AN ABSTRACT OF THE THESIS OF

David C. Collins for the degree of Master of Science in Electrical and Computer Engineering presented on February 17, 1993.

Title: Adaptive Model Reference Control of Highly Maneuverable High Performance Aircraft

Redacted for Privacy

Abstract approved:

Ron R. Mohler

This thesis presents an adaptive model reference controller for a highly maneuverable high performance aircraft, in particular, a modified F18. An adaptive controller is developed to maneuver an aircraft at a high angle of attack. Thus, the aircraft is required to fly over a highly nonlinear flight regime. The adaptive controller presented in this thesis can be viewed as a combination of a linear and a nonlinear controller. Around a fixed flight condition the adaptive controller converges to a linear controller; however, the controller remains a nonlinear controller during maneuvers.

The contributions of this thesis lie in two areas. The first area is in control. A successful application of linear adaptive control is presented for a highly nonlinear system. A new method is used to generate the reference trajectory. The reference model uses output feedback to improve the reference trajectory. It is shown that this improvement is necessary because of the control limitations. This work is also important to the control of highly maneuverable high performance aircraft. A successful adaptive controller has been developed to rapidly maneuver an aircraft to a high angle of attack. The main focus of this thesis is adaptive control.

ADAPTIVE MODEL REFERENCE CONTROL OF HIGHLY MANEUVERABLE HIGH PERFORMANCE AIRCRAFT

by

David C. Collins

A THESIS

submitted to

Oregon State University

in partial fulfillment of

the requirements for the

degree of

Master of Science

Completed February 17, 1993

Commencement June 1993

APPROVED:

Redacted for Privacy

Professor of Electrical and Computer Engineering in charge of Major

Redacted for Privacy

Head of Department of Electrical and Computer Engineering

Redacted for Privacy

Dean of Graduate School

Date thesis is presented February 17, 1993

Typed by David C. Collins for David C. Collins

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDIX FIGURES

ADAPTIVE MODEL REFERENCE CONTROL OF HIGHLY MANEUVERABLE HIGH PERFORMANCE AIRCRAFT

1 INTRODUCTION

This thesis describes the implementation of an adaptive controller for a high performance highly maneuverable aircraft. The objective is to design a controller that enables the aircraft to maneuver rapidly at and to high angles of attack. To accomplish this objective, an adaptive model reference controller is presented. The aircraft to be controlled is a modified F18 (modified to include thrust-vectoring). It has two controls -- thrust-vectoring and a stabilator, and it has two outputs for feedback -- angle of attack and pitch rate. The purpose of this thesis is to derive and analyze the controller, and to validate its performance. In so doing, several items are discussed. These include, the choice of the prediction model, the choice of the adaptation for the prediction model, the choice of the reference model, and the choice of the control calculation.

Several important contributions are made by this thesis. First, a successful design of a linear adaptive controller is applied to a highly nonlinear system. It is demonstrated that the use of feedback in the reference model can account for control limitations. Second, an adaptive controller is developed to rapidly maneuver an aircraft to a high angle of attack. For this thesis the aircraft and the differential equations used to simulate the aircraft are used interchangeably (i.e. the simulated aircraft is assumed to be the physical aircraft).

The presentation is organized in the following way. Chapter 2 gives a brief overview of adaptive control. Chapter 3 briefly describes the simulation of the aircraft. Chapter 4 describes the choice of the prediction model. Chapter 5 describes the adaption used to update the prediction model presented in chapter 4. Chapter 6 describes the control calculation. The reference model is presented in chapter 7. Chapter 8 presents the performance of the complete controller, and chapter 9 concludes this thesis and presents ideas for continued research.

2 INTRODUCTION TO ADAPTIVE MODEL REFERENCE CONTROL

The purpose of adaptive control is to provide a mechanism to account for changes in the system that is to be controlled. For the case at hand, the goal is to use concepts from linear theory to control an aircraft over a highly nonlinear flight regime. Historically, adaptive model reference control has been described in the continuous time domain. Only a discrete form is considered here. A more comprehensive overview is given in [1]-[3]; while a complete development of adaptive control is given in [4] and [5]. Some numerical aspects of implementing adaptive control are developed in [6]-[8]. Adaptive control for a small class of nonlinear and time-varying systems is investigated in [8]-[16].

The heuristic idea behind adaptive model reference control is relatively simple. An adaptive portion attempts to identify (in some respect) the system. A model system generates a desired reference trajectory. Then, a controller uses this information to calculate a command signal such that the output of the system follows the reference trajectory. A block diagram of the model reference adaptive controller presented in this thesis is displayed below. This is the first time, to my knowledge, that feedback has been used in the reference trajectory. The reasons for using feedback in the reference trajectory will be discussed in chapter 7.

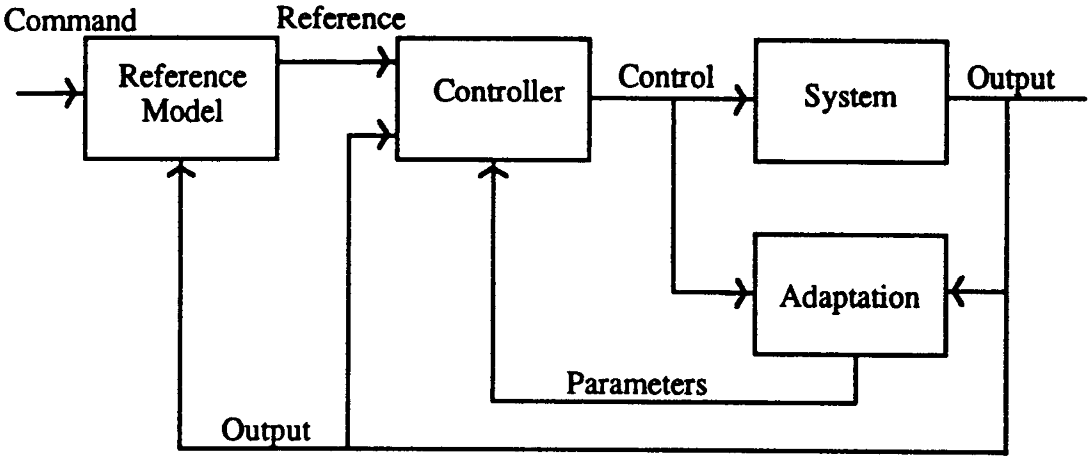

FIGURE 1 Block Diagram of Adaptive Control

Figure (1) for the particular problem presented in this thesis is explained below. The objective is have the output of the aircraft (system) follow the pilot's input (command). The pilot inputs a command to the reference model. The reference model examines the command and the aircraft position and then determines where the aircraft should move next (the reference trajectory). The controller examines the reference trajectory, the aircraft position and the aircraft model, and then determines the thrust-vectoring and stabilator input (control) that the aircraft (system) needs. The adaptation examines the control and the output, and then develops or modifies a model to approximate the local behavior of the aircraft.

Two important elements have to be developed for an effective adaptation routine. First, a class of prediction models needs to be selected. A prediction model represents the dynamics of the system, and it has parameters that are modified by an estimator. The estimator is the second part of the adaptation. It estimates the values of the parameters to improve the prediction model.

The simplest prediction models to consider are models that are linear in parameters. Models that are linear in parameters can be represented by equation (1). The f's are functions of the input and output measurements from time (t-1) and before, and the a's are parameters. Y(t) is the predicted output of the system based on past measurements. $\phi(t)$ is the regressor vector, and θ is the parameter vector to be estimated.

$$\begin{gathered} y(t)=\theta^T\phi(t) \\ \theta^T=[a_1\ a_2\ a_3\ \ldots\ a_n] \\ \phi(t)^T=[f_1\ f_2\ f_3\ \ldots\ f_n] \end{gathered} \tag{1}$$

This does not limit the model to be linear (with respect to the measurements) since elements of $\phi(t)$ could be nonlinear functions of the measurements. However, only prediction models that are linear with respect to both the measurements and the parameters will be developed in this thesis.

The most common estimation algorithm for models that are linear in parameters is the recursive-least-squares algorithm. The idea is to chose θ such that the squared difference between the prediction model and the actual system is minimized. The purpose of making the algorithm recursive is to allow for on-line identification of parameters.

The reference model is an intermediate step that allows the system to follow the command signal while meeting a variety of design criteria (for instance: rise time, overshoot, settling time, etc.). The control is calculated such that the system follows the reference trajectory, and such that the control signal remains within its constraints. Each block of the adaptive controller is described in detail in the chapters that follow. Most of the discussion focuses on the adaptation. If the prediction model or the parameter estimation is inadequate, then the controller will be unable to generate a meaningful control signal.

3 AIRCRAFT SIMULATION

3.1 AIRCRAFT DESCRIPTION

The aircraft that was controlled in this report is a modified F18 (HARV). Only the longitudinal portion of the aircraft was simulated. Seven ordinary differential equations were used to model the aircraft and the actuator dynamics. These were derived from standard rigid-body, longitudinal aircraft equations of motion. The particular form used here was developed by Cho, [17]. These differential equations contain coefficients that were determined from wind tunnel tests and actual flight tests. Thus, the simulated model was fairly accurate when compared to the performance of the actual aircraft.

All simulations were assumed to occur at a constant elevation of 15000 feet. The aircraft was controlled by feedback of the angle of attack and the pitch rate, and the controller manipulated the stabilator angle, and the thrust-vector angle (the angle of the thrust deflection from the body axis). The thrust magnitude command was set in an open loop fashion. Thus, the controller did not manipulate the thrust magnitude. The control, output, and parameter estimates were updated at a sampling rate of 20Hz.

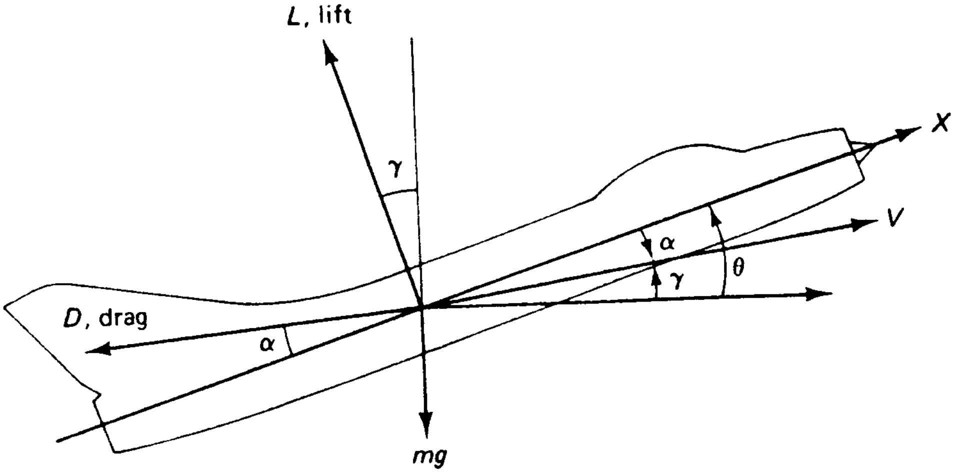

FIGURE 2 Longitudinal Airframe Geometry

3.2 LONGITUDINAL AIRCRAFT MODEL

Four states are used to represent the aircraft. They are angle of attack (α), pitch rate (q), pitch angle (θ), and velocity (v). The angle of attack is the angle between the body axis and the velocity vector. The pitch angle is the angle between the body axis and the ground (ground is assumed perpendicular to gravity). The pitch rate is the rate at which the pitch changes, and the velocity is the speed of the aircraft with respect to the air. The differential equations for the aircraft are given below.

$$\dot{\alpha}=\alpha_{11}\left[\left(1-\frac{\rho Sc}{4m}C_{L_q}\right)q-\frac{\rho S}{2m}C_{L_o}V-\frac{T\sin(\delta_v-\alpha)}{mV}+\frac{g\cos(\theta-\alpha)}{V}\right] \tag{2}$$

$$\dot{V}=-g\sin(\theta-\alpha)-\frac{\rho S}{2m}C_{D_o}V^2+\frac{T}{m}\cos(\delta_v-\alpha) \tag{3}$$

$$\begin{aligned}\dot{q}=&\frac{\rho S}{2I_{yy}}\left(cC_{M_o}+q_{11}C_{D_o}+q_{12}C_{L_o}\right)V^2+\alpha_{11}\left(1-\frac{\rho Sc}{4m}C_{L_q}\right)q\\&+\alpha_{11}\frac{\rho Sc}{4I_{yy}}\left(cC_{m_{\dot{\alpha}}}+q_{12}C_{L_{\dot{\alpha}}}\right)V\left(-\frac{\rho S}{2m}C_{L_o}V-\frac{T\sin(\delta_v-\alpha)}{mV}+\frac{g\cos(\theta-\alpha)}{V}\right)\\&+\frac{\rho Sc}{4I_{yy}}\left(cC_{m_{\dot{\alpha}}}+q_{12}C_{L_{\dot{\alpha}}}+cC_{m_q}+q_{12}C_{L_q}\right)Vq+\frac{p_{zo}}{I_{yy}}T\cos(\delta_v)-\frac{p_{xo}}{I_{yy}}T\sin(\delta_v)\end{aligned} \tag{4}$$

$$\dot{\theta}=q \tag{5}$$

where

$$\alpha_{11}=\frac{1}{\left(1+\frac{\rho Sc}{4m}C_{L_{\dot{\alpha}}}\right)} \tag{6}$$

$$q_{11}=p_{xe}\sin(\alpha)-p_{ze}\cos(\alpha) \tag{7}$$

$$q_{12}=p_{ze}\sin(\alpha)+p_{xe}\cos(\alpha) \tag{8}$$

(p_{xe},p_{ze}) is the position vector from the engine thrust to the center of mass.

I_{yy} is the moment of inertia.

ρ is the air density.

g is the gravitational acceleration.

m is the mass.

S is the effective area.

c is a moment arm.

$$\begin{aligned}
C_{L_o}&=f_1(\alpha,V,\delta_h)\\
C_{L_q}&=f_2(\alpha)\\
C_{L_{\dot{\alpha}}}&=f_3(\alpha)\\
C_{M_o}&=f_4(\alpha,V,\delta_h)\\
C_{M_q}&=f_5(\alpha)\\
C_{M_{\dot{\alpha}}}&=f_6(\alpha)\\
C_{D_o}&=f_7(\alpha,\delta_h)
\end{aligned} \tag{9}$$

The f's are found using an interpolation of flight test data.

3.3 ACTUATOR DYNAMICS

The input dynamics are described by three states--thrust magnitude (T), thrust vectoring angle (δ_v), and stabilator angle (δ_h). The stabilator and the thrust vectoring dynamics include a velocity limit of 40 degrees per second for the stabilator angle, and 80 degrees per second for the thrust vectoring angle. The differential equations are stated below.

Stabilator Angle Dynamics

$$\dot{\delta}_h = \begin{pmatrix} -40 & (\delta_{h_{cmd}} - \delta_h) < \frac{-4}{3} \\ 30(\delta_{h_{cmd}} - \delta_h) & \frac{-4}{3} \leq (\delta_{h_{cmd}} - \delta_h) \leq \frac{4}{3} \\ 40 & (\delta_{h_{cmd}} - \delta_h) > \frac{4}{3} \end{pmatrix} \qquad (10)$$

The magnitude of the stabilator angle is limited according to

$$-24.5° \leq \delta_h \leq 10.5° \qquad (11)$$

Thrust Vectoring Dynamics

$$\dot{\delta}_v = \begin{pmatrix} -80 & (\delta_{v_{cmd}} - \delta_v) < \frac{-8}{3} \\ 30(\delta_{v_{cmd}} - \delta_v) & \frac{-8}{3} \leq (\delta_{v_{cmd}} - \delta_v) \leq \frac{8}{3} \\ 80 & (\delta_{v_{cmd}} - \delta_v) > \frac{8}{3} \end{pmatrix} \qquad (12)$$

The magnitude of the thrust vectoring angle is limited according to

$$-20° \leq \delta_v \leq 20° \qquad (13)$$

Thrust Magnitude Dynamics

$$\dot{T} = (T_{cmd} - T) \tag{14}$$

The magnitude of thrust is limited according to

$$0 \le T \le 18000 \ lbs \tag{15}$$

3.4 SIMULATION

The equations were simulated using a fixed-step fourth order Runge Kutta method ([18]) with an integration time step of .01 second. A comparison was made between an integration time step of .01 and .001, and no noticeable difference was detected. Thus, numerical instabilities caused by the integration time step or the integration routines were not encountered in the results presented in the following chapters. The computer code was written in c.

4 PREDICTION MODEL

4.1 CLASS OF MODELS

Several different approaches exist to formulate a prediction model for a nonlinear system. The first method is to simply assume that the model can be adequately described by a linear time-varying system. A second method is to assume that the prediction model is a linear time-varying system with a time-varying offset. A third method is to assume a nonlinear form of the prediction model with constant or time-varying parameters. Each of these methods is described below.

A linear time-varying system can be described by the equation below,

$$A(z^{-1},t)y(t)=B(z^{-1},t)u(t) \tag{16}$$

where A and B are time varying polynomials of z^{-1}. U(t) is the input sequence. Y(t) is the output sequence, and z^{-1} is a backwards shift operator. $A(z^{-1},t)$, without loss of generality, is assumed to be monic. Thus, $A(z^{-1},t)$ could be described by

$$A(z^{-1},t)=1+a_1(t)z^{-1}+a_2(t)z^{-2}+a_3(t)z^{-3}\ \ldots\ a_n(t)z^{-n} \tag{17}$$

This leads to a simple prediction model with the following form:

$$\begin{gathered} y(t)=\phi(t)^T\theta(t) \\ \phi(t)^T=[y(t-1),y(t-2),\ldots,u(t-1),u(t-2),\ldots] \\ \theta(t)^T=[-a_1(t),-a_2(t),\ldots,b_1(t),b_2(t),\ldots] \end{gathered} \tag{18}$$

A linear time-varying system with a time varying offset can be described by the equation below.

$$A(z^{-1},t)y(t)=B(z^{-1},t)u(t)+d(t) \tag{19}$$

Several different predictor models exist for this type of model. The simplest approach is to include a one in the regressor. Thus, d(t) is calculated directly by the estimator. This prediction model is given in the equation below.

$$\begin{aligned} &y(t)=\phi(t)^T\theta(t) \\ &\phi(t)^T=[y(t-1),y(t-2),\ldots,u(t-1),u(t-2),\ldots 1] \\ &\theta(t)^T=[-a_1(t),-a_2(t),\ldots,b_1(t),b_2(t),\ldots d(t)] \end{aligned} \tag{20}$$

Another approach is to calculate the offset separately, and then use it in the prediction model. One such approach is described by Sripada et al. [7]. The idea is to calculate the offset of the elements of ϕ, and then, subtract off the offset. This prediction model is described by the equation below.

$$\begin{aligned} &y'(t)=\phi(t)^T\theta(t) \\ &\phi(t)^T=[y'(t-1),y'(t-2),\ldots,u'(t-1),u'(t-2),\ldots] \\ &\theta(t)^T=[-a_1(t),-a_2(t),\ldots,b_1(t),b_2(t),\ldots] \\ &y'(t)=y(t)-\bar{y}(t) \\ &\bar{y}(t)=estimated\ average\ of\ y(t) \\ &u'(t)=u(t)-\bar{u}(t) \\ &\bar{u}(t)=estimated\ average\ of\ u(t) \end{aligned} \tag{21}$$

Another prediction model can be formed by using differences of the input and of the output measurements, as opposed to the measurements themselves. This method was presented by Clarke et al. [19], and it eliminates the need to calculate the offset. The idea was generalized by Tuffs et al. [20]. At steady state, the elements of ϕ tend towards zero, and no offset exists. The predictor model can be represented as follows:

$$
\begin{aligned}
&\Delta y(t)=\phi(t)^T\theta(t)\\
&\phi(t)^T=[\Delta y(t-1),\Delta y(t-2),\ldots,\Delta u(t-1),\Delta u(t-2),\ldots]\\
&\theta(t)^T=[-a_1(t),-a_2(t),\ldots,b_1(t),b_2(t),\ldots] \qquad (22)\\
&\quad \Delta=1-z^{-1}\\
&y(t)=\Delta y(t)+y(t-1)
\end{aligned}
$$

The final method for forming a prediction model is to use a nonlinear prediction model. The details of developing a nonlinear prediction model will not be discussed here, but several techniques for system identification of nonlinear systems can be found in the literature. Mohler et al. [17] developed a nonlinear prediction model for this modified F18, and a comparison with this prediction model will be presented. The nonlinear prediction model will be described in section 4.3.

4.2 DATA DESCRIPTION

As can be seen from chapter 3, there are two main sources of nonlinearities. The first source is from the aircraft model itself. The second source of nonlinearities arises from the constraints on the admissible control. These constraints include velocity limits and magnitude limits. For simplicity, the dynamics on the stabilator angle and the thrust vectoring angle are ignored initially (The aircraft is simulated without these dynamics). They are included for the final prediction model presented at the end of this chapter.

Three sets of data were used to develop a prediction model. The first set included the aircraft and only the stabilator control. The thrust-vectoring angle was fixed at zero, and the stabilator angle was set to the command signal. This allowed a comparison to be made with the nonlinear prediction model that was presented in [17]. The second set of data added the thrust-vectoring control, but the control dynamics were still ignored. This displayed the effects of the nonlinearities of the aircraft on the prediction model without the control nonlinearities. The final data set included all of the nonlinearities. This data set was used to test the final form of the predictor.

Five states were used to simulate the aircraft for the first two data sets. They were angle of attack, pitch rate, pitch, velocity, and thrust magnitude. The stabilator angle and the thrust vectoring angle were set to the command signal. Thus, they were constant between each sampling instant, and the dynamics in the control were ignored. The control values were calculated using a preliminary form of the adaptive controller developed in this thesis.

For the last two data sets the change in the control command was limited to 40 degrees per second for the stabilator angle and 80 degrees per second for the thrust vectoring angle. This allowed the results to be compared more easily. A description of the data is given by the plots below. Only the angle of attack and the input signals are plotted. The other states are similar in character to the plots of the complete controller (see chapter 8).

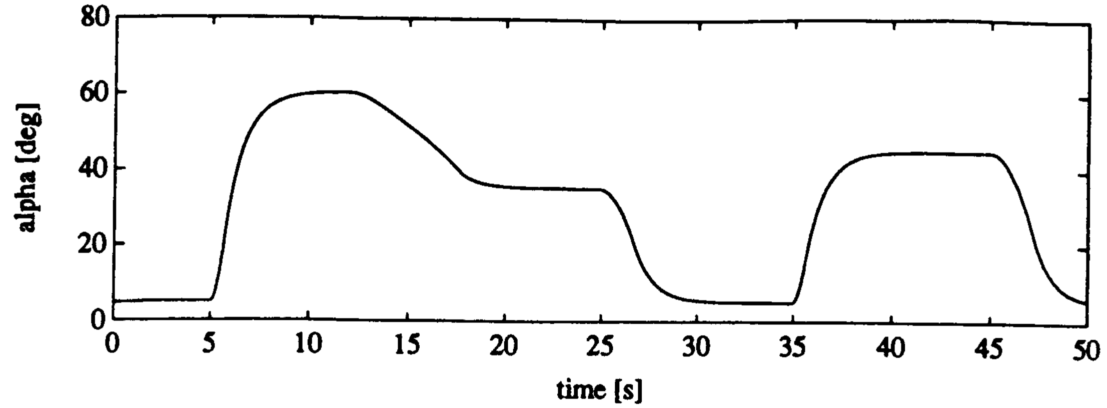

FIGURE 3 Angle of Attack for Data Set 1

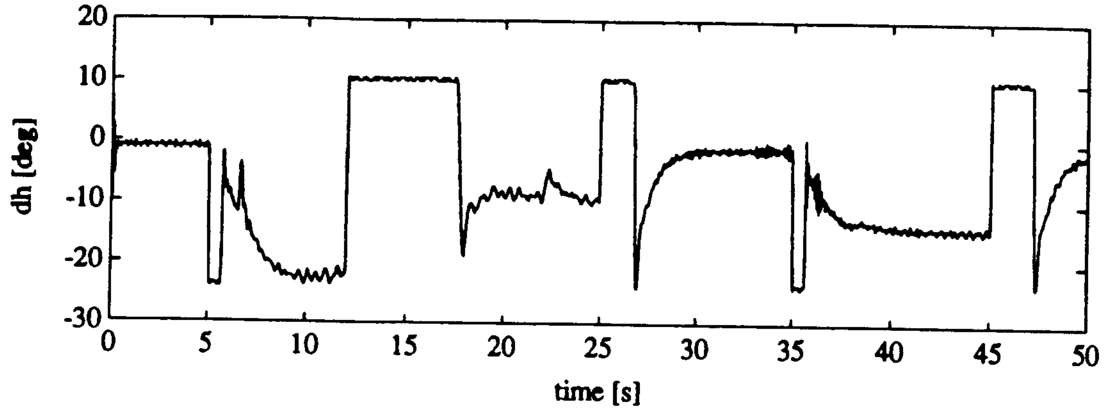

FIGURE 4 Stabilator Command for Data Set 1

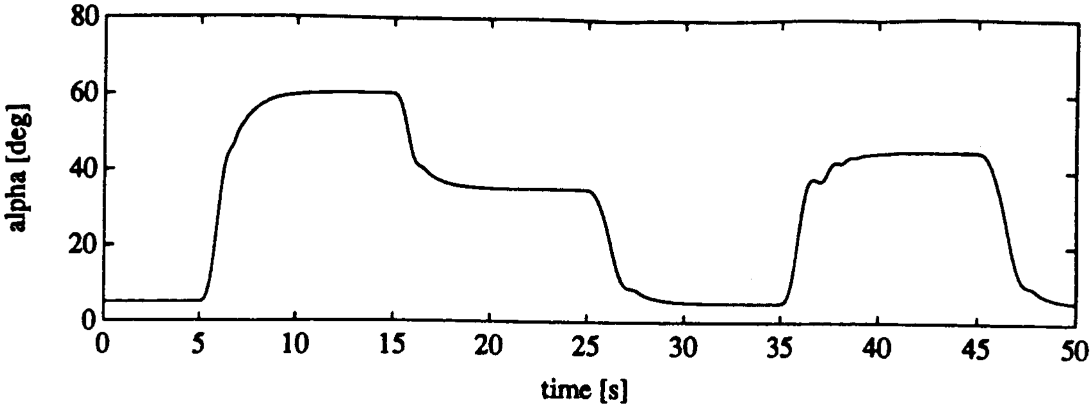

FIGURE 5 Angle of Attack for Data Set 2

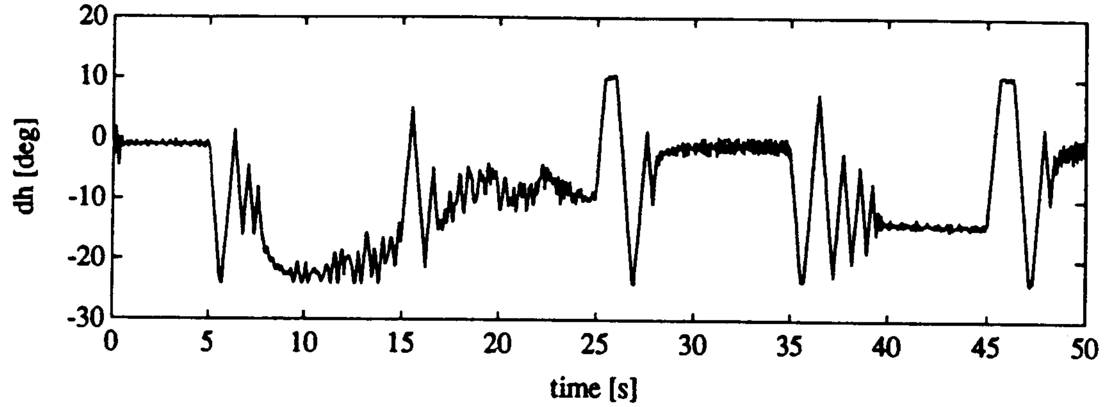

FIGURE 6 Stabilator Command for Data Set 2

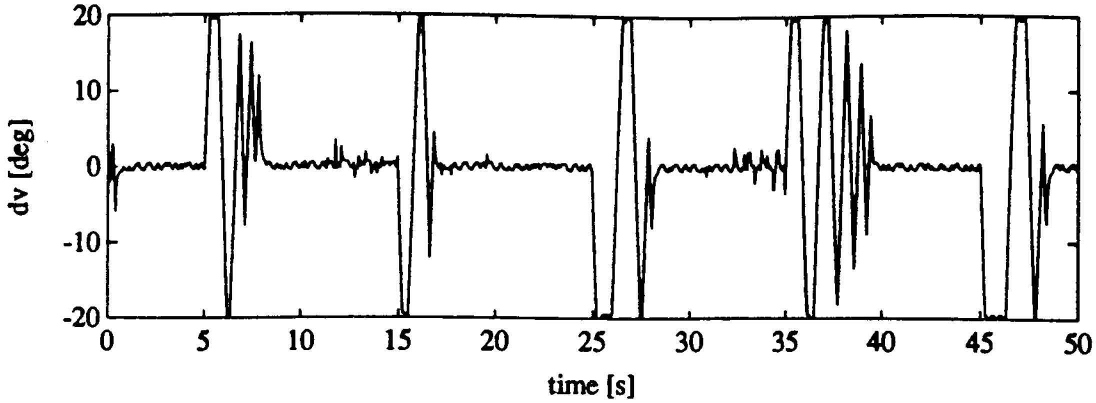

FIGURE 7 Thrust Vectoring Command for Data Set 2

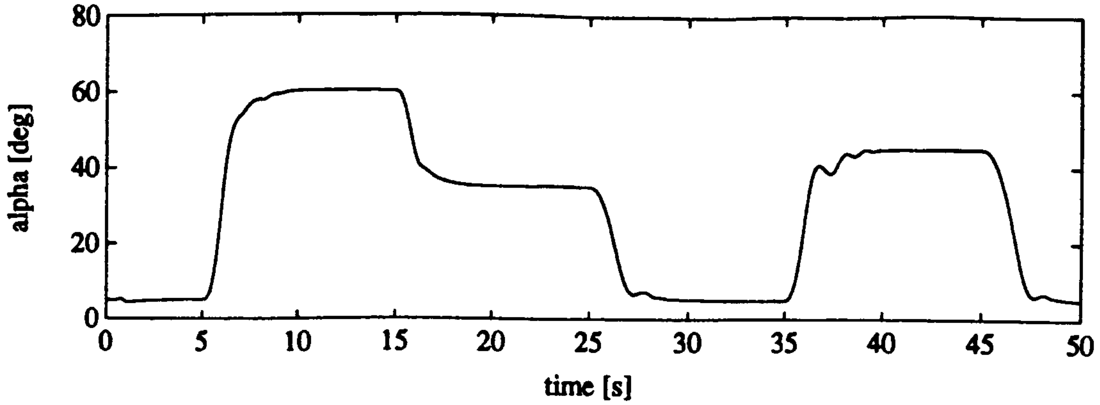

FIGURE 8 Angle of Attack for Data Set 3

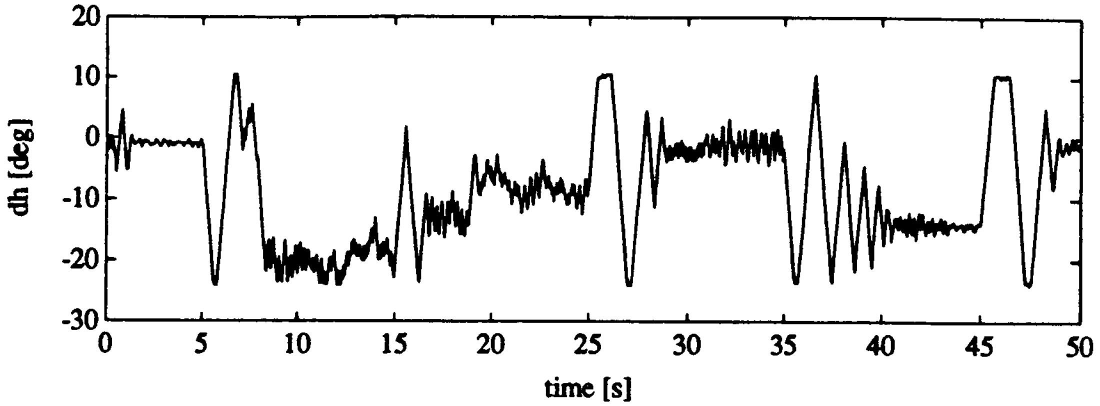

FIGURE 9 Stabilator Command for Data Set 3

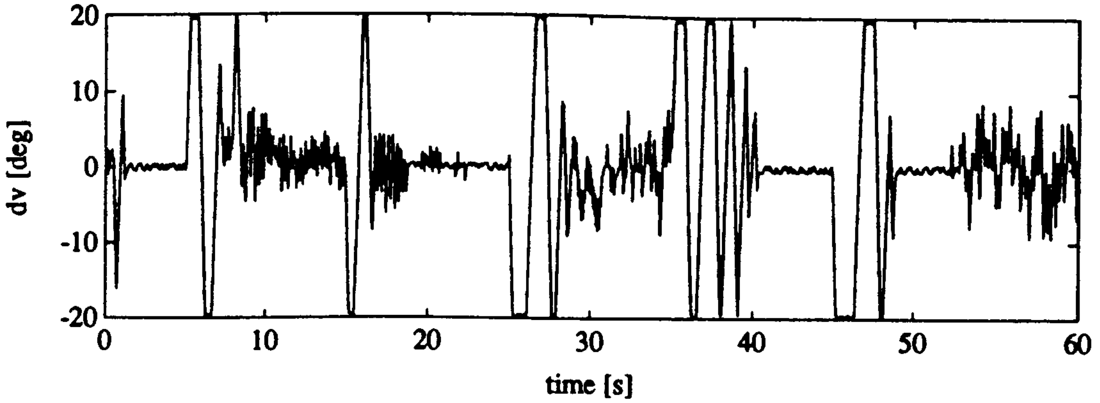

FIGURE 10 Thrust Vectoring Command for Data Set 3

4.3 PREDICTION MODELS FOR DATA SET ONE

All of the different methods described in section 4.1 will be compared on data set one. From this analysis, only the best predictors will be considered on data sets two and three. The recursive-least-squares estimator with a variable forgetting factor to maintain a constant information measure was used for all the comparison in this chapter. The recursive-least-squares algorithm is discussed in detail in chapter 5.

The first predictor model corresponds to the predictor in equation (18). This is the simplest predictor, and it can be seen that it is also the least effective. The prediction model is described below.

$$\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)$$
$$\phi(t)^T=[\alpha(t-1),q(t-1),\delta_h(t-1)] \qquad (23)$$

The prediction error (the difference between the predicted output and the actual output) for this regressor is displayed in the figure below.

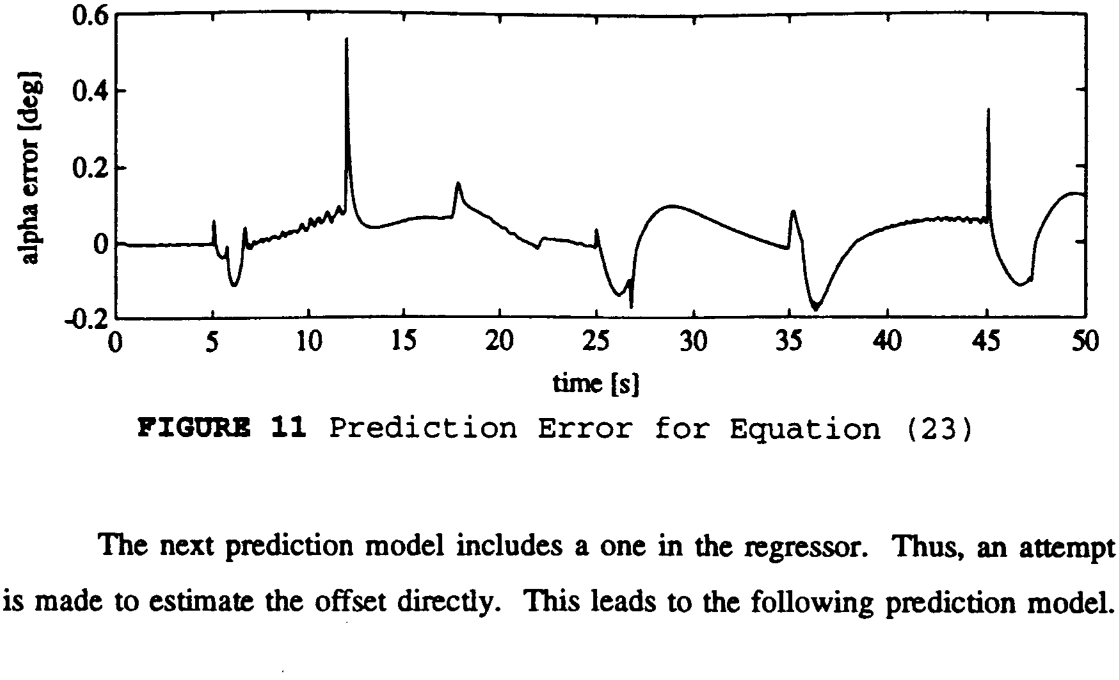

FIGURE 11 Prediction Error for Equation (23)

The next prediction model includes a one in the regressor. Thus, an attempt is made to estimate the offset directly. This leads to the following prediction model.

$$\begin{aligned}\hat{\alpha}(t)&=\phi(t)^{T}\hat{\theta}(t-1)\\ \phi(t)^{T}&=[\alpha(t-1),q(t-1),\delta_{h}(t-1),1]\end{aligned}\tag{24}$$

The prediction error improved only slightly with this addition. One explanation for this limited improvement is that the variations in the offset were too rapid. The prediction error for a one in the regressor is displayed below.

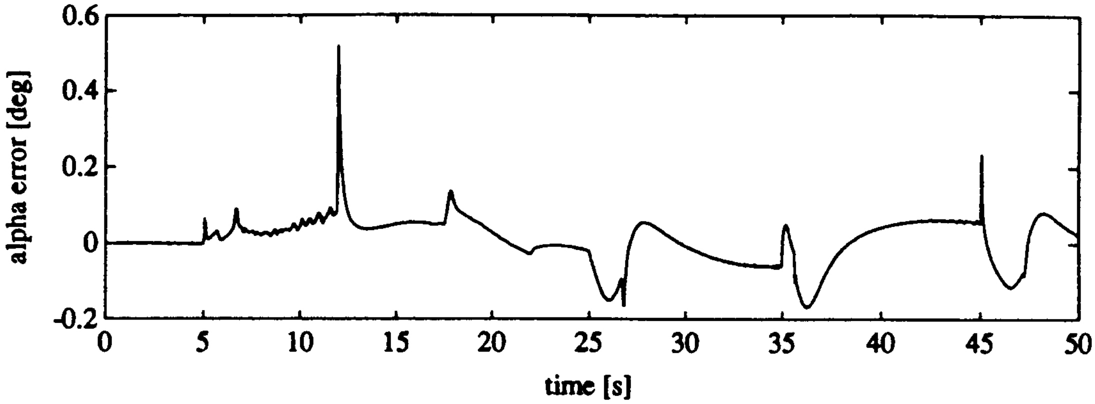

FIGURE 12 Prediction Error for Equation (24)

The difference predictor model described by equation (22) eliminates the need to estimate the offset. The predictor model is given by the equation below.

$$\begin{aligned}\Delta\hat{\alpha}(t)&=\phi(t)^{T}\hat{\theta}(t-1)\\ \phi(t)^{T}&=[\Delta\alpha(t-1),\Delta q(t-1),\delta_{h}(t-1)]\end{aligned}\tag{25}$$

The prediction error for this predictor model is given in figure (13). This predictor model offers a tremendous amount of improvement over the first two predictor models.

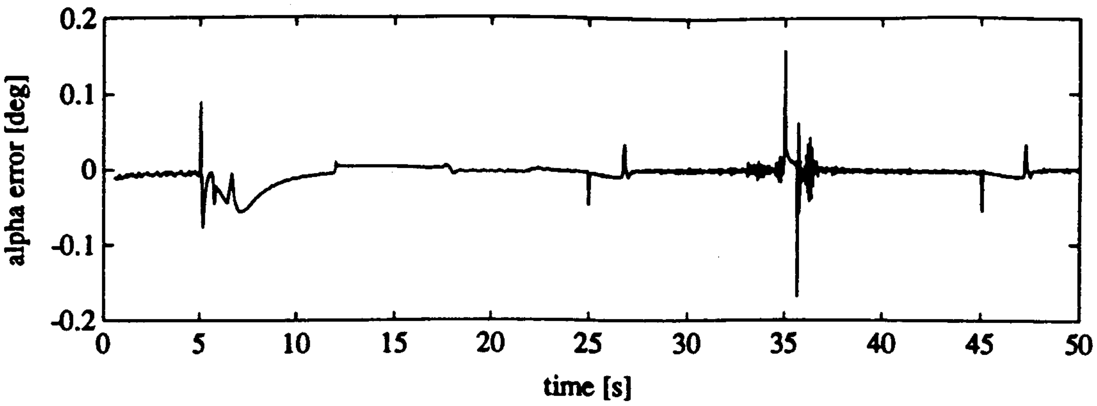

FIGURE 13 Prediction Error for Equation (25)

A fourth approach as described by equation (21), is to attempt to estimate the offset separately and then use this information in the prediction model. This leads to a prediction model of the following form:

$$\begin{aligned}&\hat{\alpha}'(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\alpha'(t-1),q'(t-1),\delta_h(t-1)]\\&\alpha'(t)=\alpha(t)-\bar{\alpha}(t)\\&q'(t)=q(t)-\bar{q}(t)\end{aligned}\qquad(26)$$

The average values of q and α are estimated as suggest by Sripada et al. [7]. Thus, the averages of q and α are estimated as follows:

$$\begin{aligned}&\bar{\alpha}(t)=\lambda_\alpha(t)\bar{\alpha}(t-1)+(1-\lambda_\alpha(t))\alpha(t-1)\\&\bar{q}(t)=\lambda_q(t)\bar{q}(t-1)+(1-\lambda_q(t))q(t-1)\\&\lambda_\alpha(t)=\lambda_q(t)=.95\end{aligned}\qquad(27)$$

The prediction error for the above predictor model is given in figure (14). It should be noted that the prediction error is better than the prediction error for a one in the regressor, but it is worse than the difference prediction error. This should come as no surprise since this method is really a combination of the two methods. To see this, let $\lambda_q = \lambda_\alpha = 1$. This leads to the following:

$$\begin{aligned}
&\bar{\alpha}(t)=\bar{\alpha}(t-1)=c_1\\
&\bar{q}(t)=\bar{q}(t-1)=c_2\\
&\alpha'(t)=\alpha(t)-c_1\\
&q'(t)=q(t)-c_2\\
&\hat{\alpha}(t)=\theta_1\alpha(t-1)+\theta_2 q(t-1)+\theta_3\delta_h+c\\
&\qquad c=c_1-\theta_1 c_1-\theta_2 c_2
\end{aligned} \tag{28}$$

This predictor model is a special case of the predictor model with a one in the regressor. Now, let $\lambda_q = \lambda_\alpha = 0$. This leads to the following:

$$\begin{aligned}
&\bar{\alpha}(t)=-\alpha(t-1)\\
&\bar{q}(t)=-q(t-1)\\
&\alpha'(t)=\alpha(t)-\alpha(t-1)\\
&q'(t)=q(t)-q(t-1)
\end{aligned} \tag{29}$$

$$\hat{\alpha}(t)-\alpha(t-1)=\theta_1(\alpha(t-1)-\alpha(t-2))+\theta_2(q(t-1)-q(t-2))+\theta_3\delta_h$$

This predictor model is equivalent to the difference predictor model.

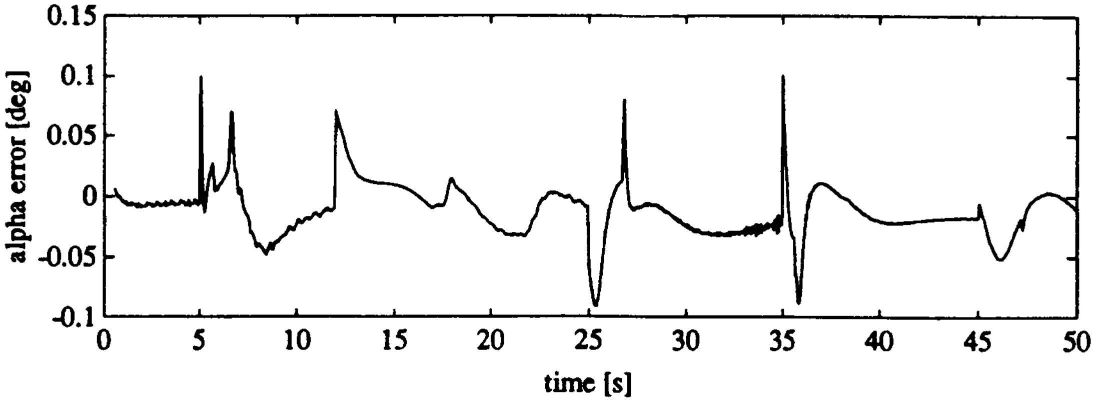

FIGURE 14 Prediction Error for Equation (26)

The following predictor model was developed by Mohler et al. [17]. This is the nonlinear predictor model, and it is included here for the sake of comparison. The predictor is given by the following equation.

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\theta(t-1)\\&\phi(t)^T=[\alpha(t-1),\alpha(t-1)^2,\alpha(t-1)^3,q(t-1),q(t-1)\alpha(t-1),\\&\quad q(t-1)\alpha(t-1)^2,q(t-1)\alpha(t-1)^3,\delta_h(t-1),\\&\quad \delta_h(t-1)\alpha(t-1),\delta_h(t-1)\alpha(t-1)^2,\delta_h(t-1)\alpha(t-1)^3,1]\end{aligned} \tag{30}$$

The prediction error for the nonlinear regressor is displayed in figure (15). As was stated in [21], the nonlinear predictor model out performed the predictor in equation (24); however, the difference predictor offers a significant improvement to both predictor models.

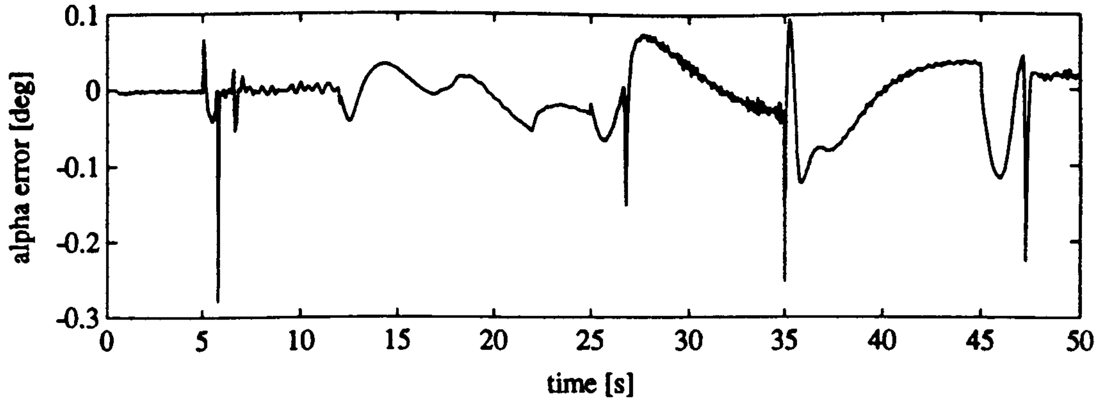

FIGURE 15 Prediction Error for Equation (30)

One more regressor was considered. This predictor model is similar to the first predictor model except that the order has been increased. The predictor model is:

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\theta(t-1)\\&\phi(t)^T=[\alpha(t-1),q(t-1),\alpha(t-2),q(t-2),\delta_h(t-1)]\end{aligned} \tag{31}$$

The prediction error is displayed below.

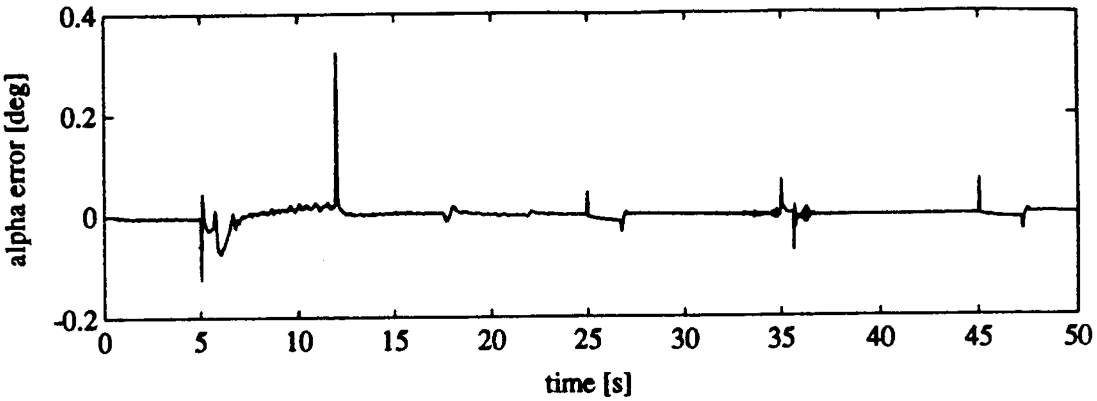

FIGURE 16 Prediction Error for Equation (31)

One possible explanation for this drastic improvement could be that the predictor model was accounting for the missing states in the predictor (i.e. the pitch, velocity, and thrust magnitude). However, when the above comparison was made using full state feedback in the predictor models, the corresponding graphs to figures (11) and (16) remained virtually unchanged. Thus, it appeared that a higher order linear model was able to identify some of the nonlinearities. What has actually happened, however, is that the higher order linear model has simply identified the offset in a similar fashion to the difference prediction model. This can be seen by examining the difference prediction model.

$$\begin{aligned}
&\Delta\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)\\
&\phi(t)^T=[\Delta\alpha(t-1),\Delta q(t-1),\delta_h(t-1)]\\
&\hat{\alpha}(t)=\Delta\hat{\alpha}(t)+\alpha(t-1)\\
&\hat{\alpha}(t)=(1+\theta_1)\alpha(t-1)-\theta_1\alpha(t-2)+\theta_2 q(t-1)-\theta_2 q(t-2)+\theta_3\delta_h(t-1)
\end{aligned} \qquad (32)$$

Thus, the difference predictor is simply a special case of a higher order linear predictor.

4.4 PREDICTION MODELS FOR DATA SET TWO

In this section only the prediction models described in equations (25), (26), and (31) will be considered since they were the most promising. Thrust-vectoring will be added to each of the prediction models, and data set two will be used in the analysis.

The addition of the thrust vectoring into the prediction models is a relatively simply matter. The prediction models that are to be compared are as follows. The first predictor model for data set two is the higher order linear model, and it is described by the following equation.

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\alpha(t-1),q(t-1),\alpha(t-2),q(t-2),\delta_h(t-1),\delta_v(t-1)]\end{aligned}\qquad(33)$$

The difference prediction model for data set two is described by the following equation.

$$\begin{aligned}&\Delta\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\Delta\alpha(t-1),\Delta q(t-1),\delta_h(t-1),\delta_v(t-1)]\end{aligned}\qquad(34)$$

The averaging predictor model for data set two is described by the following equation.

$$\begin{aligned}&\hat{\alpha}'(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\alpha'(t-1),q'(t-1),\delta_h(t-1),\delta_v(t-1)]\end{aligned}\qquad(35)$$

As expected their relative performance remained the same. The prediction error is considerably smaller for data set two. This is a result of limiting the control command velocity to 40 degrees per second for the stabilator angle and to 80 degrees per second for the thrust vectoring command. This limitation was done by the controller, and the input dynamics were ignored. The prediction errors are displayed below.

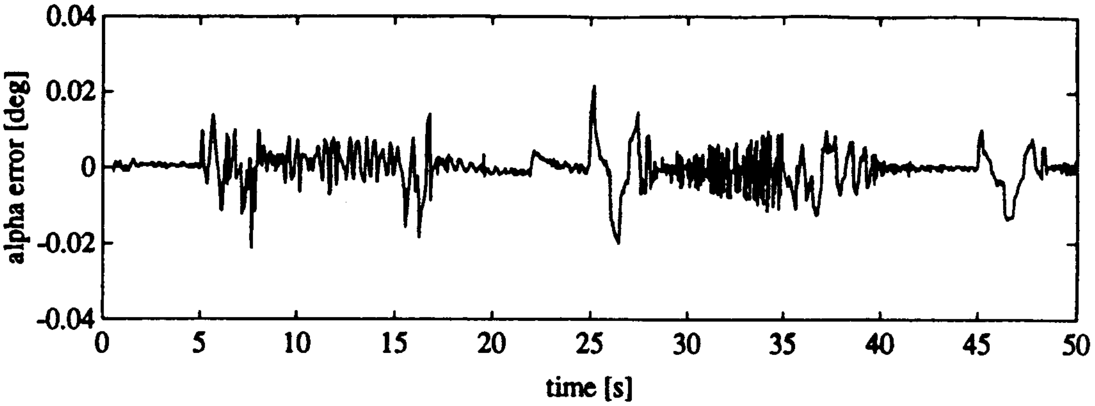

FIGURE 17 Prediction Error for Equation (33)

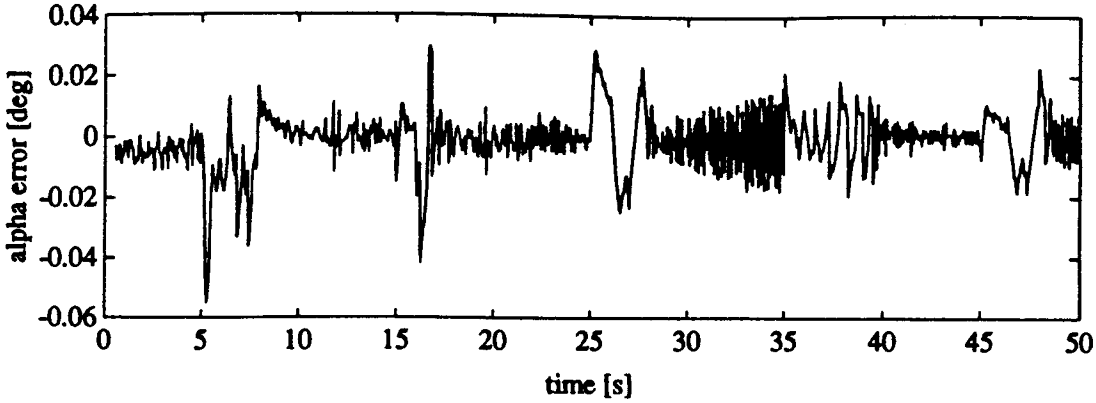

FIGURE 18 Prediction Error for Equation (34)

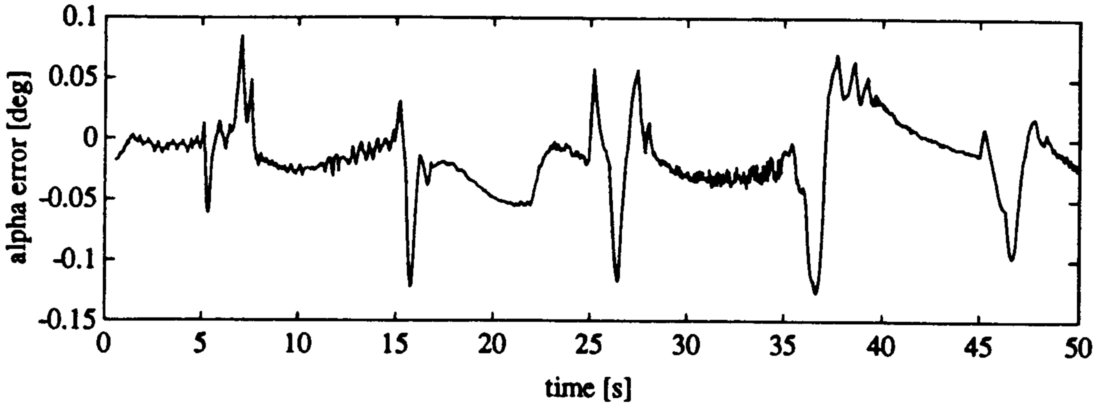

FIGURE 19 Prediction Error for Equation (35)

4.5 PREDICTION MODELS FOR DATA SET THREE

In this section the complete dynamics of the aircraft are included in the input-output data. One important characteristic of the data that needs to be pointed out is that the control command has been limited by the controller, and not just by the dynamics. This is important if the predictor is to be linear in parameters and measurements.

To see the importance of this consider the following differential equation:

$$\dot{x}=\begin{pmatrix} -40 & (u-x)<\frac{-4}{3} \\ 30\,(u-x) & \frac{-4}{3}\le(u-x)\le\frac{4}{3} \\ 40 & (u-x)>\frac{4}{3} \end{pmatrix} \qquad (36)$$

With an initial condition of zero, equation (21) has a solution at time 1/20 given by

$$x\left(\frac{1}{20}\right)=\begin{pmatrix} u(1-\exp(-1.5)) & |u|\le\frac{4}{3} \\ u-\frac{4}{3}\exp\left[-30\left(\frac{1}{20}-\frac{3u-4}{120}\right)\right] & \frac{4}{3}<|u|<\frac{10}{3} \\ 2 & u\ge\frac{10}{3} \\ -2 & u\le-\frac{10}{3} \end{pmatrix} \qquad (37)$$

A prediction model that is linear in parameters and measurements for equation (36), with x(0) = 0, is given in equation (38).

$$x\left(\frac{1}{20}\right)=cu \qquad \textit{where c is a constant} \qquad (38)$$

The predictor model is exact when

$$|\dot{u}| \le \frac{80}{3} \rightarrow |u| \le \frac{4}{3} \tag{39}$$

The predictor model remains approximately valid (± 3%) when

$$|\dot{u}| \le 40 \rightarrow |u| \le 2 \tag{40}$$

The problem is that c is not a constant, but depends on u. A predictor that is linear in parameters and measurements can be used if the value of u is limited before it enters the differential equation. It is this limited control value that is sent to the predictor model. To clarify this, equation (37) is displayed in figure (20), and a plot of c versus u is displayed in figure (21). Figure (21) shows explicitly how the 'constant' c depends on u. From the plots it can been seen that the prediction model is only linear in parameters and measurements if u is limited, and c is approximately constant if equation (40) is true.

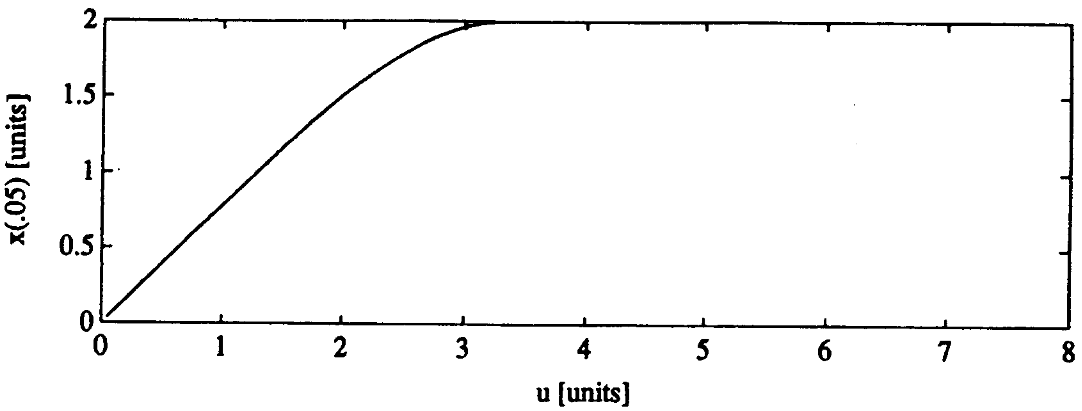

FIGURE 20 Plot of Equation (37)

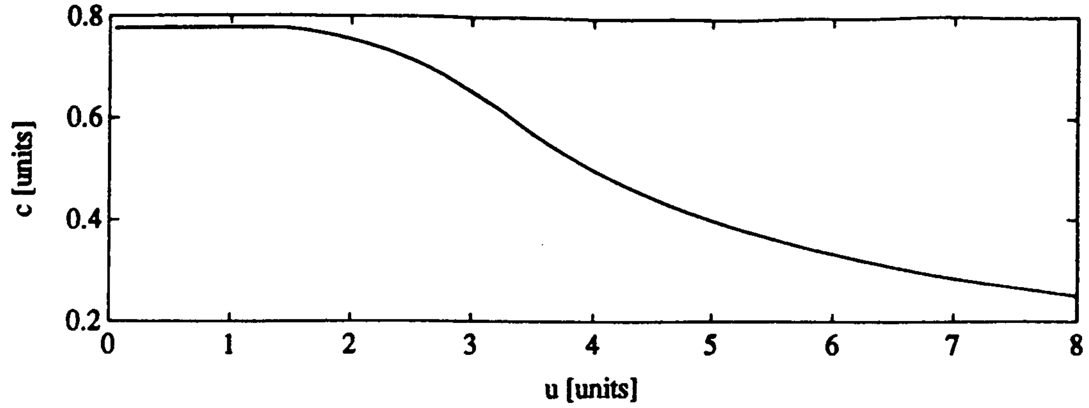

FIGURE 21 True Value of c for Equation (38)

Four predictor models are considered in this section. The first two correspond to the higher order linear predictor model, and they are given by equations (41) and (42). The only difference between them is that equation (42) has been increased in order by one. This is to account for the dynamics in the control signals.

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\theta(t-1)\\&\phi(t)^T=[\alpha(t-1),q(t-1),\alpha(t-2),q(t-2),\delta_h(t-1),\delta_v(t-1)]\end{aligned}\qquad(41)$$

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\theta(t-1)\\&\phi(t)^T=[\alpha(t-1),q(t-1),\alpha(t-2),q(t-2),\alpha(t-3),q(t-3),\\&\qquad\delta_h(t-1),\delta_v(t-1),\delta_h(t-2),\delta_v(t-2)]\end{aligned}\qquad(42)$$

The other two predictor models correspond to the difference predictor, and they are given by the following two equations.

$$\begin{aligned}&\Delta\hat{\alpha}(t)=\phi(t)^T\theta(t-1)\\&\phi(t)^T=[\Delta\alpha(t-1),\Delta q(t-1),\delta_h(t-1),\delta_v(t-1)]\end{aligned}\qquad(43)$$

$$\Delta\hat{\alpha}(t)=\phi(t)^{T}\theta(t-1)$$
$$\phi(t)^{T}=[\Delta\alpha(t-1),\Delta q(t-1),\Delta\alpha(t-2),\Delta q(t-2),\delta_{h}(t-1),\delta_{v}(t-1),\delta_{h}(t-2),\delta_{v}(t-2)] \quad (44)$$

It can be seen from the plots of the prediction error that the additional terms helped to estimate the dynamics on the control signal.

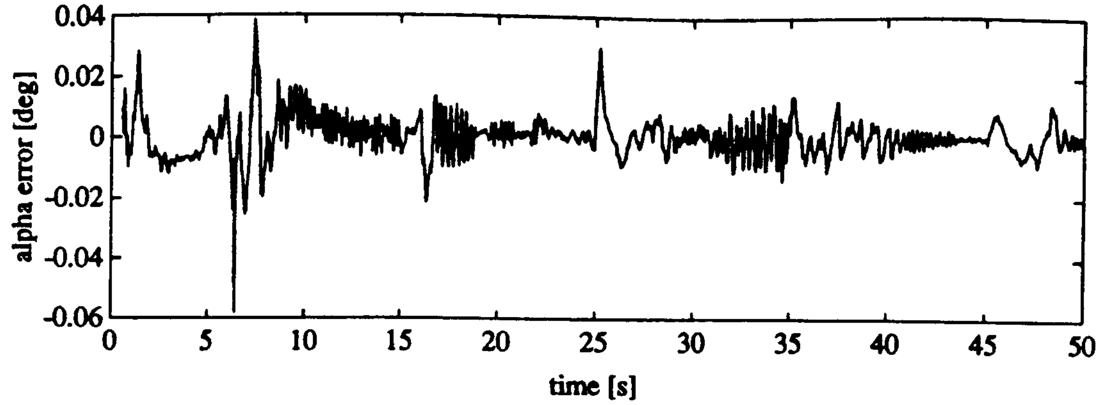

FIGURE 22 Prediction Error for Equation (41)

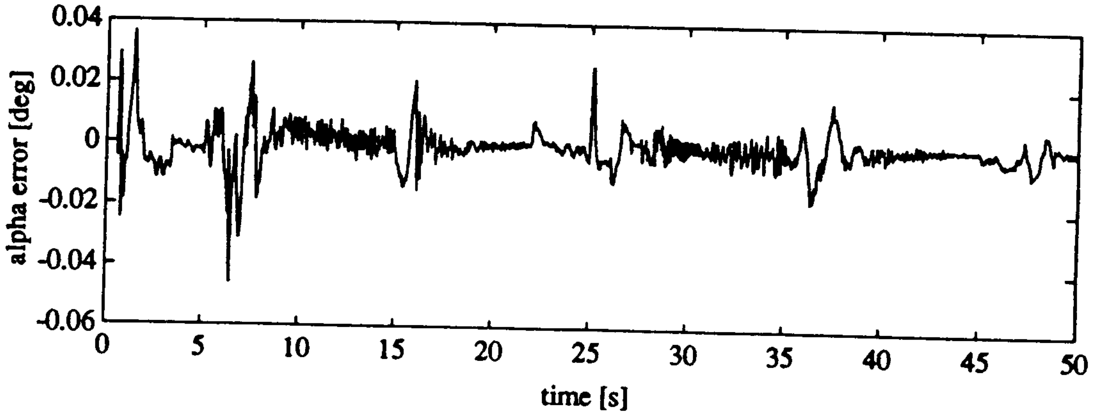

FIGURE 23 Prediction Error for Equation (42)

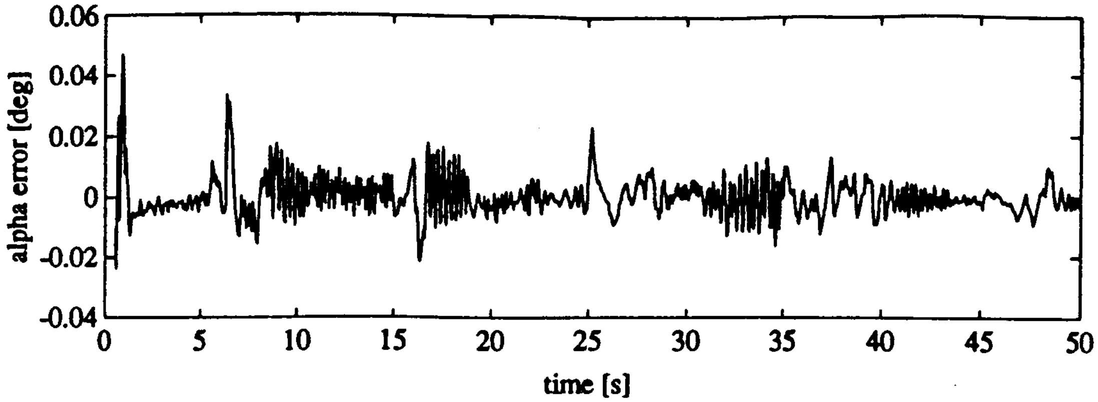

FIGURE 24 Prediction Error for Equation (43)

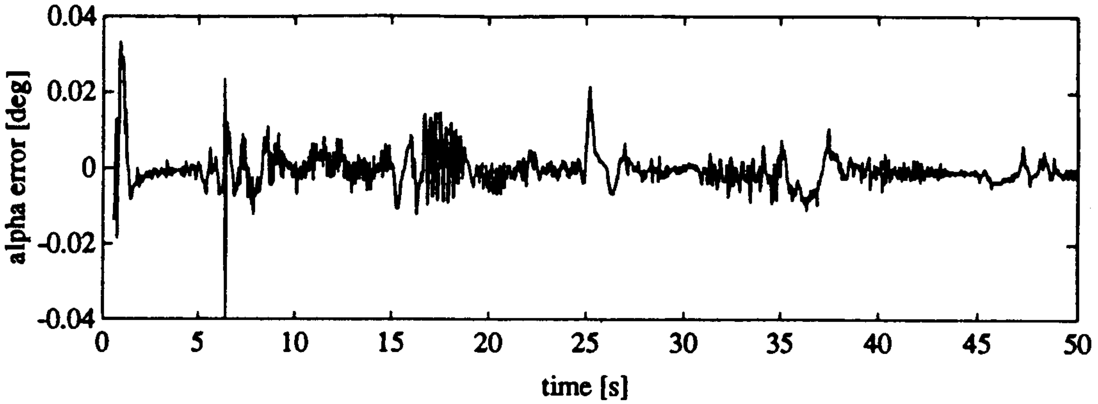

FIGURE 25 Prediction Error for Equation (44)

4.6 PREDICTOR FOR THE CONTROLLER

4.6.1 Overview of Control Calculation

Up to this point the only thing that has been considered is to provide a reasonable predictor for the next output if the past output and control values are known. This is a great first step, but it presents a problem for the control calculation. Consider the following prediction model to be used by a one step ahead controller.

$$\hat{\alpha}(t+1)=c_1\alpha(t)+c_2u(t) \tag{45}$$

Now, assume that we want $\alpha(t+1) = \alpha^*$, where α^* is some known value. Then u(t) can be calculated as follows:

$$u(t)=\frac{\alpha^*-c_1\alpha(t)}{c_2} \tag{46}$$

If the output and input are sampled at the same time, this creates a problem since u(t) is needed just prior to time t, but $\alpha(t)$ is not available until just after time t. Two solutions exist to overcome this problem. The first solution is to use an estimated value of $\alpha(t)$ to generate the control. Thus, the control calculation would be given by:

$$u(t)=\frac{\alpha^*-c_1\hat{\alpha}(t)}{c_2} \tag{47}$$

The other solution is to modify the prediction model so that it does not include $\alpha(t)$. This would lead to the following prediction model.

$$\begin{aligned}&\hat{\alpha}(t+1)=c_1\hat{\alpha}(t)+c_2u(t)\\&\hat{\alpha}(t+1)=c_1{}^2\alpha(t-1)+c_2u(t)+c_1c_2u(t-1)\\&\hat{\alpha}(t+1)=c_1'\alpha(t-1)+c_2'u(t)+c_3'u(t-1)\end{aligned} \tag{48}$$

4.6.2 Final Predictor

The purpose of this section is to develop a predictor for the aircraft that accounts for the problem that is discussed in section 4.6.1. The first solution is not adequate, since it would require an explicit predictor for q to be developed. Thus, the second method will be explored experimentally on data set three. By following the example above with the predictors in equations (42) and (44) the following predictor equations can be derived. (This assumes that an estimator for q(t-1) would be in the same form as the estimator for α(t-1). This was verified experimentally, but the results are not shown here.)

$$\begin{aligned}&\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\alpha(t-2),q(t-2),\alpha(t-3),q(t-3),\alpha(t-4),q(t-4),\\&\qquad\delta_h(t-1),\delta_v(t-1),\delta_h(t-2),\delta_v(t-2),\delta_h(t-3),\delta_v(t-3)]\end{aligned}\tag{49}$$

$$\begin{aligned}&\Delta\hat{\alpha}(t)=\phi(t)^T\hat{\theta}(t-1)\\&\phi(t)^T=[\Delta\alpha(t-2),\Delta q(t-2),\Delta\alpha(t-3),\Delta q(t-3),\\&\qquad\delta_h(t-1),\delta_v(t-1),\delta_h(t-2),\delta_v(t-2),\delta_h(t-3),\delta_v(t-3)]\end{aligned}\tag{50}$$

As can be seen from the following plots, the prediction error is not as good. This is because an estimate is implicitly being used for α(t-1) and q(t-1). The prediction error plots are displayed below.

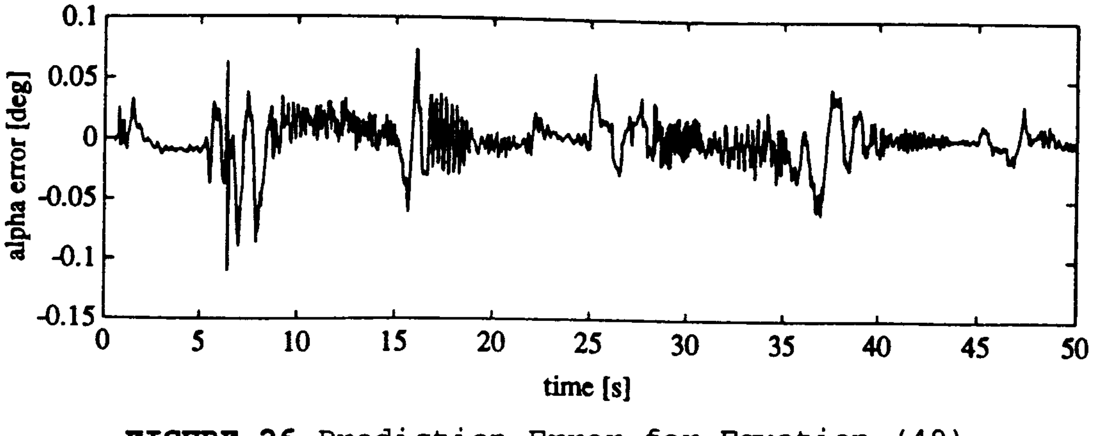

FIGURE 26 Prediction Error for Equation (49)

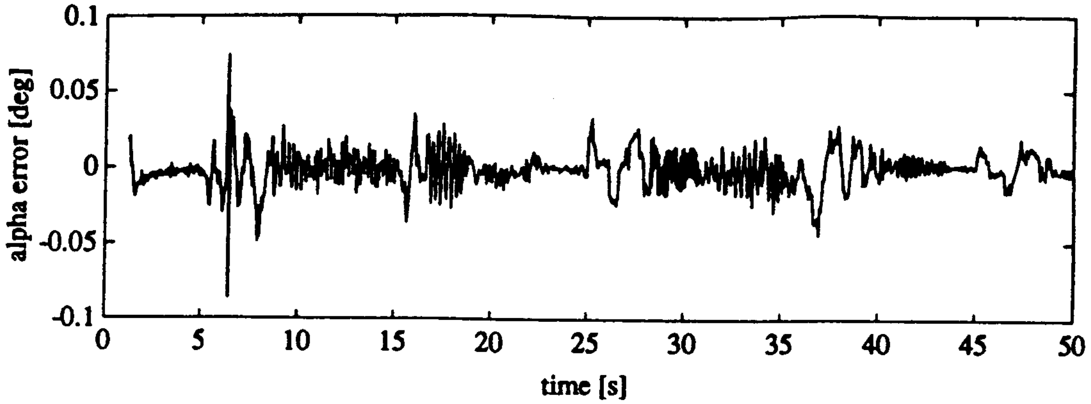

FIGURE 27 Prediction Error for Equation (50)

5 PARAMETER ESTIMATION

5.1 OVERVIEW OF THE RECURSIVE LEAST SQUARES

The recursive-least-squares (RLS) algorithm is the most common on-line parameter estimation algorithm in the literature today because it has several important properties. First, the RLS algorithm has a fast convergence rate (exponentially fast for a linear time-invariant system with proper excitation). Also, the stability of the RLS algorithm combined with direct and indirect adaptive control is well understood, and many proofs have been published in this area [4],[22]-[24].

The main disadvantage of the RLS is its inability to adequately track time-varying systems. The norm of the covariance matrix for the RLS algorithm tends towards zero which causes the adaptation to turn off. This is undesirable in the case where the parameters are time-varying. Several modifications have been made to the RLS algorithm to correct this problem. The modifications include constant forgetting factors, variable forgetting factors, and covariance modifications. Another disadvantage with the RLS algorithm is that it has a tendency to place heavy weights on outlying data points (because of the quadratic cost function). Thus, many types of data filtering and data normalization have been proposed. Since this thesis is dealing with a purely deterministic system only the modifications relating to time-varying parameters will be discussed.

5.2 DERIVATION WITH EXPONENTIAL DATA WEIGHTING

A complete derivation of the RLS algorithm with exponential data weighting can be found in [4] and [25]. A brief derivation of the RLS algorithm is given below. The idea of the RLS algorithm with exponential data weighting is to minimize the cost function in equation (51) at each time step. The weighting is included so that the older data can be forgotten.

$$J_t=\sum_{k=1}^{t}\frac{1}{2}\beta(t,k)\,[y(k)-\phi^T(k)\theta]^2 \tag{51}$$

where

$$\begin{aligned}&\beta(t,k)=\lambda(t)\beta(t-1,k) \qquad 1\le k\le t-1\\ &\beta(t,t)=1\\ &0<\lambda(t)\le 1\end{aligned} \tag{52}$$

Thus, at every step a new value is calculated for the parameters ,θ, such that the cost function is minimized.

Taking the derivative with respect to θ gives:

$$\frac{dJ_t}{d\theta}=\sum_{k=1}^{t}\beta(t,k)\phi(k)\,[y(k)-\phi^T\theta] \tag{53}$$

Setting the derivative to zero and solving for θ yields:

$$\theta(k+1)=\left[\sum_{k=1}^{t}\beta(t,k)\phi(k)\phi(k)^T\right]^{-1}\sum_{k=1}^{t}\beta(t,k)\phi(k)y(k) \tag{54}$$

Define

$$R(t)\ \triangleq\ \sum_{k=1}^{t}\beta(t,k)\phi(k)\phi(k)^T \tag{55}$$

$$F(t)\ \triangleq\ \sum_{k=1}^{t}\beta(t,k)\phi(k)y(k) \tag{56}$$

Then we have the following:

$$\theta(t)=R(t)^{-1}F(t) \tag{57}$$

$$R(t)=\lambda(t)R(t-1)+\phi(t)\phi(t)^{T} \tag{58}$$

$$F(t)=\lambda(t)F(t-1)+\phi(t)y(t)\phi \tag{59}$$

$$\theta(t)=\theta(t-1)+R(t)^{-1}\phi(t)\,[y(t)-\phi(t)^{T}\theta(t-1)] \tag{60}$$

Now let

$$P(t)=R(t)^{-1} \tag{61}$$

This leads to the recursive-least-squares algorithm with a variable forgetting factor.

$$\begin{aligned} K(t)&=\frac{P(t-1)\phi(t)}{\lambda(t)+\phi(t)^{T}P(t-1)\phi(t)} \\ \theta(t)&=\theta(t-1)+K(t)\,[y(t)-\phi(t)^{T}\theta(t-1)] \\ P(t)&=[I-K(t)\phi(t)^{T}]\,\frac{P(t-1)}{\lambda(t)} \end{aligned} \tag{62}$$

If $\lambda(t)$ equals one, then the above algorithm represents the standard recursive-least-squares algorithm. Because of applications to stochastic systems, the matrix P is commonly referred to as the covariance matrix.

5.3 MODIFICATIONS FOR TIME-VARYING PARAMETERS

One problem with the standard RLS (i.e. $\lambda(t) = 1$) is that the norm of the covariance matrix tends towards zero. This causes the gain of the algorithm to move towards zero, which causes the adaptation to turn off. This is displayed graphically in figures (28) and (29). The standard RLS algorithm is applied to data set three using the prediction model in equation (49). The trace of the covariance matrix serves as a measure for the matrix since it is a symmetric positive-definite matrix. A variety of modifications are proposed in the literature to keep the algorithm functional. The modifications are generally of two different types. The first idea is the inclusion of a forgetting factor. As explained above, this allows for past data to be forgotten. The second type of modifications that have been proposed is to manipulate the covariance matrix directly. An overview of a large variety of these modifications is presented by Shah and Cluett in [26].

The following two section will describe a variety of these methods and evaluate their performance with respect to the aircraft simulation presented in this thesis. For all of the comparisons, the prediction model in equation (49) is used. Data set three from chapter 4 is being used as the test data, and the covariance matrix is initialized to 100*I. The results of the previous chapter were developed by initializing the covariance matrix to 100000*I. This proved to be good for the prediction error, but when used in conjunction with a controller it was not as effective because of the variance of the parameters.

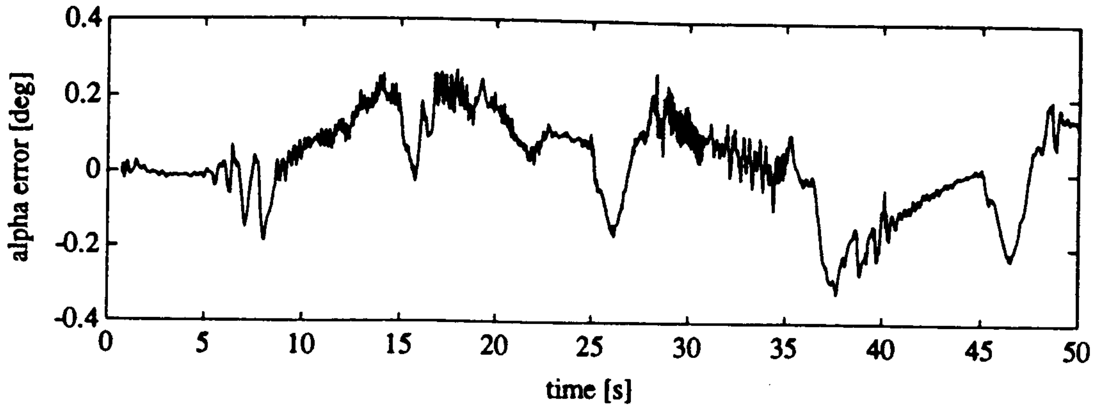

FIGURE 28 Prediction Error for the Standard RLS

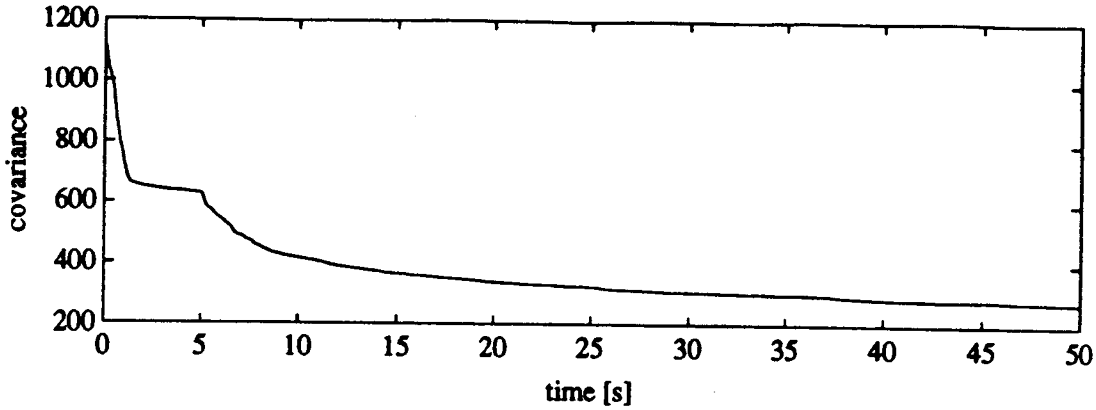

FIGURE 29 Covariance Trace for the Standard RLS

5.4 FORGETTING FACTOR

5.4.1 Constant Forgetting Factor

Several different choices exist for choosing $\lambda(t)$. The simplest is to set $\lambda(t)$ equal to a constant. This causes the data to be forgotten at an exponential rate, and the time constant of the data weighting is $1/(1-\lambda)$ (e.g. if $\lambda = .95$, the time constant would be 20 data points). This method prevents the gain from converging to zero; however, the opposite extreme can occur. If the signal does not have enough excitation, the norm of the covariance matrix tends towards infinity. This is displayed in figures (30) and (31). A maximum value was placed on the trace of the covariance matrix (200 times the initial trace), and when the trace exceeded this value the covariance matrix was reset.

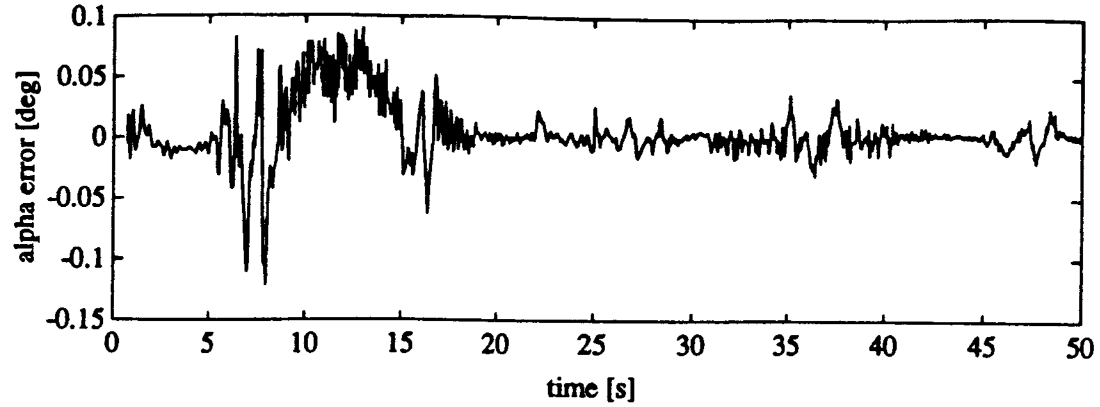

FIGURE 30 Prediction Error with λ(t) = .98

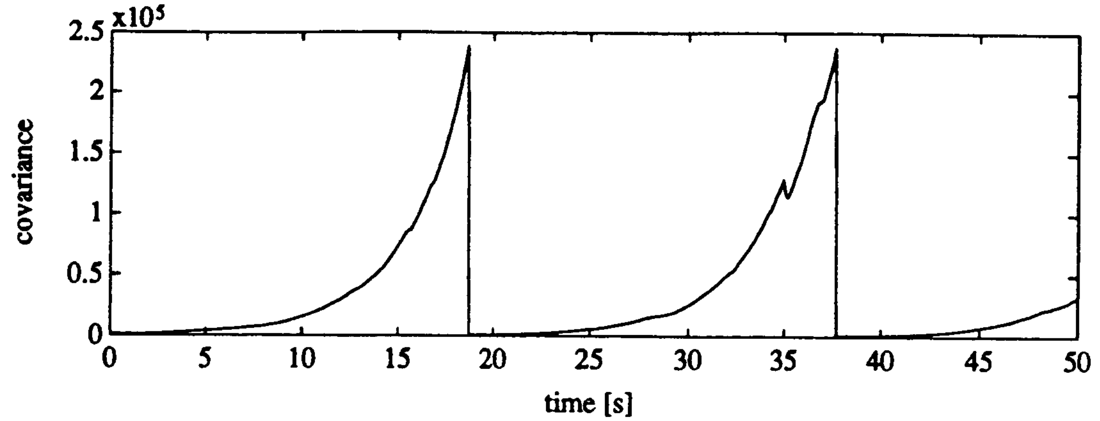

FIGURE 31 Covariance Trace for λ(t) = .98

5.4.2 Constant Information

Another method for selecting λ(t) is to choose some type of information measure, and then keep the measure constant. Thus, if the information content of the signal is large the forgetting factor will be small, and the old data will be forgotten. If the information content is small, then the forgetting factor will move towards one, and the old data will not be forgotten. This allowed the RLS algorithm to track some time-varying parameters. The problem of the covariance matrix growth, however, was not solved by this algorithm.

This idea was first presented by Fortescue et al. [27], and slight modifications were made by Ydstie et al.[28],[29]. A similar idea was presented by Sanoff et al. [31]. The information measure that was proposed in [27] was the weighted sum of the errors. Recursively, the measure was written as:

$$S(t)=\lambda(t)S(t-1)+[1-\phi(t)^{T}K(t)]\varepsilon^{2} \tag{63}$$

where

$$\varepsilon=y(t)-\phi(t)^{T}\theta(t) \tag{64}$$

Setting S(t) = to say S resulted in the following selection of a variable forgetting factor:

$$\lambda(t)=\frac{1}{2}\left(1-r-\frac{\varepsilon^{2}}{S}+\left[\left(1-r-\frac{\varepsilon^{2}}{S}\right)^{2}+4r\right]^{\frac{1}{2}}\right) \tag{65}$$

where

$$r=\phi(t)^{T}P(t-1)\phi(t) \tag{66}$$

The convergence of this algorithm for the time invariant case was derived by Cordero et al. [30]. One modification had to be made to the algorithm, however, for the convergence to be proved. The forgetting factor was set to one if the trace of the covariance exceeded a constant. The algorithm as originally stated did not limit the covariance matrix; however, it provided a great improvement over the constant forgetting factor. The results of this algorithm are displayed in figures (32)-(34).

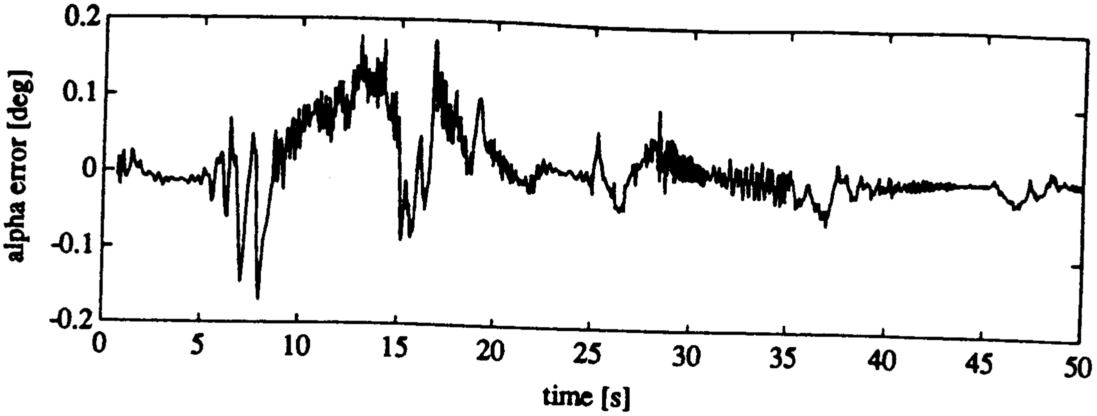

FIGURE 32 Prediction Error for Constant Information

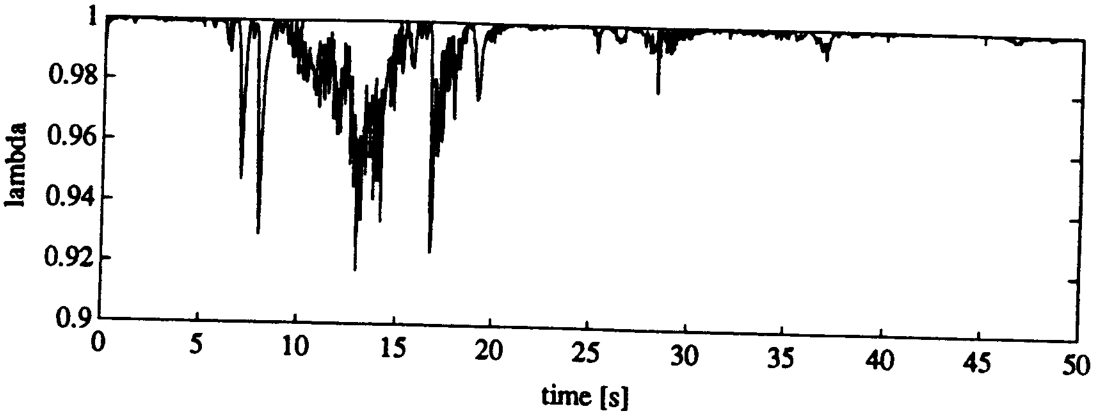

FIGURE 33 Forgetting Factor for Constant Information

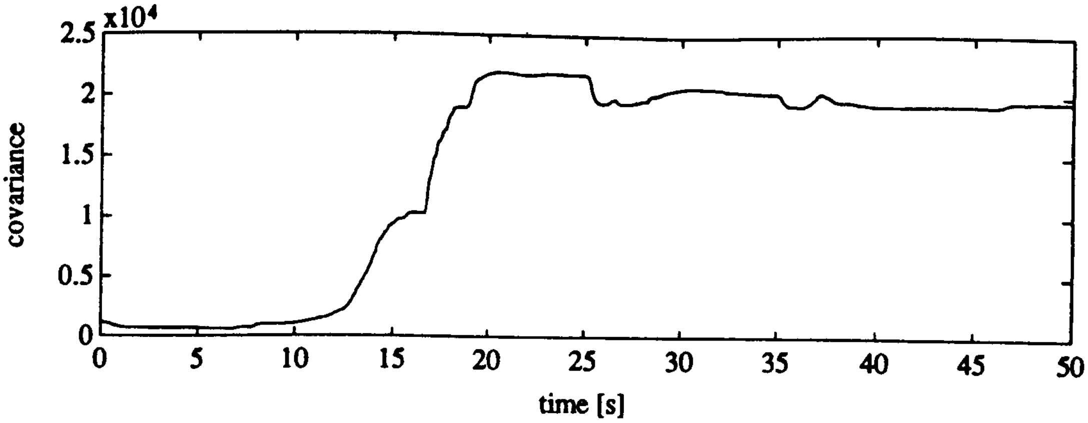

FIGURE 34 Covariance Trace for Constant Information

5.4.3 Constant Trace

Another method for selecting a variable forgetting factor was suggested by Sripada et al. [7]. This method selected a forgetting factor such that the trace of the covariance matrix remained constant. Taking the trace of both sides of the covariance matrix in equation (62), and setting the trace of P(t) equal to P(t-1) resulted in the following forgetting factor:

$$\lambda(t)=1-\frac{1}{2}\left(1+r-\left[(1+r)^2-4\frac{\|P(t-1)\phi(t)\|^2}{tr\ P(t-1)}\right]^{\frac{1}{2}}\right) \tag{67}$$

where

$$r=\phi(t)^T P(t-1)\phi(t) \tag{68}$$

The results of this algorithm are presented in figures (35) and (36).

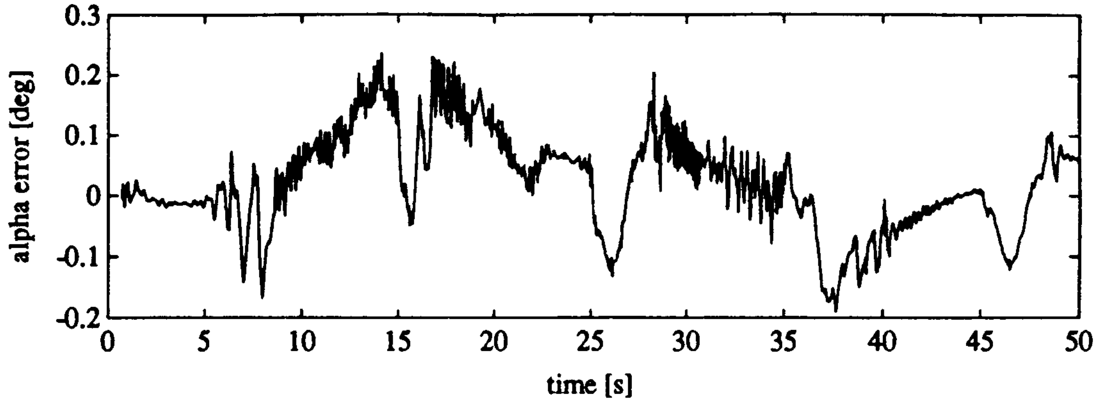

FIGURE 35 Prediction Error for Constant Trace

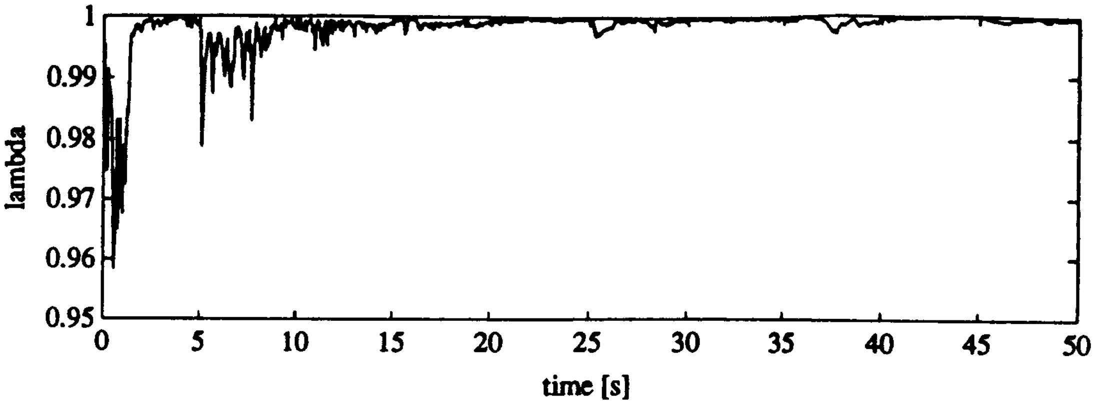

FIGURE 36 Forgetting Factor for Constant Trace

5.5 COVARIANCE MODIFICATIONS

5.5.1 Covariance Resetting

The simplest way to modify the covariance matrix was to reset it periodically. This method was suggested and the convergence for the linear time-invariant case was shown by Goodwin and Teoh [9]. The proofs presented in [9] covered most of the covariance modifications presented here. The covariance matrix in equation (62) was replaced by the following:

Let N be an integer. If t/N = integer

$$P(t)=kI \qquad 0 < k_{\min} < k < k_{\max} < \infty \tag{69}$$

Otherwise

$$P(t)=[I-K(t)\phi(t)^T]\frac{P(t-1)}{\lambda(t)} \tag{70}$$

This algorithm proved to be extremely successful. The results are displayed in figures (37) and (38).

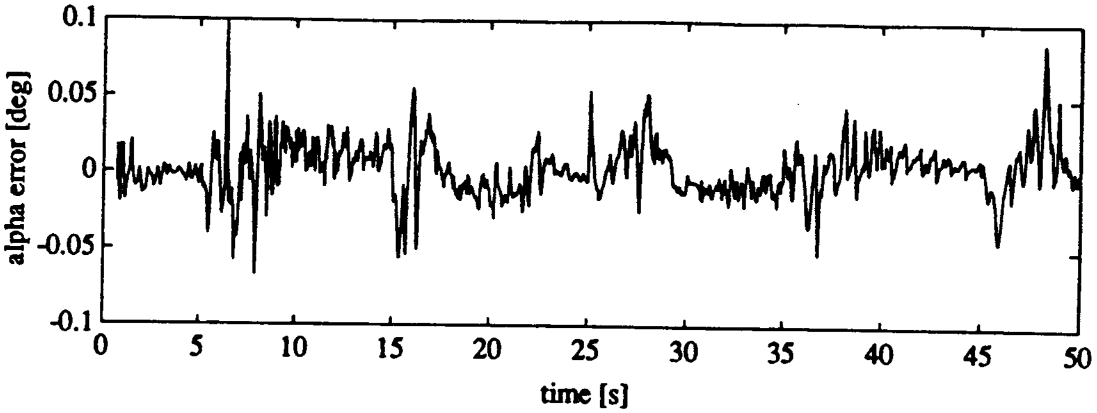

FIGURE 37 Prediction Error for Covariance Resetting

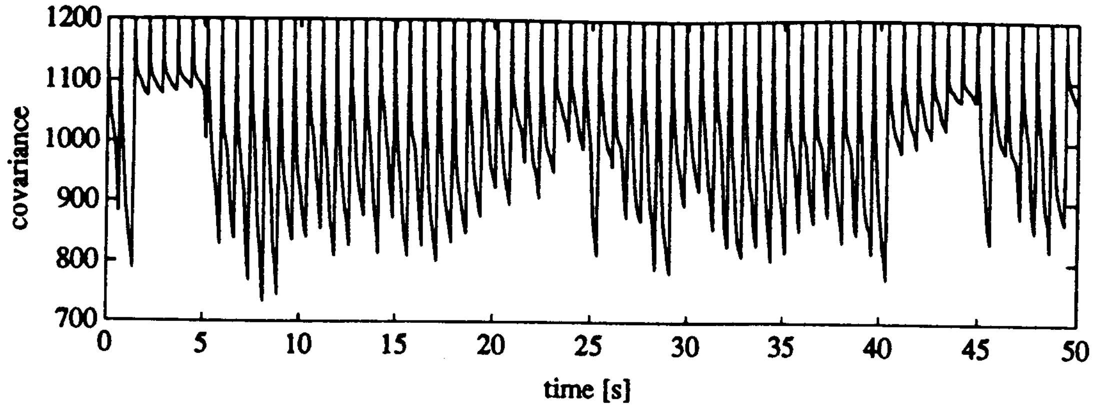

FIGURE 38 Covariance Trace for Covariance Resetting

5.5.2 Constant Covariance

Another method, proposed by Goodwin in [26], was to maintain a constant covariance trace by the addition of a properly scaled identity matrix. This leads to the following algorithm.

$$P'(t) = [I - K(t)\phi(t)^T]\frac{P(t-1)}{\lambda(t)} \tag{71}$$

Let τ = trace(P'(t)); let n = the dimension of ϕ, and C_0, C_1 denote two positive constants such that $C_1 > C_0$. If $\tau > C_0$

$$P(t) = P'(t) + \frac{C_1 - \tau}{n} I \tag{72}$$

If $\tau \le C_0$

$$P(t) = \frac{C_0}{\tau} P'(t) + \frac{C_1 - C_0}{n} I \tag{73}$$

The algorithm ensures a constant trace of C_1, and the following bounds are placed on the eigenvalues of P(t).

$$\frac{C_1 - C_0}{n} \le \lambda[P(t)] \le C_1 \tag{74}$$

The results of this algorithm are presented below.

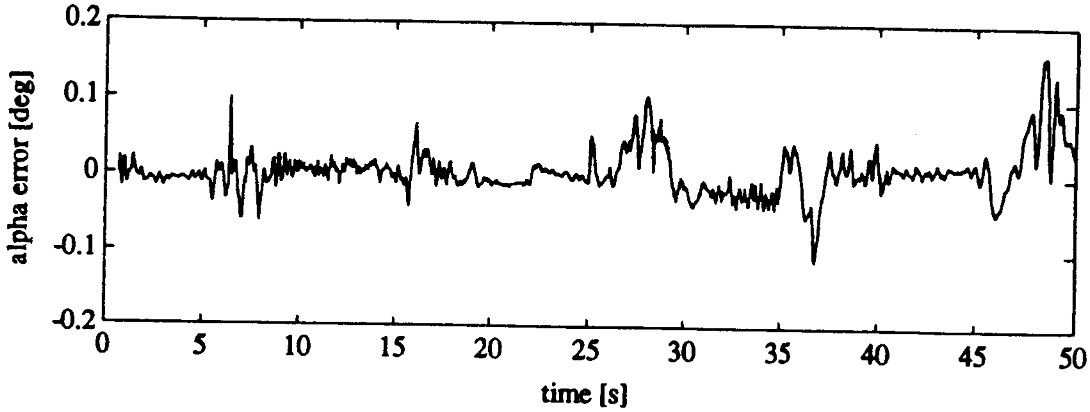

FIGURE 39 Prediction Error for Constant Trace

5.5.3 Covariance Regularization

Another method to modify the covariance was proposed by Ortega et al. [23], and convergence and robustness properties were also proved for the linear time-invariant case. It called for the following modification. Let C_0, C_1 denote two strictly positive constants such that $C_1 > C_0$. Then replace the covariance matrix in equation (62) by the following:

$$P'(t) = [I-K(t)\phi(t)^T]\,\frac{P(t-1)}{\lambda(t)} \tag{75}$$

$$P(t) = \left(1-\frac{C_0}{C_1}\right)P'(t) + C_0 I \tag{76}$$

This modification maintains the following bound on the eigenvalues of the covariance matrix.

$$C_0 \leq \lambda[P(t)] \leq C_1 \tag{77}$$

Its performance was reasonable, but the best results were obtained by combining the matrix regularization with the constant covariance. This resulted in the following algorithm.

Let τ = trace(P'(t)); let n = the dimension of ϕ, C_0, C_1 denote two positive constants such that $C_1 > C_0$, and $0 < C_2 < 1$. If $\tau > C_0$

$$P(t) = C_2 P'(t) + \frac{C_1 - C_2\tau}{n} I \tag{78}$$

If $\tau \leq C_0$

$$P(t) = \frac{C_0}{\tau} P'(t) + \frac{C_1 - C_0}{n} I \tag{79}$$

One way to interpret this algorithm is that it is a combination of the constant variance and the covariance resetting. Figure (40) displays the performance of the algorithm.

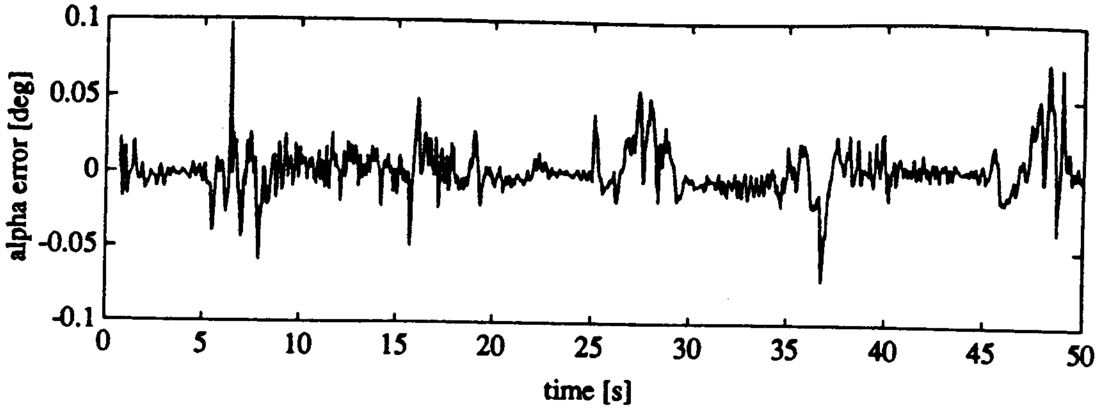

FIGURE 40 Prediction Error for Regularized RLS

It was this modified RLS algorithm that was used for the final controller. The values of the constants were $c_0 = 600$, $c_1 = 1200$ and $c_2 = .98$.

6 CONTROL CALCULATION

The controller was designed to satisfy several goals. First and most importantly, the control values were calculated such that the angle of attack of the aircraft followed the reference model. The control values were also calculated such that the thrust vectoring would return to zero if it was not needed, and a certain amount of smoothness was desired for the control signals. To be more explicit, the following cost function was minimized at each step.

$$\begin{aligned} J = {} & \frac{1}{2}\lambda_1[\alpha_{ref}(t+1)-\hat{\alpha}(t+1)]^2+\frac{1}{2}\lambda_2[\delta_{h_{cmd}}(t)-\delta_{h_{cmd}}(t-1)]^2 \\ & +\frac{1}{2}\lambda_3[\delta_{v_{cmd}}(t)-\delta_{v_{cmd}}(t-1)]^2+\frac{1}{2}\lambda_4[\delta_{v_{cmd}}(t)]^2 \end{aligned} \tag{80}$$

Let the prediction model in equation (49) be described by,

$$\begin{aligned} \hat{\alpha}(t+1) &= b_0\delta_{h_{cmd}}(t)+b_1\delta_{v_{cmd}}(t)+\Phi(t)^T\hat{\theta}(t-1) \\ \Phi(t)^T &= [\alpha(t-1),q(t-1),\alpha(t-2),q(t-2),\alpha(t-3),q(t-3), \\ & \quad \delta_{h_{cmd}}(t-1),\delta_{v_{cmd}}(t-1),\delta_{h_{cmd}}(t-2),\delta_{v_{cmd}}(t-2)] \end{aligned} \tag{81}$$

Taking the derivative of J with respect to the control gives,

$$\begin{aligned} \frac{dJ}{d\delta_{h_{cmd}}} &= \lambda_1[\alpha_{ref}(t+1)-\hat{\alpha}(t+1)]b_0+\lambda_2[\delta_{h_{cmd}}(t)-\delta_{h_{cmd}}(t-1)] \\ \frac{dJ}{d\delta_{v_{cmd}}} &= \lambda_1[\alpha_{ref}(t+1)-\hat{\alpha}(t+1)]b_1+\lambda_3[\delta_{v_{cmd}}(t)-\delta_{v_{cmd}}(t-1)] \\ & \quad +\lambda_4[\delta_{v_{cmd}}(t)] \end{aligned} \tag{82}$$

Setting the derivative to zero and solving for the control gives,

$$\begin{bmatrix} \delta_{h_{cmd}}(t) \\ \delta_{v_{cmd}}(t) \end{bmatrix} = \begin{bmatrix} \lambda_1 b_0^2+\lambda_2 & \lambda_1 b_0 b_1 \\ \lambda_1 b_0 b_1 & \lambda_1 b_1^2+\lambda_3+\lambda_4 \end{bmatrix}^{-1} \begin{bmatrix} \lambda_1 b_0\eta+\lambda_2\delta_{h_{cmd}}(t-1) \\ \lambda_1 b_1\eta+\lambda_3\delta_{v_{cmd}}(t-1) \end{bmatrix} \tag{83}$$

where

$$\eta=\alpha_{ref}(t+1)-\overline{\phi^{T}}(t)\overline{\theta}(t-1) \tag{84}$$

To include the velocity and magnitude limits in the control calculation, two extra conditions were added. The first condition requires that $\delta_{v\,cmd}(t)$ be recalculated if $\delta_{h\,cmd}(t)$ has reached the magnitude limit. The second condition requires that $\delta_{v\,cmd}(t)$ be recalculated if $\delta_{h\,cmd}(t)$ is a value requiring 80 degrees per second for the stabilator. $\delta_{v\,cmd}(t)$ is recalculated as follows:

$$\delta_{v_{cmd}}(t)=\frac{\eta-b_0\delta_{h_{cmd}}(t)}{b_1} \tag{85}$$

After the control values have been calculated, they are limited by 40 degrees per second for $\delta_{h\,cmd}(t)$ and by 80 degrees per second for $\delta_{v\,cmd}(t)$. The magnitude limits described in chapter 3 are also accounted for.

7 REFERENCE MODEL

The reference model is a critical part of the algorithm. For model reference adaptive control, a command signal is feed though a model system, and then the actual system is made to track the output of the model system. In general, the model reference signal is a feed forward signal, and it has no feedback from the aircraft. This proved to be ineffective for the aircraft when the velocity limits on the command signals were implemented. For the case when the controller dynamics (with velocity limits) were included in the simulations, I was unable to find an adequate feedforward reference model that resulted in a stable response. An example of the problem is developed below.

To simplify the example, the thrust vectoring was set to zero, and the stabilator dynamics were ignored. A comparison was made between the response of the aircraft with and without velocity limits on the stabilator angle. The velocity limits were imposed by the controller (i.e. the command signal was limited). Thus, the parameter estimation was identifying the same system for both cases, and the input dynamics were ignored. The simulation was for eight seconds. The reference trajectory for the first 3 seconds was generated using the method described later (equation (87) with λ = 0.96). It was included to damp out the disturbances that arose from the initial parameter mismatch. The reference trajectory from 3 to 8 seconds is described below, and it is a feed forward trajectory.

$$\alpha_{ref}(t) = \begin{pmatrix} 5^\circ & 3 \le t < 5 \\ 5.1^\circ & 5 \le t \end{pmatrix} \qquad (86)$$

The results are displayed in figures (41) and (42). The dashed line represents the system with the velocity constraints on the stabilator command angle. With velocity constraints on the command signal, the response was unstable. Since the output of the system follows the reference trajectory, no feedback should be needed. When the control is limited, however, it is no longer possible for the system to exactly follow the reference trajectory. Thus, feedback is included to improve the reference model and the overall performance.

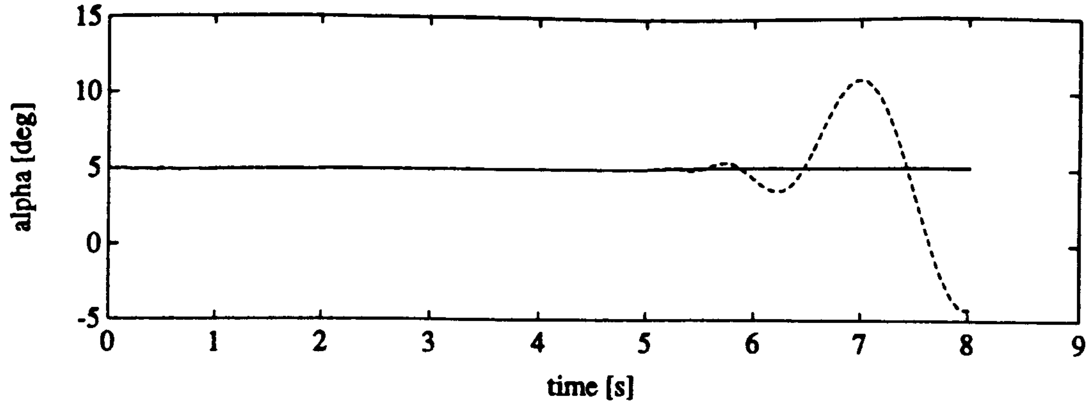

FIGURE 41 Angle of Attack for a Feedforward Reference

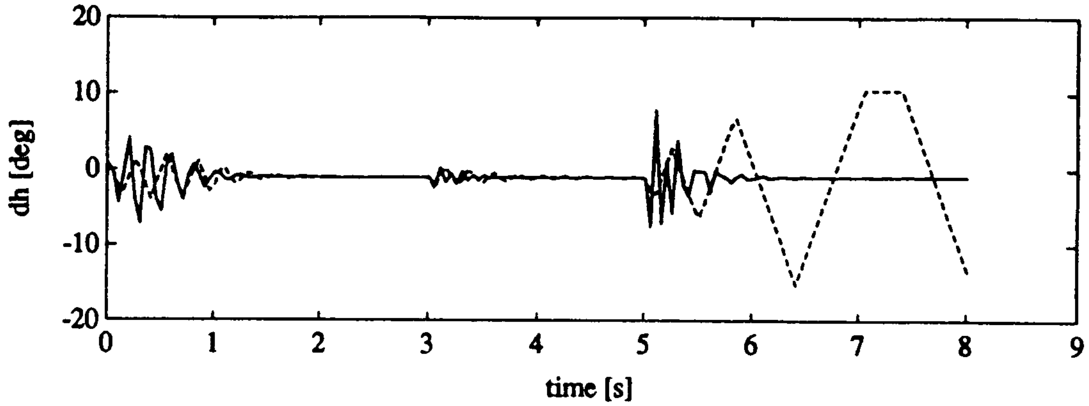

FIGURE 42 Stabilator Angle for a Feedforward Reference

The new method proposed in this paper uses feedback from the aircraft to improve the reference trajectory. To my knowledge, this method has not been reported in the literature. The class of models for the reference trajectory that were investigated are simply filters that use the past values of the angle of attack as the states. Thus, the reference model has no internal states of its own. A simple first order filter can be formed as follows:

$$\alpha_{ref}(t)=\lambda\alpha(t-1)+(1-\lambda)\alpha_{cmd} \qquad 0<\lambda<1 \tag{87}$$

If the system being controlled was a deterministic linear time-invariant system with unlimited control, the two approaches would be identical because the output of the system at (t-1) would be equal to the reference trajectory at time (t-1). Thus the reference model would not need any feedback from the system.

With a first order model reference, excellent performance was achieved when the input dynamics and velocity constraints were ignored (see data set in chapter 4). For the complete system, a second order filter was found to be sufficient to provide excellent performance. A general second order filter is described in equation (88).

$$\begin{aligned}\alpha_{ref}(t)=&(\lambda_1+\lambda_2)\alpha(t-1)-(\lambda_1\lambda_2)\alpha(t-1)\\&+(1+\lambda_1\lambda_2-\lambda_1-\lambda_2)\alpha_{cmd}(t)\end{aligned} \tag{88}$$

The results for this reference trajectory when used on the complete system are displayed in chapter 8.

8 COMPLETE RESPONSE

Each piece of the final controller will be restated below for clarity. The predictor model described in equation (49) was used. The difference predictor model in equation (50) generated similar results. One major problem with the difference predictor is that the variance of the noise on the measurements is doubled. This problem was not encountered in this thesis since no noise was considered. The predictor model is restated below.

$$\begin{aligned}
\hat{\alpha}(t+1) &= b_0(t-1)\delta_{h_{cmd}}(t) + b_1(t-1)\delta_{v_{cmd}}(t) + \bar{\Phi}(t)^T\bar{\theta}(t-1) \\
\bar{\Phi}(t)^T &= [\alpha(t-1), q(t-1), \alpha(t-2), q(t-2), \alpha(t-3), q(t-3), \\
&\quad \delta_{h_{cmd}}(t-1), \delta_{v_{cmd}}(t-1), \delta_{h_{cmd}}(t-2), \delta_{v_{cmd}}(t-2)] \\
\theta(t-1)^T &= [b_0(t-1) \quad b_1(t-1) \quad \bar{\theta}(t-1)^T] \\
\phi(t)^T &= [\delta_{h_{cmd}}(t) \quad \delta_{v_{cmd}}(t) \quad \bar{\Phi}(t)^T]
\end{aligned} \tag{89}$$

The adaptation was performed using the modified RLS described in section 5.5.3 with $c_0 = 600$, $c_1 = 1200$, and $c_2 = 0.98$. This gives the following algorithm:

$$\begin{aligned}
K(t) &= \frac{P(t-1)\phi(t)}{1+\phi(t)^T P(t-1)\phi(t)} \\
\theta(t) &= \theta(t-1) + K(t)\,[y(t) - \phi(t)^T\theta(t-1)] \\
P'(t) &= [I - K(t)\phi(t)^T]\,P(t-1)
\end{aligned} \tag{90}$$

Let τ = trace(P'(t)), if $\tau > C_0$

$$P(t) = C_2 P'(t) + \frac{C_1 - C_2\tau}{n} I \tag{91}$$

If $\tau \leq C_0$

$$P(t)=\frac{C_0}{\tau}P'(t)+\frac{C_1-C_0}{n}I \tag{92}$$

A projection was used in conjunction with the RLS algorithm to bound b_0 and b_1 away from zero. Thus, the following conditions were imposed:

$$b_0 \leq -0.0002 \qquad b_1 \geq 0.0002$$

The above values were chosen by observing the parameter variations for several different flight paths. All of the parameter variations for maneuver one are plotted in the appendix. The prediction error for maneuver one is also plotted in the appendix.

The control was calculated to minimize the cost function in equation (80) with λ_1 = 100, λ_2 = 0.001, λ_3 = 0.001, and λ_4 = 0.001. This lead to the following control calculation:

$$\begin{bmatrix}\delta_{h_{cmd}}(t)\\ \delta_{v_{cmd}}(t)\end{bmatrix}=\begin{bmatrix}\lambda_1 b_0^2+\lambda_2 & \lambda_1 b_0 b_1\\ \lambda_1 b_0 b_1 & \lambda_1 b_1^2+\lambda_3+\lambda_4\end{bmatrix}^{-1}\begin{bmatrix}\lambda_1 b_0\eta+\lambda_2\delta_{h_{cmd}}(t-1)\\ \lambda_1 b_1\eta+\lambda_3\delta_{v_{cmd}}(t-1)\end{bmatrix} \tag{94}$$

where

$$\eta=\alpha_{ref}-\bar{\Phi}(t)^{T}\bar{\theta}(t-1) \tag{95}$$

If the stabilator command exceeded a velocity of 80 degrees per second (twice its velocity limit) or its magnitude constraint, then the stabilator command was limited by 40 degrees per second or its magnitude constraint, and the thrust vectoring angle was recalculated such that the predicted output would match the reference trajectory. This is similar to the daisy chaining idea that was described in [32]. $\delta_{v\ cmd}(t)$ was recalculated as follows:

$$\delta_{v_{cmd}}(t) = \frac{\eta - b_0 \delta_{h_{cmd}}(t)}{b_1} \tag{96}$$

The limited thrust vectoring and the stabilator commands were the control signals for the aircraft. Also, to prevent the parameters for the thrust-vectoring from drifting, the following condition was imposed.

$$If\ |\delta_{v_{cmd}}| < 1° \ ,\ then\ \delta_{v_{cmd}} = 0 \tag{97}$$

A second order reference trajectory (as described in chapter 7) was used. The parameters of the reference trajectory were not fixed but varied according to the gain schedule listed in the table below. The second order reference trajectory is given in equation (98).

$$\begin{gathered} \alpha_{ref}(t) = c_1(\varepsilon)\,\alpha(t-1) + c_2(\varepsilon)\,\alpha(t-2) + (1 - c_1(\varepsilon) - c_2(\varepsilon))\,\alpha_{cmd}(t) \\ \varepsilon = |\alpha_{cmd}(t-1) - \alpha(t-1)| \end{gathered} \tag{98}$$

	$c_1(\varepsilon)$	$c_2(\varepsilon)$		$c_1(\varepsilon)$	$c_2(\varepsilon)$
$0 \le \varepsilon < 1$	1.7600	-0.7743	$6 \le \varepsilon < 8$	1.8073	-0.8221
$1 \le \varepsilon < 2$	1.7215	-0.7517	$8 \le \varepsilon < 10$	1.8241	-0.8365
$2 \le \varepsilon < 3$	1.7563	-0.7796	$10 \le \varepsilon < 15$	1.8407	-0.8509
$3 \le \varepsilon < 4$	1.7734	-0.7937	$15 \le \varepsilon < 25$	1.8572	-0.8655
$4 \le \varepsilon < 6$	1.7904	-0.8079	$25 \le \varepsilon$	1.8736	-0.8801

FIGURE 43 Table of Constants for the Equation (98)

The values were chosen such that all but the first filter correspond to a constant percent overshoot with different rise times. The first filter simply put two discrete poles on the real axis, one at .87 and one at .89. This is by no means an optimal gain schedule, and undoubtedly it can be improve. It did, however, prove to be highly successful for a wide range of maneuvers.

Several different simulations were used to evaluate the model performance. Two that are representative of the results are described below. The first maneuver corresponds to the maneuver presented by Ostroff in [33]. The angle of attack is changed from 5 degrees, to 60 degrees, to 35 degrees, and back to 5 degrees in 8 second intervals. The estimated parameters for this maneuver are displayed in the appendix. The second maneuver holds the angle of attack at 60 degrees for an extended period of time.

The character of the response for maneuver one is similar to the response reported by Ostroff in [33]. The adaptive controller, however, provided a slightly faster response. The angle of attack reached 55 degrees in 2.0 seconds. While, the variable gain approach in [33] reached 55 degrees in just under 3.5 seconds. The time-optimal control (with a limitation of 40 degrees per second on the thrust vectoring) reached 55 degrees in about 1.8 seconds [34]. The nonlinear adaptive controller reported in [35] failed when the full dynamics of the aircraft were considered. For maneuver two, the adaptive controller held the angle of attack at 60 degrees. The aircraft reached a 'steady state' position; however, the climb angle was negative (i.e. the aircraft was diving). One important characteristic about the adaptive control in this thesis was that the thrust-vectoring returned to zero when it was not needed. This prevented the thrust-vectoring vanes from constantly being exposed to the high temperature exhaust plume. The plots for maneuver one are displayed in figures (44)-(51), and the plots for maneuver two are displayed in figures (52)-(59).

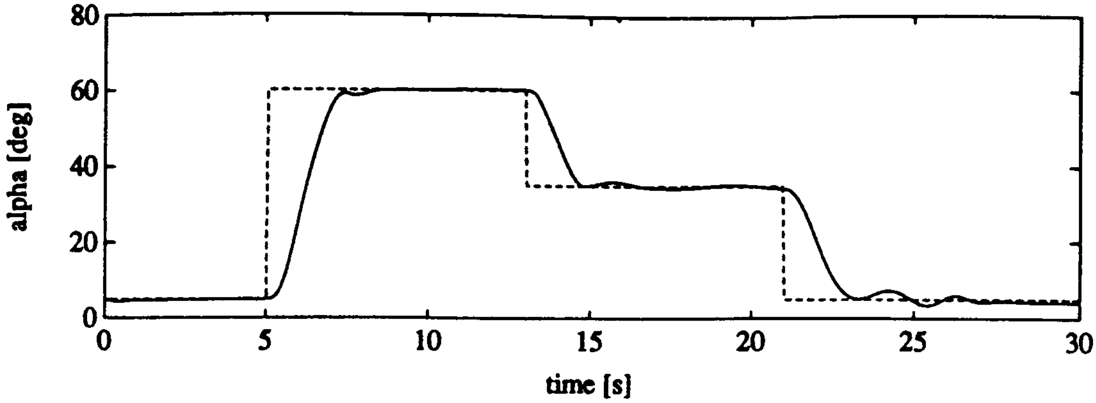

FIGURE 44 Angle of Attack for Maneuver One

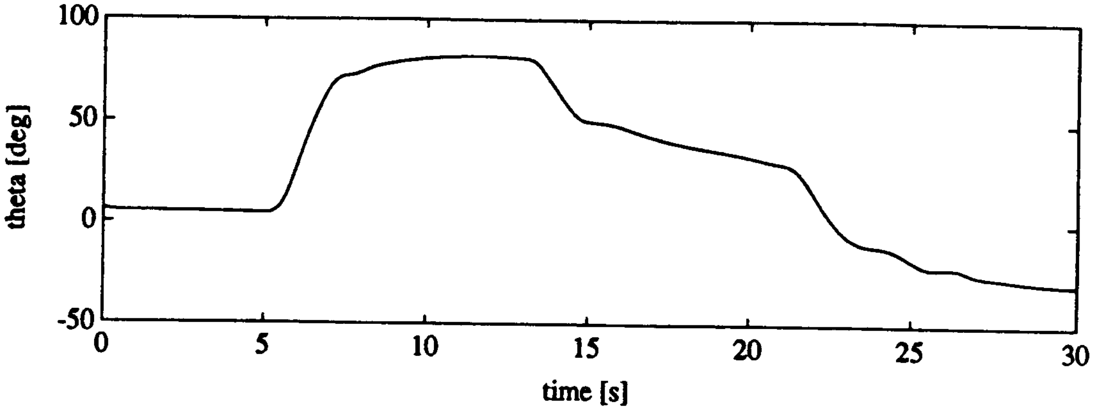

FIGURE 45 Pitch Angle for Maneuver One

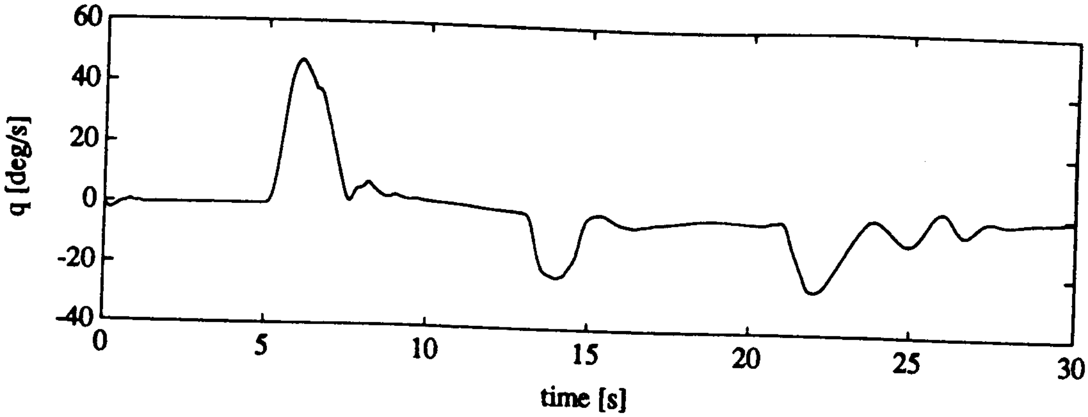

FIGURE 46 Pitch Rate for Maneuver One

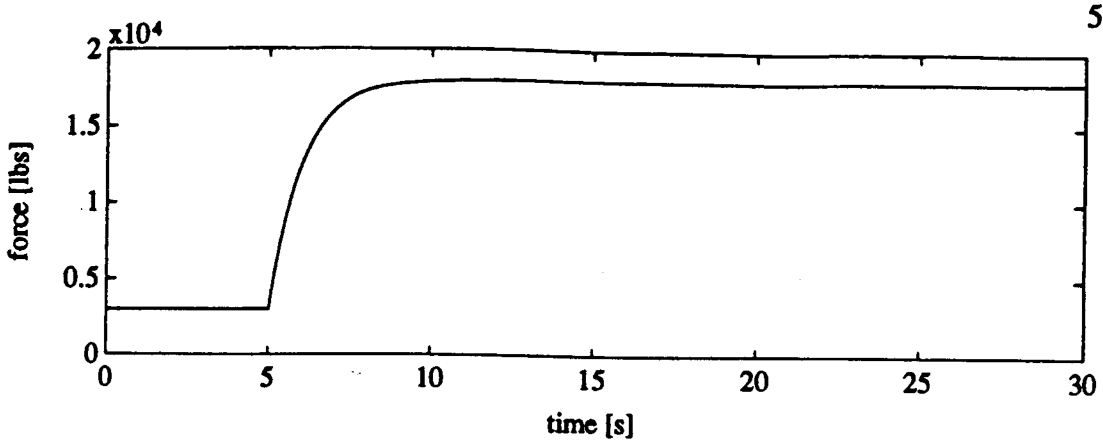

FIGURE 47 Thrust Magnitude for Maneuver One

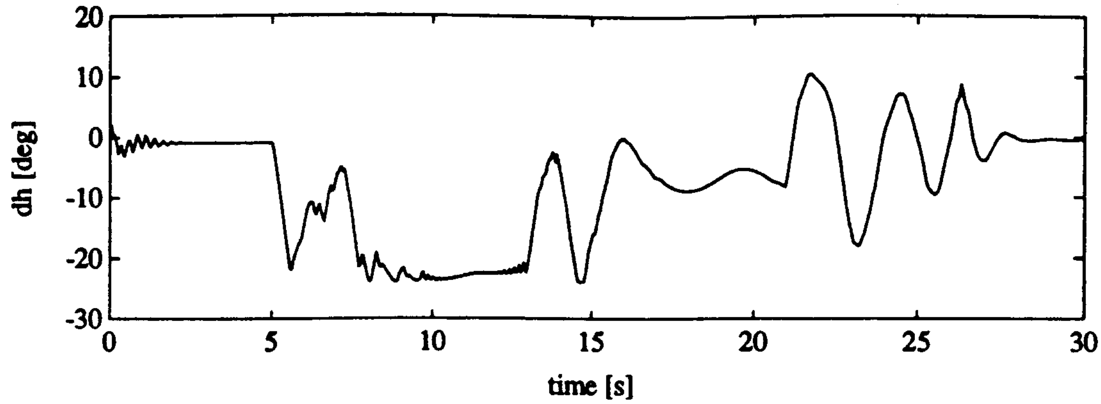

FIGURE 48 Stabilator Angle for Maneuver One

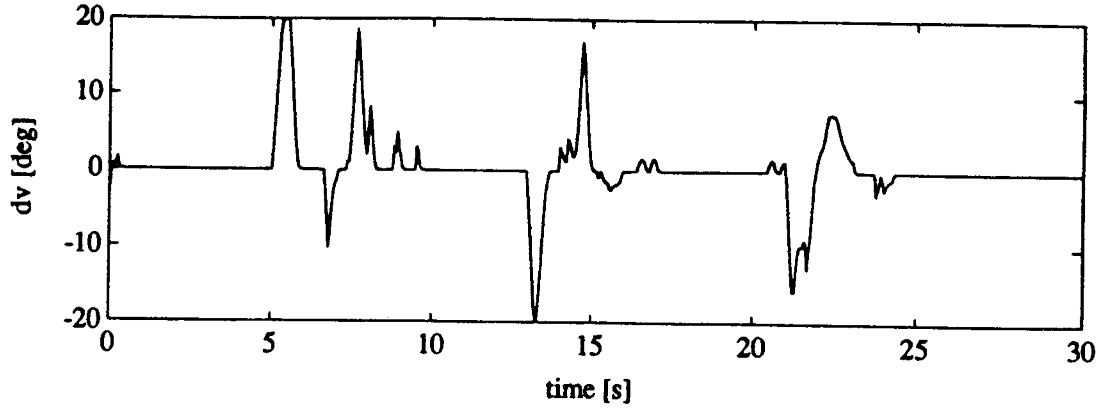

FIGURE 49 Thrust-Vectoring Angle for Maneuver One

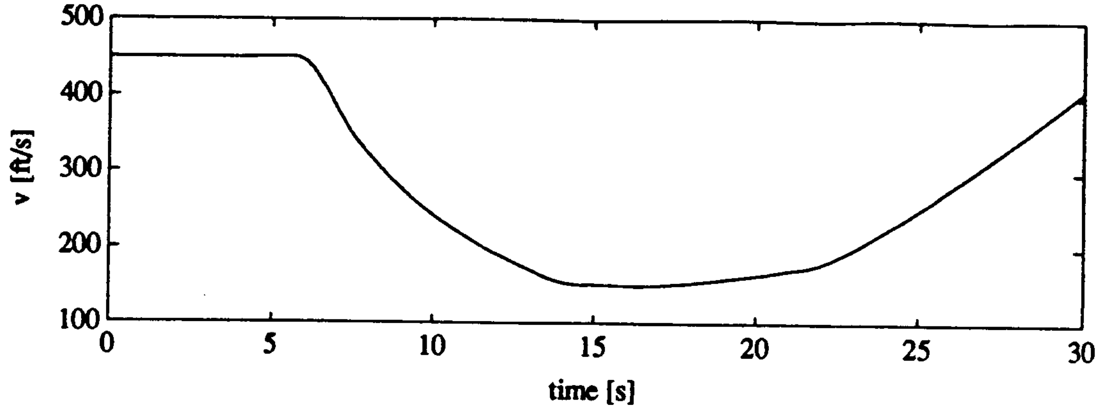

FIGURE 50 Velocity for Maneuver One

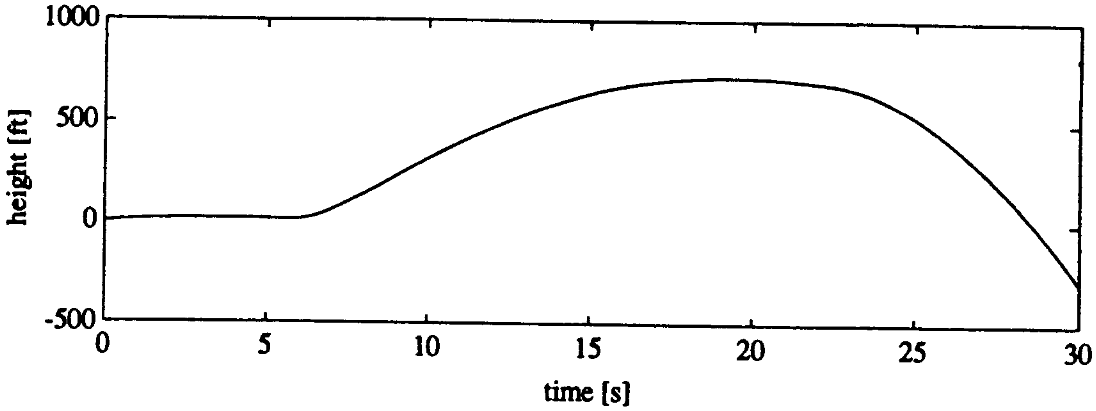

FIGURE 51 Flight Path for Maneuver One

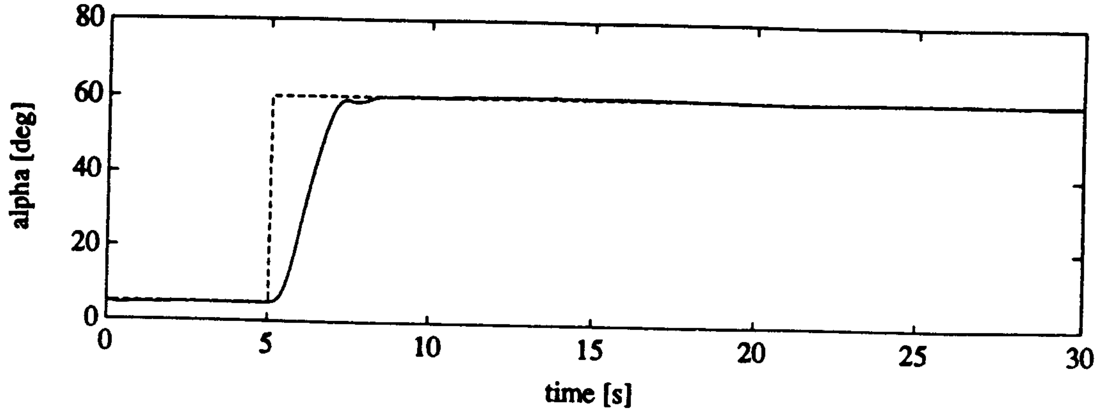

FIGURE 52 Angle of Attack for Maneuver Two

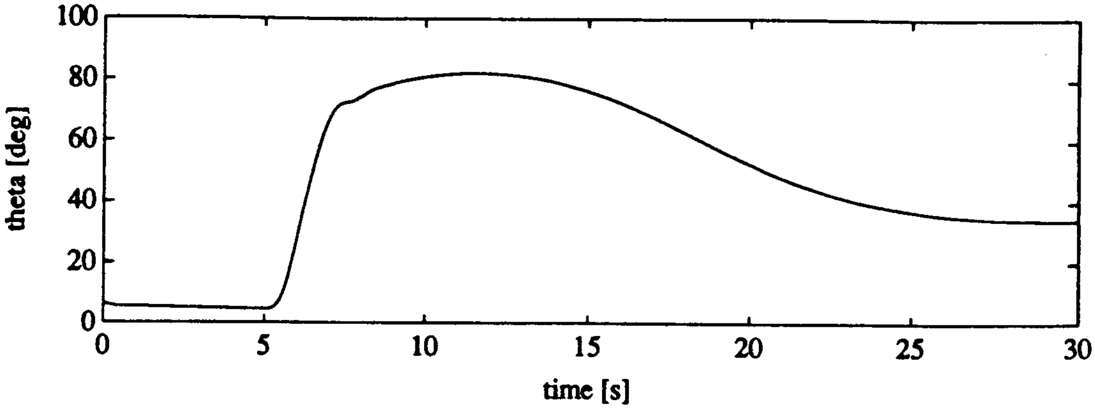

FIGURE 53 Pitch Angle for Maneuver Two

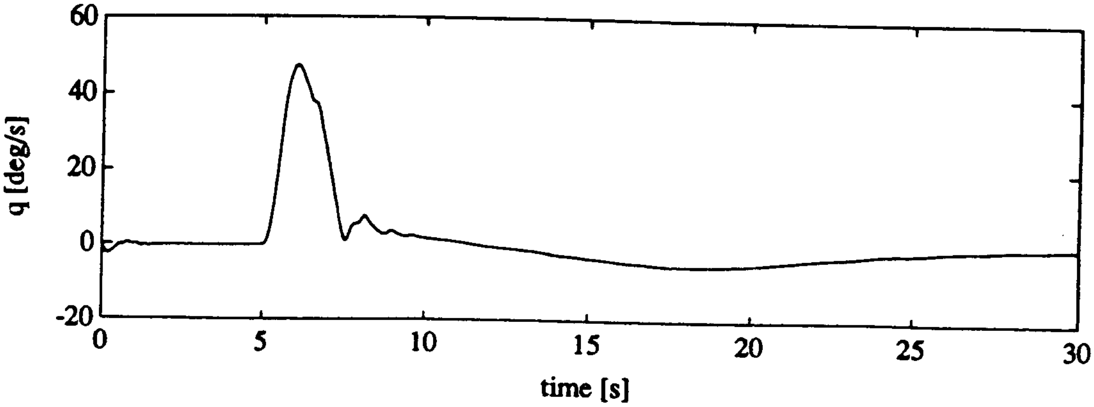

FIGURE 54 Pitch Rate for Maneuver Two

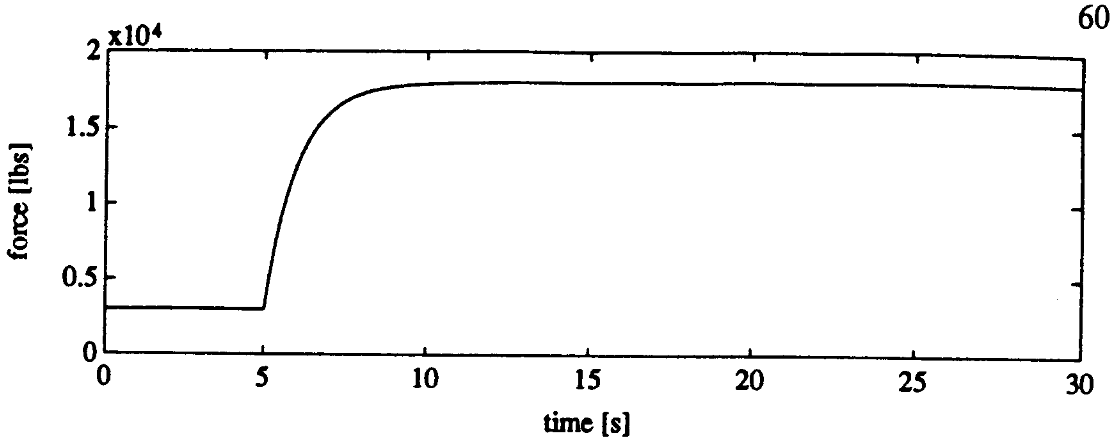

FIGURE 55 Thrust Magnitude for Maneuver Two

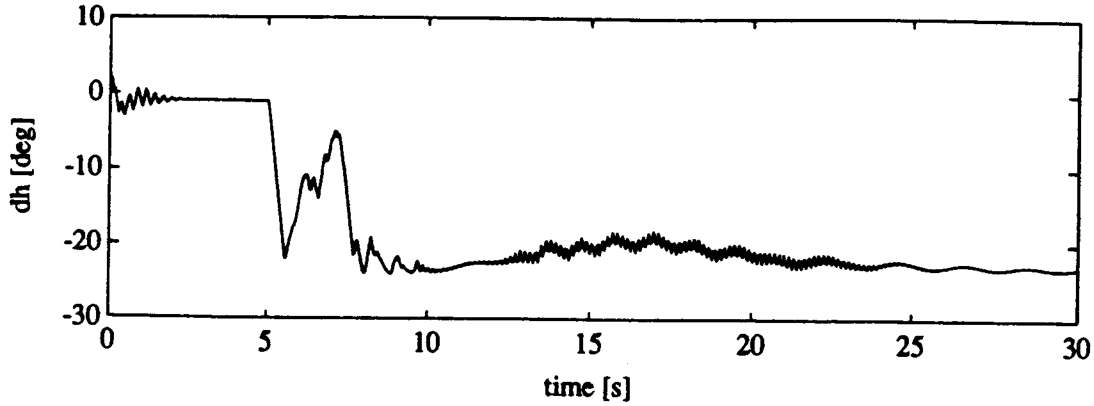

FIGURE 56 Stabilator Angle for Maneuver Two

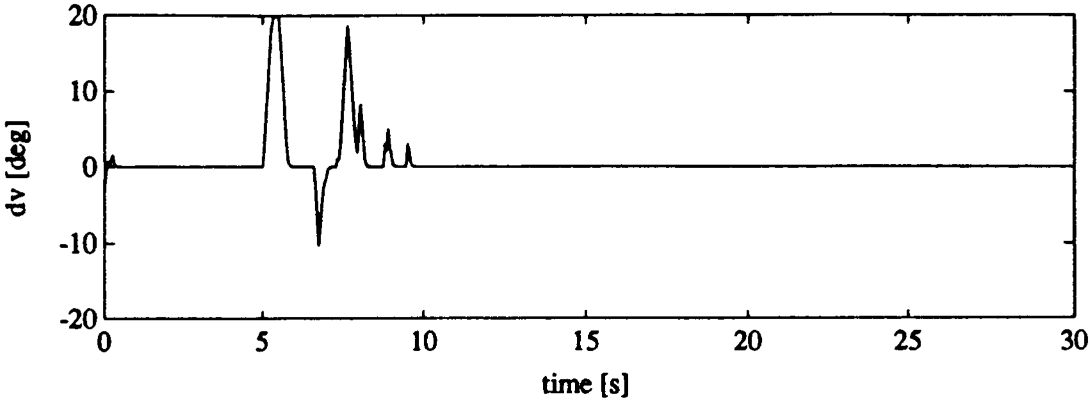

FIGURE 57 Thrust-Vectoring Angle for Maneuver Two

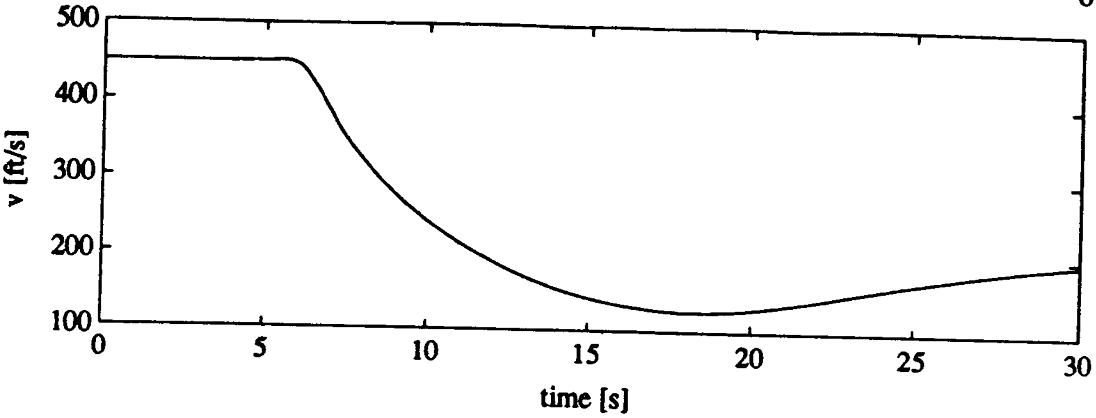

FIGURE 58 Velocity for Maneuver Two

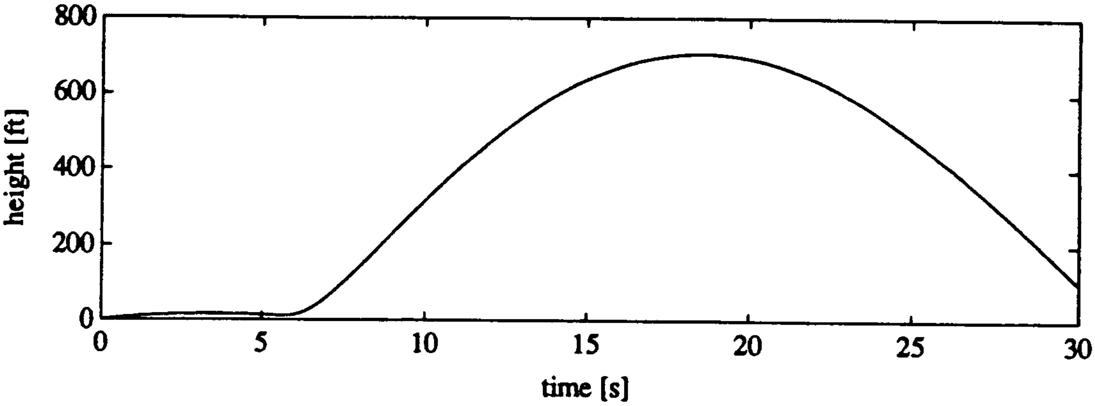

FIGURE 59 Flight Path for Maneuver Two

9 CONCLUSION AND CONTINUED RESEARCH

For the flight path conditions that were examined, the linear adaptive control performed extremely well, and close to the limits of the system (i.e. the time-optimal control [34]). Several important ideas were covered--the need to use a difference of the measurements (either implicitly or explicitly) in the prediction model, the need to modify the RLS algorithm for time-varying parameters, and the need to include feedback in the reference model to account for control limitations. A true test of the adaptive control would be to operate it in conjunction with the lateral portion of the aircraft (as Ostroff did for the variable gain controller in [33]). This would require the lateral portion of the aircraft to be modelled and controlled.

For this particular controller several things still need to be investigated--the effects of noise in the measurements, and the sensitivity of the performance with respect to the model and the flight path (i.e. robustness). Several ideas could be incorporated into the controller presented here to see if the performance is improved. A more advanced reference model could be developed--maybe even an adaptive reference model. Nonlinear terms could be added to the regressor, making the prediction model nonlinear with respect to the measurements, and the effects of an extended or variable horizon cost function for calculating the control could be investigated.

REFERENCES

[1] C. E. Rohrs, M. Athans, L. Valavani, and G. Stein, "Some Design Guidelines for Discrete-time Adaptive Controllers," *Automatica*, Vol. 20, 653-660, 1984.

[2] B. Wittenmark and K. J. Astrom, "Practical Issues in the Implementation of Self-Tuning Control," *Automatica*, Vol. 20, 595-605, 1984.

[3] K. J. Astrom, "Theory and Applications of Adaptive Control - A Survey," *Automatica*, Vol. 19, 471-486, 1983.

[4] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*, Prentice Hall, 1984.

[5] K. J. Astrom and B. Wittenmark, *Adaptive Control*, Addison Wesley Publishing Company, 1989.

[6] Kevin Warwick, *Implementation of Self-Tuning Controllers*, Peter Peregrinus Ltd., 1988.

[7] N. Rao Sripada and D. Grant Fisher, "Improved Least Squares identification," *Int. J. Control*, Vol. 46, 1889-1913, 1987.

[8] M. J. Chen and J. P. Norton, "Estimation Techniques for Tracking Rapid Parameter Changes," *Int. J. Control*, Vol. 45, 1387-1398, 1987.

[9] G. C. Goodwin and E. K. Teoh, "Adaptive Control of a Class of Linear Time Varying Systems," *IFAC Adaptive Systems in Control and Signal Processing*, San Francisco, USA, 1-6, 1983.

[10] G. Davidov, A. Shavit, and Y. Koren, "Estimation of Dynamical-Varying Parameters by the Internal Model Principle," *IEEE Transactions on Automatic Control*, Vol. 37, 498-503, 1992.

[11] R. Fernandez del Busto, P. E. Wellstead, and M. B. Zarrop, "Parameter Transient Tracking for Recursive Estimation," *IEE Proceedings Pt. D*, Vol. 138, 68-74, 1991.

[12] W. E. Moore, "The Parallel Implementation of a Cascade Adaptive Identification Algorithm," *IEEE Transaction of Automatic Control*, Vol. 31, 1074-1076, 1986.

[13] C. Wen and D. J. Hill, "Adaptive Linear Control of Nonlinear Systems," *IEEE Transaction of Automatic Control,* Vol. 35, 1253-1257, 1990.

[14] G. C. Goodwin, H. Elliot, and E. K. Teoh, "Deterministic Convergence of a Self-Tuning Regulator with Covariance Resetting," *IEE Proc. Pt. D,* Vol. 130, 6-8, 1983.

[15] B. D. O. Anderson and R. M. Johnstone, "Adaptive Systems and Time-varying Plants," *Int. J. Control,* Vol. 37, 367-377, 1983.

[16] J. M. Martin-Sanchez, "Adaptive Control for Time-Variant Processes," *Int. J. Control,* Vol. 44, 315-329, 1986.

[17] R. R. Mohler, ed., Semi-Annual Report on "Nonlinear Stability and Control Study of Highly Maneuverable High Performance Aircraft," OSU-ECE Report NASA 9101, Corvallis, OR, August 1991.

[18] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C,* Cambridge University Press, 1990.

[19] D.W. Clarke, A.J.F. Hodgson, and P.S. Tuffs, "Offset Problem and k-incremental predictors in self-tuning control," *IEE Proceedings Pt. D,* Vol. 130, 217-225, 1983.

[20] P.S. Tuffs and D.W. Clarke, "Self-Tuning Control of offset: a unified approach," *IEE Proceedings Pt. D,* Vol. 132, 100-110, 1985.

[21] R. R. Mohler, V. Rajkumar, and R. R. Zakrzewski, "Nonlinear Time-Series-Based Adaptive Control Applications," *Proceedings of the 30th Conference of Decision and Control,* Brighton, England, 2917-2919, 1991.

[22] P. Ioannou and J. Sun, "Theory and Design of Robust Direct and Indirect Adaptive-Control Schemes," *Int. J. Control,* Vol. 47, 775-813, 1988.

[23] R. Ortega, L. Praly, and I. D. Landau, "Robustness of Discrete-Time Direct Adaptive Controllers," *IEEE Transaction on Automatic Control,* Vol. 30, 1179-1187, 1985.

[24] W. R. Cluett, J. M. Martin-Sanchez, S. L. Shah, and D. G. Fisher, "Stable Discrete-Time Adaptive Control in the Presence of Unmodeled Dynamics," *IEEE Transaction on Automatic Control,* Vol. 33, 410-414, 1988.

[25] L. Ljung, *System Identification: Theory for the User,* Prentice Hall, 1987.

[26] S. L. Shah and W. R. Cluett, "Recursive Least Squares Based Estimation Schemes for Self-Tuning Control," *The Canadian Journal of Chemical Engineering*, Vol. 69, 89-96, 1991.

[27] T. R. Fortescue, L. S. Kershenbaum, and B. E. Ydstie, "Implementation of Self-Tuning Regulators with Variable Forgetting Factors", *Automatica*, Vol. 17, 831-835, 1981.

[28] B. E. Ydstie, L. S. Kershenbaum and R. W. Sargent, "Theory and Application of an Extended Horizon Self-Tuning Controller," *AIChe Journal*, Vol. 31, 1771-1780, 1985.

[29] B. E. Ydstie and R. W. H. Sargent, "Convergence and Stability Properties of an Adaptive Regulator with Variable Forgetting Factor," Automatica, Vol. 22, 749-751, 1986.

[30] A. O. Cordero and D. Q. Mayne, "Deterministic Convergence of a Self-Tuning Regulator with Variable Forgetting Factor," *IEE Proceedings Pt. D*, Vol. 128, 19-23, 1981.

[31] S. P. Sanoff and P. E. Wellstead, "Comments of: Implementation of Self Tuning Regulators with Variable Forgetting Factors," *Automatica*, Vol. 19, 345-346, 1983.

[32] J. M. Buffington, A. G. Sparks, and S. S. Banda, "Full Envelope Robust Longitudinal Axis Design of a Fighter Aircraft with Thrust Vectoring," Technical Report, 1992.

[33] A. J. Ostroff, "High-Alpha Applications of Variable-Gain Output Feedback Control," *Journal of Guidance, Control, and Dynamics*, Vol. 15, 491-497, 1992.

[34] R. R. Mohler, ed., Semi-Annual Report on "Nonlinear Stability and Control Study of Highly Maneuverable High Performance Aircraft," OSU-ECE Report NASA 9301, Corvallis, OR, February 1993.

[35] R. R. Mohler, ed., Semi-Annual Report on "Nonlinear Stability and Control Study of Highly Maneuverable High Performance Aircraft," OSU-ECE Report NASA 9202, Corvallis, OR, August 1992.

APPENDIX

APPENDIX

This appendix displays the estimated parameter variations and the prediction error for maneuver one of chapter 8. The prediction error is displayed in figure (60), and the parameter variations are displayed in figures (61) - (64). For the parameter variations the prediction model is described in the following form:

$$\hat{\alpha}(t)=\phi(t)^T\theta(t-1)$$

$$\phi(t)^T=[\alpha(t-2),\alpha(t-3),\alpha(t-4),q(t-2),q(t-3),q(t-4),\\ \delta_{h_{cmd}}(t-1),\delta_{h_{cmd}}(t-2),\delta_{h_{cmd}}(t-3),\\ \delta_{v_{cmd}}(t-1),\delta_{v_{cmd}}(t-2),\delta_{v_{cmd}}(t-3)] \tag{99}$$

Thus, θ_1, θ_2, and θ_3 correspond to the angle of attack. θ_4, θ_5, and θ_6 correspond to the pitch rate. θ_7, θ_8, and θ_9 correspond to the stabilator command, and θ_{10}, θ_{11}, and θ_{12} correspond to the thrust-vectoring angle.

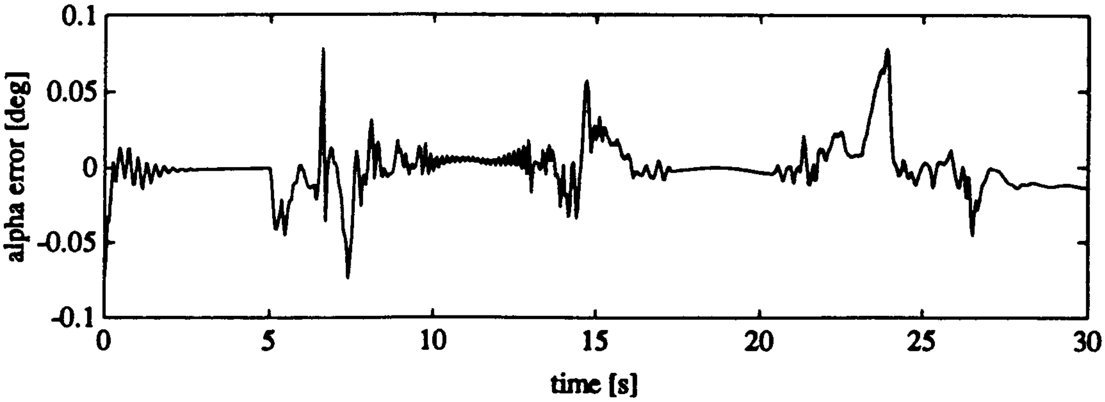

FIGURE 60 Prediction Error for Maneuver One

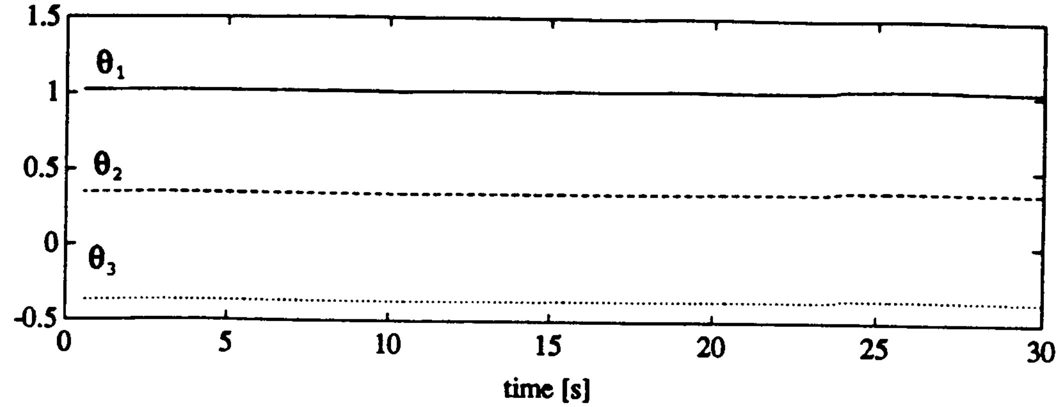

FIGURE 61 Parameter Variation for Maneuver One

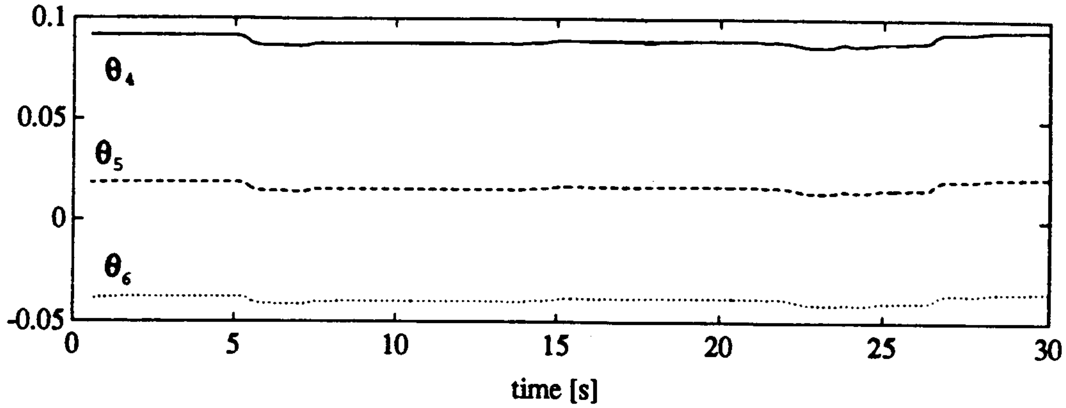

FIGURE 62 Parameter Variation for Maneuver One

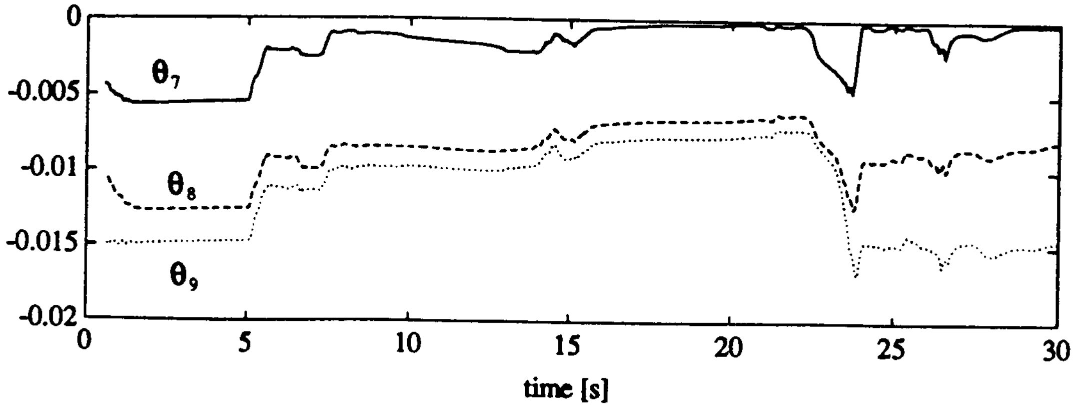

FIGURE 63 Parameter Variation for Maneuver One

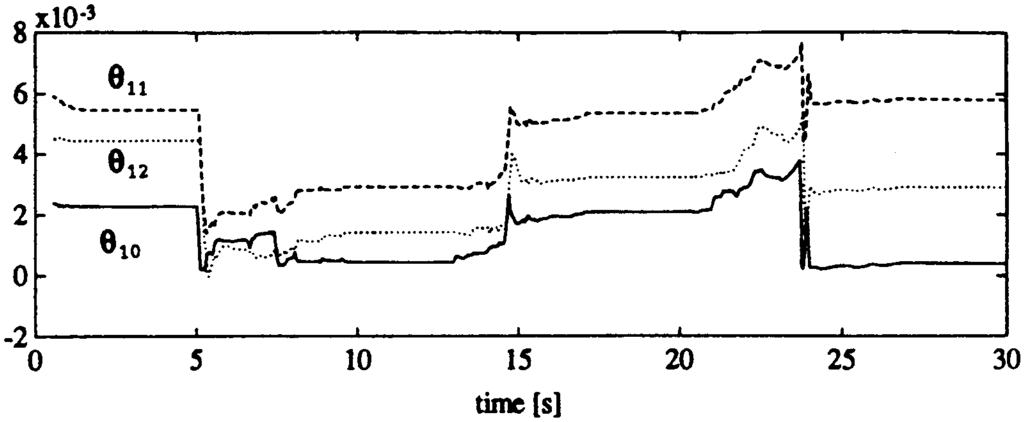

FIGURE 64 Parameter Variation for Maneuver One