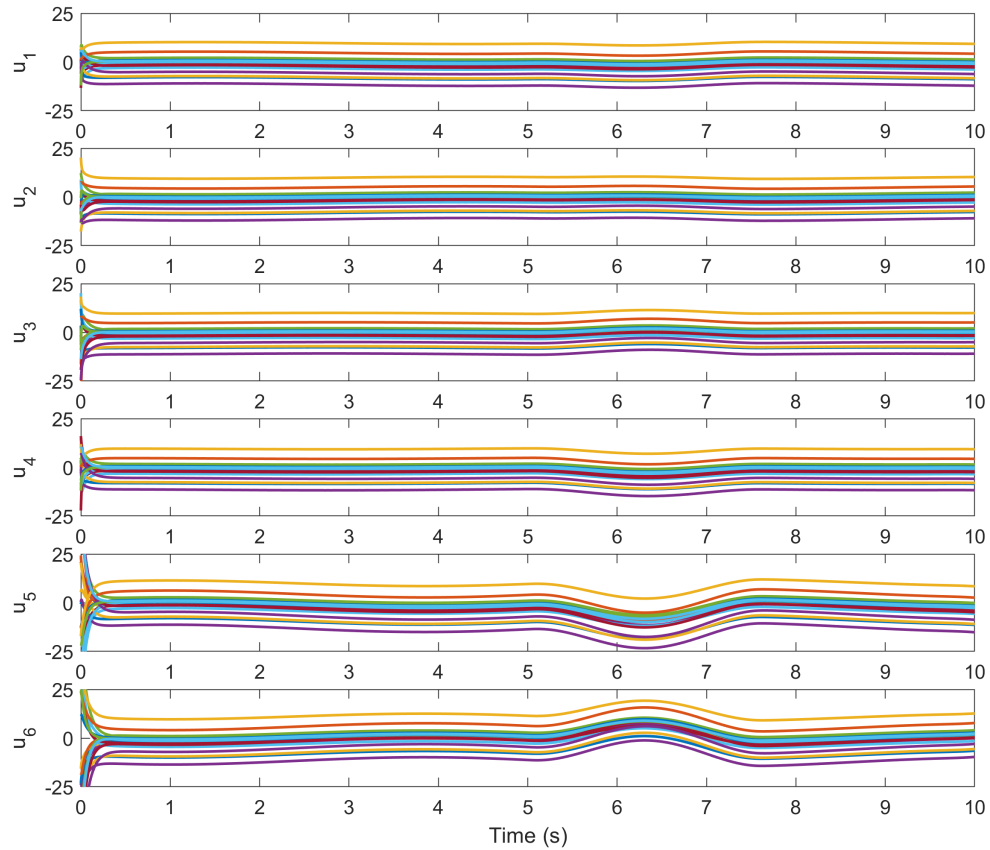


Figure 3.2: Closed-loop regulation of the virtual deflection angle for 20 runs

will demonstrate that this steady-state error is reduced over this robust EMK method. Figure 3.4 shows the convergence of the actual UAV states to the model reference states during closed-loop operation for the first iteration of our Monte Carlo-type simulation. The control commands remain within reasonable limits in all 20 cases.

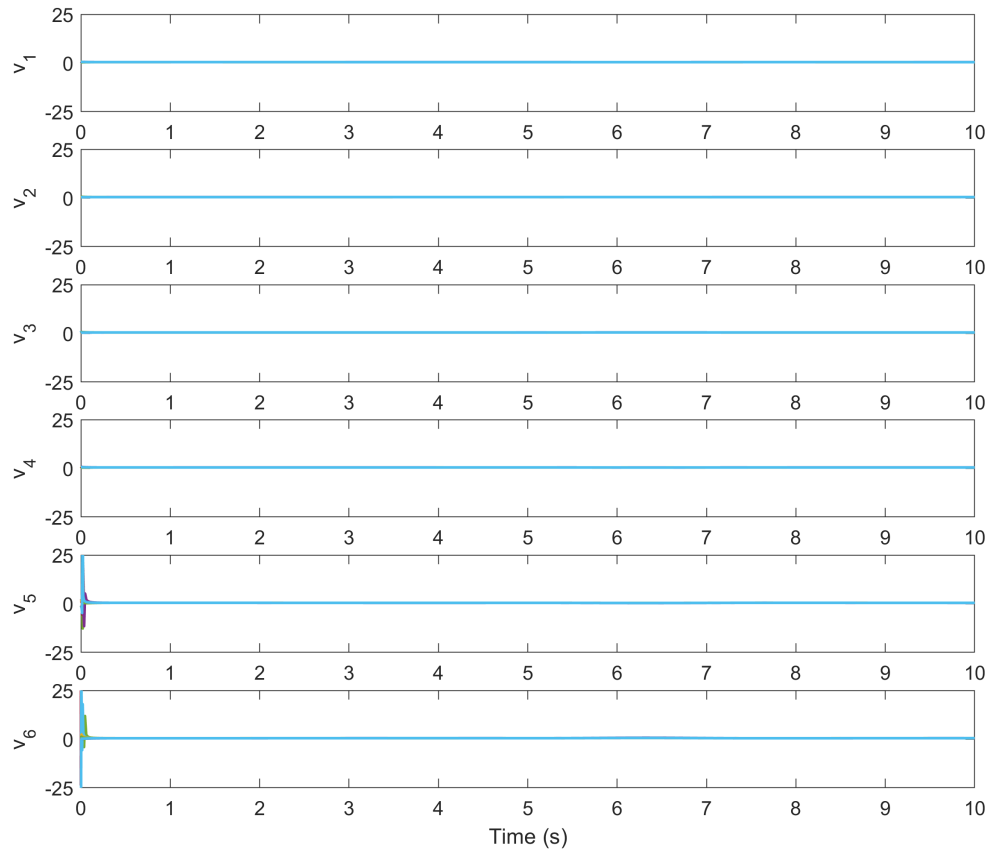


Figure 3.3: Control voltage signals commanded for the six SJA arrays

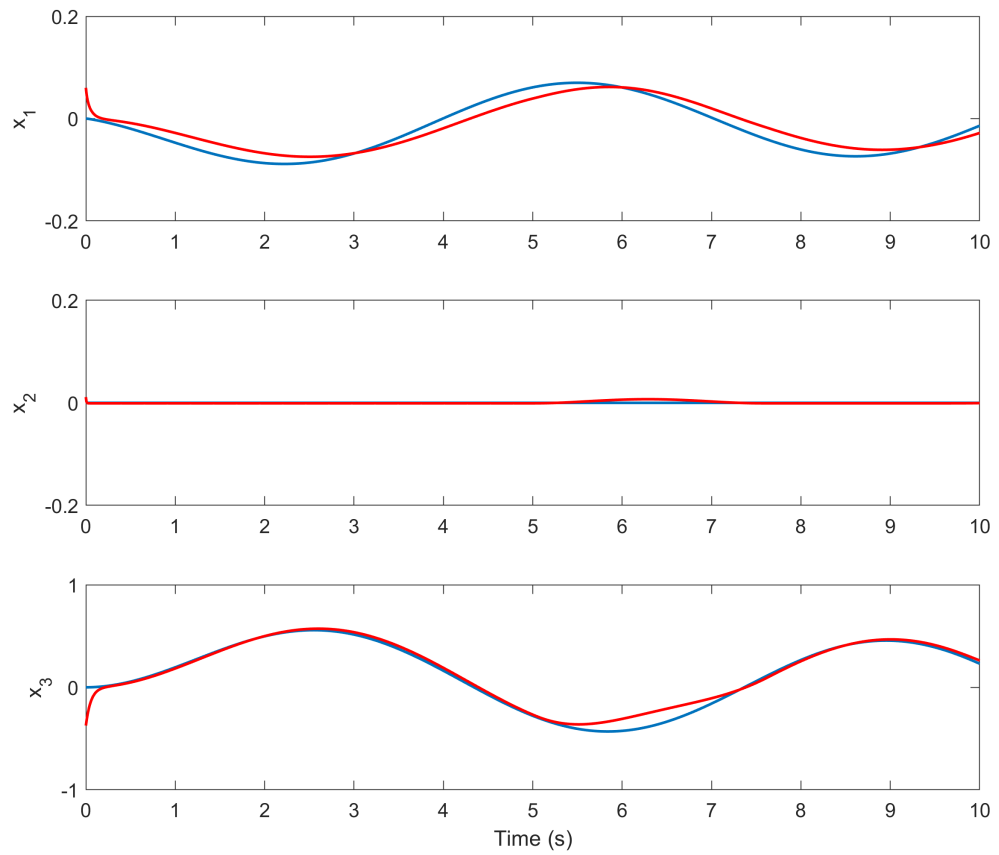


Figure 3.4: Model reference (blue) and actual state (red) during closed-loop exact-model knowledge control

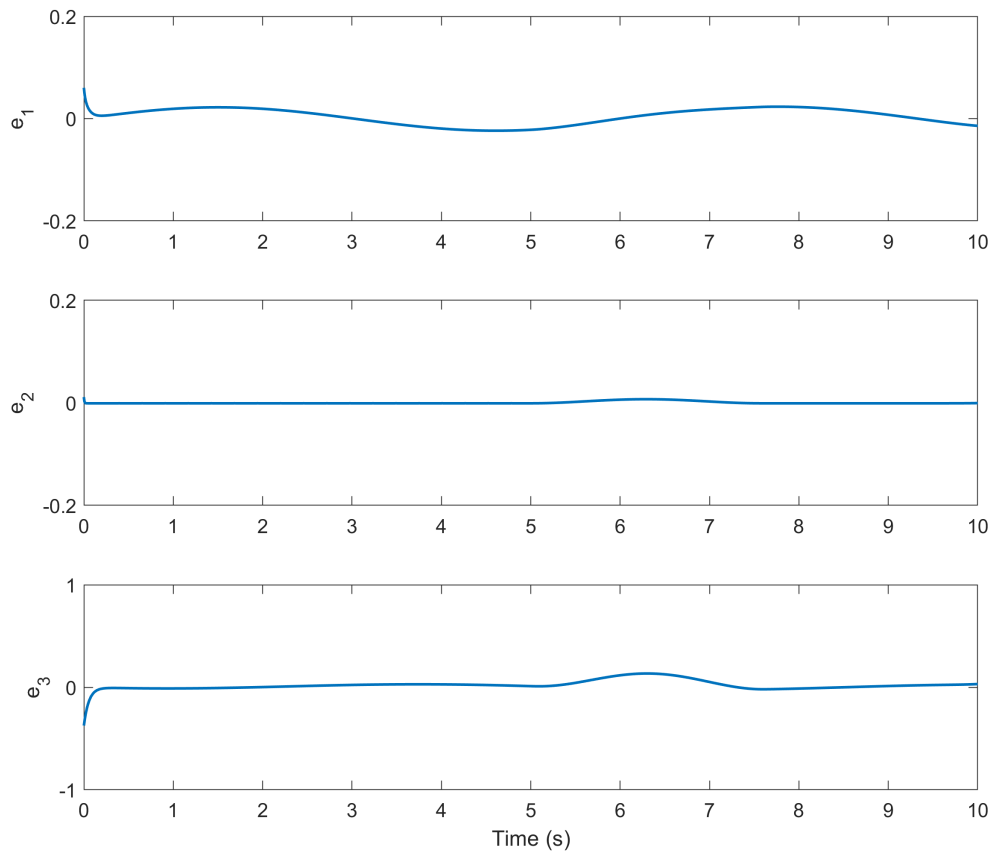


Figure 3.5: State error during closed-loop exact-model knowledge control

Chapter 4

Adaptive Control

In this chapter, the EMK assumption will be relaxed, and an adaptive control law will be designed to compensate for the input-multiplicative parametric uncertainty in the SJA model.

4.1 Dynamic Model and Properties

Similar to Chapter 3, the quasi-linear model of a SJA-based aircraft is given as

$$\dot{x} = Ax + Bu + f(x, t) = Ax + \sum_{i=1}^m b_i u_i + f(x, t) \quad (4.1)$$

In (4.1), the state and input gain matrices A and B contain parametric uncertainty, and $f(x, t)$ is an unknown nonlinear function.

Assumption 1: The unknown nonlinear function $f(x, t)$ in (4.1) is bounded and sufficiently smooth in the sense that $f(x, t), \dot{f}(x, t) \in L_\infty$.

4.2 Control Objective and Open Loop Error System

The objective is to design the voltage input $v(t)$ defined in (3.3) such that $x(t)$ follows a desired model reference state $x_m(t)$, where $x_m(t)$ is generated via the state space

model in (3.4).

To quantify the control objective, the tracking error $e(t)$ is defined as

$$e = x - x_m. \quad (4.2)$$

Thus, the control objective can be mathematically stated as $e(t) \rightarrow 0$.

The open loop dynamics can be obtained by taking the time derivative of (4.2) and using (3.1) - (3.4) as follows:

$$\dot{e} = \dot{x} - \dot{x}_m$$

$$\dot{e} = Ax + Bu_d - A_m x_m - B_m \delta(t) + f(x, t)$$

$$\dot{e} = A(e + x_m) - A_m x_m + Bu_d - B_m \delta(t) + f(x, t)$$

$$\dot{e} = Ae + Bu_d - B_m \delta(t) + Ax_m - A_m x_m + f(x, t) - f(x_m, t) + f(x_m, t)$$

where (4.2) was used, and $u_d(t)$ was introduced in (3.3).

The open loop error system can now be expressed in compact form as

$$\dot{e} = \tilde{N} + N_d + Bu_d + Y_1 \theta_1 \quad (4.3)$$

where the uncertain auxiliary terms $\tilde{N}(t)$ and $N_d(t)$ are explicitly defined as

$$\tilde{N} \triangleq Ae + f(x, t) - f(x_m, t) \quad (4.4)$$

$$N_d \triangleq -A_m x_m + f(x_d, t) - B_m \delta(t) \quad (4.5)$$

The motivation for the separation of terms in (4.4) and (4.5) is based on the fact that the following inequalities can be developed:

$$\tilde{N} \leq \rho(\|e\|)\|e\| \quad (4.6)$$

$$N_d \leq \zeta_d \quad (4.7)$$

where $\zeta_d \in \mathbb{R}^+$ is a known bounding constant, and $\rho(\cdot)$ denotes a positive, globally invertible nondecreasing function. Note that the bound in (4.6) can be obtained by using the mean value theorem along with Assumption 1.

In (4.3), $Y_1(x_m) \in \mathbb{R}^{(n \times n^2)}$ denotes a measurable regression matrix; and $\theta_1 \in \mathbb{R}^{n^2}$ is a vector of uncertain constants, which are defined via the linear parameterization

$$Y_1\theta_1 \triangleq Ax_m \quad (4.8)$$

To facilitate the subsequent adaptive control law design and stability analysis, the term Bu_d in (4.3) will be reparameterized as

$$Bu_d = Y_2\theta_2 \quad (4.9)$$

where $Y_2(u_d) \in \mathbb{R}^{(n \times nm)}$ is a measurable regression matrix, and $\theta_2 \in \mathbb{R}^{nm}$ is a vector of uncertain constants (i.e., θ_2 contains the uncertain elements of the B matrix). By using the parameterization in (4.9), Bu_d can be rewritten as

$$Bu_d = Y_2\tilde{\theta}_2 + \hat{B}u_d \quad (4.10)$$

where $\tilde{\theta}_2(t) \triangleq \theta_2 - \hat{\theta}_2(t)$ is the parameter estimate mismatch, and $\hat{\theta}_2(t) \in \mathbb{R}^{nm}$ denotes a subsequently defined adaptive estimate of θ_2 . In (4.10), $\hat{B}(t) \in \mathbb{R}^{(n \times m)}$ is an adaptive estimate of the uncertain input gain matrix B . The elements of the input gain matrix estimate $\hat{B}(t)$ are generated via the subsequently defined adaptive update law for $\hat{\theta}_2(t)$.

By utilizing (4.10), the open loop dynamics can be expressed as

$$\dot{e} = \tilde{N} + N_d + Y_1\theta_1 + Y_2\tilde{\theta}_2 + \hat{B}u_d \quad (4.11)$$

4.3 Control Design and Closed-loop Error System

By virtue of the parameterization in (4.10) the auxiliary control input $u_d(t)$ is pre-multiplied by the measurable input gain estimate $\hat{B}(t)$ in the open loop error system.

Based on the open loop error system in (4.11) and the subsequent stability analysis, the control input is designed as

$$u_d = \hat{B}^\#(-(k_1 + k_2)e - k_3 \text{sgn}(e) - Y_1 \hat{\theta}_1) \quad (4.12)$$

where the adaptive update laws are designed as

$$\dot{\hat{\theta}}_1 = -\Gamma_1 e^T Y_1 \quad \dot{\hat{\theta}}_2 = -\Gamma_2 e^T Y_2 \quad (4.13)$$

After substituting the control input in (4.12) into the open loop error system in (4.11), the closed-loop error system is obtained as

$$\dot{e} = \tilde{N} + N_d - (k_1 + k_2)e - k_3 \text{sgn}(e) + Y_1 \tilde{\theta}_1 + Y_2 \tilde{\theta}_2. \quad (4.14)$$

4.4 Stability Analysis

Theorem 2: The adaptive control law defined in (4.12) with parameter adaptation laws given by (4.13) achieves asymptotic trajectory tracking of the model reference in the sense that $e(t) \rightarrow 0$ as $t \rightarrow \infty$, provided the control gain k_2 is selected sufficiently large (see the subsequent stability proof).

Proof: Consider a non-negative function defined as

$$V(t) = \frac{1}{2} e^T e + \frac{1}{2} \tilde{\theta}_1^T \Gamma_1^{-1} \tilde{\theta}_1 + \frac{1}{2} \tilde{\theta}_2^T \Gamma_2^{-1} \tilde{\theta}_2 \quad (4.15)$$

where $\Gamma_1 \in \mathbb{R}^{n^2 \times n^2}$ and $\Gamma_2 \in \mathbb{R}^{nm \times nm}$ denote positive definite, diagonal adaptive gain matrices. After taking the time derivative of (4.15) along trajectories of the closed-loop error system in (4.14), $\dot{V}(t)$ can be expressed as

$$\dot{V}(t) = e^T (\tilde{N} + N_d - (k_1 + k_2)e - k_3 \text{sgn}(e) + Y_2 \tilde{\theta}_2) - \tilde{\theta}_1^T \Gamma_1^{-1} \dot{\tilde{\theta}}_1 - \tilde{\theta}_2^T \Gamma_2^{-1} \dot{\tilde{\theta}}_2 \quad (4.16)$$

The expression in (4.16) can be rewritten as follows:

$$\dot{V}(t) = -k_1 \|e\|^2 + e^T Y_1 \tilde{\theta}_1 + \tilde{\theta}_1^T \Gamma_1^{-1} \dot{\tilde{\theta}}_1 + e^T Y_2 \tilde{\theta}_2 - \tilde{\theta}_2^T \Gamma_2^{-1} \dot{\tilde{\theta}}_2 + e^T \tilde{N} - k_2 \|e\|^2 + e^T N_d - k_3 \|e\| \quad (4.17)$$

After substituting the adaptive update laws in (4.13) and canceling common terms, the $\dot{V}(t)$ expression in (4.17) can be expressed as

$$\dot{V}(t) = -k_1 \|e\|^2 + e^T \tilde{N} - k_2 \|e\|^2 + e^T N_d - k_3 \|e\| \quad (4.18)$$

By utilizing the bounding inequalities in (4.6) and (4.7), the expression in (4.18) can be upper bounded as

$$\dot{V}(t) \leq -k_1 \|e\|^2 + \rho(\|e\|) \|e\|^2 - k_2 \|e\|^2 + \zeta_d \|e\| - k_3 \|e\| \quad (4.19)$$

where the fact that $|\gamma| \geq \|\gamma\| \forall \gamma \in \mathbb{R}^n$ (the triangle inequality) was used. By combining terms, the upper bound in (4.19) can be expressed as follows:

$$\dot{V}(t) \leq -k_1 \|e\|^2 - (k_2 - \rho(\|e\|)) \|e\|^2 - (k_3 - \zeta_d) \|e\| \quad (4.20)$$

By selecting $k_3 > \zeta_d$, the following expression can be obtained from (4.20):

$$\dot{V}(t) \leq -c \|e\|^2 \quad (4.21)$$

for some constant $c \in \mathbb{R}^+$, where k_2 is selected to satisfy

$$k_2 \geq \rho(\|e\|). \quad (4.22)$$

Thus, (4.21) is satisfied on the domain \mathcal{D} that is defined as

$$\mathcal{D} \triangleq \{e \in \mathbb{R}^n \mid \|e\| \leq \rho^{-1}(k_2)\}. \quad (4.23)$$

The expressions in (4.15) and (4.21) can now be used to conclude that $V(t) \in \mathcal{L}_\infty$ in \mathcal{D} ; hence, $e(t)$, $\tilde{\theta}_1(t)$, and $\tilde{\theta}_2(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $e(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that $x_m(t) \in \mathcal{L}_\infty$ can be used along with (4.2) to prove that $x(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given

that $\tilde{\theta}_1(t)$ and $\tilde{\theta}_2(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that θ_1 and $\theta_2 \in \mathcal{L}_\infty$ in \mathcal{D} can be used to conclude that $\hat{\theta}_1(t)$ and $\hat{\theta}_2(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $x_m(t) \in \mathcal{L}_\infty$ in \mathcal{D} , (4.8) can be used to prove that $Y_1(x_m) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $e(t)$, $\hat{\theta}_1(t)$, $\hat{\theta}_2(t)$, and $Y_1(x_m) \in \mathcal{L}_\infty$ in \mathcal{D} , it follows that $\hat{B}(t) \in \mathcal{L}_\infty$ in \mathcal{D} ; and (4.12) can be used to conclude that $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the SJA equations in (3.2) and (3.3) can be used to conclude that $v(t)$ and $u(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $e(t)$, $\hat{\theta}_2(t)$, $Y_1(x_m)$, $Y_2(u_d)$, $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} , Inequalities (4.6) and (4.7) can be used along with (4.11) to conclude that $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that $x_m(t) \in \mathcal{L}_\infty$ in \mathcal{D} can be used along with (4.2) to conclude that $\dot{x}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} , it follows that $e(t)$ is uniformly continuous in \mathcal{D} . Barbalat's lemma can now be invoked to conclude that $e(t) \rightarrow 0$ as $t \rightarrow \infty$ for $e(0) \in \mathcal{D}$. Since the domain \mathcal{D} defined in (5.16) can be made arbitrarily large by judicious selection of the control gain k_2 , this is a semi-global result.

4.5 Numerical Simulation Results

A numerical simulation was created to test the performance of the proposed control law described in this section. The results of 20 Monte Carlo simulations are shown in Figures 4.1 to 4.3. The results were obtained using control gains selected as $k_1 = \text{diag}\{5, 5, 5\}$, $k_2 = \text{diag}\{15, 15, 15\}$, and $\gamma = .007$. Each set of axes shows the control performance for 20 different scenarios, where each plot shows the closed-loop response in the presence of 20 different sets of off-nominal values for the actual (plant) SJA parameters θ_{1i} , θ_{2i} , and B for $i = 1, \dots, 6$. The 20 sets of parameter values were randomly generated, which resulted in deviations of the actual parameters by up to 20% off nominal.

Figure 4.1 shows the closed-loop tracking error response and demonstrates rapid convergence of the tracking error to zero in all 20 cases. Figure 4.2 shows the virtual surface deflection control commands during closed-loop operation, and Figure 3.3 shows the SJA voltage control inputs commanded during closed-loop operation. The results demonstrate that the closed-loop system remains stable in all 20 cases, and asymptotic tracking is achieved throughout the range of uncertainty tested. Figures

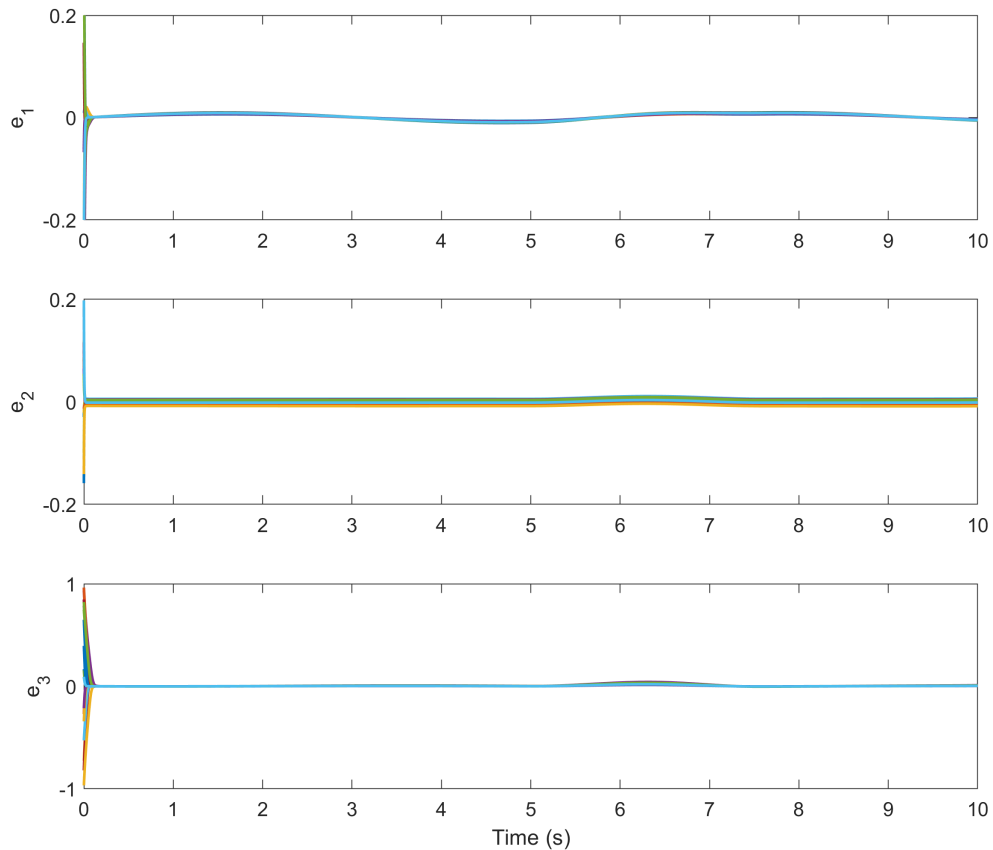


Figure 4.1: Closed-loop regulation of the steady state error for 20 runs

4.4 and 4.5 show the convergence of the actual states to the model reference states during closed-loop operation for the first iteration of our Monte Carlo-type simulation. The control commands remain within reasonable limits in all 20 cases. Most importantly, these results demonstrate a significant improvement in the steady-state tracking error is achieved using the adaptive control method, as opposed to the robust EMK method in Chapter 3. However, the disturbance is still creating a noticeable deviation in the flight trajectory, which is particularly evident in the $e_3(t)$ plot in Figure 4.1. It will be shown in the following chapter that the disturbance rejection is significantly improved using a NN-based control method.

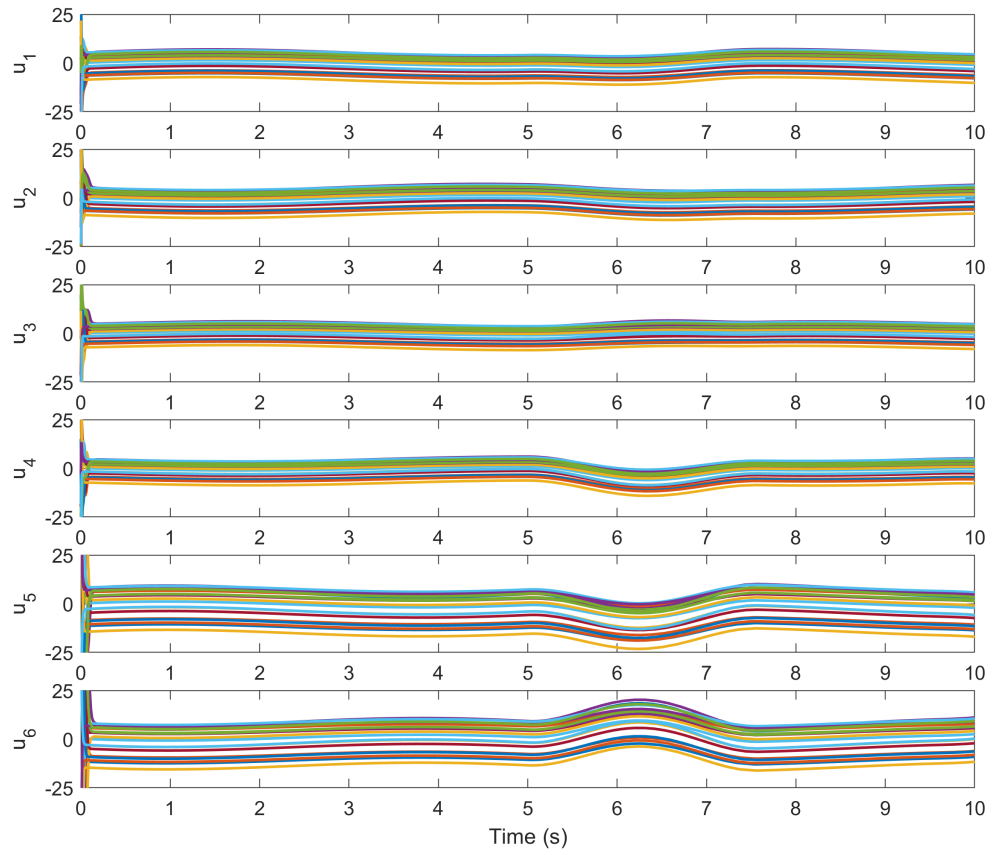


Figure 4.2: Closed-loop regulation of the virtual deflection angle for 20 runs

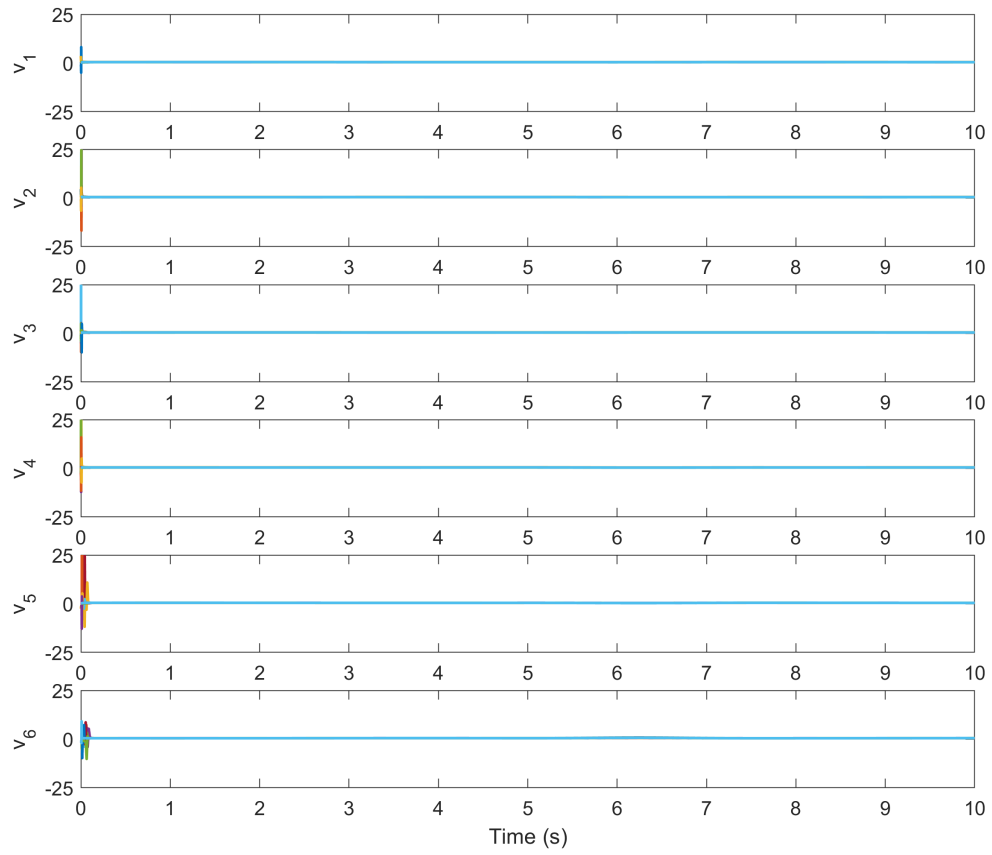


Figure 4.3: Control voltage signals commanded for the six SJA arrays

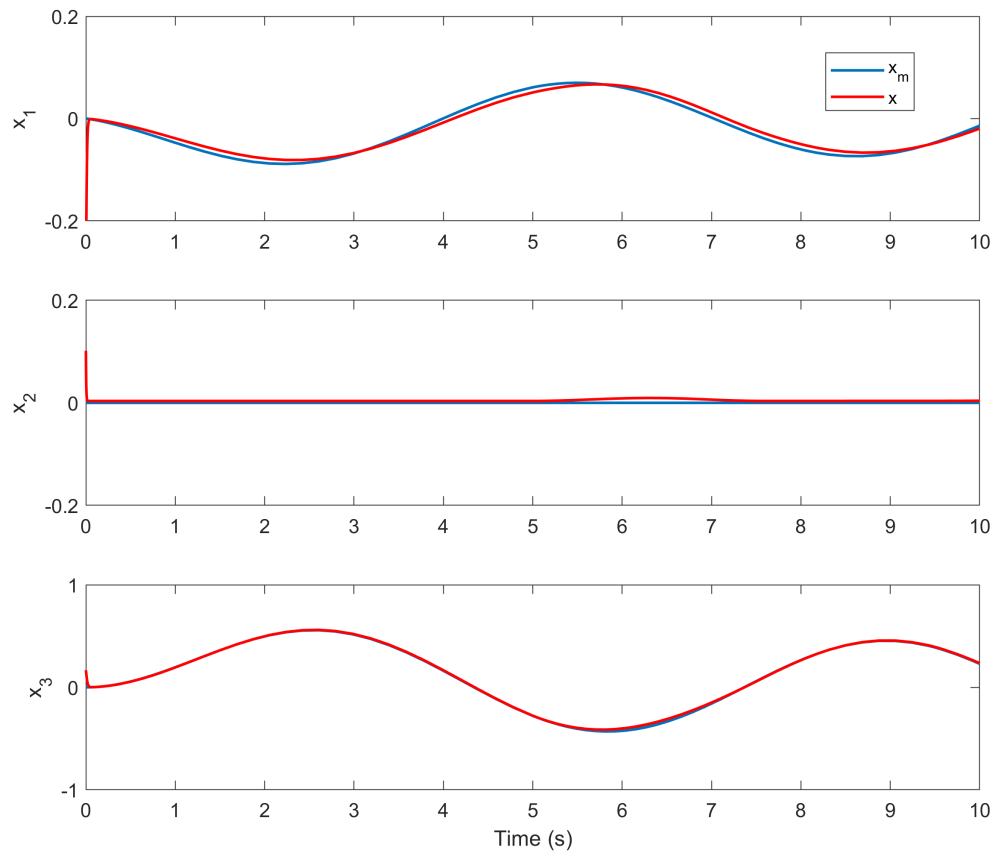


Figure 4.4: Model reference (blue) and actual state (red) with adaptive control

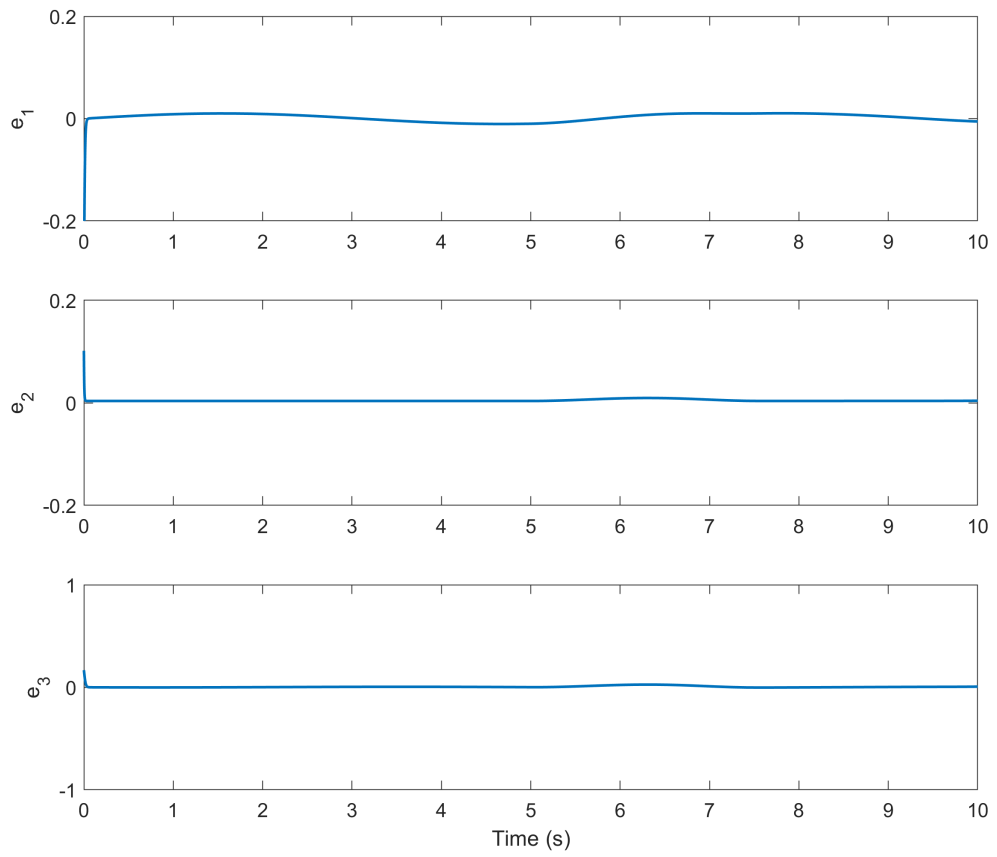


Figure 4.5: State error with adaptive control

Chapter 5

Neural Network Based Adaptive Control

5.1 Dynamic Model and Properties

In this chapter, a controller will be designed for a SJA-based UAV model in the presence of unmodeled disturbances by using an adaptive control law and a NN to compensate for both the uncertain parameters and disturbances. The numerical simulation results in this chapter will demonstrate the improved disturbance rejection that is achieved using the NN.

We again consider a quasi-linear model of the aircraft dynamics with SJA given by

$$\dot{x} = Ax + Bu + f(x, t) = Ax + \sum_{i=1}^m b_i u_i + f(x, t) \quad (5.1)$$

The objective is to design $u(t)$ such that $x(t)$ achieves a desired model reference state $x_m(t)$. The model reference state is generated via

$$\dot{x}_m = A_m x_m + B_m \delta(t) \quad (5.2)$$

where $\delta(t)$ is a reference input signal.

5.2 Control Objective and Open Loop Error System

Similar to the procedure in Chapters 3 and 4, the control objective is quantified by defining the tracking error $e(t)$ as

$$e = x - x_m \quad (5.3)$$

Thus, the objective can be stated mathematically as

$$e \rightarrow 0. \quad (5.4)$$

By taking the time derivative of (5.3) and following a procedure identical to that in Chapter 4 (i.e., see (4.3) - (4.10)), the open loop error system can be formulated as

$$\dot{e} = \tilde{N} + N_{d1} + Y_1\theta_1 + Y_2\tilde{\theta}_2 + \hat{B}u_d + f(x, t) \quad (5.5)$$

5.3 Control Design and Closed-loop Error System

The NN-based control design in this chapter is motivated by the desire to compensate for unmodeled, nonlinear external disturbances. In such a scenario, the standard adaptive control approach described in Chapter 4 is not sufficient to achieve reliable control performance. This section presents a non-trivial extension to the adaptive control approach in Chapter 4, which rigorously develops a NN-based control law to formally compensate for unknown nonlinear disturbances in the plant dynamics.

5.3.1 NN Properties

By virtue of the well-known universal approximation property of NN, there exists ideal weights and thresholds such that the function $f(x, t)$ can be represented by a three-layer NN as

$$f(x, t) = W^T \sigma(V^T x) + \varepsilon(x) \quad (5.6)$$

for some given input $x(t)$ (i.e., the state vector in this case). In (5.6), $V \in \mathbb{R}^{(N_1+1) \times N_2}$ and $W \in \mathbb{R}^{(N_2+1) \times n}$ are bounded constant ideal weight matrices for the first-to-second and second-to-third layers, respectively, where N_1 is the number of neurons in the input layer, N_2 is the number of neurons in the hidden layer, and n is the number of neurons in the third layer (or the output layer, which corresponds to the dimension of the state vector $x(t)$). The activation function in (5.6) is denoted by $\sigma(\cdot) \in \mathbb{R}^{(N_2+1) \times n}$, and $\varepsilon(x) \in \mathbb{R}^n$ is the functional reconstruction error. The activation function $\sigma(\cdot)$ can be defined using any of the functions described in the Introduction (e.g., sigmoid, hyperbolic tangent, rectifier).

Based on the open loop error dynamics in (5.5), the control input is designed to include a NN-based estimate of the nonlinear function $f(x, t)$ as

$$u_d = \hat{B}^{-1}(-(k_1 + k_2)e - k_3 \text{sgn}(e) - Y_1 \hat{\theta}_1 - \hat{f}(x, t)) \quad (5.7)$$

where $\hat{B}(t)$, $\hat{\theta}_1(t)$, and $Y_1(x_m)$ are defined as in Chapter 4. In (5.7), $\hat{f}(x, t)$ is a NN estimate of the unknown function $f(x, t)$ defined as

$$\hat{f}(x, t) = \hat{W}^T \sigma(\hat{V}^T x) \quad (5.8)$$

where $\hat{W}(t)$ and $\hat{V}(t)$ are adaptive estimates of the constant ideal weight matrices W and V (with appropriate dimensions), which are generated via the adaptive update laws

$$\dot{\hat{W}} = \Gamma_3 e^T (\hat{\sigma} + \hat{\sigma}' \hat{V}^T x) \quad (5.9)$$

$$\dot{\hat{V}} = \Gamma_4 e^T \hat{W}^T \hat{\sigma}' x \quad (5.10)$$

The closed-loop dynamics can be expressed by substituting (5.7) and (5.8) into (5.5) and following a few algebra steps as follows:

$$\dot{e} = \tilde{N} + N_{d1} - (k_1 + k_2)e - k_3 \text{sgn}(e) - \hat{W}^T \sigma(\hat{V}^T x) + W^T (V^T x) + \varepsilon(x) + Y_1 \tilde{\theta}_1 + Y_2 \tilde{\theta}_2$$

By using the following Taylor series approximation of $\sigma(V^T x)$ about $\hat{V}^T x$

$$\sigma(V^T x) = \sigma(\hat{V}^T x) + \sigma'(\hat{V}^T x)\tilde{V}^T x + o(\tilde{V}^T x)^2$$

By using the Taylor series approximation and substituting the parameter mismatch matrices defined as

$$\tilde{W} = W - \hat{W} \quad \text{and} \quad \tilde{V} = V - \hat{V} \quad (5.11)$$

The closed-loop error dynamics can be rewritten as follows:

$$\begin{aligned} \dot{e} = & \tilde{N} + N_d - (k_1 + k_2)e - k_3 \text{sgn}(e) + \tilde{W}^T(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x) + \dots \\ & \tilde{W}^T \hat{\sigma} V^T x + \hat{W}^T(\hat{\sigma} + \hat{\sigma}'\tilde{V}^T x) + Y_1 \tilde{\theta}_1 + Y_2 \tilde{\theta}_2 \end{aligned}$$

where $N_d \triangleq N_{d1} + \tilde{W}^T \sigma' V^T x + o(\tilde{V}^T x)^2 + \varepsilon(x)$.

5.4 Stability Analysis

Consider a non-negative function defined as

$$V_L(x, t) = \frac{1}{2} e^T e + \frac{1}{2} \tilde{\theta}_1^T \Gamma_1^{-1} \tilde{\theta}_1 + \frac{1}{2} \tilde{\theta}_2^T \Gamma_2^{-1} \tilde{\theta}_2 + \frac{1}{2} \text{tr}(\tilde{W}^T \Gamma_3^{-1} \tilde{W}) + \frac{1}{2} \text{tr}(\tilde{V}^T \Gamma_4^{-1} \tilde{V}) \quad (5.12)$$

The time derivative along trajectories of the closed-loop error dynamics can be calculated as

$$\begin{aligned} \dot{V}_L(x, t) = & e^T (\tilde{N} + N_d - (k_1 + k_2)e - k_3 \text{sgn}(e) + \tilde{W}^T(\hat{\sigma} - \hat{\sigma}'\hat{V}^T x_d) \\ & + \tilde{W}^T \hat{\sigma} V^T x + \hat{W}^T(\hat{\sigma} + \hat{\sigma}'\tilde{V}^T x) + Y_1 \tilde{\theta}_1 + Y_2 \tilde{\theta}_2) \\ & - \tilde{\theta}_1^T \Gamma_1^{-1} \dot{\tilde{\theta}}_1 - \tilde{\theta}_2^T \Gamma_2^{-1} \dot{\tilde{\theta}}_2 - \text{tr}(\tilde{W}^T \Gamma_3^{-1} \dot{\tilde{W}}) - \text{tr}(\tilde{V}^T \Gamma_4^{-1} \dot{\tilde{V}}) \end{aligned}$$

After substituting the weight adaptation laws in (5.8) and (5.9) and canceling common terms, $\dot{V}_L(t)$ can be expressed as

$$\dot{V}_L = e^T(\tilde{N} + N_d) - (k_1 + k_2)\|e\|^2 - k_3\|e\|$$

By using the bounding inequalities in (4.6) and (4.7), \dot{V}_L can be upper bounded as

$$\dot{V}_L \leq \rho(\|e\|)\|e\|^2 + \zeta\|e\| - (k_1 + k_2)\|e\|^2 - k_3\|e\| \quad (5.13)$$

By following a procedure similar to that in Chapter 4, the upper bound of \dot{V}_L can be expressed as

$$\dot{V}_L \leq -c\|e\|^2. \quad (5.14)$$

for some constant $c \in \mathbb{R}^+$, where k_2 is selected to satisfy

$$k_2 \geq \rho(\|e\|). \quad (5.15)$$

Thus, (5.12) is satisfied on the domain \mathcal{D} that is defined as

$$\mathcal{D} \triangleq \{e \in \mathbb{R}^n \mid \|e\| \leq \rho^{-1}(k_2)\}. \quad (5.16)$$

The expressions in (5.12) and (5.14) can now be used to conclude that $V_L(t) \in \mathcal{L}_\infty$ in \mathcal{D} ; hence, $e(t), \tilde{\theta}_1(t), \tilde{\theta}_2(t), \tilde{W}(t), \tilde{V}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $e(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that $x_m(t) \in \mathcal{L}_\infty$ can be used along with (5.3) to prove that $x(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $\tilde{\theta}_1(t), \tilde{\theta}_2(t), \tilde{W}(t)$, and $\tilde{V}(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that θ_1, θ_2, W , and $V \in \mathcal{L}_\infty$ in \mathcal{D} can be used to conclude that $\hat{\theta}_1(t), \hat{\theta}_2(t), \hat{W}(t)$, and $\hat{V}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $x_m(t) \in \mathcal{L}_\infty$ in \mathcal{D} , (4.8) (which also applies to Chapter 5) can be used to prove that $Y_1(x_m) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $e(t), \hat{\theta}_1(t), \hat{\theta}_2(t), \hat{W}(t), \hat{V}(t)$, and $Y_1(x_m) \in \mathcal{L}_\infty$ in \mathcal{D} , it follows that $\hat{B}(t) \in \mathcal{L}_\infty$ in \mathcal{D} ; and (5.7) can be used to conclude that $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the fundamental SJA equations in (3.2) and (3.3) can be used to conclude that $v(t)$ and $u(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $e(t), \hat{\theta}_2(t), \tilde{W}(t), \hat{W}(t), \tilde{V}(t), \hat{V}(t), Y_1(x_m), Y_2(u_d)$, and $u_d(t) \in \mathcal{L}_\infty$ in \mathcal{D} , Inequalities (4.6) and (4.7) can be used along with the closed-loop error dynamics to conclude that $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Since $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} , the assumption that $x_m(t) \in \mathcal{L}_\infty$ in \mathcal{D}

can be used along with (4.2) to conclude that $\dot{x}(t) \in \mathcal{L}_\infty$ in \mathcal{D} . Given that $\dot{e}(t) \in \mathcal{L}_\infty$ in \mathcal{D} , it follows that $e(t)$ is uniformly continuous in \mathcal{D} . Barbalat's lemma can now be invoked to conclude that $e(t) \rightarrow 0$ as $t \rightarrow \infty$ for $e(0) \in \mathcal{D}$. Since the domain \mathcal{D} defined in (5.16) can be made arbitrarily large by judicious selection of the control gain k_2 , this is a semi-global result.

5.5 Simulation Results

A numerical simulation was created to test the performance of the proposed control law described in this section. The results of 20 Monte Carlo simulations are shown in Figures 5.1 to 5.3. The results were obtained using control gains selected as $k_1 = \text{diag}\{5, 5, 5\}$, $k_2 = \text{diag}\{15, 15, 15\}$, and $\gamma = .007$. Each set of axes shows the control performance for 20 different scenarios, where each plot shows the closed-loop response in the presence of 20 different sets of off-nominal values for the actual (plant) SJA parameters θ_{1i}^* , θ_{2i}^* , and B for $i = 1, \dots, 6$. The 20 sets of parameter values were randomly generated, which resulted in deviations of the actual parameters by up to 20% off nominal.

Figure 5.1 shows the closed-loop tracking error response and demonstrates rapid convergence of the tracking error to zero in all 20 cases. Moreover, this figure clearly shows a reduction in the trajectory deviation caused by the external disturbance. Figure 5.2 shows the virtual surface deflection control commands during closed-loop operation, and Figure 5.3 shows the SJA voltage control inputs commanded during closed-loop operation. The results demonstrate that the closed-loop system remains stable in all 20 cases, and asymptotic tracking is achieved throughout the range of uncertainty tested. Figure 5.4 shows the convergence of the actual states to the model reference states during closed-loop operation for the first iteration of our Monte Carlo-type simulation. The control commands remain within reasonable limits in all 20 cases. These results clearly show that the NN effectively rejects the disturbance that is injected around 6 seconds into the simulation.

Figures 5.5, 5.6, and 5.7 show a comparison of the mean squared error for each method. It is clear that the inclusion of adaptive control and the neural network

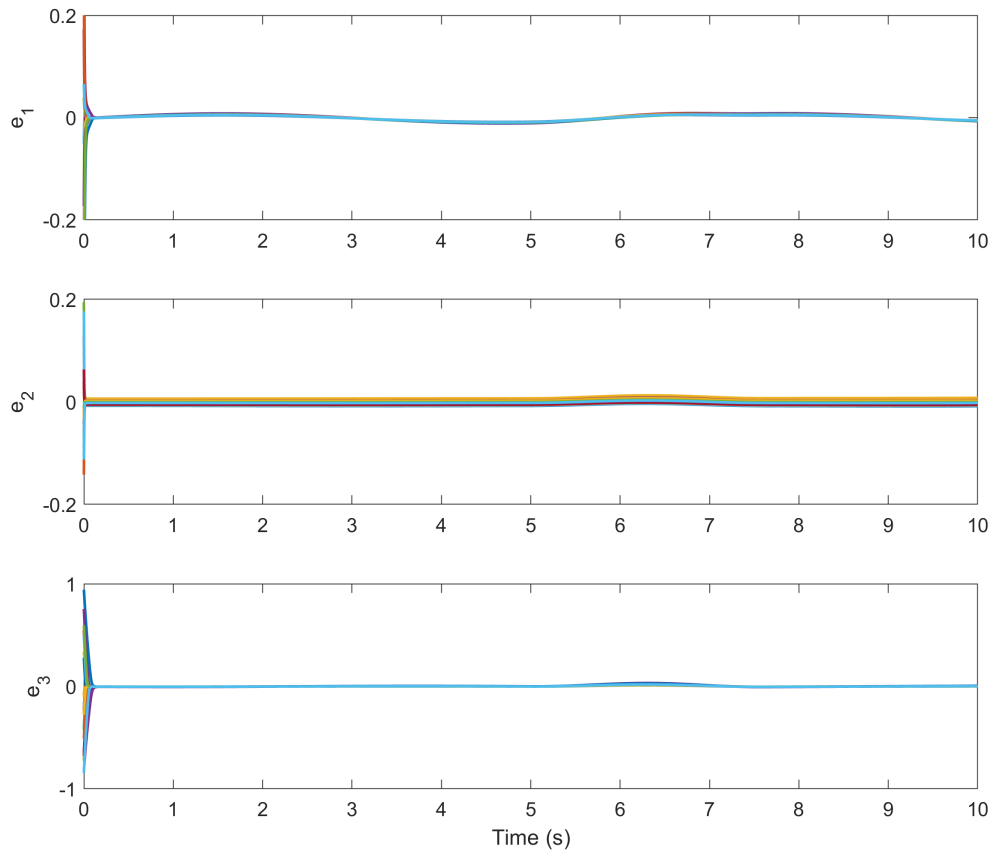


Figure 5.1: Closed-loop regulation of the steady state error for 20 runs

introduced improved performance for SJA-based flight control. Further, Figure 5.7 shows the significant improvement that is achieved in the disturbance rejection using the NN. Indeed, the MSE in the disturbed state e_3 is significantly reduced in using the NN over the adaptive control law.

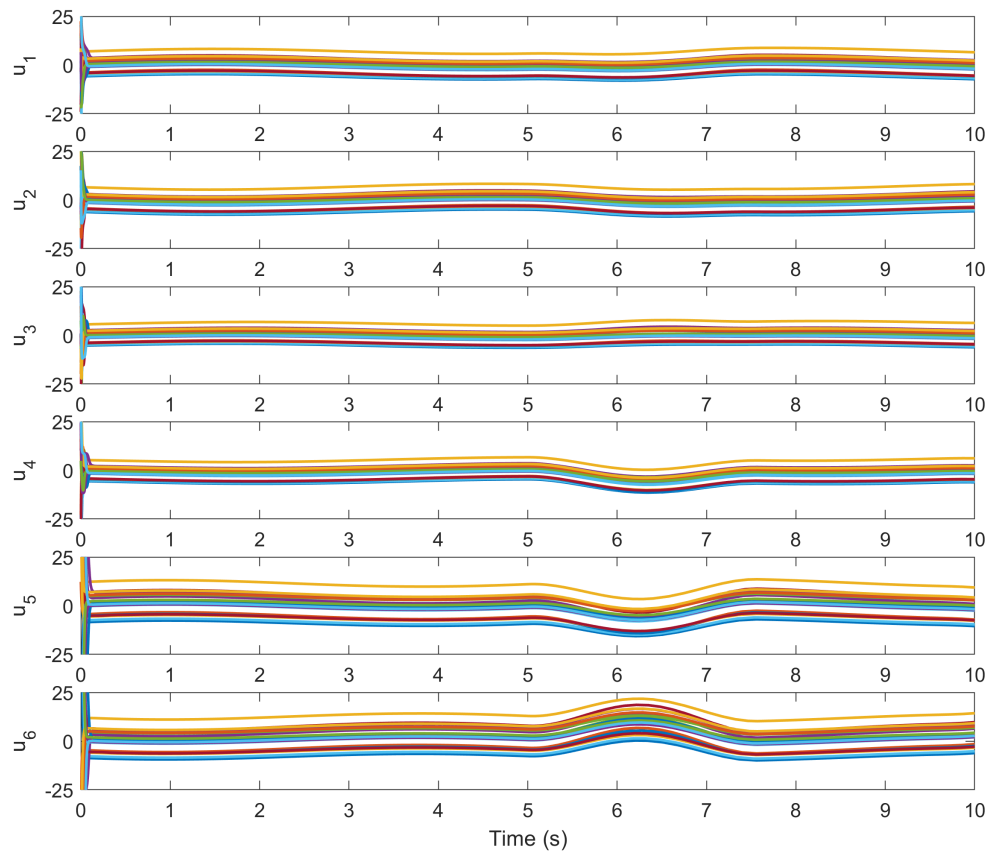


Figure 5.2: Closed-loop regulation of the virtual deflection angle for 20 runs

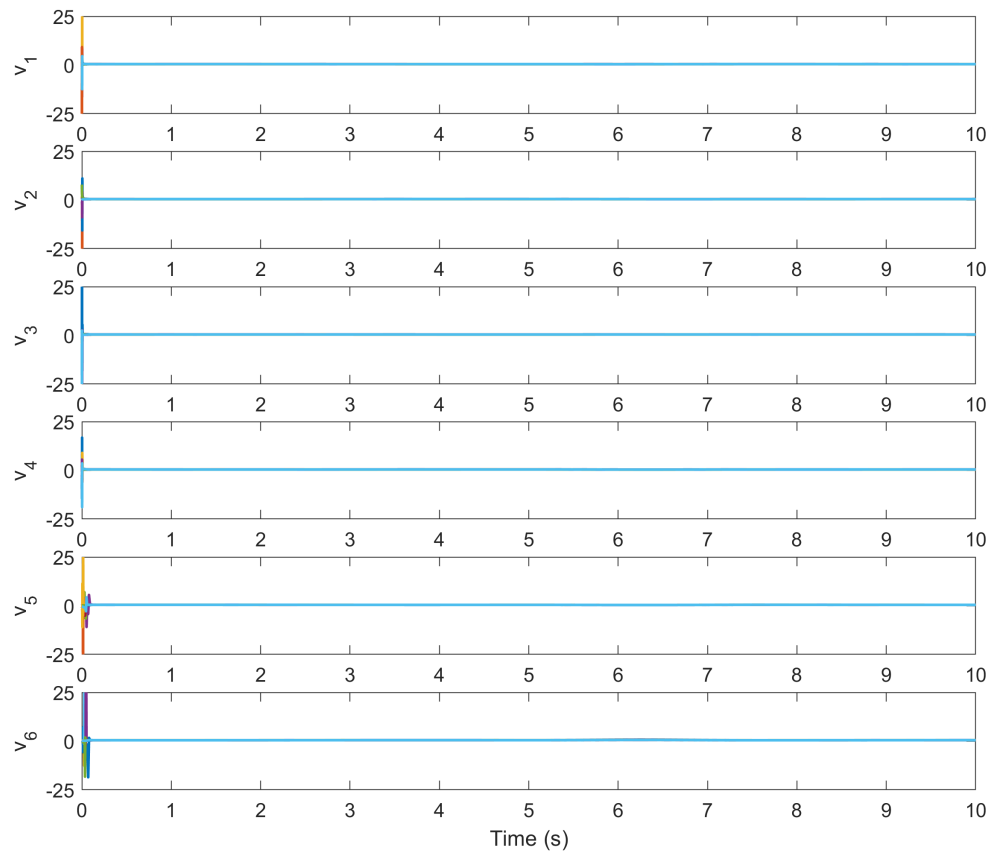
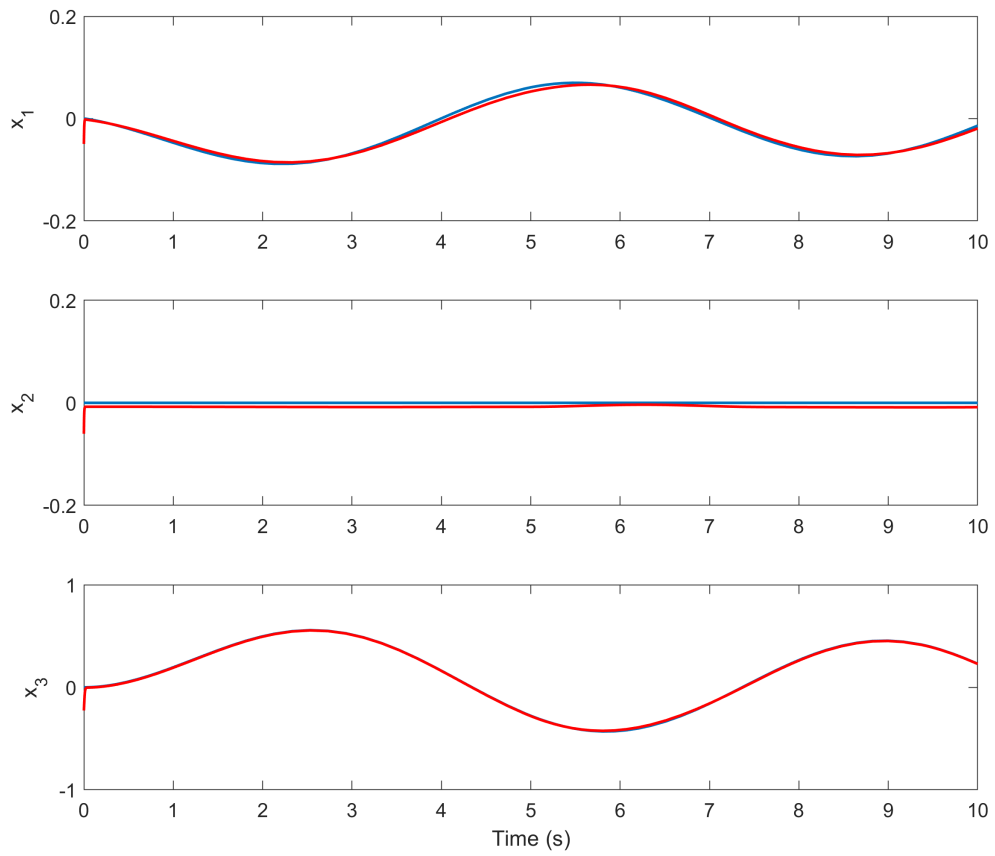


Figure 5.3: Control voltage signals commanded for the six SJA arrays



Model reference (blue) and actual state (red) during neural network based adaptive control

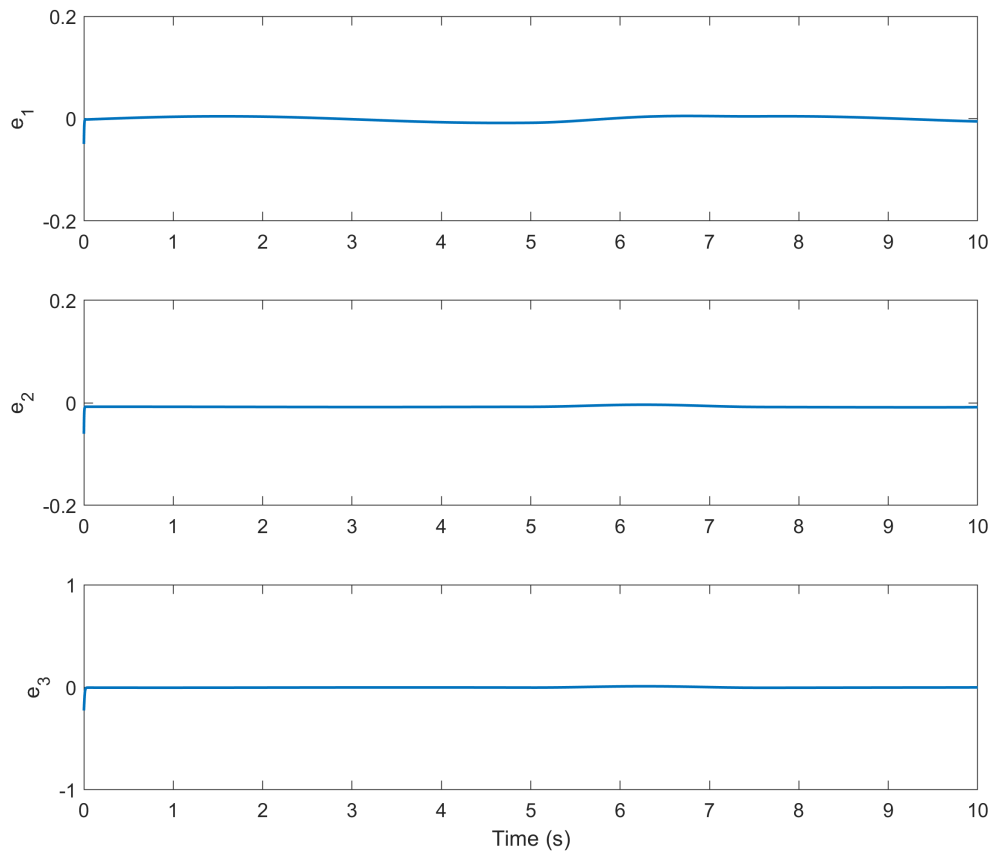
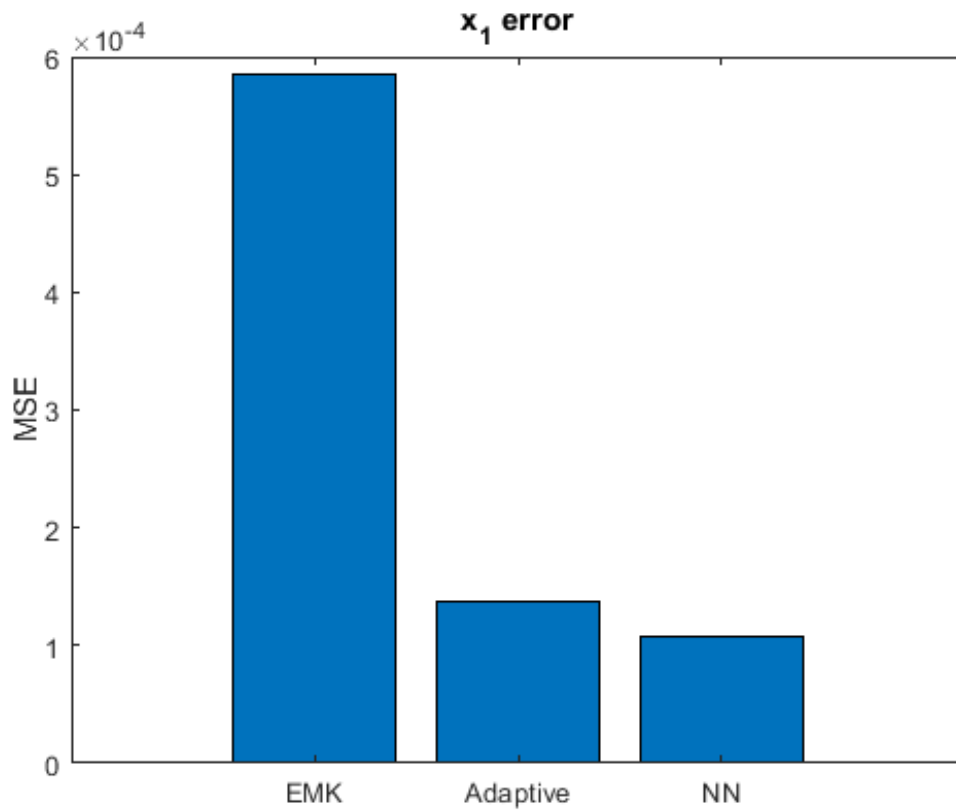


Figure 5.4: State error during neural network based adaptive control

Figure 5.5: Comparison of mean squared error for x_1

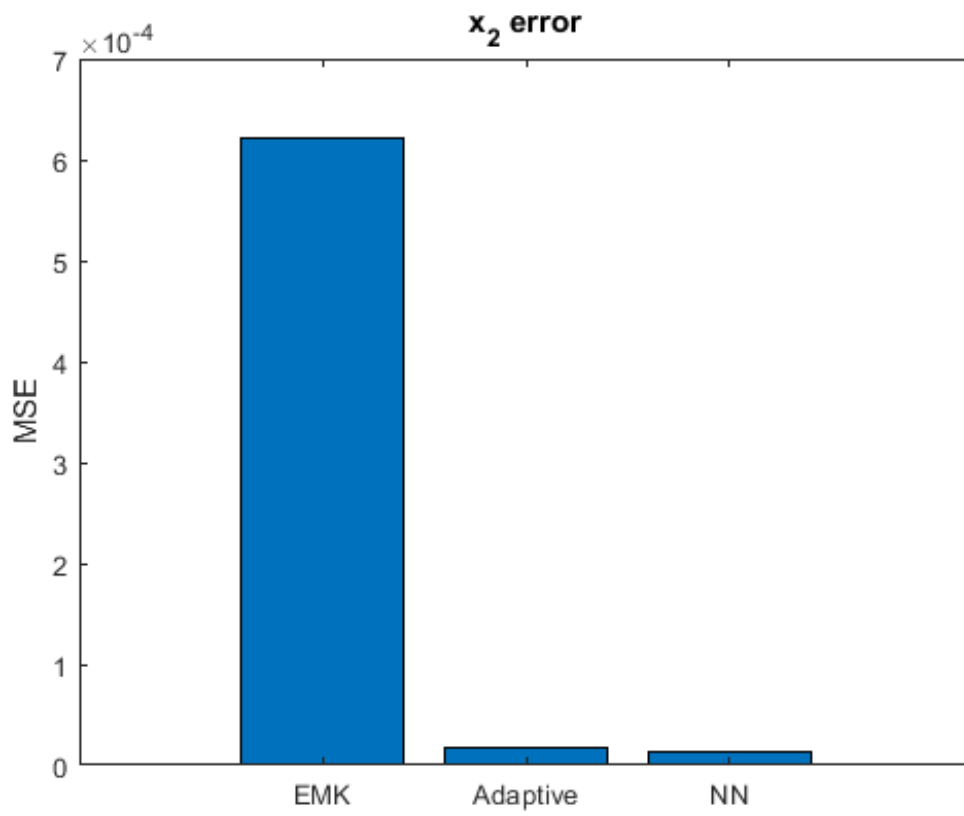
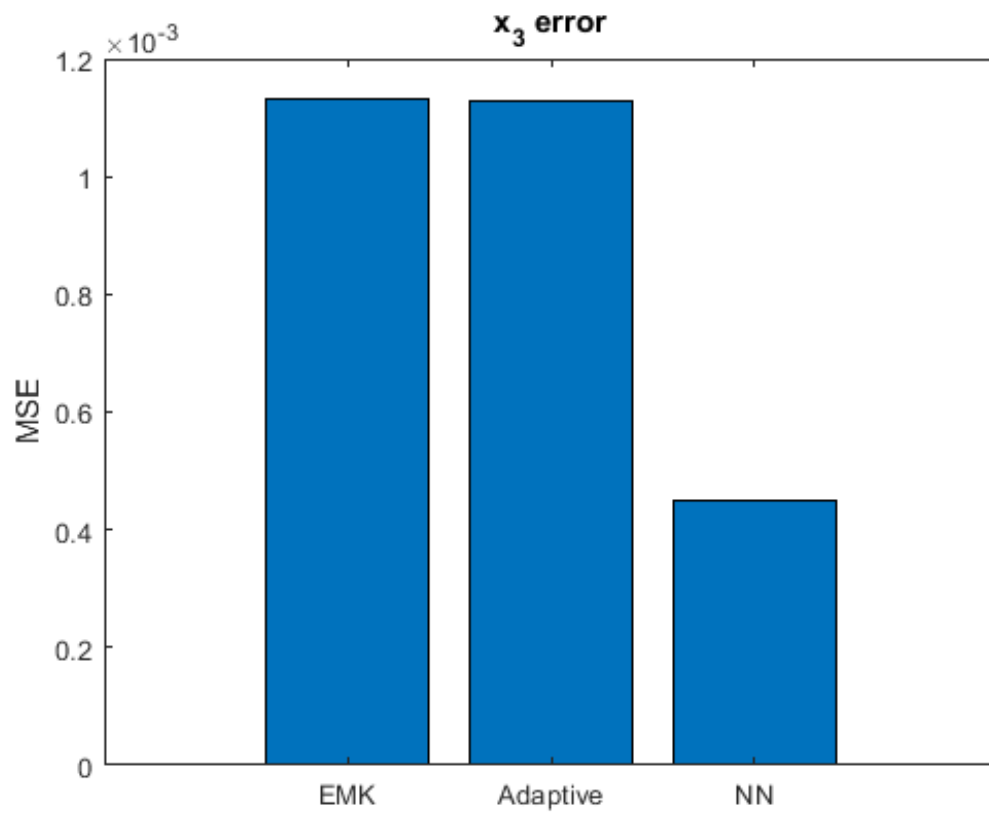


Figure 5.6: Comparison of mean squared error for x_2

Figure 5.7: Comparison of mean squared error for x_3

Chapter 6

Conclusion

This thesis developed adaptive and NN-based nonlinear SJA-based control methods to compensate for uncertain SJA actuator parameters and unmodeled nonlinear disturbances in the aircraft plant dynamics. The first result utilized a purely robust EMK design method in the controller development as a comparison and to motivate the adaptive and NN-based control designs.

The second approach was based on an adaptive control law to compensate for the input-multiplicative parametric uncertainty in the SJA model. The adaptive law was designed using a Lyapunov-based approach, and disturbance rejection was not formally addressed.

In the third approach, an NN-based adaptive controller was designed to formally compensate for both the uncertain SJA parameters and an unmodeled external nonlinear disturbance (e.g., a numerically simulated wind gust). For each of the three control laws, a Monte Carlo simulation was utilized to show the performance of the closed-loop system in the presence of pseudorandom uncertain parameters in the SJA model. The numerical simulation results clearly demonstrate that a significant reduction in the MSE is achieved using the NN-based control method over the adaptive and robust control methods.

Rigorous Lyapunov-based stability analyses were utilized to prove the theoretical results, and the numerical simulation results confirm the theoretical predictions. Future work will investigate additional intelligent control designs, such as recurrent

neural networks, and will also address experimental and high-fidelity numerical validation studies of the SJA model.

Bibliography

- [1] A. Shmilovich, Y. Y. and Whalen, E. (2016). Computational evaluation of flow control for enhanced control authority of a vertical tail. *AIAA Journal*, 54(8):2211–2220.
- [2] Cattafesta, L. N. and Sheplak, M. (2011). Actuators for active flow control. *Annual review of fluid mechanics*, 43(1):247–272.
- [3] DARPA (2020). Darpa awards contracts for new x-plane program based on active flow control.
- [4] Deb, D. and Tao, G. (2007). Adaptive compensation control of synthetic jet actuator arrays for airfoil virtual shaping. *Journal of Aircraft*, 44(2):616–626.
- [5] Glezer, A. and Amitay, M. (2002). Synthetic jets. *Annual review of fluid mechanics*, pages 503–529.
- [6] Golubev, V. and Mankbadi, R. (2012). High fidelity modeling of synthetic jet actuators for airfoil flow and noise control. *HEFAT*.
- [7] H. Tang, P. Salunkhe, Y. Z. J. D. Y. W. (2014). On the use of synthetic jet actuator arrays for active flow separation control. *Experimental Thermal and Fluid Science*, 57:1–10.
- [8] Jitendra Sharma, A. A. and Sur, M. (2000). Induction of visual orientation modules in auditory cortex. *Nature*, 404:841–847.

- [9] K. B. Kimdabi, N. Ramos-Pedroza, W. M. and Drakunov, S. V. (2016). Robust nonlinear estimation and control of fluid ow velocity fields. *IEEE Conference on Decision and Control*.
- [10] K. Hornik, M. S. and White, H. (1985). Multiplayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366.
- [11] K. Kidambi, W. MacKunis, S. D. and Golubev, V. (2020). Sliding mode estimation and closed-loop active ow control under actuator uncertainty. *International Journal of Robust and Nonlinear Control*, 30(16):6645–6660.
- [12] K. Kidambi, N. Ramos-Pedroza, W. M. and Drakunov, S. (2019). A closed-loop nonlinear control and sliding mode estimation strategy for fluid flow regulation. *International Journal of Robust and Nonlinear Control*, 29(3):779–792.
- [13] Khalil, H. K. (2002). *Nonlinear Systems 3rd edn*. Prentice Hall.
- [14] Lewis, F. L. (1999). Nonlinear network structures for feedback control. *Asian Journal of Control*, 1(4):205–228.
- [15] M. De Giorgi, C. De Luca, A. F. and Marra, F. (2015). Comparison between synthetic jets and continuous jets for active flow control: application on a naca 0015 and a compressor stator cascade. *Aerospace Science and Technology*, 43:256–280.
- [16] Marques, P. (2017). Aerodynamics of uav configurations. *Advanced UAV Aerodynamics Flight Stability and Control*.
- [17] N. Ramos-Pedroza, W. M. and Golubev, V. (2015). A new method of synthetic jet actuator-based lco suppression using an output feedback control strategy. *International Conference on Control, Automation and Systems (ICCAS)*, pages 1463–1468.
- [18] N. Ramos-Pedroza, W. M. and Golubev, V. (2017). A robust nonlinear output feedback control method for limit cycle oscillation suppression using synthetic jet actuators. *Aerospace Science and Technology*, 64:16–23.

- [19] N. Ramos-Pedroza, W. M. and Reyhanoglu, M. (2015a). Sliding mode control-based limit cycle oscillation suppression for uavs using synthetic jet actuators. *International Workshop on Recent Advances in Sliding Modes (RASM)*, pages 1–5.
- [20] N. Ramos-Pedroza, W. M. and Reyhanoglu, M. (2015b). A sliding mode lco regulation strategy for dual-parallel underactuated uav systems using synthetic jet actuators. *International Journal of Aerospace Engineering*.
- [21] N. Ramos-Pedroza, K. B. Kidambi, W. M. and Reyhanoglu, M. (2016). Nonlinear tracking control and structural vibration suppression for aircraft using synthetic jet actuators. *International Conference on Control, Automation, Robotics, and Vision (ICARCV)*.
- [22] R. Seele, P. T. and Wygnanski, I. J. (2009). Discrete sweeping jets as tools for improving the performance of the v-22. *Journal of Aircraft*, 46(6):2098–2160.
- [23] Ramos Pedroza, N., Mackunis, W., and Golubev, V. (2014). Robust nonlinear regulation of limit cycle oscillations in uavs using synthetic jet actuators. 3.
- [24] Sun, J. and Ioannou, P. A. (1996). *Robust Adaptive Control*. Prentice Hall.
- [25] Tomac, M. N. and Gregory, J. W. (2019). Phase-synchronized fluidic oscillator pair. *AIAA Journal*, 57(2):670–681.
- [26] V. Golubev, P. Kazarin, W. M. S. B. D. H. A. S. and Fraga, R. (2015). Analysis of safety implications for sjma-based robust uas flight control technology. *34th digital avionics systems conference*.
- [27] Venkateswaran Narayanaswamy, L. L. R. and Clemens, N. T. (2010). Characterization of a high-frequency pulsed-plasma jet actuator for supersonic flow control. *AIAA Journal*, 48(2):297–305.
- [28] W. MacKunis, S. Subramanian, S. M. J. W. C. M. R. and Ton, C. (2013). Robust nonlinear aircraft tracking control using synthetic jetactuators. *Ieee conference on decision and control*, pages 220–225.