3-1994

Direct Reduced Order Mixed H_2/H_∞ Control for the Short Take-Off and Landing-Maneuver Technology Demonstrator (STOL-MTD)

William C. Reigelsperger Jr.

AFIT/GAE/ENY/94M-3

DIRECT REDUCED ORDER MIXED H_2 / H_∞ CONTROL FOR THE SHORT TAKE-OFF AND LANDING / MANEUVER TECHNOLOGY DEMONSTRATOR (STOL/MTD)

THESIS

Presented to the Faculty of the Graduate School of Engineering

of the Air Force Institute of Technology

Air Training Command

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Aeronautical Engineering

William C. Reigelsperger Jr., B.S.

Second Lieutenant, USAF

March 1994

Approved for public release; distribution unlimited

Acknowledgments

There are many people who helped me over the course of this research project to whom I am indebted. The first I would like to thank is my advisor Dr. Ridgely. I benifited greatly from his knowledge and experience. I also owe a lot to the sponsor of this research, David Moorhouse for presenting the idea and always having the answers to my questions. I would also like to thank the people down at WL/FIGC for volunteering the use of computer resources, adding comments about my research, and putting up with my many questions. The two people that helped me get the mixed H_2 / H_- code up and running are Lt. Colonel David Walker, who developed the code, and Captain Jim Luke, who helped me understand how to use it. Finally, I would like to thank my Parents and Grandparents for their support over the past few months.

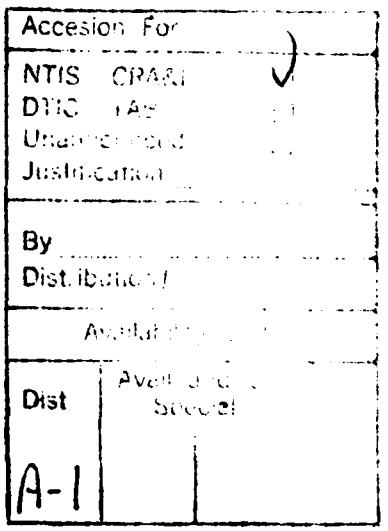


Table of Contents

List of Figures

List of Tables

AFIT/GAE/ENY/94M-3

DIRECT REDUCED ORDER MIXED H_2 / H_∞ CONTROL FOR THE SHORT TAKE-OFF AND LANDING / MANEUVER TECHNOLOGY DEMONSTRATOR (STOL/MTD)

THESIS

William C. Reigelsperger Jr., 2Lt, USAF

AFIT/GAE/ENY/94M-3

Abstract

One of the conclusions from the STOL/MTD program was the need for a multivariable method of designing controllers of low order. This research investigated that problem by studying reduced order mixed H_2 / H_∞ control theory applied to the STOL Landing configuration which employs both thrust vectoring and the use of a canard. Model matching techniques were used to obtain responses that met handling qualities criteria and reduced pilot workload by decoupling pitch rate and velocity commands. The time responses were found through nonlinear simulation and showed that the full order designs did match the ideal models very well and had good noise and wind rejection. Singular value analysis showed that the commands were decoupled very well. The reduced order method was mixed H_2 / H_∞ optimization. A fourth order controller that had good performance was found by using a performance constraint, and a fourth order controller that provided good margins was found using a robustness constraint. A third order controller was also found with a performance constraint. Recommendations for finding a low order controller, with good performance and robustness are given.

DIRECT REDUCED ORDER MIXED H_2 / H_∞ CONTROL FOR THE SHORT TAKE-OFF AND LANDING / MANEUVER TECHNOLOGY DEMONSTRATOR (STOL/MTD)

Chapter 1. Introduction

1.1 Background

Over the past several years, great steps have been taken in the area of multivariable control. The designer of today has methods like H_2, H_∞, μ-synthesis, and others to deal with multivariable problems. Besides being able to handle multivariable problems, these methods also offer benefits such as robust stability and performance. There are, however, drawbacks to using these methods. One of the biggest is the fact that the controllers these methods produce are of order greater than or equal to the design model. When one considers a complicated problem with dynamic weightings, the controllers are of such a high order that actually applying them is not practical or possible. As a result, there has been a great deal of work on finding ways of producing low or reduced order multivariable controllers. Some progress has been made. This work is very important since one could have the best theory in the world but it would not do much good if it could not be applied in a real application. This applicability should be a long term goal of any engineering research. This, and conclusions from the STOL and Maneuver Technology Demonstrator (SMTD), which will be discussed further in the next section, are the motivation behind this thesis. The goal of this thesis is to apply reduced order mixed H_2/H_∞ theory to the SMTD and to analyze the results with respect to both robustness and handling qualities.

1.2 STOL and Maneuver Technology Demonstrator

The SMTD program was a Wright Laboratory development that was structured to:

> ... develop and validate through analysis, experiment and flight test, specific technologies intended to provide current and future high-performance fighters with both STOL capability and enhanced combat mission performance.[1:1]

This program used a modified F-15B. Some of these modifications are shown in Figure 1.1.

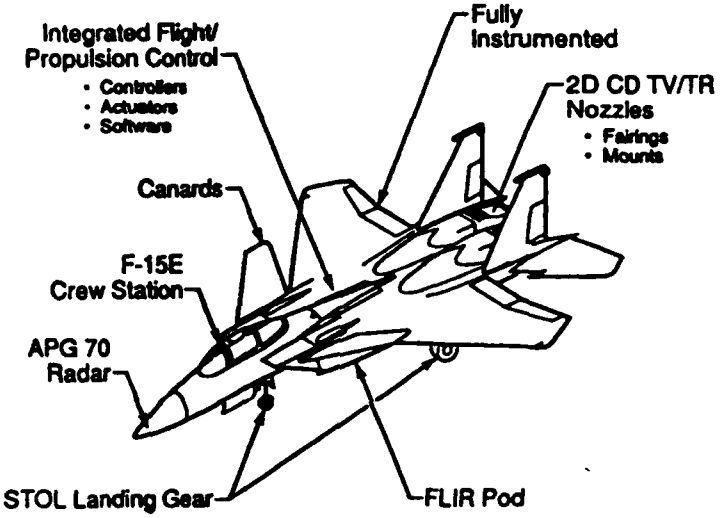


Figure 1.1. SMTD Aircraft Modifications

The main modifications that are important to this research involve the canards, Integrated Flight/Propulsion Control (IFPC) , and the thrust vectoring. Just as the name suggests, the IFPC's purpose is to integrate the aerodynamic control surfaces as well as the engine thrust and vectoring to achieve the required performance which includes good stability and "positive manual control [2:473]".

The handling qualities criteria for the program were obtained from MIL-F-8785C. Throughout the program, however, manned flight simulation showed deviations from this guideline which were incorporated into MIL-F-1797, which

replaced the previous document. More specific information can be found in [1], [2], [3], and [5].

The flight control system was divided into a number of pilot designated modes that accomplished different tasks. Table 1.1 shows a summary of these modes and various aspects involved.

Table 1.1 Mode Summary for SMTD[4:4]

Mode	Mach	Altitude (ft)	Effectors	Regulated Variables
STOL landing (pitch)	0.15 - 0.38	0 - 10,000	Stabilator + canard + flaperon + aileron + vanes Vanes	Pitch rate Velocity
STOL takeoff (pitch)	0.15 - 0.38	0 - 10,000	Stabilator + canard + nozzle	Pitch rate + AOA
STOL rollout (pitch)	0.0 - 0.18	0	Stabilator + canard + bottom vane	Pitch rate
STOL rollout (lateral-directional)	0.0 - 0.18	0	Stabilator + aileron + flaperon Rudder + nose gear + canard	Roll rate Yaw rate
Cruise/combat (pitch)	0.2 - 2.0	0 - 50,000	Stabilator + canard + nozzle	Load factor + pitch rate

The main modes of concern are the Cruise, Combat, and STOL Landing. The Combat mode was designed to improve combat maneuvering and weapons delivery by using the new technologies. Similarly, the Cruise mode was designed to improve cruise capability with the added technologies. There was also a Conventional mode that was used as a baseline to compare to the other modes as well as a backup in case of failure. The mode that was the focus for this research was the STOL Landing (Sland) mode. Not only was this one of the more interesting modes as far the number of effectors, but it was also the mode with the most amount of readily available data. The object of this mode was to produce "precise manual control of flight path trajectory, airspeed, and aircraft attitudes [1:2]," which minimized pilot workload in precise landing. This was accomplished through decoupling stick and throttle inputs. In other

words, stick inputs would not affect airspeed and throttle inputs would not affect flight path, allowing the pilot to worry only about the flight path with the stick once a desired approach speed is achieved [1:3].

The actual control laws were developed to allow a touchdown in an area that is 60 ft long and 20 ft wide. The approach is done by spooling the engine to 100% and closing off the rear nozzle. This forces the exhaust out the upper and lower vanes which can be vectored both fore and aft. This allows the aircraft to have precise control of airspeed and immediate thrust reversing upon touchdown. More information on this can be found in [3].

The Statement of Work on the SMTD program encouraged, but did not limit, the study to using multivariable control techniques. The two contractors were McDonnell Douglas Aerospace (McAir), who used a classical approach, and Honeywell, who used a multivariable approach. The classical technique used was an Inverse Equivalent System method that used theories similar to equivalent system fits [1:5]. This method was applied to various modes of the flight, but this thesis is more concerned with Honeywell's multivariable approach. Honeywell used the well known LQG/LTR method. This method, like many other multivariable methods, produces a compensator of order that is often not practical. Such was the case here, and modal truncation was used followed by balancing the compensator to produce a practical controller. The final reduction of the compensator brought the order from fifth to second order. There was some degradation in the decoupling of the system and care was taken at each step to check the margins and closed-loop transfer-functions [1:8]. Although pilot simulation showed that the reduced order compensator was satisfactory, compensator reduction is a bit of an "art" which is less desirable than a direct method to find reduced order compensators.

One of the conclusions obtained by comparing the classical and multivariable approaches was that, for the simple systems, there were no real benefits to using the multivariable design. However, for the complex modes with many effectors, a multivariable method was preferred [1:9]. Considering that more and more aircraft are relying on such things as fly-by-wire and thrust vectoring, and with the advent of new effectors (vortex control), the importance of reduced order (practical) multivariable design techniques increases tremendously. This thesis will attempt to address this aspect of control design by first applying full order H_2 and H_∞ theory as a baseline and then reduced order mixed H_2/H_∞ theory to the Sland mode.

1.3 Previous Work

This section will look into five approaches currently being used in the area of reduced or fixed order control theory. The first is a look into parameterizing low order H_∞ controllers [6]. The second uses a differential game approach [7]. The third deals with minimal order controllers based on the bounded real lemma [8]. The fourth is a Lagrange multiplier approach [10] and the last is a mixed H_2 / H_∞ approach [11].

Iwasaki and Skelton [6] present a parameterization of stabilizing controllers that satisfies a specific H_∞ norm bound and has order less than or equal to the plant. This is done by taking a Lyapunov approach to the H_∞ control problem and applying theories that have been developed in covariance control. (see references 2,10,11,13,21,and 25 in [7]) The Riccati equation in the bounded real lemma (see reference 24 in [7]) is considered to be a special case of the Lyapunov equation

$$AP + PA^T + W = 0$$

where W is a given symmetric nonnegative definite matrix. The system is stabilized iff P is nonnegative definite and (A,W) is stabilizable. The matrix P is referred to as a "Lyapunov matrix" in [6]. This P matrix specifies the order of the controller in that its dimension is (np+nc), where np is the order of the plant and nc is the order of the controller. Physical importance is added through W using covariance theory.

The connection between the bounded real lemma Riccati equation and the Lyapunov equation above occurs when W includes a quadratic in P. In this case a nonnegative definite solution also meets an H_∞ norm bo[illegible] P is then referred to as an "H_∞ Lyapunov matrix" and the parameterization is based on this P.

Three main advantages are presented in [6], but two of them are mostly advantages to the full order case. The reduced order case results in two coupled Riccati equations that are similar to those in the fixed order mixed LQG / H_∞ problem. Although [6] does not go into detail about solving these equations, it does, however, show that under this parameterization, low order controllers have an observer based structure, which is useful insight into the problem.

Sweriduk and Calise [7] present a method that improves the computation of the fixed order problem. A differential game approach is taken to solve the fixed order H_∞ problem. Sweriduk and Calise derive a performance index expressed in terms of the states. From this, three necessary conditions (equations 30-32 in [7:3]) are found from the resulting Lagrangian (see Section 2.3 for more on Lagrange theory). The solution is obtained by a conjugate gradient method and iteratively reduces the infinity-norm γ until it gets close to optimal. To reduce the number of parameters in the search, the controller is put into canonical form. [7] also allows the controller to have a nonzero D

term, so that it is not restricted to being strictly proper. This procedure shows good results for the examples presented. Two issues of concern that were presented were the fact that it remained to be shown whether solutions to the necessary conditions always exist (given assumptions used) and computational time, since the gradient search slowed down near the minimum.

The third methodology, by Hsu, Yu, Yeh, and Banda, is presented in [8] and is somewhat different from the last in that it is not computationally based. Low order controllers were found to be observer-based in [6] and this is used in the development of [8]. A Luenberger observer-based controller is applied in light of the bounded real lemma. The minimal order of this controller is classically known to be n-p, where n is the number of states in the design model including any weightings, and p is the number of measurements. The result is a stabilizing controller that yields a closed loop infinity norm ($\|T_{ed}\|_\infty$) less than some given γ. The main advantage to this method is that the primary equations are two uncoupled Riccati equations and a Sylvester equation. This means that the solution process is easy compared to solving two coupled Riccati equations as in [6]. The disadvantage is that for a large problem the order of the controller will not be reduced significantly unless there are a large number of measurements, which may not necessarily be the case. More information on the application of this method is given in [9].

The fourth method, by DeShetler and Ridgely [10], is similar to the second in that it uses Lagrange multiplier theory. Using a controller of arbitrary but fixed order, the closed loop state space is formed. From this, a minimization problem is set up. The performance index is simply the entropy of the closed loop system, which is constrained by a Riccati equation based on the bounded real lemma. Given γ, the performance index is minimized by a gradient technique involving the necessary conditions from the Lagrangian. By

iterating on γ, the optimal H_∞ level is approached to a given tolerance. Like other computational methods based on gradients, it was found that, as the solution approached the minimum, the gradient search got slower. Also, the actual numerics became less stable as the optimal solution is approached. An interesting observation was that the minimum γ achieved by a strictly proper controller of order nc is found to be attained with a relative degree zero (the D term in the controller state space, D_C, is not equal to 0) controller of order nc-1, although [10] did not prove this was always true.

The final approach, by Walker and Ridgely, is given in [11] and is the basis for the research in this thesis. This method of reduced order control uses a mixed H_2 / H_∞ performance criterion. The basic idea behind this approach is that a function is minimized given both H_2 and H_∞ objectives. The resulting compensator order may be low in that it may be chosen to be any order desired. If the chosen compensator order is greater than or equal to the order of the H_2 part of the problem, some very strong properties of the solution are known. In the case where a relatively simple H_2 problem yet complicated H_∞ constraints are developed, choosing the compensator order equal to the H_2 order results in a reduced order controller. A more in-depth discussion of this method and how it applies to this research is given in the next chapter.

1.4 Overview

This thesis is broken into six main chapters. Following this introductory chapter, Chapter 2 is a theory section that goes over the theory used in this work. Chapter 3 goes through the development of the full and reduced order design models. The plant, Sland flight, and ideal handling qualities model are also discussed. The results that were obtained are discussed in Chapter 4 and

Chapter 5. Conclusions and recommendations based on these results are presented in the sixth chapter.

Chapter 2. Theory

2.1 H_2 *Optimization*

Before attempts to find a reduced order mixed controller were made, a full model H_2 controller was found. To provide a better understanding of H_2, this part of the section examines the H_2 problem based on the discussion in [12:68-74]. See [12] as well as [13] and [14] for more details. Consider Figure 2.1 which shows the general H_2 problem.

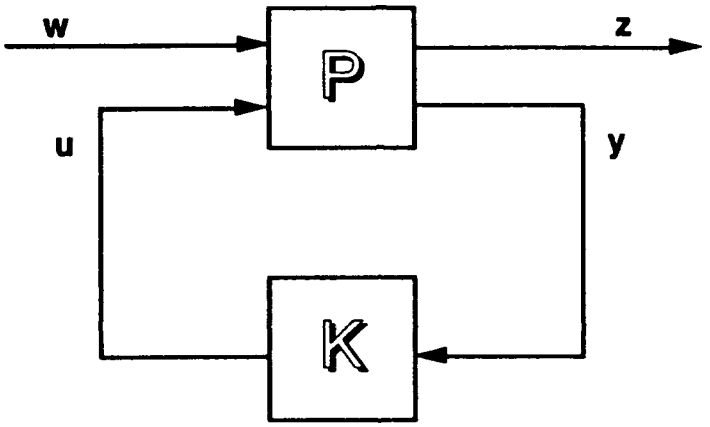


Figure 2.1. General H_2 feedback problem

The plant P can be written as

$$P = \begin{bmatrix} P_{zw} & P_{zu} \\ P_{yw} & P_{yu} \end{bmatrix}$$

or,

$$z = P_{zw}w + P_{zu}u$$

$$y = P_{yw}w + P_{yu}u$$

The output is shown as z. The feedback measurements, y, are fed into the controller which produces the control commands u. The exogenous input w is assumed to be a zero-mean white Gaussian noise of unit intensity. It is desired

to minimize the energy or two-norm of the controlled output z; that is, find an admissible (internally stabilizing) K(s) such that $\| z \|_2$ is minimized.

$$\inf_{K \text{ admissible}} \| z \|_2 = \inf_{K \text{ admissible}} \| T_{zw} \|_2$$

$$= \inf_{K \text{ admissible}} \| P_{zw} + P_{zu}K[I - P_{yu}K]^{-1}P_{yw} \|_2$$

where

$$\| T_{zw} \|_2^2 \equiv \frac{1}{2\pi} \int_{-\infty}^{\infty} tr[T_{zw}^*(j\omega)T_{zw}(j\omega)]\, d\omega$$

P can be written in state space form as

$$\dot{x} = Ax + B_w w + B_u u$$

$$z = C_z x + D_{zw} w + D_{zu} u$$

$$y = C_y x + D_{yw} w + D_{yu} u$$

The following assumptions are made:

1. $D_{zw} = 0$
2. $D_{yu} = 0$
3. (A, B_u) stabilizable & (C_y, A) detectable
4. $D_{zu}^T D_{zu}$ full rank ; $D_{yw} D_{yw}^T$ full rank
5. $\begin{bmatrix} A-j\omega I & B_u \\ C_z & D_{zu} \end{bmatrix}$ has full column rank for all ω
6. $\begin{bmatrix} A-j\omega I & B_w \\ C_y & D_{yw} \end{bmatrix}$ has full row rank for all ω

To make the formulation of the resulting controller simpler, the scalings S_u and S_y are added to the plant. These are shown in Figure 2.2 and are simply added into the controller after the synthesis is complete.

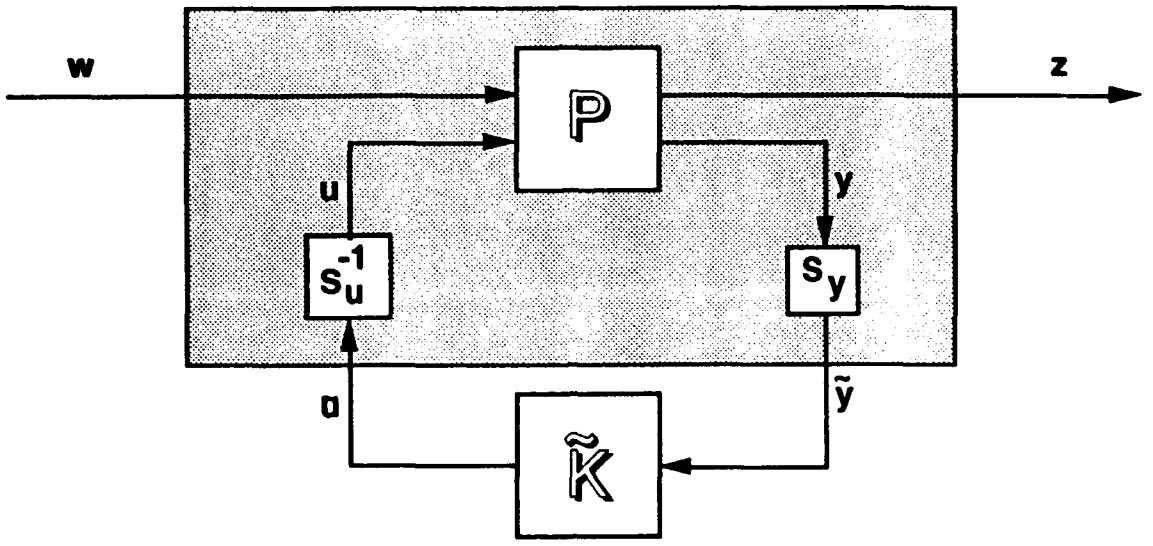


Figure 2.2. H_2 Feedback Problem with Scalings

The scaled equations are

$$u = S_u^{-1}\,\tilde{u}$$

$$\tilde{y} = S_y\,y$$

where S_u and S_y are such that

$$S_u^T S_u = D_{zu}^T D_{zu}$$

$$S_y^{-1}(S_y^{-1})^T = D_{yw} D_{yw}^T$$

If $A \in R^{nxn}$ is symmetric positive definite, then there exists an upper triangular $S \in R^{nxn}$ with positive diagonal entries such that $A = S^T S$. This is known as a *Cholesky decomposition*, which may be used to find S_u and S_y. The "new" plant state and output equations become

$$\dot{x} = Ax + B_w w + \tilde{B}_u \tilde{u}$$

$$z = C_z x + \tilde{D}_{zu} \tilde{u}$$

$$\tilde{y} = \tilde{C}_y x + \tilde{D}_{yw} w$$

where

$$\tilde{B}_u = B_u S_u^{-1} \qquad \tilde{C}_y = S_y C_y$$

$$\tilde{D}_{zu} = D_{zu} S_u^{-1} \qquad \tilde{D}_{yw} = S_y D_{yw}$$

The optimal compensator is unique, $\|T_{zw}\|_2 = \alpha_o$, and is given by

$$\tilde{K}_{opt}(s) = \begin{bmatrix} A_J & K_f \\ -K_c & 0 \end{bmatrix}$$

$$A_J = A - K_f \tilde{C}_y - \tilde{B}_u K_c$$

$$K_c = \tilde{B}_u^T X_2 + \tilde{D}_{zu}^T C_z$$

$$K_f = Y_2 \tilde{C}_y^T + B_w \tilde{D}_{yw}^T$$

Here, X_2 and Y_2 are the real, unique, symmetric positive semidefinite solutions to the algebraic Riccati equations:

$$(A - \tilde{B}_u \tilde{D}_{zu}^T C_z)^T X_2 + X_2 (A - \tilde{B}_u \tilde{D}_{zu}^T C_z) - X_2 \tilde{B}_u \tilde{B}_u^T X_2 + \hat{C}_z^T \hat{C}_z = 0$$

where

$$\hat{C}_z = (I - \tilde{D}_{zu} \tilde{D}_{zu}^T) C_z$$

and

$$(A - B_w \tilde{D}_{yw}^T \tilde{C}_y) Y_2 + Y_2 (A - B_w \tilde{D}_{yw}^T \tilde{C}_y)^T - Y_2 \tilde{C}_y^T \tilde{C}_y Y_2 + \hat{B}_w \hat{B}_w^T = 0$$

where

$$\hat{B}_w = B_w (I - \tilde{D}_{yw}^T \tilde{D}_{yw})$$

After the scalings are put back into the compensator, the resulting $K2_{opt}(s)$ is given by

$$K2_{opt}(s) = \begin{bmatrix} A_J & \tilde{K}_f \\ -\tilde{K}_c & 0 \end{bmatrix}$$

where

$$\tilde{K}_f = K_f S_y \qquad \tilde{K}_c = S_u^{-1} K_c$$

and can be seen in Figure 2.3.

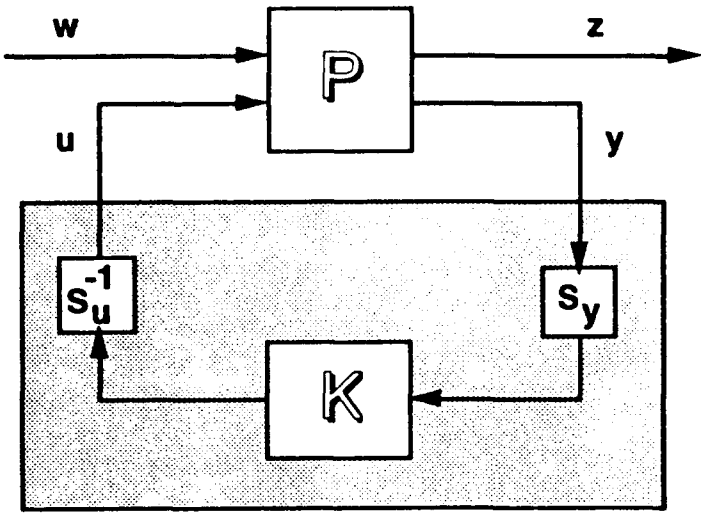


Figure 2.3. H_2 Optimal Diagram with Scalings to Compensator

2.2 H_∞ *Optimization*

The other form of optimization that is dealt with in this work is H_∞ optimization. The theory behind this is given below and is taken from [12:74-81].

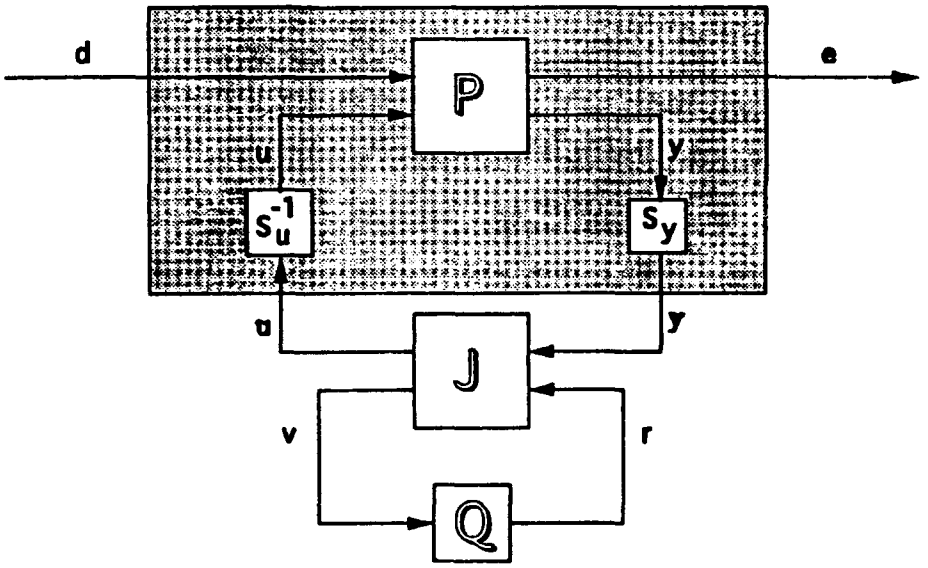


Figure 2.5. H_∞ Feedback Problem

Consider the problem shown in Figure 2.5. In this set-up d is assumed to be an exogenous input with unknown but bounded energy. The problem is to find an internally stabilizing K(s) that minimizes the energy of the output e. This can be written as:

$$\inf_{K \text{ stabilizing}} \|T_{ed}\|_\infty = \inf_{K \text{ stabilizing}} \|P_{ed} + P_{eu}K(I - P_{yu}K)^{-1}P_{yd}\|_\infty$$

where

$$\|T_{ed}\|_\infty = \sup_{\omega} \bar{\sigma}[T_{ed}] \quad \text{and} \quad P = \begin{bmatrix} P_{ed} & P_{eu} \\ P_{yd} & P_{yu} \end{bmatrix}$$

Given the state space for P

$$\dot{x} = Ax + B_d d + B_u u$$

$$e = C_e x + D_{ed} d + D_{eu} u$$

$$y = C_y x + D_{yd} d + D_{yu} u$$

the following assumptions are made:

1. $D_{ed}=0$
2. $D_{yu}=0$
3. (A,B_u) is stabilizable and (C_y,A) is detectable
4. $D_{eu}^T D_{eu}$ and $D_{yd} D_{yd}^T$ have full rank
5. $\begin{bmatrix} A - j\omega I & B_u \\ C_e & D_{eu} \end{bmatrix}$ has full column rank for all ω
6. $\begin{bmatrix} A - j\omega I & B_d \\ C_y & D_{yd} \end{bmatrix}$ has full row rank for all ω

The scalings S_u^{-1} and S_y shown in Figure 2.5 make the formulation of the resulting controller simpler. They are then incorporated into the controller after the synthesis is complete.

Finding solutions to the optimal H_∞ problem is based on the parameterization of all suboptimal compensators and iterating the norm down to arbitrarily close to optimal. Let

$K(s)=F_l(J(s),Q(s))$; family of all admissible compensators that satisfy $\|T_{ed}\|_\infty < \gamma$

where

$$J(s)=\begin{bmatrix} J_{uy} & J_{ur} \\ J_{vy} & J_{vr} \end{bmatrix}=\left[\begin{array}{c|cc} A_J & K_f & K_{fl} \\ \hline -K_c & I & 0 \\ K_{cl} & 0 & I \end{array}\right]$$

and

$$A_j = A - K_f\tilde{C}_y - \tilde{B}_u K_c + \gamma^{-2}Y_\infty C_e^T(C_e - \tilde{D}_{eu}K_c)$$

$$K_c = (\tilde{B}_u^T X_\infty + \tilde{D}_{eu}^T C_e)(I - \gamma^{-2}Y_\infty X_\infty)^{-1}$$

$$K_f = Y_\infty \tilde{C}_y^T + B_d\tilde{D}_{yd}^T$$

$$K_{cl} = -(\gamma^{-2}\tilde{D}_{yd}B_d^T X_\infty + \tilde{C}_y)(I - \gamma^{-2}Y_\infty X_\infty)^{-1}$$

$$K_{fl} = Y_\infty C_e^T\tilde{D}_{eu} + \tilde{B}_u$$

where the tilde matrices are those that include the scaling in the state space. X_∞ and Y_∞ are the solutions to the algebraic Riccati equations.

$$(A - \tilde{B}_u\tilde{D}_{eu}^T C_e)^T X_\infty + X_\infty(A - \tilde{B}_u\tilde{D}_{eu}^T C_e) + X_\infty(\gamma^{-2}B_dB_d^T - \tilde{B}_u\tilde{B}_u^T)X_\infty + \hat{C}_e^T\hat{C}_e = 0$$

where $\hat{C}_e = (I - \tilde{D}_{eu}\tilde{D}_{eu}^T)C_e$

and

$$(A - B_d\tilde{D}_{yd}^T\tilde{C}_y)^T Y_\infty + Y_\infty(A - B_d\tilde{D}_{yd}^T\tilde{C}_y) + Y_\infty(\gamma^{-2}C_eC_e^T - \tilde{C}_y^T\tilde{C}_y)Y_\infty + \hat{B}_d\hat{B}_d^T = 0$$

where $\hat{B}_d = B_d(I - \tilde{D}_{yd}^T\tilde{D}_{yd})$

Finally,

$Q \in RH_\infty$; $\|Q\|_\infty < \gamma$ *which means that the optimal is not unique*

Given the Hamiltonian matrices in [12:79-80], the parameterization above is valid if and only if the following are satisfied:

1. $H_X \in \text{dom(Ric)}$ with $X_- = \text{Ric}(H_X) \geq 0$
2. $H_Y \in \text{dom(Ric)}$ with $Y_- = \text{Ric}(H_Y) \geq 0$
3. $\rho(Y_- X_-) < \gamma^2$

See [12] for more discussion on Hamiltonians and dom(Ric). This forms the basis for finding H_- controllers. If one of these conditions is not satisfied, then γ must be increased. Therefore, one can get arbitrarily close to the optimal by iteration. See [14] for more on this subject.

2.3 Mixed H_2 / H_- Optimization

After designing a full order controller, this thesis examined the application of mixed H_2 / H_- theory to find a low order solution. The basic idea is to push the mixed problem close to the optimal H_- solution (for good tracking and/or margins) while retaining the low order and noise and disturbance rejection properties of the H_2 problem. The theory behind this is given in [11] and is discussed below.

For the mixed H_2 / H_- problem we have the system shown in Figure 2.5:

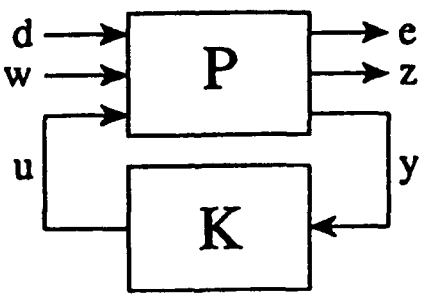


Figure 2.5. Mixed H_2 / H_- System

where,

$$P = \left[\begin{array}{c|ccc} A & B_d & B_w & B_u \\ \hline C_e & D_{ed} & D_{ew} & D_{eu} \\ C_z & D_{zd} & D_{zw} & D_{zu} \\ C_y & D_{yd} & D_{yw} & D_{yu} \end{array}\right]$$

In this system, P is the entire weighted system consisting of both the H_2 and H_∞ parts of the problem. The signals w and z are the inputs and controlled outputs of the H_2 part of the problem, respectively. Likewise, d and e are the inputs and outputs of the H_∞ problem, respectively. It is not necessary to have any correlation between e and z or d and w. W is assumed to be zero mean, unit intensity, white Gaussian noise and d is of bounded energy. The measured output is y and the control input is u, which are the same for both the H_2 and H_∞ parts of the problem.

The state space equations for the H_2 problem are:

$$\dot{x}_2 = A_2 x_2 + B_w w + B_{u2} u$$

$$z = C_z x_2 + D_{zw} w + D_{zu} u$$

$$y = C_{y2} x_2 + D_{yw} w + D_{yu} u$$

The state space equations for the H_∞ problem are:

$$\dot{x}_\infty = A_\infty x_\infty + B_d d + B_{u\infty} u$$

$$e = C_e x_\infty + D_{ed} d + D_{eu} u$$

$$y = C_{y\infty} x_\infty + D_{yd} d + D_{yu} u$$

The mixed H_2 / H_∞ problem is: Find an internally stabilizing K(s) that achieves

$$\inf_{K\ \text{stabilizing}} \|T_{zw}\|_2 \quad \text{subject to } \|T_{ed}\|_\infty \le \gamma$$

where the closed loop transfer functions are

$$T_{zw} = C_z(sI - A_2)^{-1} B_w + D_{zw}$$

$$T_{ed} = C_e(sI - A_\infty)^{-1} B_d + D_{ed}$$

These transfer functions are found from closing the loop for each system, using the controller state space.

$$\dot{x}_c = A_c x_c + B_c y$$
$$u = C_c x_c$$

One can see that there is no D_C term in the controller. This is due to the fact that for $\|T_{zw}\|_2$ to be finite, D_C must be zero.

The assumptions made to solve this problem are:

1. $D_{zw} = 0$
2. $D_{yu} = 0$
3. (A_2, B_{u2}) stabilizable (C_{y2}, A_2) detectable
4. $(A_\infty, B_{u\infty})$ stabilizable $(C_{y\infty}, A_\infty)$ detectable
5. $D_{zu}^T D_{zu}$ full rank and $D_{yw}^T D_{yw}$ full rank
6. $\begin{bmatrix} A_\infty - j\omega I & B_{u\infty} \\ C_e & D_{eu} \end{bmatrix}$ has full column rank for all ω
7. $\begin{bmatrix} A_\infty - j\omega I & B_d \\ C_{y\infty} & D_{yd} \end{bmatrix}$ has full row rank for all ω
8. $\begin{bmatrix} A_2 - j\omega I & B_{u2} \\ C_z & D_{zu} \end{bmatrix}$ has full column rank for all ω
9. $\begin{bmatrix} A_2 - j\omega I & B_w \\ C_{y2} & D_{yw} \end{bmatrix}$ has full row rank for all ω

There are no restrictions on D_{ed}, D_{eu}, or D_{yd}. The closed loop system is written as

$$\dot{x}_2 = A_2 x_2 + B_w w$$
$$z = C_z x_2$$
$$\dot{x}_\infty = A_\infty x_\infty + B_d d$$
$$e = C_e x_\infty + D_{ed} d$$

In order to be a solution to the mixed problem the following must be satisfied:

1. A_∞ and A_2 are stable
2. $\|T_{ed}\|_\infty \le \gamma$ for some given $\gamma > \gamma_0$

3. $\|T_{zw}\|_2$ is minimized

With this in mind, and using Theorem 1 in [11], the problem can be rewritten as: find K(s) that minimizes

$$J(A_c, B_c, C_c) = tr[Q_2 C_z^T C_z]$$

where Q_2 is the real, symmetric, positive semidefinite solution to the Lyapunov equation

$$A_2 Q_2 + Q_2 A_2^T + B_w B_w^T = 0$$

and such that there exists a real, symmetric, positive semidefinite solution to the Riccati equation

$$A_- Q_- + Q_- A_-^T + (Q_- C_e^T + B_d D_{ed}^T) R^{-1} (Q_- C_e^T + B_d D_{ed}^T)^T + B_d B_d^T = 0$$

where $R = (\gamma^2 I - D_{ed} D_{ed}^T) > 0$

This is a minimization problem with two equality constraints. One way to solve this is by using the method of Lagrange multipliers. With this method, a function F can be minimized with an equality constraint G=0 by setting up the Lagrangian

L=F + Gλ; where λ is a Lagrange multiplier

First order necessary conditions are found by taking

$$\frac{\partial L}{\partial x_i} = 0; \quad \text{where } x_i \text{ are the unknowns in the problem}$$

and solving for the x_i's, which include λ.

For this problem, the Lagrangian is (see [11])

$$\begin{aligned} L = {} & tr[Q_2 C_z^T C_z] + tr\{[A_2 Q_2 + Q_2 A_2^T + B_w B_w^T] X\} \\ & + tr\{[A_- Q_- + Q_- A_-^T + (Q_- C_e^T + B_d D_{ed}^T) R^{-1} (Q_- C_e^T + B_d D_{ed}^T)^T \\ & + B_d B_d^T] Y\} \end{aligned} \tag{2.1}$$

with the symmetric Lagrange multiplier matrices X and Y.

The following definitions are made when finding the first order necessary conditions:

$$M = R^{-1}D_{ed}D_{yd}^T$$

$$P_1 = D_{eu}^T R^{-1} D_{ed} B_d^T$$

$$P_2 = D_{eu}^T M B_c^T$$

$$Q_2 = \begin{bmatrix} Q_1 & Q_{12} \\ Q_{12}^T & Q_2 \end{bmatrix}$$

$$X = \begin{bmatrix} X_1 & X_{12} \\ X_{12}^T & X_2 \end{bmatrix}$$

$$Q_{uu} = \begin{bmatrix} Q_a & Q_{ab} \\ Q_{ab}^T & Q_b \end{bmatrix}$$

$$Y = \begin{bmatrix} Y_1 & Y_{12} \\ Y_{12}^T & Y_2 \end{bmatrix}$$

$$B_w B_w^T = \begin{bmatrix} B_w \\ B_c D_{yw} \end{bmatrix} \begin{bmatrix} B_w^T & D_{yw}^T B_c^T \end{bmatrix} = \begin{bmatrix} V_1 & V_{12} B_c^T \\ B_c V_{12}^T & B_c V_2 B_c^T \end{bmatrix}$$

$$B_d (D_{ed}^T R^{-1} D_{ed} + I) B_d^T = \begin{bmatrix} B_d \\ B_c D_{yd} \end{bmatrix} (D_{ed}^T R^{-1} D_{ed} + I) \begin{bmatrix} B_d^T & D_{yd}^T B_c^T \end{bmatrix}$$

$$= \begin{bmatrix} V_a & V_{ab} B_c^T \\ B_c V_{ab}^T & B_c V_b B_c^T \end{bmatrix}$$

$$C_z^T C_z = \begin{bmatrix} C_z^T \\ C_c^T D_{zu}^T \end{bmatrix} \begin{bmatrix} C_z & D_{zu} C_c \end{bmatrix} = \begin{bmatrix} R_1 & R_{12} C_c \\ C_c^T R_{12}^T & C_c^T R_2 C_c \end{bmatrix}$$

$$C_e^T R^{-1} C_e = \begin{bmatrix} C_e^T \\ C_c^T D_{zu}^T \end{bmatrix} R^{-1} \begin{bmatrix} C_e & D_{zu} C_c \end{bmatrix} = \begin{bmatrix} R_a & R_{ab} C_c \\ C_c^T R_{ab}^T & C_c^T R_b C_c \end{bmatrix}$$

With these definitions, the first order necessary conditions for the Lagrangian are:

$$\frac{\partial L}{\partial A_c} = X_{12}^T Q_{12} + X_2 Q_2 + Y_{12}^T Q_{ab} + Y_2 Q_b = 0$$

$$\begin{aligned}\frac{\partial L}{\partial B_c} = {} & X_{12}^T Q_1 C_{y2}^T + X_2 Q_{12}^T C_{y2}^T + X_{12}^T V_{12} + X_2 B_c V_2 + Y_{12}^T Q_a C_{y-}^T \\ & + Y_2 Q_{ab}^T C_{y-}^T + Y_{12}^T V_{ab} + Y_2 B_c V_b \\ & + (Y_{12}^T Q_a + Y_2 Q_{ab}^T) C_e^T M \\ & + (Y_{12}^T Q_{ab} + Y_2 Q_b) C_e^T D_{eu}^T M = 0\end{aligned}$$

$$\begin{aligned}\frac{\partial L}{\partial C_c} = {} & B_{u2}^T X_1 Q_{12} + B_{u2}^T X_{12} Q_2 + R_{12}^T Q_{12} + R_2 C_c Q_2 + B_{u-}^T Y_1 Q_{ab} \\ & + B_u^T Y_{12} Q_b + R_{ab}^T Q_a Y_1 Q_{ab} + R_{ab}^T Q_{ab} Y_{12}^T Q_{ab} + R_{ab}^T Q_a Y_{12}^T Q_b \\ & + R_{ab}^T Q_{ab} Y_2 Q_b + R_b C_c Q_{ab}^T Y_1 Q_{ab} + R_b C_c Q_b Y_{12}^T Q_{ab} \\ & + R_b C_c Q_{ab}^T Y_{12} Q_b + R_b C_c Q_b Y_2 Q_b + P_1 (Y_1 Q_{ab} + Y_{12} Q_b) \\ & + P_2 (Y_{12}^T Q_{ab} + Y_2 Q_b) = 0\end{aligned}$$

$$\frac{\partial L}{\partial X} = A_2 Q_2 + Q_2 A_2^T + B_w B_w^T = 0$$

$$\frac{\partial L}{\partial Q_2} = A_2^T X + X A_2 + C_z^T C_z = 0$$

$$\begin{aligned}\frac{\partial L}{\partial Y} = {} & A_- Q_- + Q_- A_-^T \\ & + (Q_- C_e^T + B_d D_{ed}^T) R^{-1} (Q_- C_e^T + B_d D_{ed}^T)^T \\ & + B_d B_d^T = 0\end{aligned} \qquad (2.2)$$

$$\begin{aligned}\frac{\partial L}{\partial Q_-} = {} & (A_- + B_d D_{ed}^T R^{-1} C_e + Q_- C_e^T R^{-1} C_e)^T Y \\ & + Y (A_- + B_d D_{ed}^T R^{-1} C_e + Q_- C_e^T R^{-1} C_e) = 0\end{aligned} \qquad (2.3)$$

Unfortunately, these equations have not been solved analytically. Equation (2.1), however, shows that for Y=0 the solution is not on the H_- constraint boundary since the third term in (2.1) will be zero for any solution to the H_- Riccati equation in that third term. Equation (2.2) implies that if $\mathrm{Re}[\lambda_i(A_- + B_d D_{ed}^T R^{-1} C_e + Q_- C_e^T R^{-1} C_e)] = 0$ for some i, the solution is on the

boundary of the H_- constraint and that Q_- is the neutrally stabilizing solution to (2.3).

Some interesting aspects of the problem can be seen in Figure 2.6.

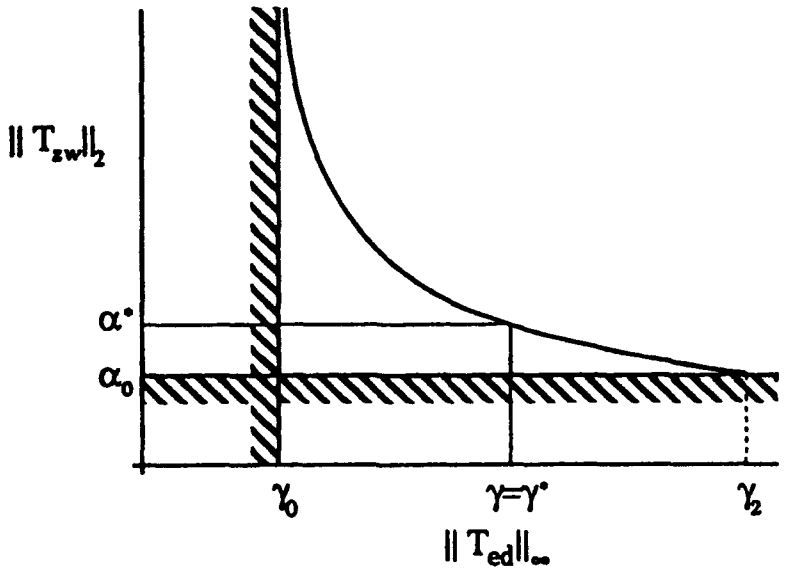


Figure 2.6. Mixed Boundary Plot

The optimal H_2 norm of T_{zw} is α_0 and results in an infinity-norm of T_{ed} equal to γ_2. Since there is no controller of any order below γ_0 (the optimal $\|T_{ed}\|_-$), the mixed problem will not have a solution with γ below γ_0. Therefore, γ chosen between γ_0 and γ_2 results in an active constraint. As one can see, the 2-norm of T_{zw} at the optimal H_- controller is usually infinite. By posing a simple (statically weighted) H_2 problem, one can get a reduced order controller that is close to α_0 and γ_0 by moving along the curve to the "corner". This is the basic idea behind the reduced order method examined in this research.

As mentioned earlier, the first order necessary conditions have no analytical solution. Therefore, this problem must be attacked numerically. For the method used in this research, the following approach is taken.

Define the performance index

$$J_\gamma = \|T_{zw}\|_2^2 + \lambda(1 - \frac{\|T_{ed}\|_\infty}{\gamma})^2$$

where λ is a penalty on the error between the desired and actual infinity-norm of the transfer function. In minimizing this index, the two-norm of the H_2 problem will be minimized and the infinity-norm of the constraint will try and match the desired infinity-norm value γ. For each value of γ a new controller is found by minimizing J_γ. Let

$$X = \left[a1^T \cdots an^T \quad b1^T \cdots bp^T \quad c1^T \cdots cm^T \right]^T$$

where ai, bi, and ci are the columns of A_C, B_C, and C_C. Given this, the first order necessary conditions are:

$$\frac{\partial J_\gamma}{\partial x_i} = 0 \quad \text{for } i = 1,\ldots,(n^2 + n*p + n*m)$$

$$= \frac{\partial \|T_{zw}\|_2^2}{\partial x_i} + \frac{\partial \lambda(1 - \frac{\|T_{ed}\|_\infty}{\gamma})^2}{\partial x_i} \qquad (2.4)$$

where x_i represents the i th element of the solution vector X.

By using the fact that $\|T_{zw}\|_2^2 = tr\{Q_2 C_z^T C_z\}$ the first part of the right hand side of equation (2.4) can be found analytically. In the past, the second required the use of numerical computations of modified central differences which often led to numerical difficulties. A method of analytically calculating the derivative of an infinity-norm was found in [15] which not only helped the accuracy of the calculations but decreased the numerical difficulties associated with using Hamiltonians and increased the speed of the optimization.

A Davidon-Fletcher-Powell(DFP) routine is used to find the X that minimizes the performance index. One improvement over previous mixed H_2/H_∞ solvers [11:1,2] is that a parabola fit is used instead of a bisection

method in the search. This results in lower computational effort. Another aspect of the solver is that, if an unstable solution is found, the step size is decreased and the search is continued. This allows the use of large step sizes in γ, but as γ approaches optimal, γ is kept above γ_0. This is necessary since we are only interested in stabilizing controllers.

The algorithm for this solver is outlined below.

1. Set up initial X vector from a controller that stabilizes both problems
2. Compute γ_2 and set $\gamma=\gamma_2$
3. Decrease γ
4. DFP search over X vector space to minimize performance index
5. Store X and repeat from 3.

Although the size of X can be reduced using canonical forms of the controller, it would have made little difference in this research since it would only reduce the size of X by 10% at the most. Also, the program can be run using lower order controllers than that of the H_2 problem, but then Theorem 2 in [11:10] does not necessarily apply, and there is no guarantee of existence of a solution.

This chapter has outlined H_2, H_∞,and mixed H_2/H_∞ theory. The designs that utilized these theories are developed in the next chapter.

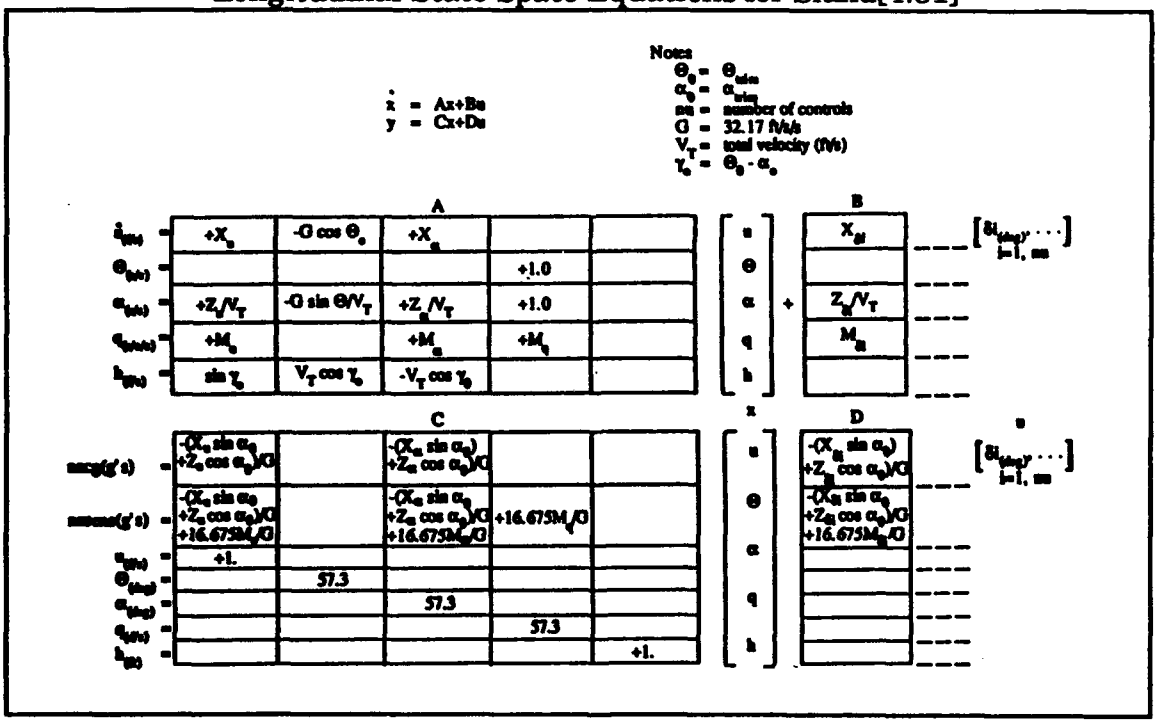
Chapter 3. Model Development

3.1 Plant and Sland Flight Condition

As mentioned before, the primary flight condition used in this research was the Sland mode. This mode utilizes the stabilator, canard, flaperon/aileron, and vanes. These effectors allow the pilot to utilize speed hold so that he is only concerned about the pitch attitude of the aircraft. Once the pilot is on approach, the control system keeps the vehicle speed constant even though its attitude is changed.

The flight data used in this research can be found in Table 15 and 16 in [4:87,89]. This is for the longitudinal state space which is based on the equations given below in Table 3.1.

Table 3.1
Longitudinal State Space Equations for Sland[4:81]



The effectors and their maximum deflections are shown below.

$$\text{Effectors} = \begin{Bmatrix} \text{Canard} & \text{deg} & -35/+15 \\ \text{Stabilator} & \text{deg} & -29/+15 \\ \text{Flap} & \text{deg} & -20/+20 \\ \text{Aileron} & \text{deg} & -20/+20 \\ \text{Right Top Vane} & \text{deg} & +35/+75 \\ \text{Right Bottom Vane} & \text{deg} & +35/+75 \\ \text{Left Top Vane} & \text{deg} & +35/+75 \\ \text{Left Bottom Vane} & \text{deg} & +35/+75 \end{Bmatrix}$$

The actual state space data is shown below and is from Table 16 in [4:89].

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

$$A = \begin{bmatrix} -0.1729 & -30.62 & -12.49 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -0.00264 & -0.0583 & -0.3129 & 1 & 0 \\ 0.001067 & 0 & 0.7453 & -0.3813 & 0 \\ 0 & 168.7 & -168.7 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0.003912 & -0.08876 & -0.042 & -0.042 \\ 0 & 0 & 0 & 0 \\ -0.000837 & -0.001365 & -0.0002601 & -0.0002601... \\ 0.0117 & -0.02452 & 0.0008338 & 0.0008338 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} -0.5352 & -0.09201 & -0.5352 & -0.09201 \\ 0 & 0 & 0 & 0 \\ ... \ 0.0005339 & -0.0001733 & 0.0005339 & -0.0001733 \\ 0.007877 & -0.007061 & 0.007877 & -0.007061 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$C = \begin{bmatrix} 0.0148 & 0 & 1.681 & 0 & 0 \\ 0.01538 & 0 & 2.067 & -0.1975 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 57.3 & 0 & 0 & 0 \\ 0 & 0 & 57.3 & 0 & 0 \\ 0 & 0 & 0 & 57.3 & 0 \\ 0 & 0 & 0 & 0 & 57.3 \end{bmatrix}$$

$$D = \begin{bmatrix} 0.004142 & 0.00766 & 0.00169 & 0.00169 \\ 0.0102 & -0.00504 & 0.002131 & 0.002131 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \cdots$$

$$\cdots \begin{bmatrix} -0.002154 & 0.001734 & -0.002154 & 0.001734 \\ 0.001926 & -0.001915 & 0.001926 & -0.001915 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Pertinent information about the flight condition at which this state space was calculated is shown in Table 3.2.

Table 3.2
Initial Conditions for Sland State Space Calculations

Item	Value
Mach	0.151
Altitude	0.0 ft
Weight	34,265 lb.
AOA	17.9 deg
Pitch Attitude	17.9 deg
Flight Path Angle	0.0 deg
Stabilator	+4.1 deg
Flap/Aileron	+20.0 deg
Top Vanes	+52.1 deg
Bottom Vanes	+52.1 deg
Canard	-14.0 deg

From the table, one can see that the Sland flight condition is basically steady level flight at sea level conditions at a typical approach speed. Even though in Section 3.3.2 this state space will be modified for use in the design model, it forms the basis for the work in this thesis.

3.2 Ideal Model Development

The ideal response is given in [4:5,6] and is based on the desire to perform precise landings in the manner that was discussed before. There are two main inputs into the system: the stick and throttle. The stick is to control the pitch and the throttle to control the speed. The three outputs of concern for this mode are the pitch rate, angle of attack, and velocity. The desired responses [4:3-7] to stick inputs are shown below:

$$\frac{\alpha}{\text{stick}} = \frac{G[\frac{\omega^2}{n/a}](\frac{v}{g})}{s^2 + 2\varsigma\omega s + \omega^2} e^{-T_d s} \quad \text{deg/in.}$$

$$\frac{q}{\text{stick}} = \frac{G\omega^2[\{(\frac{v}{g})/(\frac{n}{\alpha})\}s + 1]}{s^2 + 2\varsigma\omega s + \omega^2} e^{-T_d s} \quad (\text{deg/s})/\text{in.}$$

$$\frac{U}{\text{stick}} = 0 \quad (\text{ft./s})/\text{in}$$

where,

$$\frac{n}{\alpha} = \frac{-Z_\alpha}{g} \quad g/rad$$

$$0.6 \le \varsigma \le 0.8$$

$$\omega \ge 2.0 \ \ rad/sec$$

$$[\frac{w^2}{n/a}](\frac{v}{g}) = 7.5$$

and,

T_d = time delay < 100 msec

G = stick gain = 2.0 (deg / sec) / in.

The response should follow these transfer functions from 0.5 to 20 rad/sec. By a simple inspection of these, one can see that the ideal model is a decoupled response, which is what was mentioned earlier. Also notice that, by describing the angle of attack response, the pilot is indirectly controlling the rate of change of the flight path angle.

The response to the throttle is slightly different. In this case the desired response [4:6] is first order instead of second and is shown below:

$$\frac{U}{throttle} = G\frac{a}{s+a}e^{-T_d s} \quad (ft/sec)/deg$$

$$\frac{q}{throttle} = 0; \qquad \frac{\alpha}{throttle} = 0$$

where,

$G = 5 \ \ (ft/sec)/deg$

$0.1 \le a \le 0.2 \ \ rad/sec$

$T_d \le 150 \ \ msec$

One can see that this response is also decoupled. In both sets of responses there is a G term that is a gain. This value will be omitted for the actual design model. The resulting responses will then be to pitch rate command and velocity command. The actual stick and throttle deflections can be found from the respective G term. For example, a two deg/sec pitch rate command corresponds to a one inch deflection of the stick.

Notice that there are ranges in the values in the responses. These values needed to be specified at particular values from the admissible ranges and are as follows:

$$\omega = 5 \ \text{rad/sec}$$

$$\varsigma = .7$$

$$a = .15$$

$$T_d = 0$$

The resulting transfer function matrix is found by substituting these values into the desired response equations (while dropping the stick and throttle gain) and is:

$$G_i = \begin{bmatrix} \frac{\alpha}{q_c} & \frac{\alpha}{U_c} \\ \frac{q}{q_c} & \frac{q}{U_c} \\ \frac{U}{q_c} & \frac{U}{U_c} \end{bmatrix} = \begin{bmatrix} \frac{7.5}{s^2+7s+25} & 0 \\ \frac{25(.3s+1)}{s^2+7s+25} & 0 \\ 0 & \frac{.15}{s+.15} \end{bmatrix}$$

This is the transfer function that is the ideal model that the design model will attempt to match.

3.3 Full Design Model

The actual aircraft model that is the basis for the full design model is shown in Figure 3.1. By "full" model, it is meant the model that would be used if no separation of the problem into H_2 and H_∞ sub-parts were used.

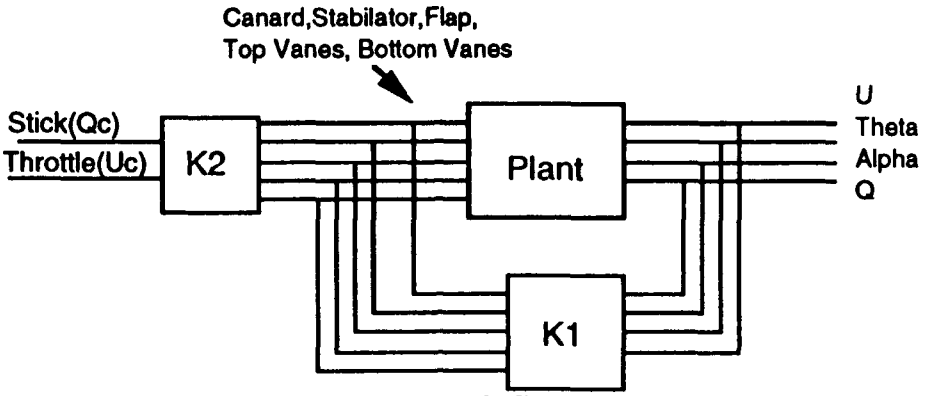


Figure 3.1. Aircraft Control Model

As one can see, the controller is a two degree of freedom (DOF) controller and some of the effectors have been combined. Examining the state space given in Section 3.1, one can see that the right and left top vanes have the same effect on the system. The same can be said for the right and left bottom vanes and the flap and aileron. Therefore, these were combined. This helps reduce the number of parameters in the controller that must be found, and thereby helps reduce computation times.

For any design model there are some given goals. For most cases this can be broken down into tracking, noise rejection, control usage, and disturbance rejection. This model is slightly different in that it is concerned with a model match instead of tracking, but is standard in that it is concerned with control usage and rejecting noises and disturbances. With these things in mind, Figure 3.2 was developed.

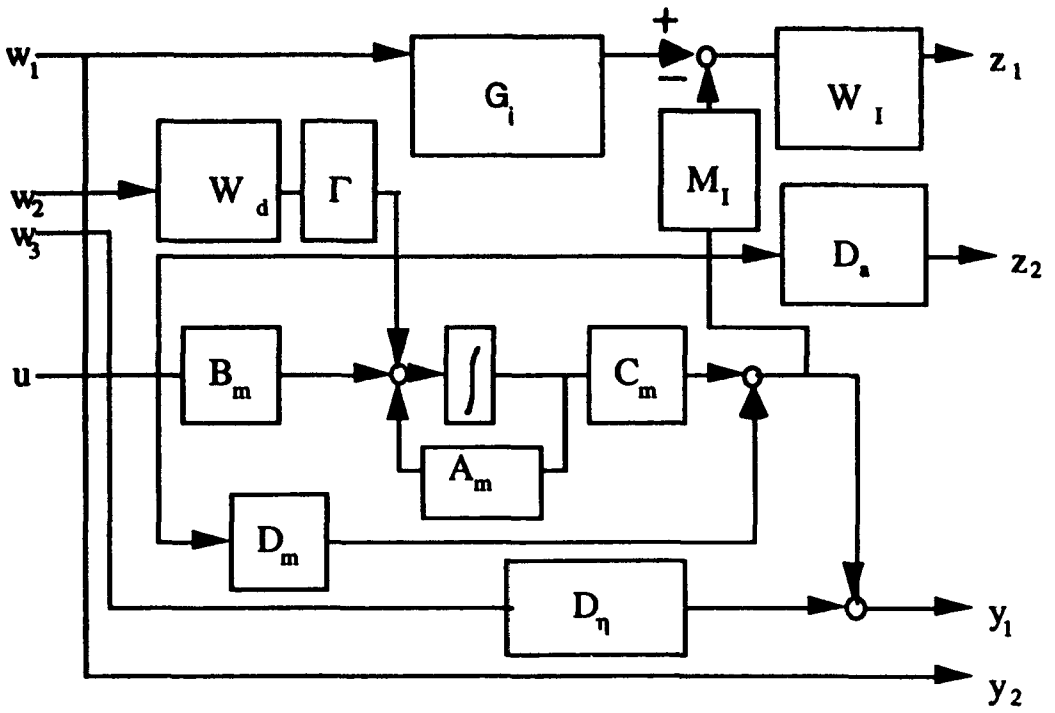


Figure 3.2. Design Model

This figure is based on the PK form of creating a design model. The y's are fed back into the controller which gives the effector commands, u. As one can see, this model attacks the four concerns mentioned earlier. The examination of this model will consist of first looking at the inputs and outputs, then the plant and weightings, and finally a look at the total state space derivation and transfer function analysis.

3.3.1 Inputs and Outputs of the System

The inputs and outputs of the system are very straightforward and are as follows:

$$\text{Design States: } X = \begin{Bmatrix} x_p \\ x_i \\ x_d \\ x_l \end{Bmatrix} = \begin{Bmatrix} \text{Sland plant states} \\ \text{ideal closed loop states} \\ \text{wind disturbance states} \\ \text{matching weighting states} \end{Bmatrix}$$

$$\text{Controlled Outputs: } z = \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} \text{Weighted Error on } \alpha, q, U \\ \text{Weighted Actuators} \end{Bmatrix}$$

$$\text{Inputs: } w = \begin{Bmatrix} w_1 \\ w_2 \\ w_3 \end{Bmatrix} = \begin{Bmatrix} \frac{q_c}{U_c} \\ \text{Disturbance} \\ \text{Sensor Noise} \end{Bmatrix}$$

$$\text{Measurements: } y = \begin{Bmatrix} y_1 \\ y_2 \end{Bmatrix} = \begin{Bmatrix} \begin{pmatrix} U_m \\ q_m \\ a_m \\ q_m \end{pmatrix} \\ \begin{pmatrix} q_c \\ U_c \end{pmatrix} \end{Bmatrix}$$

An early design included the outputs of the plant as actual outputs in the design. This proved to be unwise since it directly conflicted with the ideal model match by forcing all the outputs to zero. Also, on early models the normal acceleration was included in the measurements. The responses were good but the plant had a non-zero D term which caused the D_{yu} term of the design model to be non-zero. This violated the conditions in most of the reduced order control methods described in Chapter 1. This choice of measurements (without normal acceleration) also more closely resembled what was done by Honeywell in the SMTD program.

3.3.2 Plant and Weightings

The plant (A_p,B_p,C_p,D_p) was changed slightly from what was given in Section 3.1. It was found early on that the altitude state caused problems in

that it interfered with the tracking of the resulting controller. As a result this state was removed from the original plant to produce the design plant (A_m,B_m,C_m,D_m). Other changes made to the original included removing the normal acceleration outputs and combining some of the effectors. These changes produced the following state space of the plant:

$$A_m = \begin{bmatrix} -0.1729 & -30.62 & -12.49 & 0 \\ 0 & 0 & 0 & 1 \\ -0.00264 & -0.0583 & -0.3129 & 1 \\ 0.001067 & 0 & 0.7453 & -0.3813 \end{bmatrix}$$

$$B_m = \begin{bmatrix} 0.0039 & -0.0888 & -0.0840 & -.1070 & -.1840 \\ 0 & 0 & 0 & 0 & 0 \\ -0.0008 & -0.0014 & -0.0005 & 0.0011 & -0.0003 \\ 0.0117 & -0.0245 & 0.0017 & 0.0158 & -0.0141 \end{bmatrix}$$

$$C_m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 57.3 & 0 & 0 \\ 0 & 0 & 57.3 & 0 \\ 0 & 0 & 0 & 57.3 \end{bmatrix}$$

$$D_m = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The outputs from this system are the states (in degrees instead of radians) which are U, θ, α, and q. The inputs are canard, stabilator, flap/aileron, top vane, and bottom vane. This means that u and z_2 are 5 dimensional vectors and y_1 is a 4 dimensional vector.

To keep the order of this system down to a reasonable level, it was decided to use static weights in certain areas since adding dynamics to each control input and measurement would greatly increase the order of the controller. The two static weights are

$$D_a = \begin{bmatrix} 0.28 & 0 & 0 & 0 & 0 \\ 0 & 0.14 & 0 & 0 & 0 \\ 0 & 0 & 0.35 & 0 & 0 \\ 0 & 0 & 0 & 0.14 & 0 \\ 0 & 0 & 0 & 0 & 0.14 \end{bmatrix} \begin{Bmatrix} \text{canard} \\ \text{stabilator} \\ \text{flap / aileron} \\ \text{top vanes} \\ \text{bottom vanes} \end{Bmatrix}$$

$$D_\eta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0.2236 & 0 & 0 \\ 0 & 0 & 0.3162 & 0 \\ 0 & 0 & 0 & 0.1414 \end{bmatrix} \begin{Bmatrix} U \\ \theta \\ \alpha \\ q \end{Bmatrix}$$

The first is the actuator (control) weighting and the second is a noise weighting. The actuator weighting was found after many iterations while the noise weighting was the square root of the expected magnitude of the noise in each channel. Variations in the actuator weightings were needed once the rate and deflection limiters were put into the model. The noise weight became important once noises were introduced.

Looking at Figure 3.2, one can see that there is also dynamics in the wind disturbance model. It consists of the following:

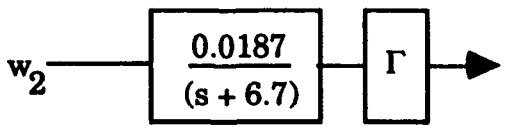


Figure 3.3. Wind Disturbance Model

This is a low pass transfer function, and the wind becomes an α disturbance by letting Γ equal the column corresponding to α in A_m.

The transfer function G_i is the ideal model that was discussed earlier in Section 3.2. This leaves W_I and M_I. M_I is a selector matrix that selects α, q, and U of the plant, and is

$$M_I = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

W_I is the weighting on the error between the output of the ideal model and the design plant. This is a 3 x 3 diagonal transfer function given by

$$W_I = \frac{s+5}{s+0.0005} \begin{bmatrix} 5 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 0.1 \end{bmatrix}$$

This was based on the weight shown in [9] and has the singular value plot shown in Figure 3.4. Although runs were made with variations of this, it was

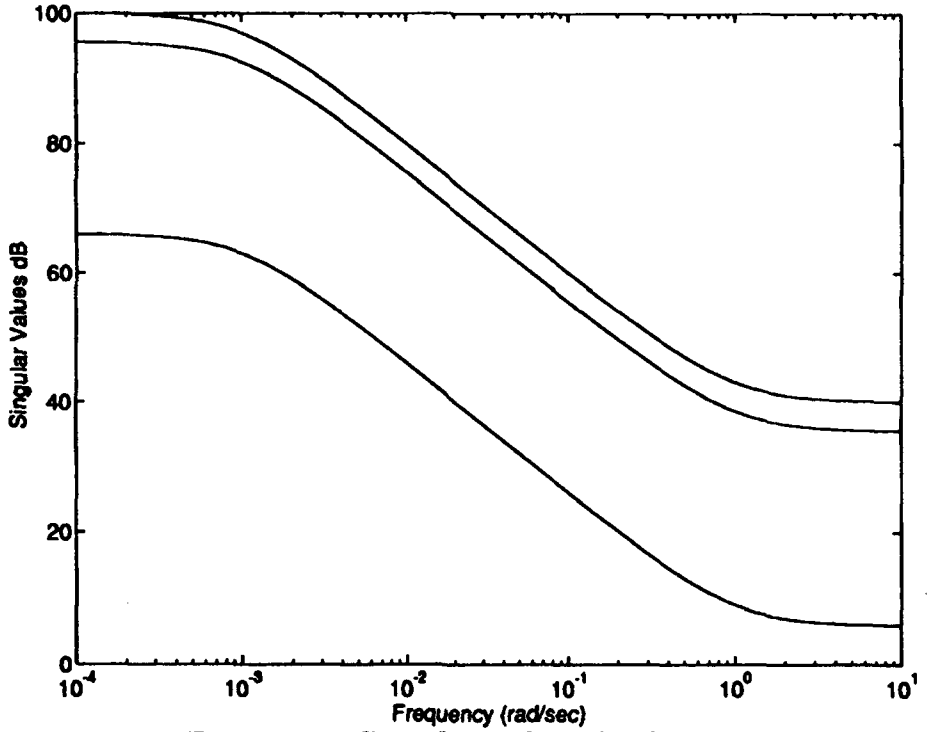


Figure 3.4. Singular Value Plot for W_I

found that this weighting produced very good results. It was found that the channels could be changed by changing the gain in the numerator pertaining to that channel. For example, from flight testing it was found that the most important response to the pilots was the q response. Therefore, the other two responses were weighted less.

3.3.3 Equation Development and Transfer Function Analysis

In order for this system to be analyzed, it must be put into a state space of the form

$$P = \left[\begin{array}{c|cc} A & B_w & B_u \\ \hline C_z & D_{zw} & D_{zu} \\ C_y & D_{yw} & D_{yu} \end{array}\right]$$

The definitions for the various inputs , outputs, and states are in Section 3.3.1. From these definitions and Figure 3.2, the state equations can be derived as

$$\dot{x}_p = A_m x_p + B_m u + GC_d x_d + GD_d w_2$$
$$\dot{x}_i = A_i x_i + B_i w_1$$
$$\dot{x}_d = A_d x_d + B_d w_2$$
$$\dot{x}_I = A_I x_I + B_I C_i x_i + B_I D_i w_1 - B_I M_I C_m x_p - B_I M_I D_m u$$

The equations for the output e and feedback y are

$$z_1 = C_I x_1 + D_I C_i x_i + D_I D_i d_1 - D_I M_I C_m x_p - D_I M_I D_m u$$
$$z_2 = D_a u$$

$$y_1 = C_m x_p + D_m u + D_h w_3$$
$$y_2 = w_1$$

The PK state space is found from the above equations, and is

$$\dot{X} = \begin{bmatrix} A_m & 0 & GC_d & 0 \\ 0 & A_i & 0 & 0 \\ 0 & 0 & A_d & 0 \\ -B_I M_I C_m & B_I C_i & 0 & A_I \end{bmatrix} X + \begin{bmatrix} 0 & GD_d & 0 \\ B_i & 0 & 0 \\ 0 & B_d & 0 \\ B_I D_i & 0 & 0 \end{bmatrix} w + \begin{bmatrix} B_m \\ 0 \\ 0 \\ -B_I M_I D_m \end{bmatrix} u$$

$$z = \begin{bmatrix} -D_I M_I C_m & D_I C_i & 0 & C_I \\ 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} D_I D_i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} w + \begin{bmatrix} -D_I M_I D_m \\ D_a \end{bmatrix} u$$

$$y = \begin{bmatrix} C_m & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 & 0 & D_h \\ I & 0 & 0 \end{bmatrix} w + \begin{bmatrix} D_m \\ 0 \end{bmatrix} u$$

This model has a total of 11 states, 8 outputs, 6 measurements, 5 u-inputs, and 7 d-inputs. This system was formed by two computer programs run in Matlab [16] (the Matlab programs developed for this research can be found in Appendix B). The first one, ACDF2.M, formed the design plant and set up the weightings. The second, PFF.M, formed the PK form of the state space based on the above equations and packed it into a system matrix called 'sys'. This was then used in the Matlab routine H2SYN.M to get the full order design.

To get a better idea of what the design model was trying to accomplish, a transfer function analysis was done. This was done in a manner similar to that of the state space derivation, but kept in transfer function form. First, let G denote the design plant and y_g the design plant output.

Then define the following:

$$z = G_i w_1 - M_I y_g$$

$$\dot{x} = A_m x + B_m u + G W_d w_2$$

$$x(s) = (sI - A_m)^{-1} B_m u + (sI - A_m)^{-1} G W_d w_2$$

$$y_g = C_m x + D_m u$$

The output y_g is then

$$y_g(s) = [C_m(sI - A_m)^{-1} B_m + D_m]u + C_m(sI - A_m)^{-1} G W_d w_2$$

or

$$y_g = Gu + G_\Gamma W_d w_2, \quad \text{where } G_\Gamma = C_m(sI - A_m)^{-1} G$$

Z_1, z_2, y_1, and y_2 can be found using the previous expressions.

$$z_1 = W_1 G_i w_1 - W_1 M_1 G u - W_1 M_1 G_\Gamma W_d w_2$$

$$z_2 = D_a u$$

$$y_1 = y_g + D_\eta w_3$$

$$y_2 = w_1$$

Now the loop is closed and the transfer functions for the outputs and measurements are found. Letting

$$u = K_1 y_1 + K_2 y_2 = K_1 y_1 + K_2 w_1$$

and closing the loop on y_1 gives

$$y_1 = [I - GK_1]^{-1} GK_2 w_1 + [I - GK_1]^{-1} G_\Gamma W_d w_2 + [I - GK_1]^{-1} D_\eta w_3$$

$$z_1 = W_1 G_i w_1 - W_1 M_1 GK_1 y_1 - W_1 M_1 GK_2 w_1 - W_1 M_1 G_\Gamma W_d w_2$$

$$z_2 = D_a K_1 y_1 + D_a K_2 w_1$$

Substituting for y_1 yields

$$\begin{aligned} z_1 = W_1 \{ & G_i - M_1 GK_2 - M_1 GK_1 [I - GK_1]^{-1} GK_2 \} w_1 \\ & - W_1 \{ M_1 G_\Gamma W_d + M_1 GK_1 [I - GK_1]^{-1} G_\Gamma W_d \} w_2 \\ & - \{ W_1 M_1 GK_1 [I - GK_1]^{-1} D_\eta \} w_3 \end{aligned}$$

$$\begin{aligned} z_2 = D_a \{ & (K_1 [I - GK_1]^{-1} GK_2 + K_2) w_1 + K_1 [I - GK_1]^{-1} G_\Gamma W_d w_2 \\ & + K_1 [I - GK_1]^{-1} D_\eta w_3 \} \end{aligned}$$

Let

$$S_o \equiv [I - GK_1]^{-1} = \text{Output Sensitivity}$$

$$T_o \equiv S_o - I = [I - GK_1]^{-1} GK_1 = \text{Output Complementary Sensitivity}$$

Therefore

$$z_1 = W_1 \{ [G_i - M_1 S_o GK_2] w_1 - [M_1 S_o G_\Gamma W_d] w_2 - [M_1 T_o D_\eta] w_3 \}$$

$$z_2 = D_a \{ S_i K_2 w_1 + K_1 S_o G_\Gamma W_d w_2 + K_1 S_o D_\eta w_3 \}$$

One can see that this model boils down to model match (first term) vs. disturbance rejection (second term) vs. noise rejection (third term). This type of problem usually leads to some sort of compromise between the three, except that there is also the control usage aspect. This can lead to conflicts. An advantage of the mixed H_2 / H_- method is that the H_- constraint can be singular, which helps to reduce these conflicts.

3.4 Mixed H_2 / H_- Models

The heart of the mixed H_2 / H_- synthesis is the H_2 model. The main concern in this design model was to serve as a provider of a low order problem that also provided noise and disturbance rejection. The design was based on Figure 3.5.

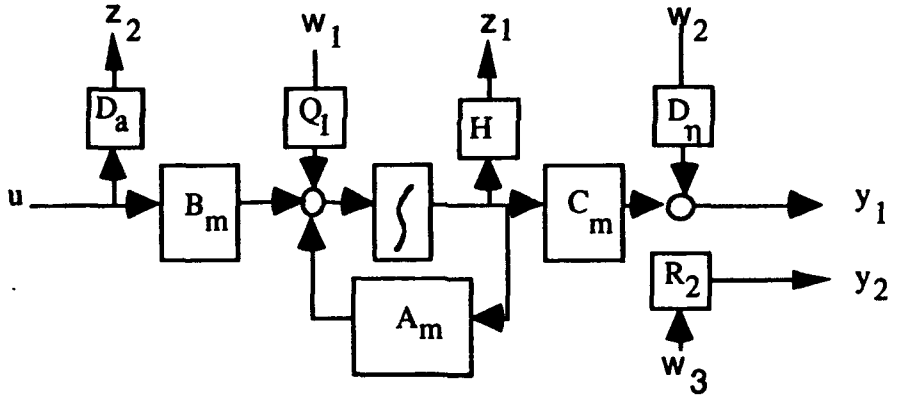


Figure 3.5. H_2 Design Model

The key phrase when describing this model is: simple. Although the model was designed to produce good noise and disturbance rejection, its main goal was to produce a low order controller. With this in mind, all the weightings were gain matrices and the plant was the same as that used in the full problem. U, y_1 and y_2 were the same as the full model. w_1 was a disturbance that was fed into the plant. w_2 was a noise that was fed into the measurement y_1. The input w3 represented the stick and throttle. Note that this is "detached" as this is a 1 DOF design, but the H_- part will be 2 DOF. H and D_a

were simply the weighted state and control usage. Since the order of this problem was four, the resulting controller was also fourth order. This was a reduction of seven from the full case.

The resulting equations were

$$\dot{x} = A_m x + Q_1 w_1 + B_m u$$
$$y_1 = D_\eta w_2 + C_m x$$
$$y_2 = R_2 w_3$$
$$z_1 = Hx$$
$$z_2 = D_a u$$

which forms the state space

$$A_2 = A_m \qquad B_w = [Q_1 \;\; 0 \;\; 0] \qquad B_u = B_m$$
$$C_z = \begin{bmatrix} H \\ 0 \end{bmatrix} \quad D_{zw} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad D_{zu} = \begin{bmatrix} 0 \\ D_a \end{bmatrix}$$
$$C_y = \begin{bmatrix} C_m \\ 0 \end{bmatrix} \quad D_{yw} = \begin{bmatrix} 0 & D_\eta & 0 \\ 0 & 0 & R_2 \end{bmatrix} \quad D_{yu} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

The only two parameters that were tunable were H and D_a. The others were fixed given the noise and disturbance, and are shown below.

D_η = Noise matrix given previously

$R_2 = I_{2x2}$

$Q_1 = 0.00279 * \Gamma$ (dc gain of $\Gamma * W_d(s)$)

The tunable parameters were adjusted so that a regulator with good noise and disturbance rejection was found. They are shown below.

$$H = 0.9 * I_{4x4}$$

$$D_a = \begin{bmatrix} 0.05 & 0 & 0 & 0 & 0 \\ 0 & 0.05 & 0 & 0 & 0 \\ 0 & 0 & .25 & 0 & 0 \\ 0 & 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0 & 0.1 \end{bmatrix}$$

The system shown was produced by the Matlab routines ACDFH2.M and PFH2.M, shown in Appendix B.

The H_∞ constraints (both tracking and robustness) can be derived from Figure 3.6:

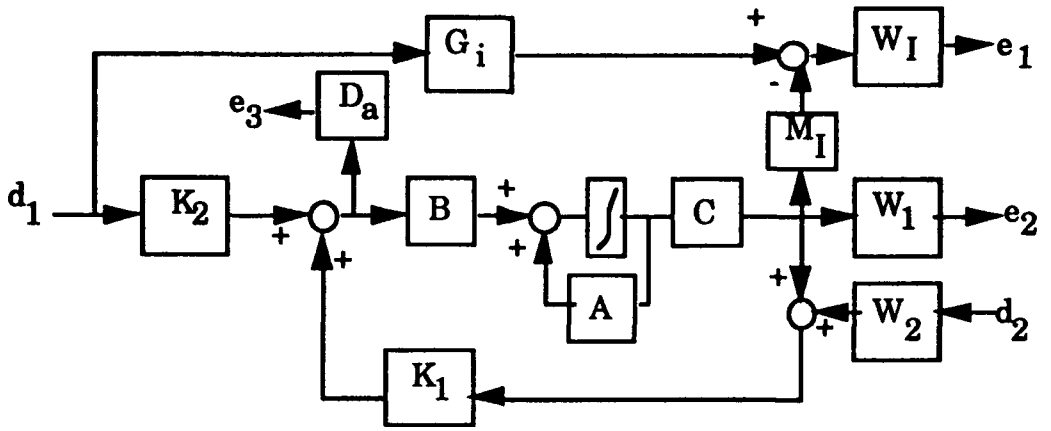


Figure 3.6. General Constraint Diagram

The matrix equations for the design were found using state space manipulation and are shown below.

$$\begin{Bmatrix} \dot{x} \\ \dot{x}_i \\ \dot{x}_I \\ \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} = \begin{bmatrix} A & 0 & 0 & 0 & 0 \\ 0 & A_i & 0 & 0 & 0 \\ -B_I M_I C & B_I C_i & A_I & 0 & 0 \\ B_I C & 0 & 0 & A_1 & 0 \\ 0 & 0 & 0 & 0 & A_2 \end{bmatrix} X + \begin{bmatrix} 0 & 0 \\ B_i & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & B_2 \end{bmatrix} \begin{Bmatrix} d_1 \\ d_2 \end{Bmatrix} + \begin{bmatrix} B \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u$$

$$\begin{Bmatrix} e_1 \\ e_2 \\ e_3 \end{Bmatrix} = \begin{bmatrix} -D_I M_I C & D_I C_i & C_I & 0 & 0 \\ D_I C & C_i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} d + \begin{bmatrix} 0 \\ 0 \\ D_a \end{bmatrix} u$$

$$\begin{Bmatrix} y_1 \\ y_1 \end{Bmatrix} = \begin{bmatrix} C & 0 & 0 & 0 & C_2 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 & D_2 \\ I & 0 \end{bmatrix} d + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u$$

One can choose any of the possible transfer function combinations depending on what aspects of the problem are desired. For example, this

research desired good model following and stability margins. The tracking consists of the transfer function e_1/d_1 and the margins are related by the complementary sensitivity which is e_2/d_2. Both of these are singular problems that cannot be solved by the standard approach described in Chapter 2. However, if one desires to get an idea of how the singular problems would perform, one can simply add in the necessary inputs or outputs but use very small weightings (10^{-5}) so that the additions are not noticed. In order to do H_∞ optimization, the number of outputs e must be greater or equal to the number of inputs d which must be greater than or equal to the number of measurements y so certain outputs and inputs must be added to meet this requirement. This approach was taken in this work.

The only tunable parameter in the tracking constraint was the weight W_I. By doing full order H_∞ synthesis (in the same manner as mentioned above), a good weight was found and is shown below.

$$W_I = 100\frac{(s+10^{-2})}{(s+10^{-1})(s+10)}I_{3x3}$$

The singular value plot for this is shown in Figure 3.7.

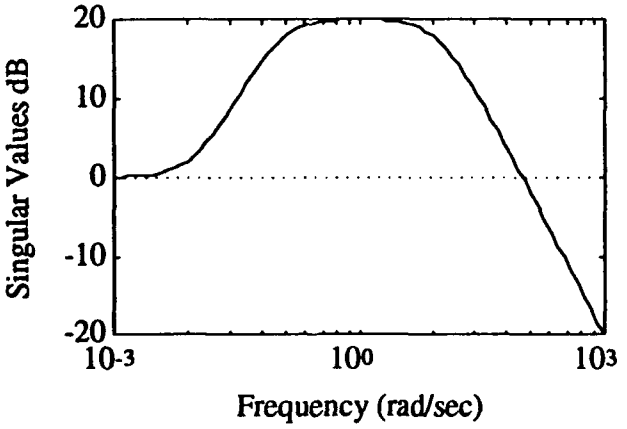


Figure 3.7. Singular Value Plot for W_I

This weight is somewhat different from the standard weighting used to produce good tracking or model following. Through the full model analysis, it

was found that the dc value of the pitch rate magnitude is always 0.0. Since the ideal model dc value is 1, there is always an error of 1. If one uses a standard weight that has a high gain at very low frequency, the absolutely lowest achievable infinity norm of any synthesis done (given the setup of the model following) would be the dc value of the weight. This leads to a poorly scaled problem. Although satisfactory responses can be found using the standard weighting and full model synthesis, it was found that in the mixed problem it did not produce acceptable results.

Like the tracking problem, the only tunable aspect of the margin constraint was the weight W_1. An acceptable weight is

$$W_1 = \frac{70(s+100)}{(s+10000)} I_{4x4}$$

and has the following magnitude plot.

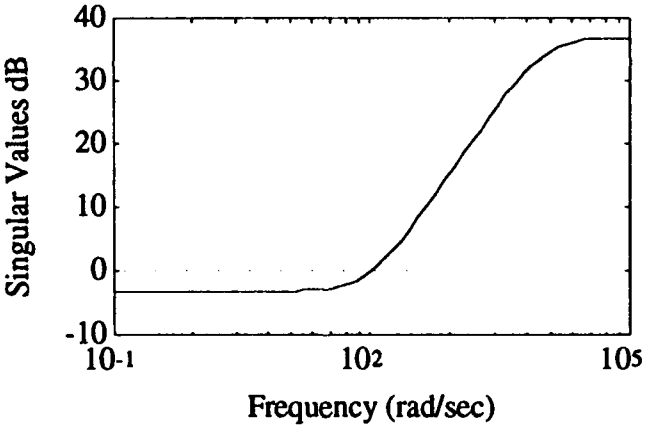


Figure 3.8. Margin Output Weighting

This weight tells the system that there is potentially 70% modeling error (unmodeled dynamics, errors in aerodynamic derivatives, etc.) at low frequency which increase with increasing frequency.

There were two primary ways to run the program. If one was starting with an initial guess, RUNOPT.M would be used. This would call programs that set up the H_2 problem and the constraint and would start the program

from the H_2 optimal controller. If it was desired to restart from a previous run, RROPT.M was used. In this program, the H_2 problem and the constraint are set up, and a point from a previous run is specified as the starting point. In both programs, there are four main parameters that can be adjusted. They are the γ step size, the number of steps, the number of iterations, and λ. The first two are self explanatory, the third is how many steps are to be taken in the Davidon-Fletcher-Powell (DFP) search, and λ is the factor in the performance index given in Section 2.2. This determines how accurate the γ step is to be.

3.5 Evaluation Model and Closed Loop System

A very important part of any design is the analysis of the response. For this work, this will consist of looking at the closed loop system and also using an evaluation model with nonlinearities. This section will examine both by deriving the closed loop system, and then examining the Simulink [16] evaluation model.

The closed loop state space is found by taking the plant and forming a PK system based on Figure 3.1 and closing the loop. The plant states can be written as follows:

$$\dot{x} = Ax + B_e d + B_u u \tag{3.1}$$

$$e = C_e x + D_{ed} d + D_{eu} u \tag{3.2}$$

$$y = C_y x + D_{yd} d + D_{yu} u \tag{3.3}$$

The controller state space is

$$\dot{x}_c = A_c x_c + B_c y \tag{3.4}$$

$$u = C_c x_c + D_c y \tag{3.5}$$

The desired form of the closed loop system is

$$\begin{bmatrix}\dot{x}\\ \dot{x}_c\end{bmatrix}=[A_{cl}]\begin{bmatrix}x\\ x_c\end{bmatrix}+[B_{cl}]d$$

$$e=[C_{cl}]\begin{bmatrix}x\\ x_c\end{bmatrix}+[D_{cl}]d$$

Substituting (3.3) into (3.5) yields

$$u = C_c x_c + D_c C_y x + D_c D_{yd} d + D_c D_{yu} u$$

or

$$(I - D_c D_{yu})u = C_c x_c + D_c C_y x + D_c D_{yd} d$$

Letting $S = (I - D_c D_{yu})^{-1}$ yields

$$u = S(C_c x_c + D_c C_y x + D_c D_{yd} d) \qquad (3.6)$$

Substituting (3.6) into (3.1) and (3.2) and (3.3) into (3.4) yields

$$\dot{x} = Ax + B_d d + B_u S(C_c x_c + D_c C_y x + D_c D_{yd} d)$$

$$e = C_e x + D_{ed} d + D_{eu} S(C_c x_c + D_c C_y x + D_c D_{yd} d)$$

$$\dot{x}_c = A_c x_c + B_c C_y x + B_c D_{yd} d + B_c D_{yu} u$$

$$= A_c x_c + B_c C_y x + B_c D_{yd} d + B_c D_{yu} S(C_c x_c + D_c C_y x + D_c D_{yd} d)$$

$$= [A_c + B_c D_{yu} S C_c]x_c + [B_c C_y + B_c D_{yu} S D_c C_y]x + [B_c D_{yd} + B_c D_{yu} S D_c D_{21}]d$$

The above equations form the closed loop state space, given by

$$\dot{X} = \begin{bmatrix} A + B_u S D_c C_y & B_u S C_c \\ B_c C_y + B_c D_{yu} S D_c C_y & A_c + B_c D_{yu} S C_c \end{bmatrix} X + \begin{bmatrix} B_d + B_u S D_c D_{yd} \\ B_c D_{yd} + B_c D_{yu} S D_c D_{yd} \end{bmatrix} d$$

$$e = [C_e + D_{eu} S D_c C_y \quad D_{eu} S C_c] X + [D_{ed} + D_{eu} S D_c D_{yd}] d$$

This is easily implemented into a Matlab program so that the closed loop system is found. This was found not only to be valuable in analyzing the singular value plots, but also in confirming that any reduction in a controller order was still stabilizing. This derivation has been somewhat generic, in that

it can be applied to just about any system as long as it is in PK form. The evaluation model will be a bit more complicated and specific to this problem.

The evaluation model was made using Simulink and is shown in Figure 3.9:

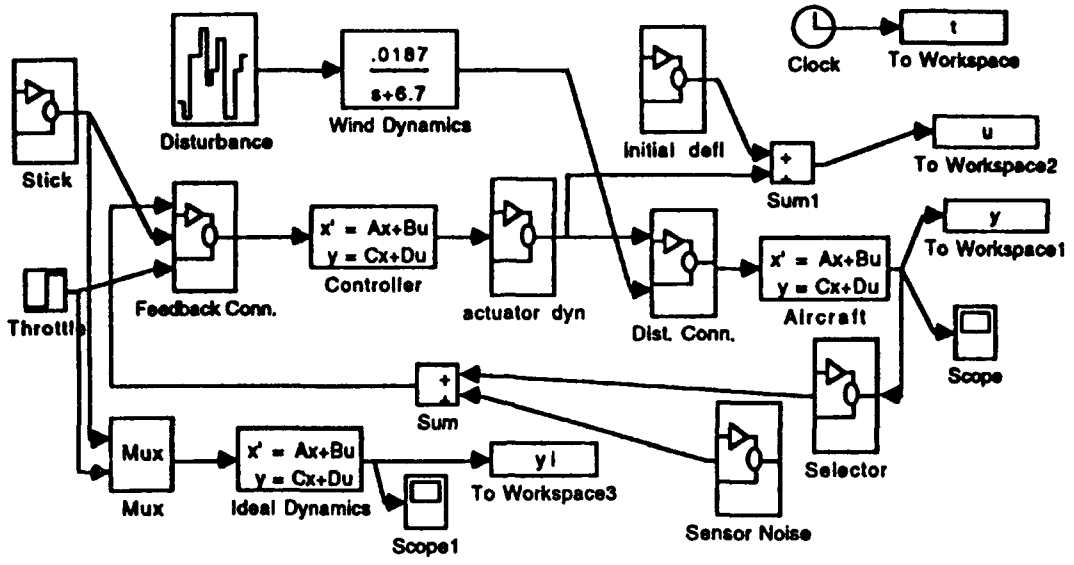


Figure 3.9. Simulink Evaluation Model

The purpose of this model is to produce responses of the aircraft to stick and throttle inputs (q and U commands) in the presence of sensor noise, wind disturbance, and actuator limitations. The implementation of the last three aspects is the main part of this discussion, but first the other parts of the model will be discussed.

As one can see there are three state spaces: Aircraft, Controller, and Ideal Dynamics. Aircraft is the aircraft model that was given as Ap, Bp, Cp, and Dp in Section 3.2 with a couple of changes. The various similar effectors are combined into five effectors just like in the design state space. There is also an extra column in the B and D matrices. The column added to B is the same as the Γ mentioned earlier which was just the α column of Ap. This is where the wind disturbance is fed into the state space. A zero column is added to D. The actual names for the different matrices in the Matlab workspace are

atm, btm, ctm, and dtm which are produced by the program SIMSET2.M. The Controller state space is simply the controller state space of the designs. It is ak, bk, ck, and dk in the workspace. The Ideal Dynamics is the exact same state space ai, bi, ci, and di that is used for the design model. Both the ideal response and the aircraft response are saved to the workspace as yi and y, respectively. They can also be observed on the two scopes.

As mentioned previously, the real essense of the model is in the noises, wind disturbance, and actuator limitations. Figure 3.10 shows what is used to simulate the sensor noise.

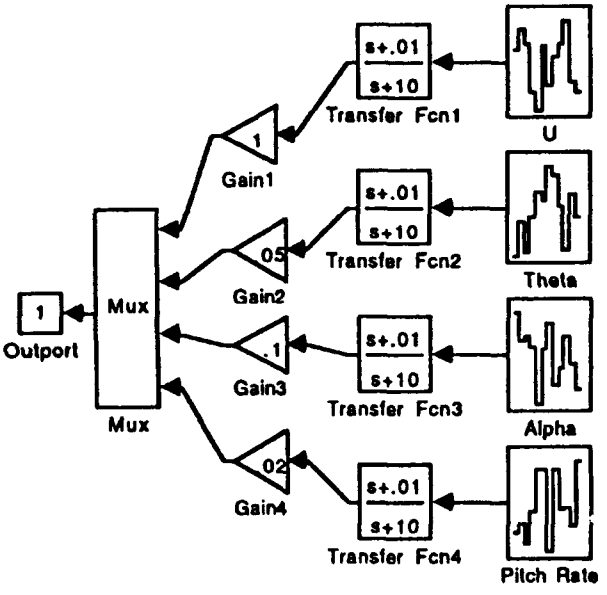


Figure 3.10. Simulink Sensor Noise Model

What is really needed as an input is colored Gaussian noise. This is accomplished by feeding the source, "White Noise for Continuous Systems" into the high pass transfer function

$$G(s) = \frac{s+0.01}{s+10}$$

The gain by which the output is multiplied is dependent upon two things. One is the expected variance of the measurement and the sample rate chosen in

the White Noise source. A sample rate of 100 Hz was used and this corresponded to a gain of:

$$\text{Gain} = \frac{\text{Expected variance}}{10}$$

The desired deviations for U, θ, α, an q were 10 fps, 0.5 deg, 1 deg, and .2 deg/s, respectively. These produced the gains shown in Figure 3.8.

The wind disturbance is implemented in much the same way as the noise. White Noise for Continuous Systems is fed through a transfer function and then into btm as the last input. As can be seen in Figure 3.9, the transfer function is

$$G(s) = \frac{0.0187}{s+6.7}$$

Like the sensor noises, the gain is dependent upon the sample rate which is also 100 Hz for this model.

The third important part of the model is the actuator dynamics and limitations, which are shown in Figure 3.11. This proved to be an important

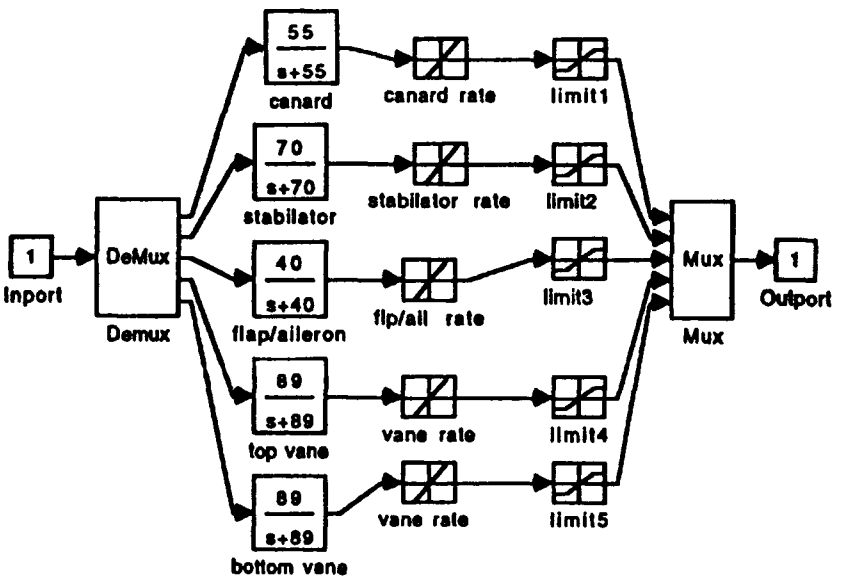


Figure 3.11. Simulink Actuator Dynamics and Limits

part of the evaluation since a perfectly stable system would be destabilized by the nonlinear rate and deflection limits. The dynamics of each effector were obtained from [4:141] and can be seen in the above figure. The rate limits were obtained from David Moorhouse [17] and are shown below in Table 3.3.

Table 3.3
Rate and Deflection Limits for Effectors

Effector	Rate Limit	Deflection Limit(from steady)
Canard	46 deg/s	-21 + 29 deg
Stabilator	46 deg/s	-33 +11 deg
Flap/Aileron	30 deg/s	-35 +5 deg
Top Vanes	180 deg/s	-17 +23 deg
Bottom Vanes	180 deg/s	-17 +23 deg

The deflection limits are given as the deflection from steady state position. These are based on the initial positions given in Table 3.2, although there is one exception. The listed initial value for the flaps is 20 deg. It was found that this setting had an adverse affect on the alpha response, since the design method has no real way of weighting signs of outputs. Therefore, the author made the adjustment of setting the flaps at 15 deg to see what could be done. In reality, the pilot is just as likely to command a negative pitch rate (this was the case in flight test data) as a positive since in landing the flight path angle is usually negative. Considering this, the change did not adversely affect the analysis.

Another change that was made, but in the actual program itself, has to do with the vane deflections. In the original document that listed the deflections for the effectors [4], the vane limits were given as +45 to +75 degrees. Over the course of developing the full order design it was found that this hindered the response since the vanes only had seven degrees to come back for positive pitch rate commands. When this was presented to David Moorhouse [17], he stated that contractors had the same problem, and as a

result the deflections on the vanes on the aircraft were adjusted to allow +35 to +75. It was interesting to find that some of the things found by this research were also found in the actual program development.

As shown in Figure 3.9, the deflections are split off and added to the steady state values to produce the actual positions of the effectors. This was then sent to the workspace as u. Therefore, the actuator deflection plots will show the actual deflection as it would be on the aircraft.

This chapter has described the models used for both design and evaluation. Before any type of mixed designs were found, the full, regulator, tracking, and margin designs were analyzed. The results of this analysis are found in the next chapter.

Chapter 4. Preliminary Results

4.1 Full Model Design

The process of finding a full model controller that satisfied the ideal model and also performed well in the face of wind disturbances, noises, and actuator limits became quite a problem. Early models that had the altitude state in the plant would not track since that put a column of zeros in the A matrix of the model. Once that state was taken out, a response that followed the ideal was found. The early models used educated guesses at what the weightings on the noises and actuators should be. When the actuator limits and noises were added to the evaluation model it was found that these initial guesses were not very good. The nonlinear simulation responses were often very unstable. Through many changes in the two weights, an acceptable response was found. The final weightings were documented in Section 3.3.2.

As mentioned earlier, an 11th order controller was found using the H2SYN.M Matlab routine.

4.1.1 Closed Loop Frequency Responses for Full Model Design

The first aspect of this synthesis to be analyzed is the set closed loop frequency responses of the system. The two inputs to the system are the stick and the throttle, which will be referred to as Qc (pitch rate command) and Uc (velocity command) respectively. The frequency responses to the stick are shown in Figure 4.1.

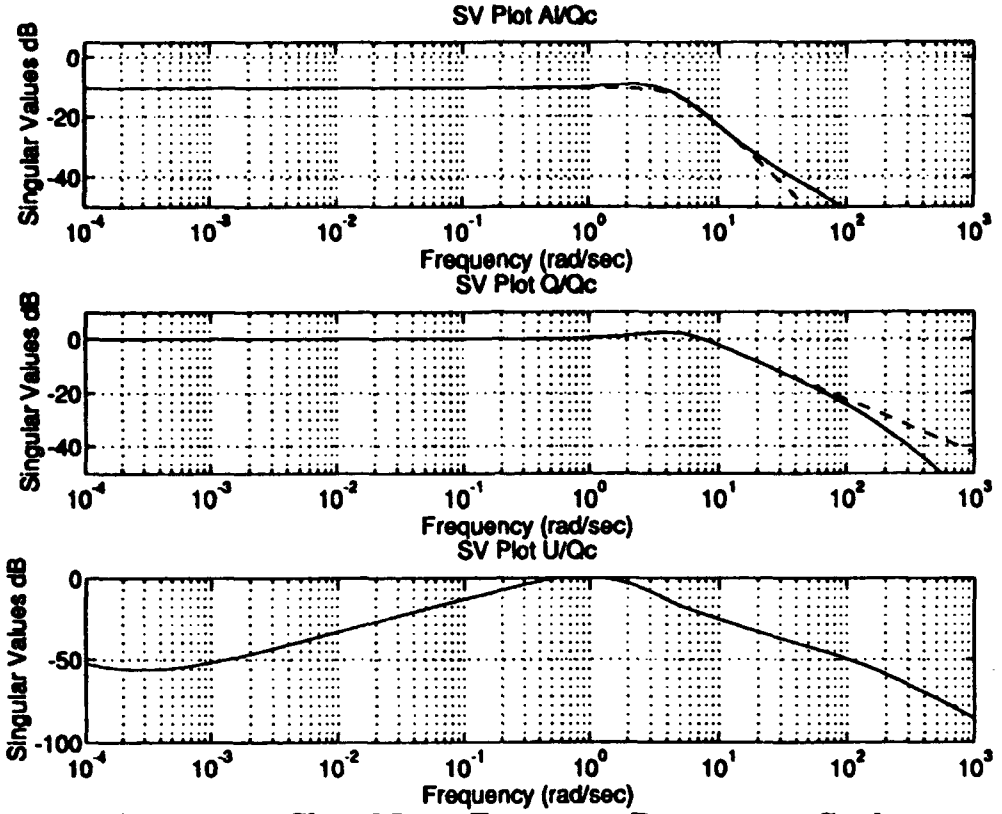


Figure 4.1. Closed Loop Frequency Response to Stick

The dashed lines are the responses for the ideal models and the solid lines are the actual. From this figure, one can conclude a few things about the response of the system. The alpha and q responses are very close to the ideal except for high frequencies. This is not bad since the weighting did not specify close tracking of ideal for very high frequency. There is some deviation in the alpha response between 1 and 10 rad/sec. This is the region of concern for the pilots are concerned, but the deviation is quite small and should have little effect on the performance. The velocity response is not bad. The ideal is that this would be totally decoupled from the stick. The peak occurs in an undesirable area but it only goes up to zero dB. This will be acceptable considering that the velocity is represented in ft/sec and Qc is in deg/sec. The worst case is that the velocity tracks Qc which will be on the order of three to

five ft/sec, which is a very small deviation considering that the equilibrium velocity is on the order of 200 ft/sec.

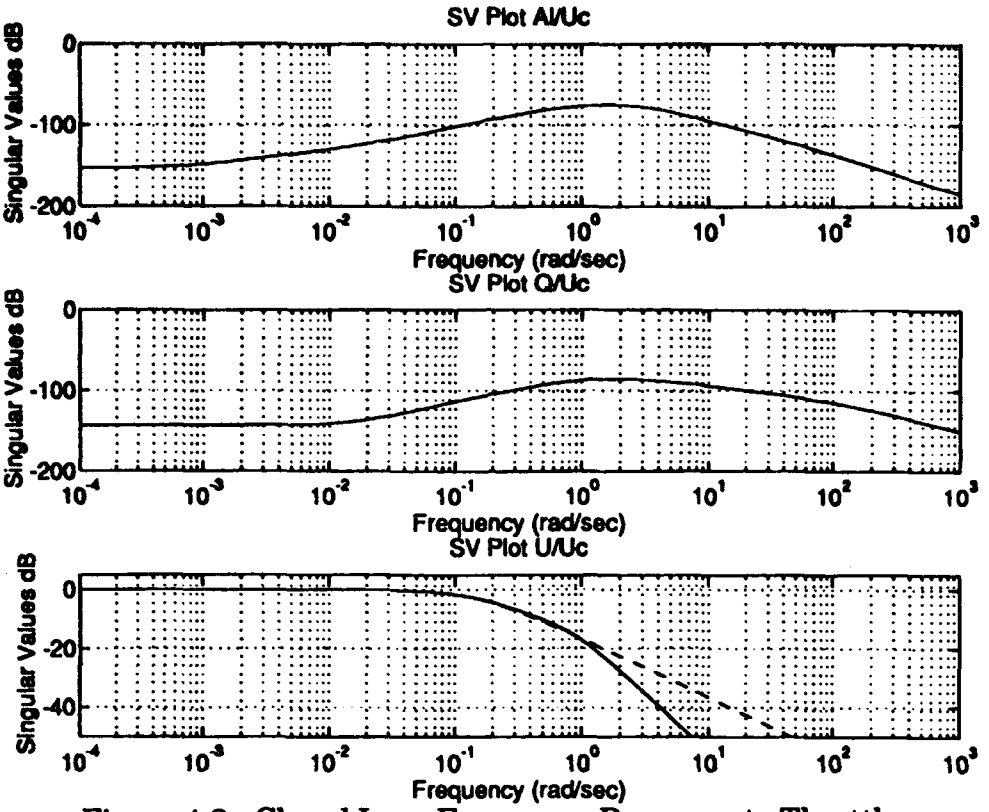


Figure 4.2. Closed Loop Frequency Response to Throttle

The frequency responses to throttle input are shown in Figure 4.2. Again, the ideal model is shown by the dashed line. The velocity response follows the ideal very well up to about 10 rad/sec. The main use of the throttle is for a speed hold, so there will not be high frequency throttle inputs and this response would be expected to be acceptable. The other two responses are fairly well decoupled. Neither goes past -70 dB, which corresponds to a gain of 0.0003. Even though the velocity inputs are much larger in scale than the others, this is quite acceptable.

These two figures show that the full order design should produce very good responses that are close to ideal. This will be shown to be true in a later section. Although these figures show the performance of the aircraft, they do

not lend any insight into the potential noise and disturbance rejection characteristics. This will be examined in the next section by looking at the open loop.

4.1.2 Open Loop Analysis for Full Model Design

One of the goals of this design was to reject disturbances and noise. Besides doing simulations in the time domain, one can examine the rejection properties of the system by looking at the open loop transfer function. In general, it has been found that the closed loop system will have good rejection properties if the open loop has the proper bandwidth and performance characteristics. Even though the system in this research uses a two degree of freedom controller, the basic ideas stated above still hold. Figure 4.3 shows the singular value plots for the open loop (GK1) and for the two controllers.

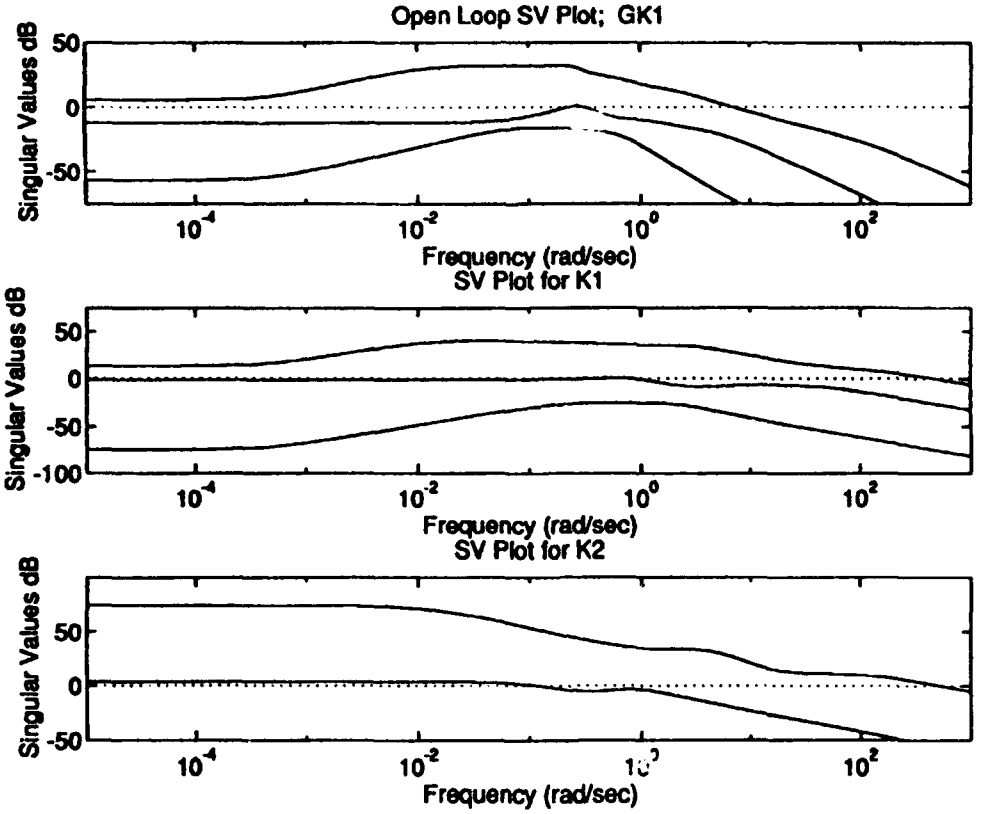


Figure 4.3. Singular Value Plots for Open Loop Transfer Function and Controllers K1 and K2

The bandwidth of the open loop system can be seen on the plot to be around 8 rad/sec. This is fairly high, but is acceptable and should prove to produce good noise rejection characteristics. For a single degree of freedom controller, the low frequency gain would be expected to be high for good tracking. This is not necessary for the two degree of freedom controller and is also evident in the plot of GK1 [18]. One interesting aspect of this plot is that GK1 is a four by four transfer function matrix, but only three singular values show up on the plot. There is a fourth singular value, but it has very low gain at all frequencies. The same thing can be said for K1, so one could conclude that the measurements can be somehow reduced to three since the controller seems to only need that many.

Another interesting aspect of these plots is that both the controllers have very high bandwidths. In most cases this would be something to be avoided, but for the two degree of freedom controller it can be acceptable.

From the frequency domain analysis in the previous two sections, it has been concluded that the performance of the system will be good both with respect to tracking and rejection. The next section will examine this conclusion through time domain analysis.

4.1.3 Step Responses for Full Model Design

The step responses were found using the nonlinear Simulink evaluation model discussed in Section 3.5. The step responses played a large part in the design process. They helped determined whether or not a design was satisfactory. The criteria for a satisfactory response were that it had to match the ideal pitch rate response for a 1 deg/sec command closely and have small steady state error. Also, for the 1 deg/sec command, the response had to be well behaved with respect to wind and noise. These were somewhat

subjective but were deemed to be acceptable. Another thing that had to be kept in mind was that the pitch rate response was more important from a handling qualities point of view. Comments from test pilots in the SMTD program showed that pitch rate following was more important than α following [17]. Therefore, if needed, the weighting on the pitch rate error would be increased by some scalar multiple. Figures 4.4 and 4.5 show the responses to a 1 deg/s pitch rate command without the disturbances. The dashed line in all

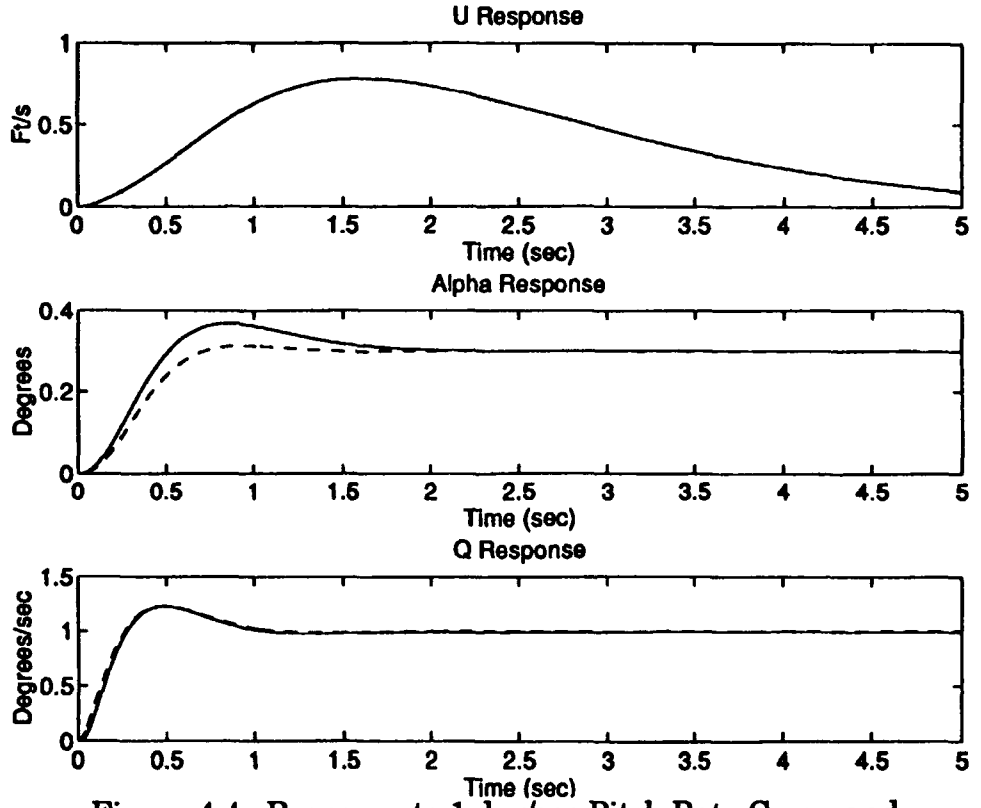


Figure 4.4. Response to 1 deg/sec Pitch Rate Command (no Disturbances)

the response figures represents the ideal response. As one can see, all three quantities are very close to the ideal. As expected, there is some transient deviation in the alpha response, but it tracks very well. The U response also deviates but settles out fairly quickly. A change in velocity of less than one ft/sec is almost unnoticeable to a pilot. It appears that the conclusions drawn

from the frequency plots were correct and that this design does very well to follow the ideal model.

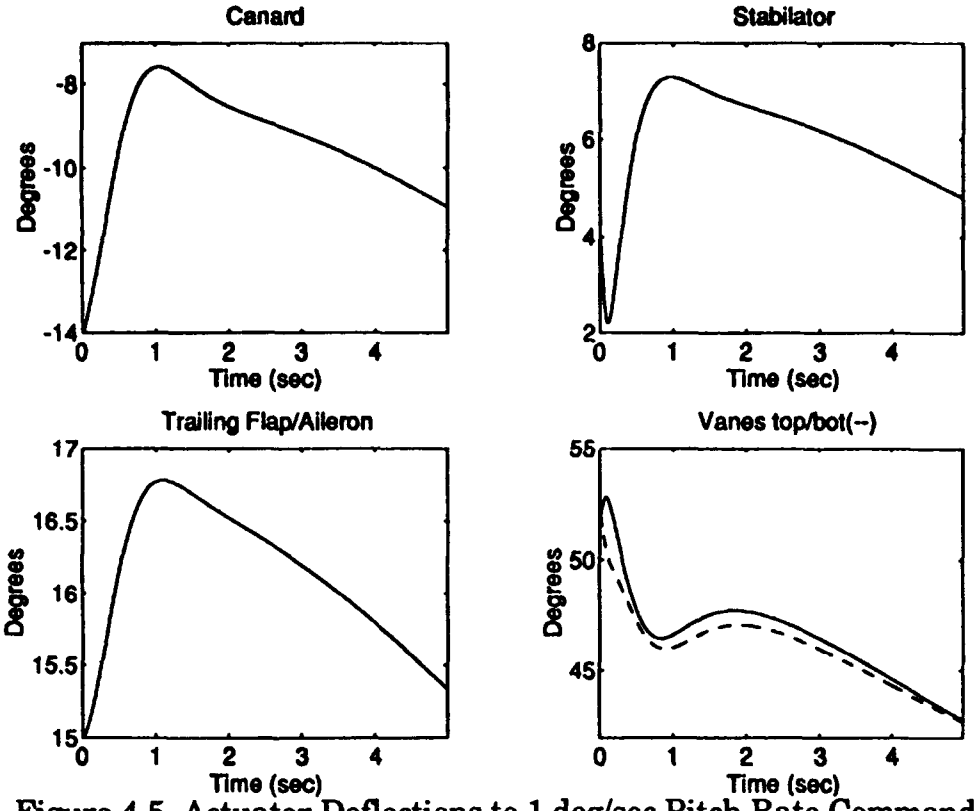


Figure 4.5. Actuator Deflections to 1 deg/sec Pitch Rate Command

The actuator deflections show a couple of interesting things. It appears that the stabilator rate is initially very close to maximum. As mentioned in Chapter 3, the flap is important for the angle of attack response, which is responsible for the pitch rate commands to command flight path angle rate indirectly since flight path rate is pitch rate minus angle of attack rate (angle of attack rate is zero after about 1.5 seconds). It was found that when the angle of attack match is taken out of the design model, the angle of attack responds just like pitch angle, resulting in no change in flight path angle. Another interesting thing is that the top and bottom vanes were not deflected the same. This shows that the vanes helped produce some pitching moment when the stabilator was at its maximum rate. The vanes also play a major

role in holding the velocity. As pitch angle is increased, the thrust must also be increased to sustain velocity and this is exactly what the vanes do.

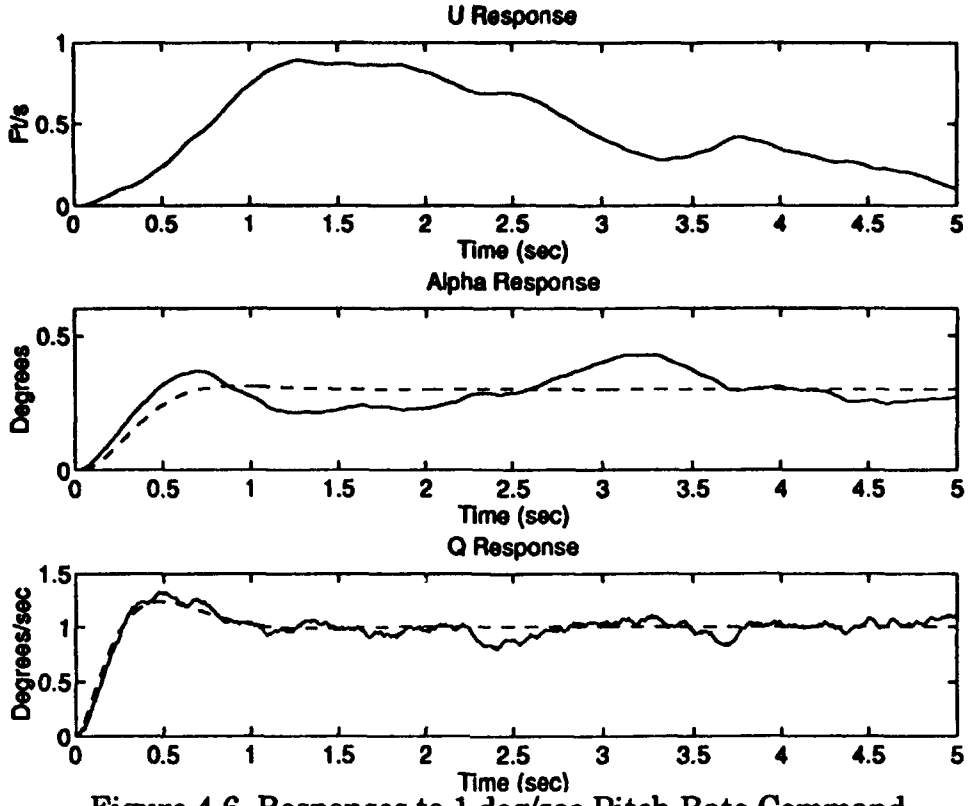


Figure 4.6. Responses to 1 deg/sec Pitch Rate Command (with Wind Disturbance and Sensor Noise)

The next thing to examine is the effect of the wind disturbance and noise. These are seen in Figures 4.6 and 4.7. As expected the response is degraded by the disturbances. As one may expect, the alpha response is affected more in the low frequencies (wind disturbance) and q shows the effects of sensor noise. The response is far from clean but considering that the noise in the model is very conservative (probably worse than what will actually be experienced), the controller does quite well.

The actuator deflections (Figure 4.7) show the same general actuation trends as the clean response. The maximum actuator deflections are larger than the clean actuators but this is also to be expected since they must counteract the disturbances and noises.

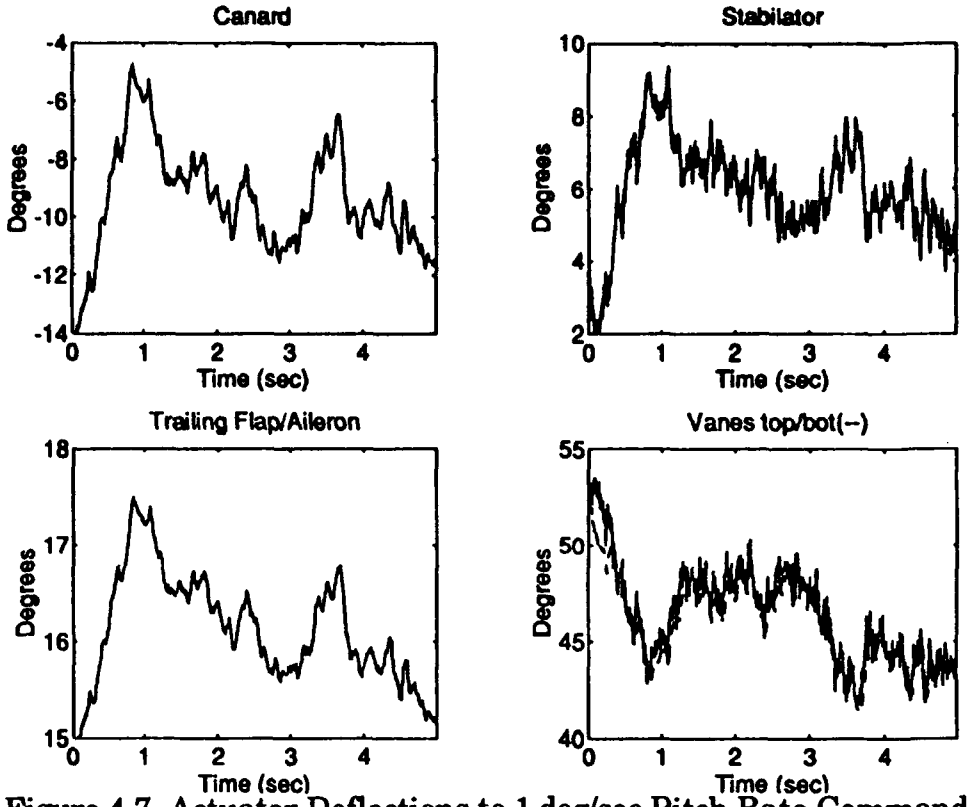


Figure 4.7. Actuator Deflections to 1 deg/sec Pitch Rate Command (with Wind Disturbance and Sensor Noise)

Another aspect of the design was the question: what was the pilot going to ask the aircraft to do? This was a very important question since one would like to use the full range of effector deflection without getting into trouble. From flight test data showing stick and actuator deflections, a good idea of what was needed from the aircraft was obtained. It was observed that the maximum the pilot would command would be a 3 deg/s pitch rate held for around 5 seconds. This was a fairly conservative guideline since the maximum range of pitch angles was only 10 degrees. Applying this to older designs showed they were using too much control power, which destabilized the system. This was found to be the limiting factor as far as the actuator weighting was concerned.

4.1.4 Complementary Sensitivity and Margins

A very important aspect of control design is the stability robustness of the system. The design methods mentioned in this research are based on models that are linearized around a specific flight condition. Unfortunately, the real world is not linear, and it is necessary that the control design stabilize the system within a certain region around the specific flight condition. Calculating the stability robustness for a single input single output (SISO) system is a fairly simple operation involving Nyquist or Bode plots. Multiple input multiple output (MIMO) systems are a bit more difficult to analyze. The biggest difficulty is how to address simultaneous changes in the MIMO system. Vector margins, which are based on the infinity norms of the sensitivity or complementary sensitivity, are used to assess the stability robustness of the designs in this work.

The complementary sensitivity singular value plot is shown in Figure 4.8.

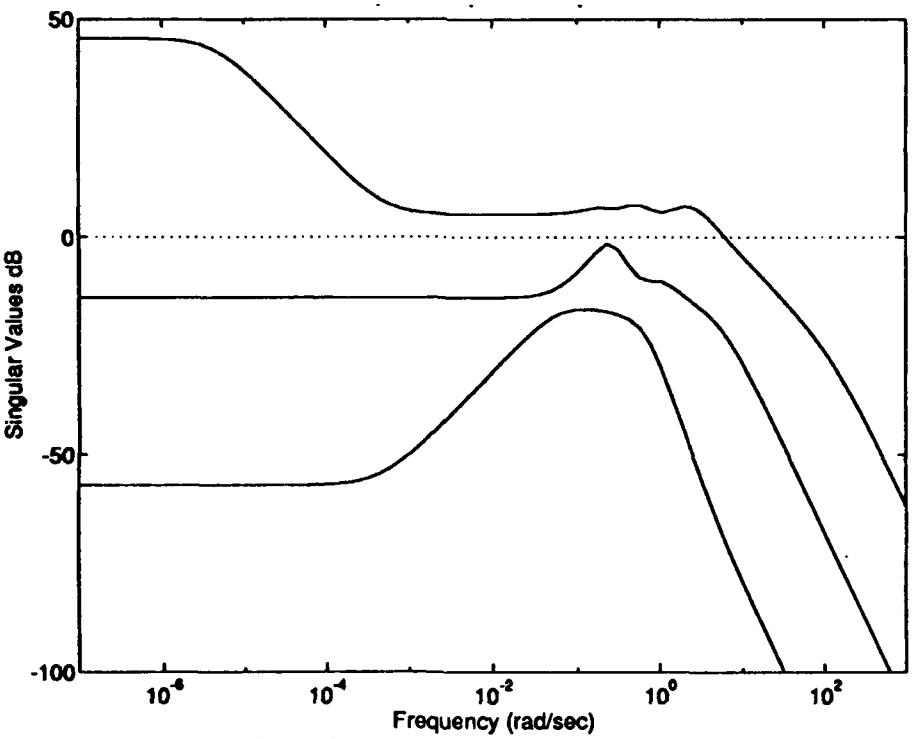


Figure 4.8. Complementary Sensitivity for Full Design

The margins are based on the infinity norm of the complementary sensitivity which can be seen in the following equations.

$$\alpha_0 = \frac{1}{\|T\|_\infty}$$

$$\text{Lower Gain Margin} = 1 - \alpha_0$$

$$\text{Upper Gain Margin} = 1 + \alpha_0$$

$$\text{Phase Margin} = \pm 2\sin^{-1}\left(\frac{\alpha_0}{2}\right)$$

Looking at the complementary sensitivity plot, one would assume that this design would have small margins. This is indeed the case as can be seen below.

Gain Margin=[-0.0458,.0455] dB

Phase Margin=± 0.3 degrees

Note that these tiny margins are caused by the complementary sensitivity being large at low frequency; this is due to the rate tracking loop. It can easily be concluded that this problem needs to address robustness, since these margins are unacceptable for any type of application.

4.2 The H_2 Regulator

As mentioned earlier, the heart of the mixed H_2 / H_∞ synthesis is the H_2 problem. It has several important objectives in this research which are reflected in its design (see Chapter 3). These relate to the main goal of this work: to get a d reduced order controller, and provide disturbance and noise rejection characteristics. Based on these, the H_2 problem had to provide a simple (4th order) regulator with good disturbance and noise rejection. This regulator was found by using H2SYN.M in Matlab.

4.2.1 Open Loop Analysis for Regulator

The open loop singular values are shown in Figure 4.9. The main points that can be drawn from these are that the regulator has a lower bandwidth than the full design; it can achieve what it needs to with only one signal; the

stick and throttle inputs are not used at all. For this reason there is no tracking analysis, as no information from the commands gets into the system.

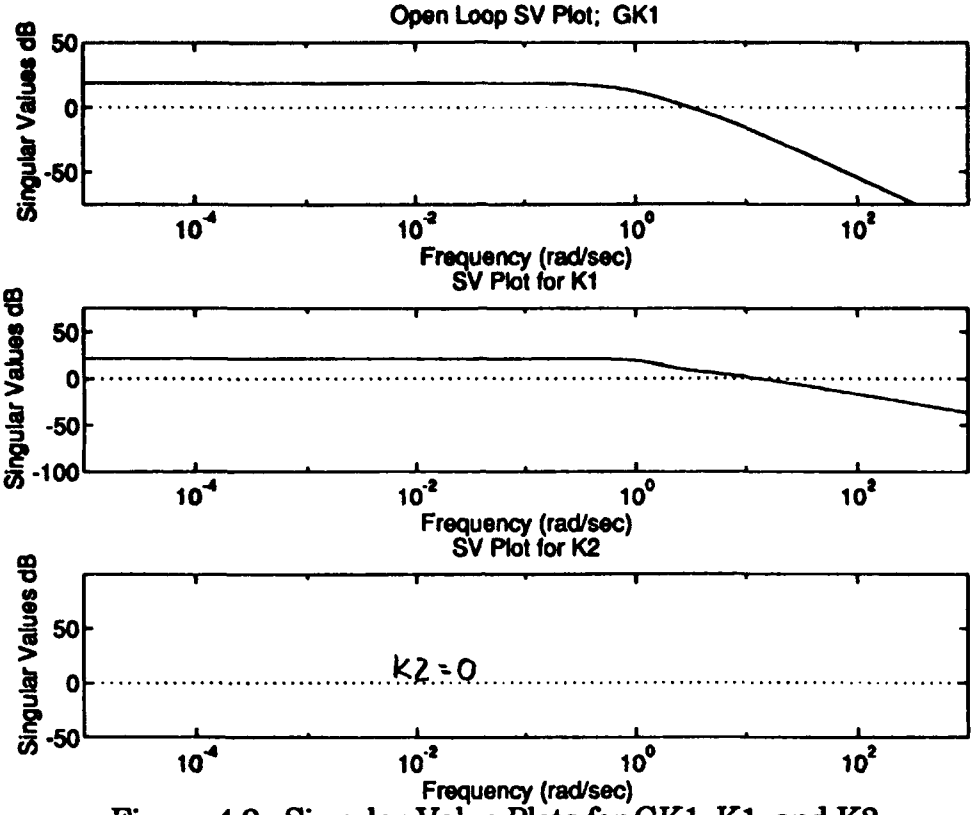


Figure 4.9. Singular Value Plots for GK1, K1, and K2

Based on this, the closed loop system will not track but will reject disturbances and noises and regulate the states.

4.2.2 Time Responses for Regulator

Since the closed loop tracking was not relevant to this problem, most of the decisions as to what weights were to be used were based on initial condition responses. These responses were produced by perturbing the initial state vector in the aircraft state space. The two perturbed states are the alpha state (5 deg) and the pitch rate (3 deg/sec). The response to an alpha perturbation is shown in Figure 4.10.

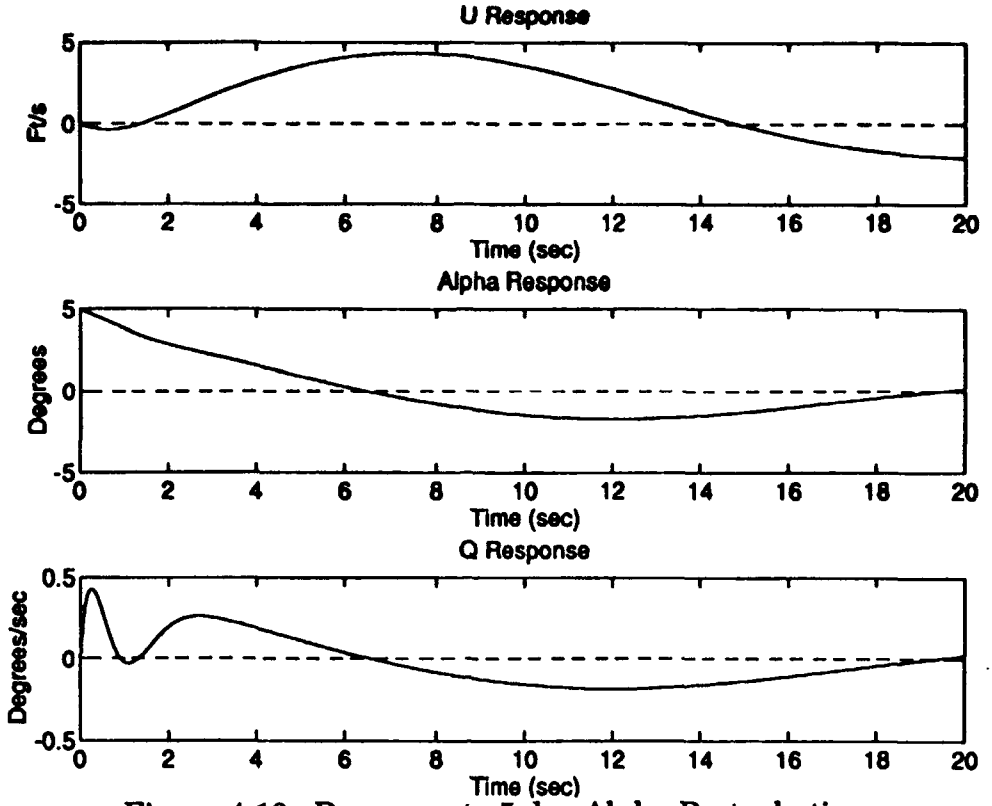


Figure 4.10. Response to 5 deg Alpha Perturbation
(no Disturbance or Noise)

The weights in the design model were chosen so that the pitch rate response was fairly fast and had good damping (ζ=.7) with alpha and velocity being regulated. Given the perturbation of 5 deg alpha, the plots show that this is accomplished.

As mentioned earlier, one of the important aspects of the regulator design is to produce good disturbance and noise rejection. The response and actuator deflections with wind and noise are shown in Figure 4.11 and 4.12.

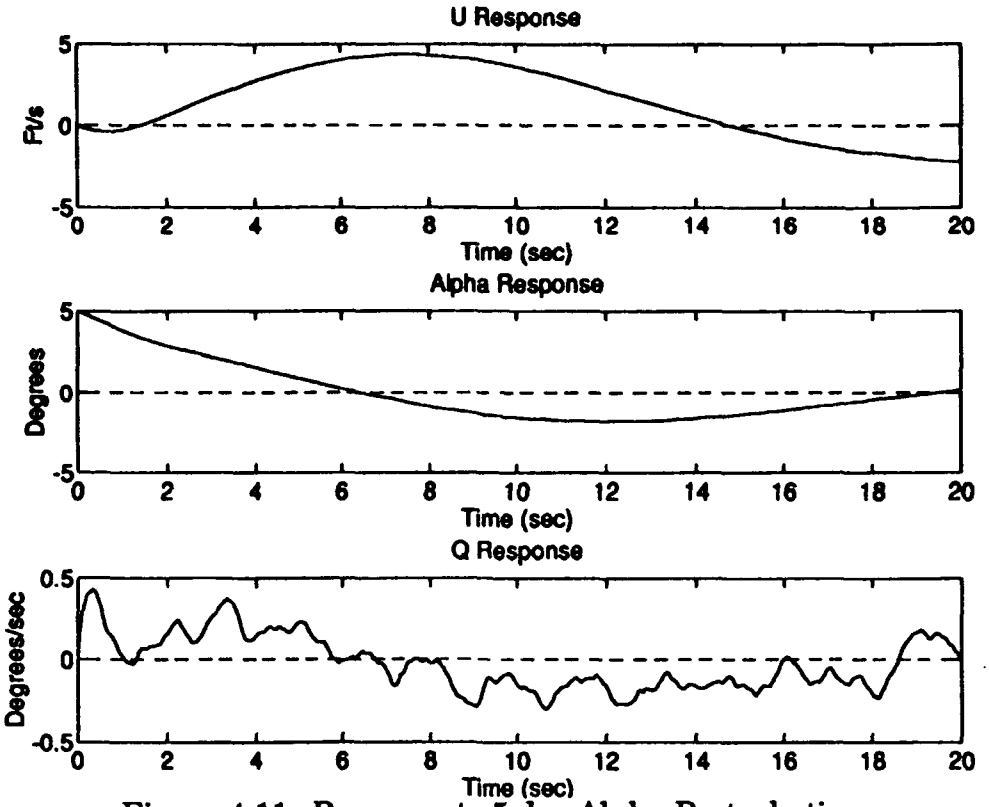


Figure 4.11. Response to 5 deg Alpha Perturbation (Wind Disturbance and Sensor Noise)

The velocity and alpha responses show practically no deviation from the clean responses. The pitch rate is noisier but well regulated. The actuator deflections also show some of the effects of the noise but are also quite small. The interesting thing to note is how the deflections seem to have the same shape curve. This makes sense, considering what was seen in the open loop frequency plots.

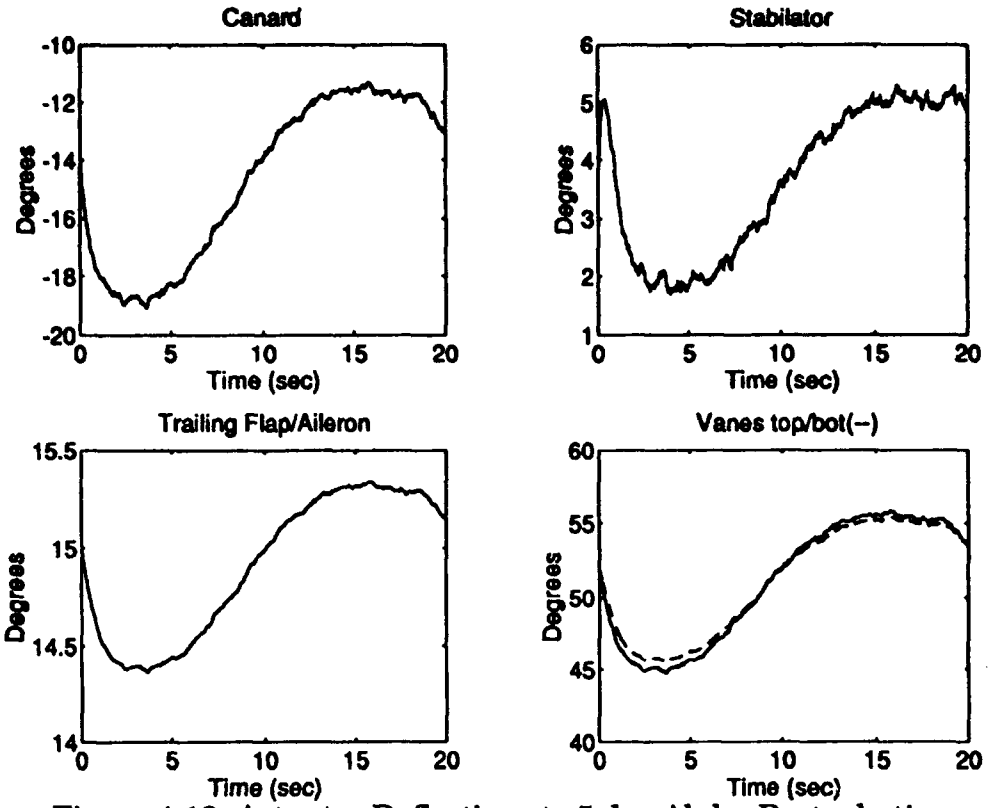


Figure 4.12 Actuator Deflections to 5 deg Alpha Perturbation (Wind Disturbance and Sensor Noise)

The other perturbation that was investigated was an initial pitch rate deviation. The desire here was to produce a response that was somewhat similar to the ideal pitch rate response. An initial pitch rate of 3 deg/sec was used to examine this. The result is shown in Figure 4.13.

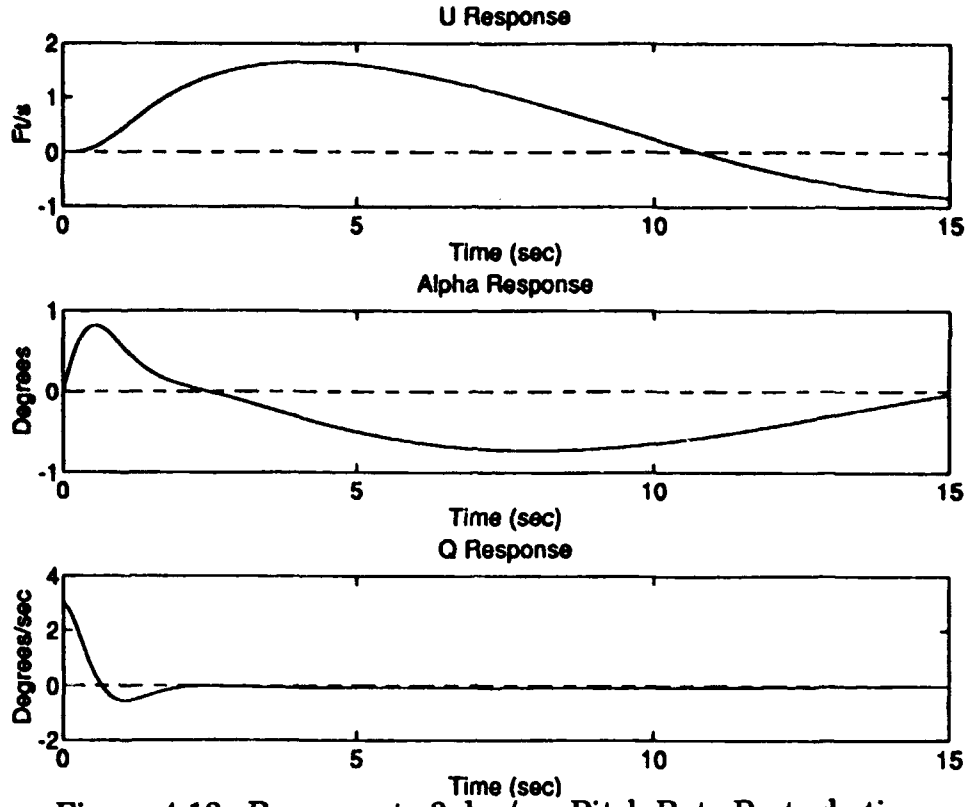


Figure 4.13. Response to 3 deg/sec Pitch Rate Perturbation (no Disturbance or Noise)

The figure shows that the desired response is accomplished. The velocity deviations are smaller than those in Figure 4.10. Obviously, velocity is more sensitive to alpha changes than pitch rate changes. This would make sense, considering that drag is dependent upon alpha.

The response to this perturbation with disturbances and noise was also examined. These are seen in Figures 4.14 and 4.15.

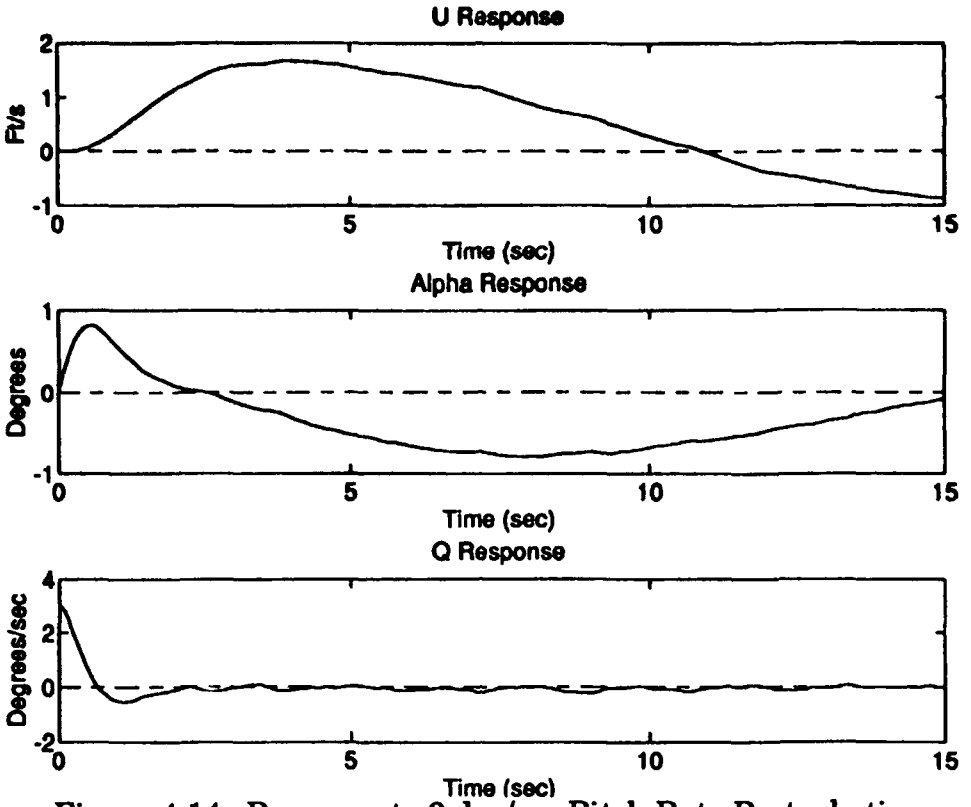


Figure 4.14. Response to 3 deg/sec Pitch Rate Perturbation (Wind Disturbance and Noise)

Figure 4.14 shows that the rejection properties are very good. The pitch rate response looks a lot better than the pitch rate response due to an alpha perturbation, but this is mainly due to the scales of the plot. The actuators also show the same trends that were seen in the alpha perturbation.

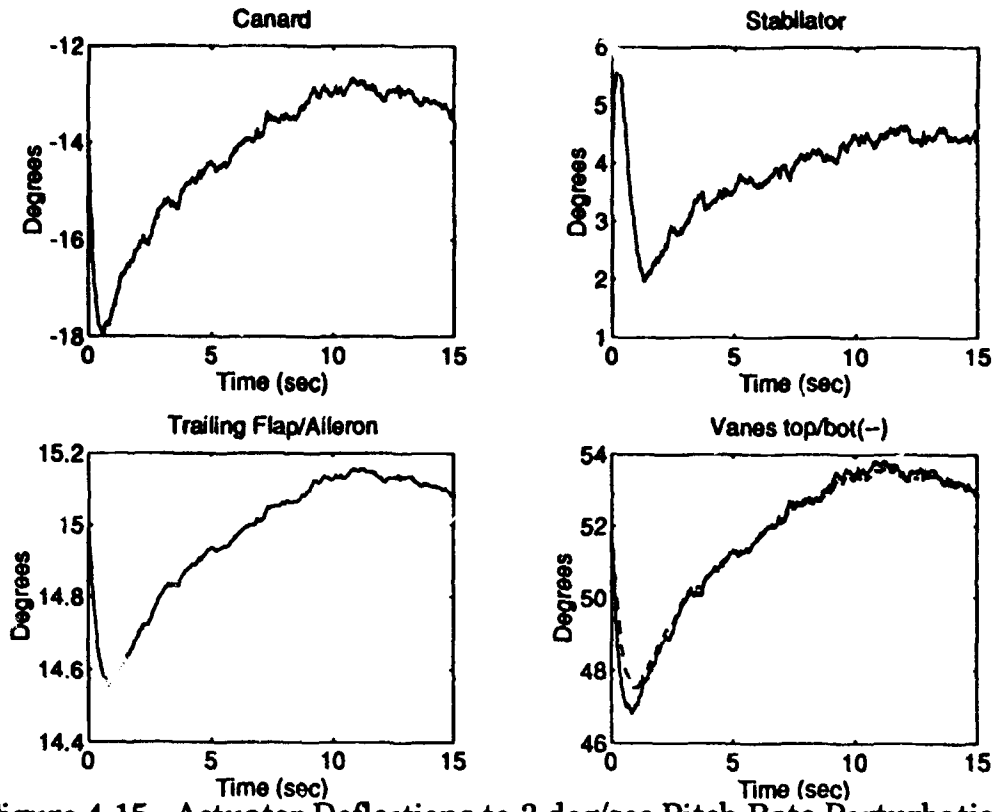


Figure 4.15. Actuator Deflections to 3 deg/sec Pitch Rate Perturbations (Wind Disturbance and Noise)

4.2.3 Complementary Sensitivity and Margins

Although the regulator was not meant to provide the system with good margins, it is important to look at them to provide insight into the mixed solution. The margin calculation is based on the complementary sensitivity which can be seen in Figure 4.16.

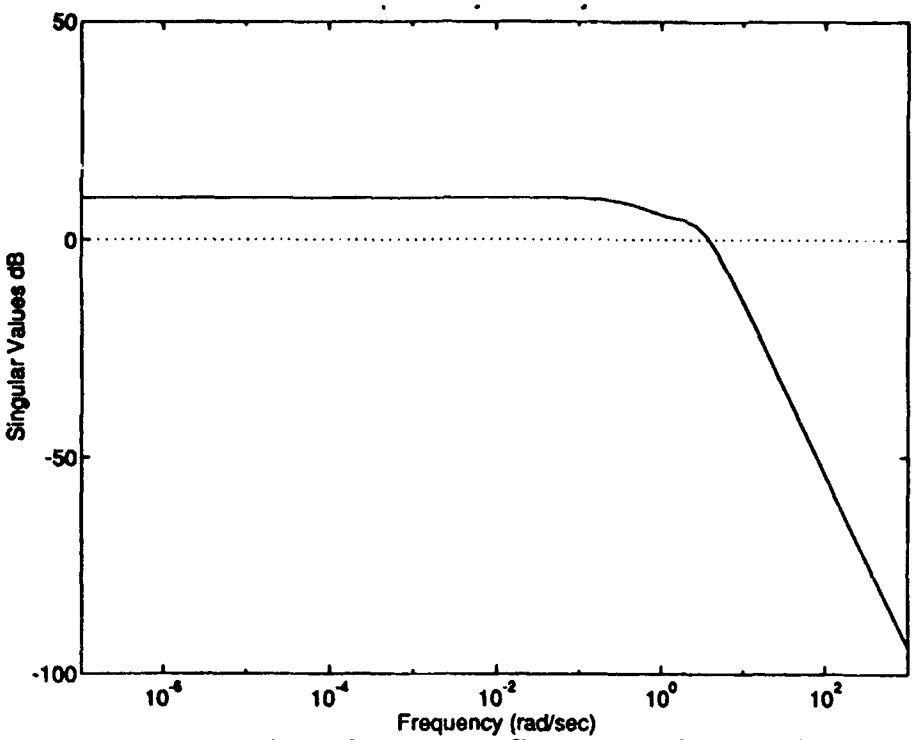


Figure 4.16. Complementary Sensitivity for Regulator

The margins for this are calculated the same way as for the full order problem and are as follows:

Gain Margin= [-3.5,2.49] dB

Phase Margin= $\pm$ 19.1 degrees

These are much better than the full order design but are still a little small for an aircraft application. Hopefully, the margins from this will be carried over to the mixed problem, even though it is not one of the things the regulator is trying to accomplish.

4.3 H_∞ *Tracking Design*

In the mixed synthesis, the constraints are set up as singular H_∞ problems. It is important that the designer have some knowledge about what they actually do. This gives the designer an idea of how much they are losing by trading off between the H_∞ constraint and the H_2 problem. Therefore, they will first be examined as H_∞ optimization problems, rather than mixed

constraints. As mentioned in Chapter 3, the H_∞ constraints need to be augmented with extra inputs/outputs to make them nonsingular. A full order synthesis was then performed. For the tracking constraint, control weighting and weighted sensor noise inputs were added. Both weights were chosen small so that they had little effect on the solution. The resulting full order controller was 13th order with an optimal infinity norm of 1.0.

The purpose of this constraint was to provide the problem with model following characteristics. The specifics on the design setup and weights to accomplish this are detailed in Chapter 3.

4.3.1 Closed Loop Frequency Responses for Tracking Design

Considering the purpose of the constraint, this section will tell the most about how well the mixed controller will do. The frequency responses for stick input are shown in Figure 4.17.

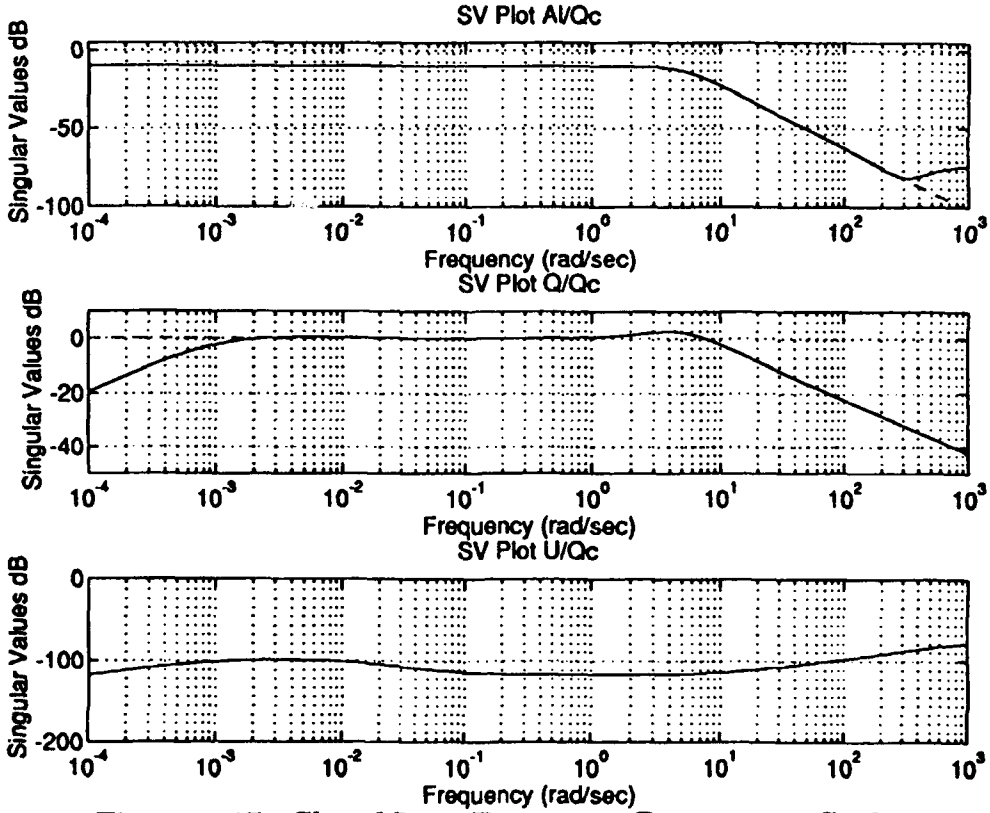


Figure 4.17. Closed Loop Frequency Response to Stick

The ideal model is shown as the dashed lines. The models are followed very closely by alpha and q. The velocity is also decoupled very well. The main deviations occur in the q response below 0.001 rad/sec and in the alpha response above 300 rad/sec. Both of these can be attributed to the fact that the weighting is low in both of these areas (see Figure 3.7). It has already been mentioned that the pitch rate is going to deviate at low frequencies regardless of weighting. It was found that following the model down to 0.001 rad/sec provided excellent tracking, so this is not a large concern. The deviation in the alpha response is most likely due to a high frequency zero in that particular transfer function. This also should not be a problem.

The frequency responses for the throttle command are shown in Figure 4.18. These responses do not appear to do as well as the stick responses. The

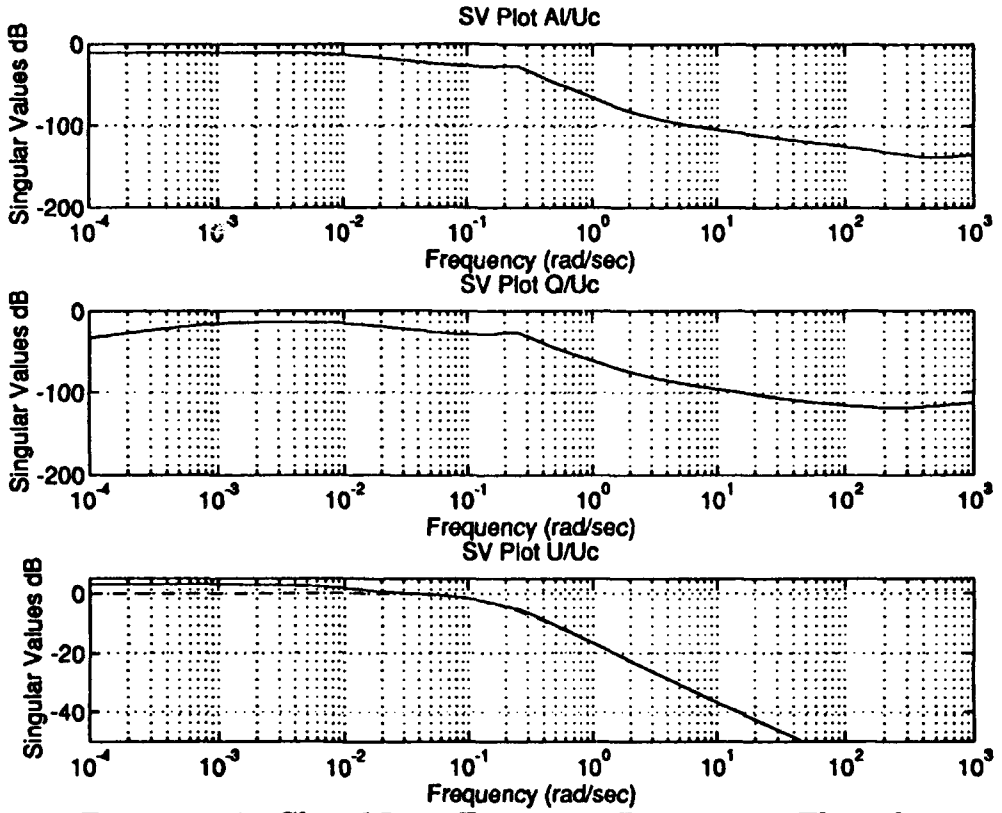


Figure 4.18. Closed Loop Frequency Response to Throttle

decoupling is not as good as the H_2 full order case, but this is because different weights were used. The poor decoupling occurs at low frequency where the weighting is low. There is also a bit of deviation in the U response at low frequency. In the future, this problem can be attacked through the use of different weights in the necessary channels.

4.3.2 Open Loop Analysis for the Tracking Design

The tracking constraint was not meant to provide any type of rejection properties, so it is expected that the open loop analysis would show deficiencies in this area. The open loop plots are shown in Figure 4.19. One thing that

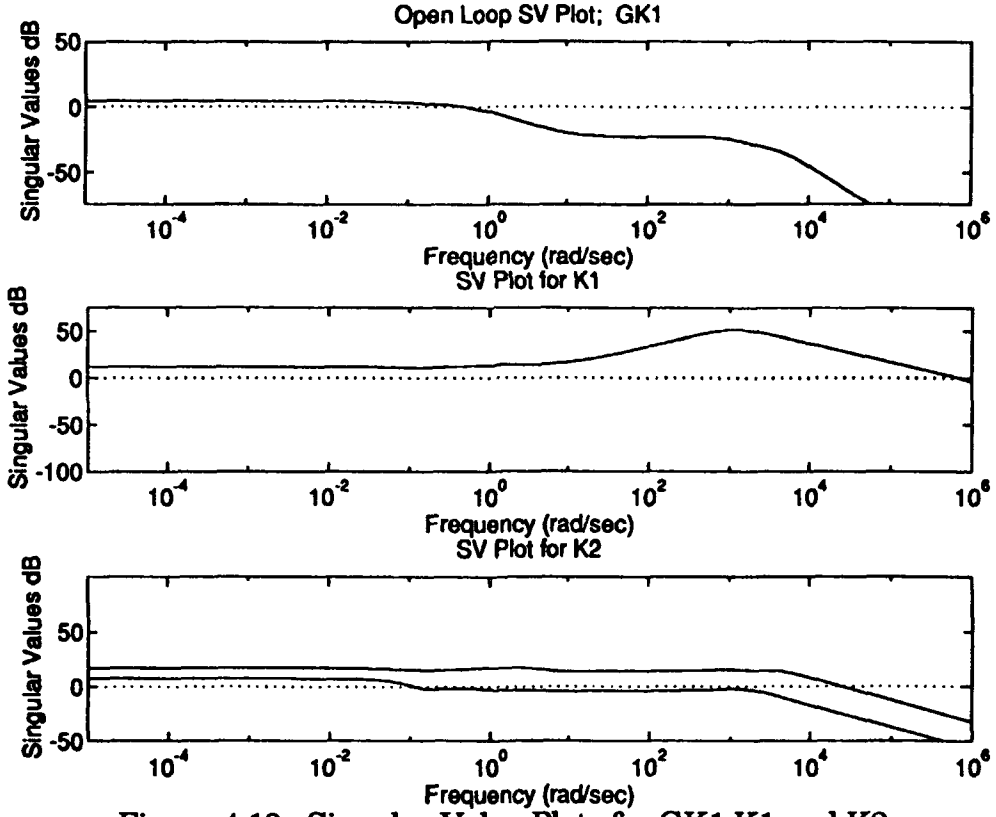


Figure 4.19. Singular Value Plots for GK1,K1,and K2

is noticeable about these plots is the very high bandwidth of the controllers, especially K1. This type of gain at high frequencies usually means poor noise rejection. The open loop transfer function is somewhat deceiving. It would appear that the bandwidth is less than 1 rad/sec which would normally mean

good rejection properties. The thing to notice is that the transfer function does not actually drop off until it reaches 1000 rad/sec, even though it goes below 0 dB much sooner. If a high frequency noise barrier, as used in LQG/LTR, were to be put on the plot it would extend into that barrier. With these things in mind, it is expected that the rejection properties of this controller will be poor.

4.3.3 Step Responses for the Tracking Design

The step responses were found using the same Simulink model as in previous sections. The noiseless responses of the system are expected to quite good based on the results in the closed loop analysis. Figure 4.20 shows that

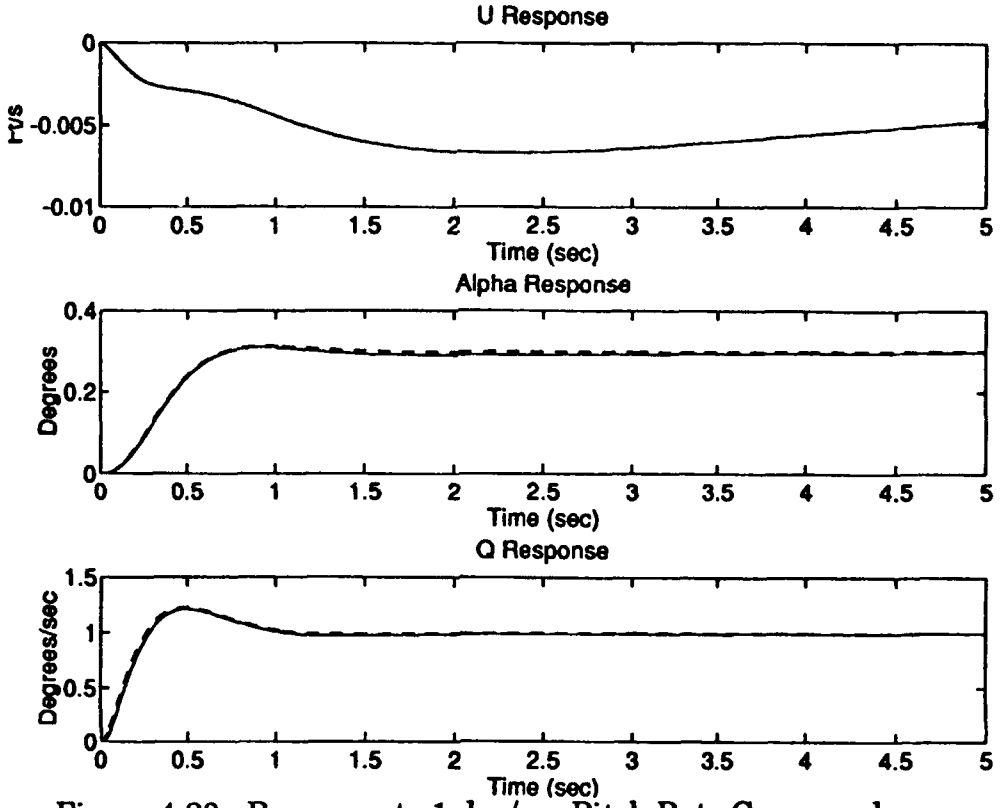


Figure 4.20. Response to 1 deg/sec Pitch Rate Command (no Disturbance or Noise)

this was indeed the case. The ideal models are followed very well. This is what the tracking constraint was supposed to provide.

The actuator deflections are very similar to the H_2 full order response except that the vane provided more velocity control. This can be seen in Figure 4.21.

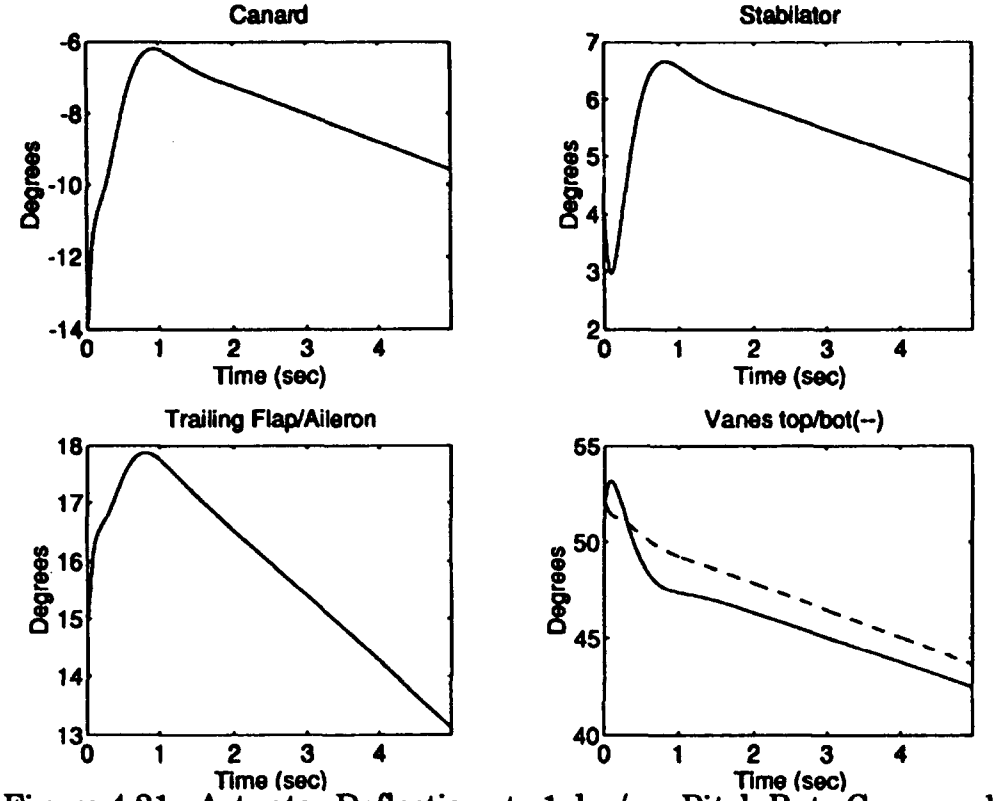


Figure 4.21. Actuator Deflections to 1 deg/sec Pitch Rate Command (no Disturbance or Noise)

As mentioned earlier, this constraint was not meant for any type of noise rejection. Figure 4.22 shows that the design does not provide any type of noise rejection properties. The disturbance and noise make the performance horrible. It does not track (recall that limiters are in place, so this is a nonlinear simulation) and has large amounts of noise in the response. The actuator deflections were found to be very noisy also.

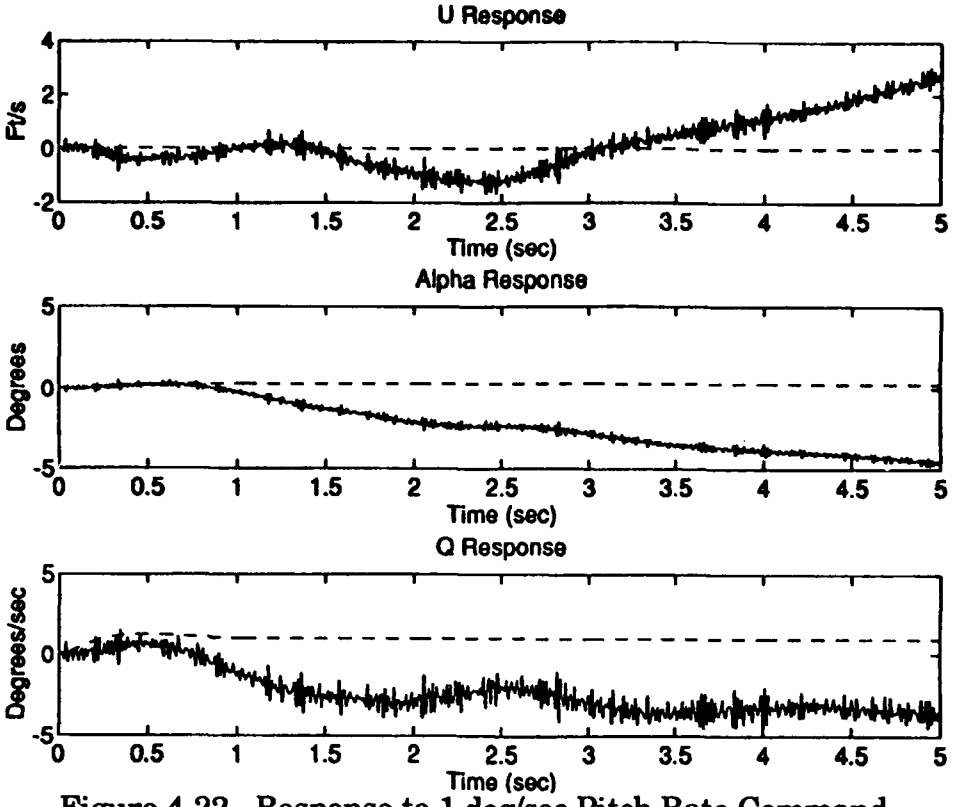


Figure 4.22. Response to 1 deg/sec Pitch Rate Command (Wind Disturbance and Noise)

4.3.4 Complementary Sensitivity and Margins

Looking at Figure 4.23 and based on what was seen in 4.1.4, one would expect these margins to be even worse, since the complementary sensitivity is so large at low frequency. The results are below.

Gain Margin= [-0.00172,0.00172] dB

Phase Margin=±0.011 degrees

These margins are very bad. Therefore, it is expected that the mixed design will also have poor margins with good tracking performance.

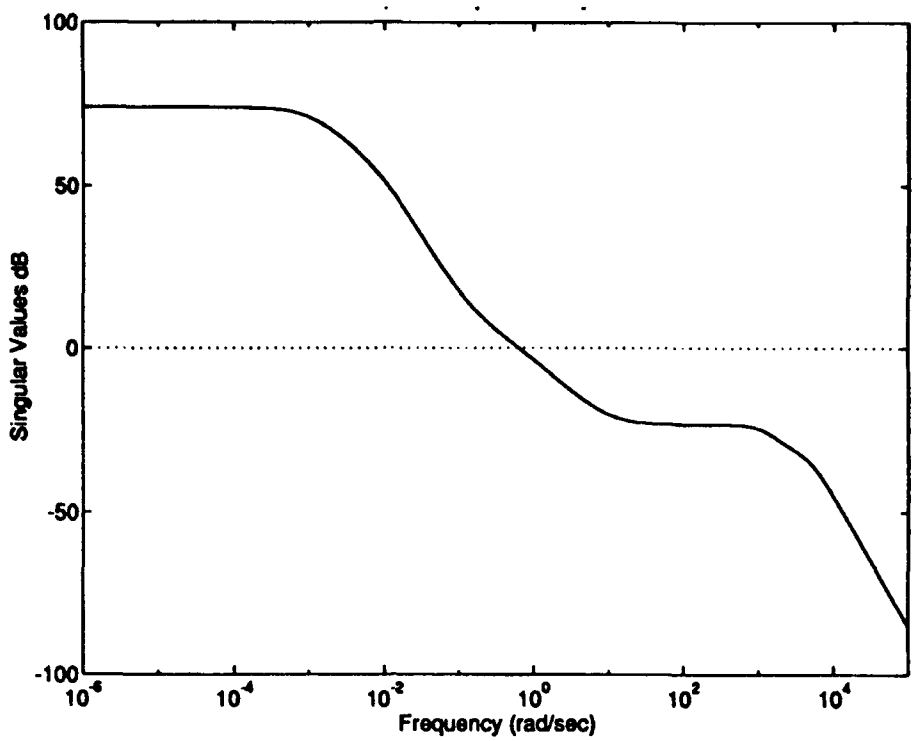


Figure 4.23. Complementary Sensitivity for Tracking Design

4.4 Stability Margin Design

As we have seen previously, the margins of this system have to be addressed in some way. This would be accomplished in a mixed design by including another constraint that minimizes the infinity norm of the complementary sensitivity. The setup for this problem is discussed in Chapter 3. This section examines the complementary sensitivity problem by doing full order H_∞ synthesis. Like the tracking problem, the design model for this problem is singular, but can be made regular by adding in small inputs and outputs.

4.4.1 Open Loop Analysis for the Margin Design

The open loop plots are shown below in Figure 4.24. The most notable

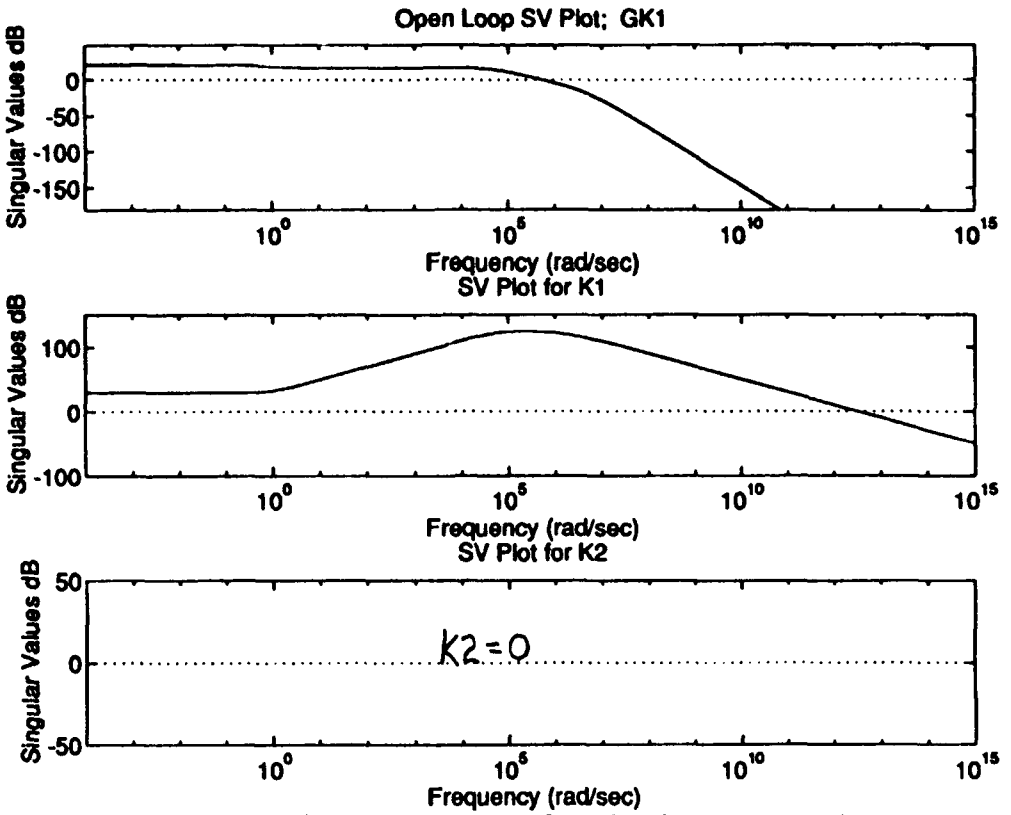


Figure 4.24. Singular Value Plots for GK1, K1, and K2

aspects of these plots are the very large bandwidths. Since noise rejection is not addressed in the design, this is not very surprising. In all control designs, to get one thing you usually have to give up another. It appears that this design gives up noise rejection for increased margins.

4.4.2 Step Responses for the Margin Design

Even though this problem is not designed to give good tracking or regulation, the results may be enlightening. The design was given a 1 deg alpha perturbation (initial condition) to examine this aspect. The design does

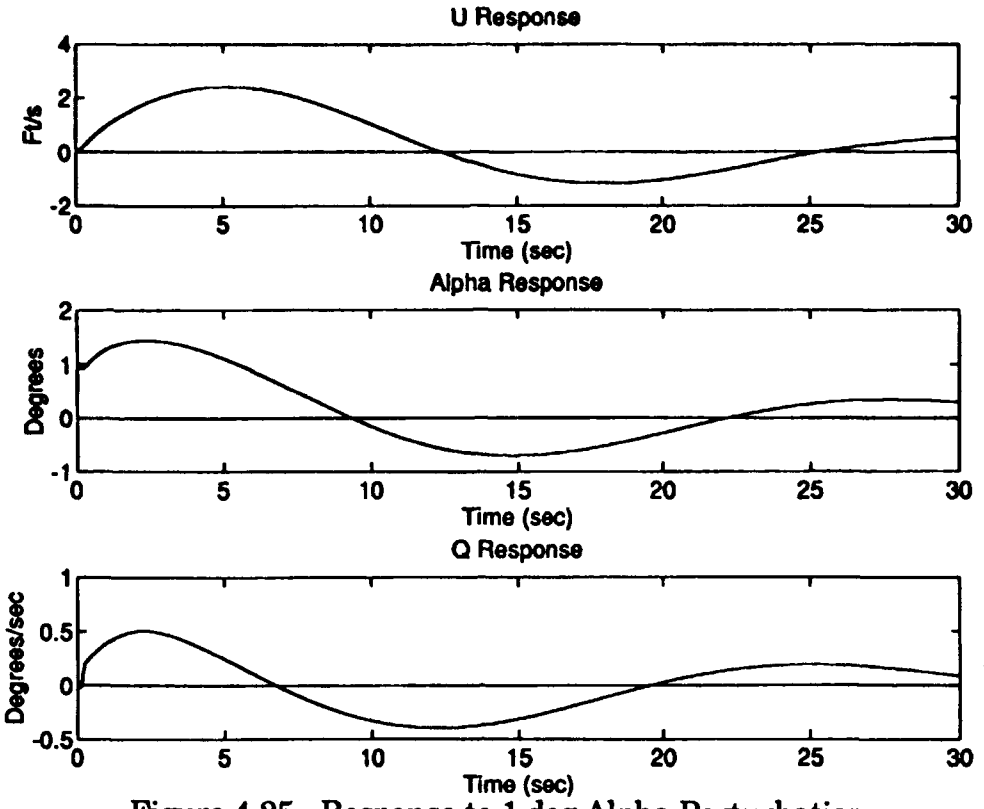


Figure 4.25. Response to 1 deg Alpha Perturbation
(no Disturbance or Noise)

regulate the system (although not well), but there are some interesting things going on. One should notice that the alpha response initially increases before being regulated. This is a result of the nonlinear actuator model that is used. Figure 4.26 shows the actuator deflections. Initially, the actuators are at their maximum rates. This is also not surprising since the actuator weighting on the design model was very small. An analysis was done without the nonlinear actuator dynamics, for which the regulation was better, but the initial actuator deflections were in the hundreds, which is clearly unacceptable.

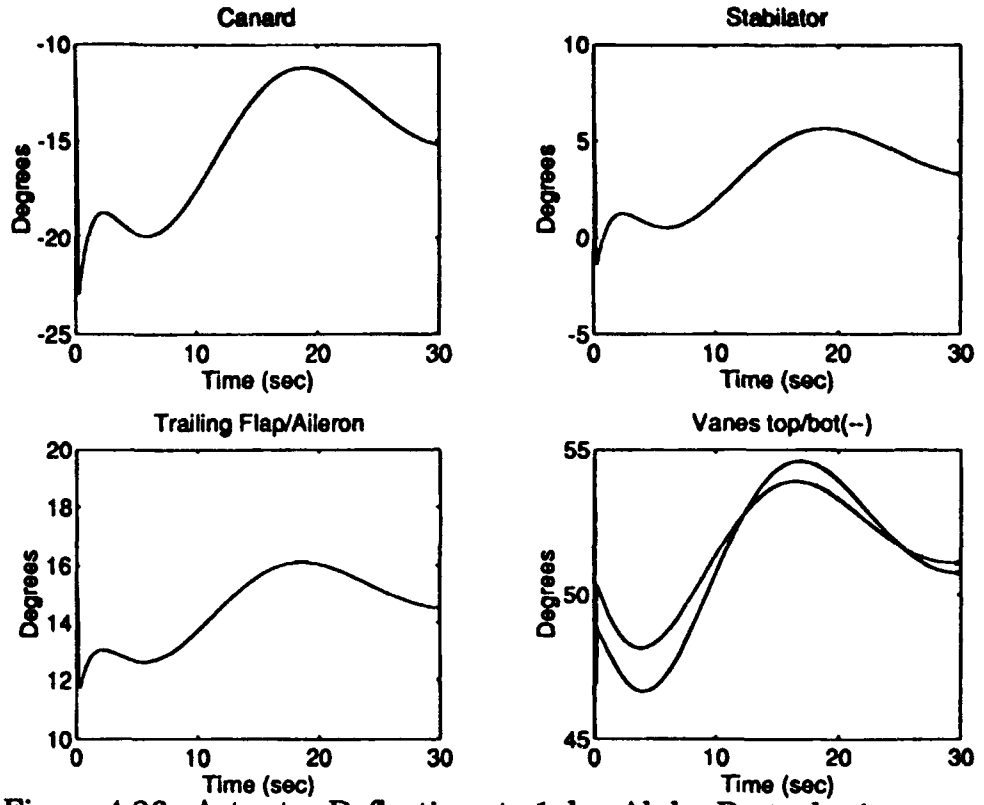


Figure 4.26. Actuator Deflections to 1 deg Alpha Perturbation (no Disturbance or Noise)

As mentioned earlier, this design is not expected to provide any type of rejection properties. This is indeed the case and can be seen in Figures 4.27 and 4.28.

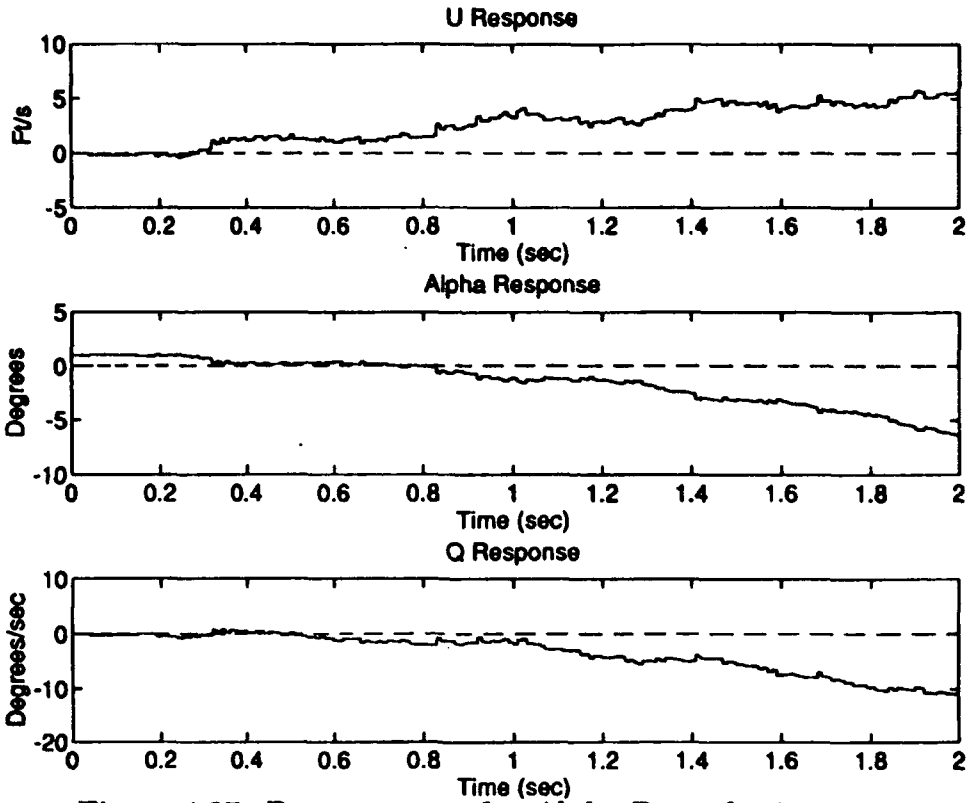


Figure 4.27. Response to 1 deg Alpha Perturbation
(with Wind Disturbance and Noise)

The response becomes unstable and the actuators are at maximum rate all the time. It was found that, with systems like this, the time step size of the simulation makes a big difference in the results. Therefore, these noise responses may not be totally accurate, but it is still obvious that the design does not have good rejection properties. This should be corrected in the mixed design.

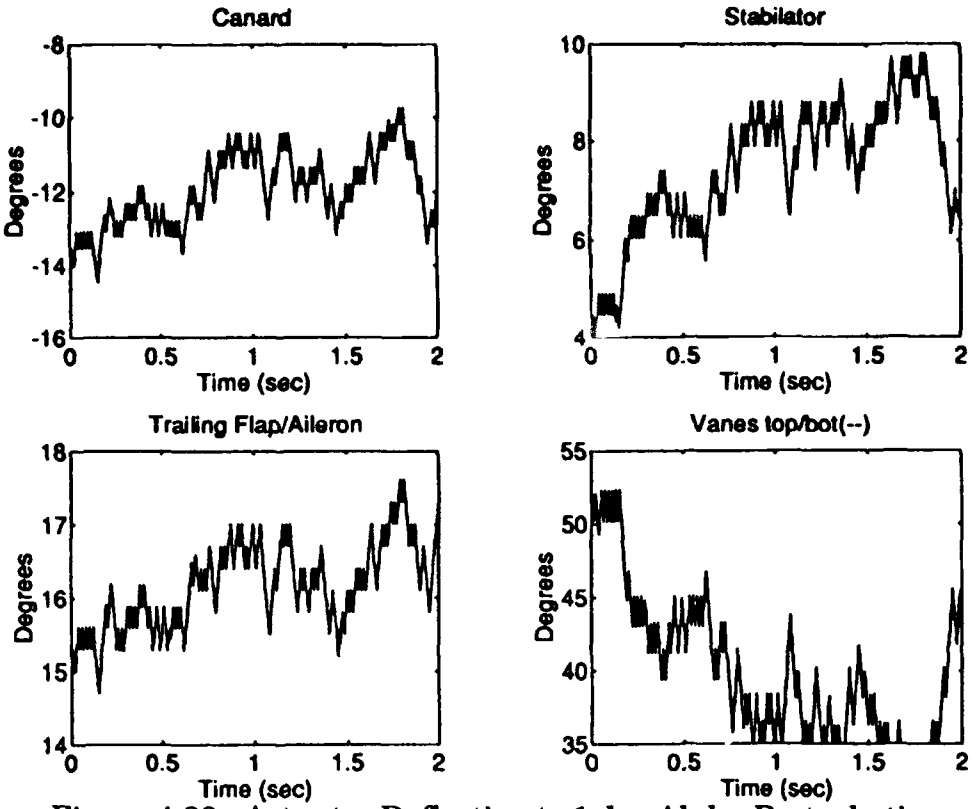


Figure 4.28. Actuator Deflection to 1 deg Alpha Perturbation (with Wind Disturbance and Noise)

4.4.1 Complementary Sensitivity and Margins

This section is the most important aspect of this design since its sole purpose is to improve the margins. The plot of complementary sensitivity is seen in Figure 4.29.

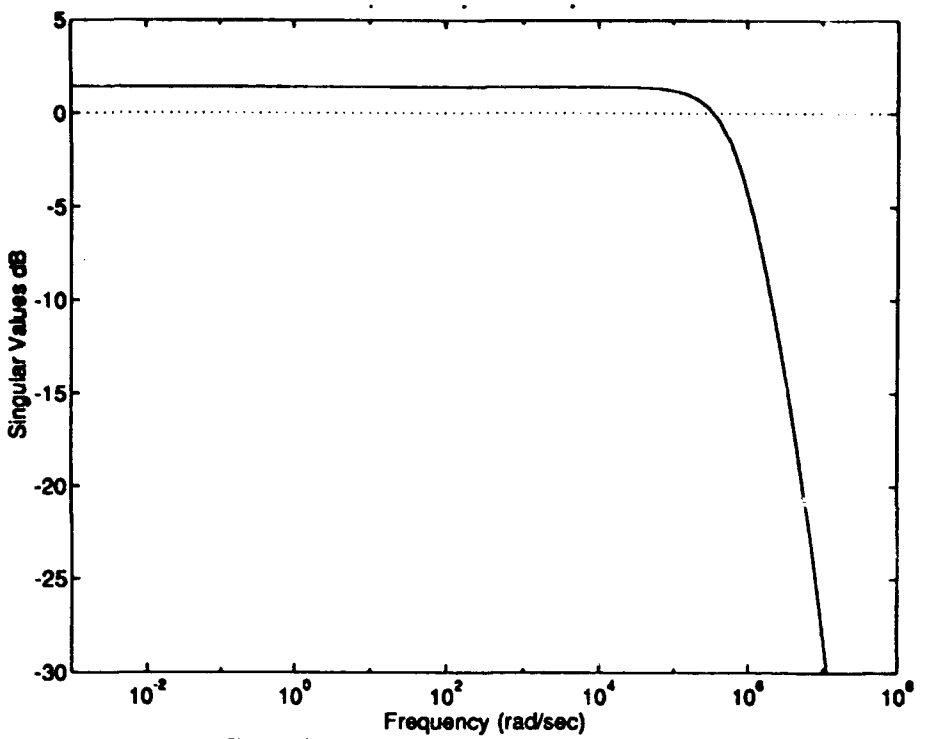


Figure 4.29. Complementary Sensitivity for Margin Design

By inspection, it can be seen that the margins are going to be much better since the complementary sensitivity has been reduced. Like the open loop plots, the frequencies of concern here are very high. This is consistent with other problems. Usually, when one lowers a norm in one region, it increases in some other. It appears that lowering the norm on the complementary sensitivity has spread it out over a larger frequency range. The results of this are improved margins, which are as follows:

Gain Margin=[-16.51,5.35] dB

Phase Margin= ±50.3 degrees

These margins are quite good and should help the mixed problem with its margins.

Using the information found in the analysis of the previous designs, the mixed solutions were found. The results of the mixed synthesis are documented in the next chapter.

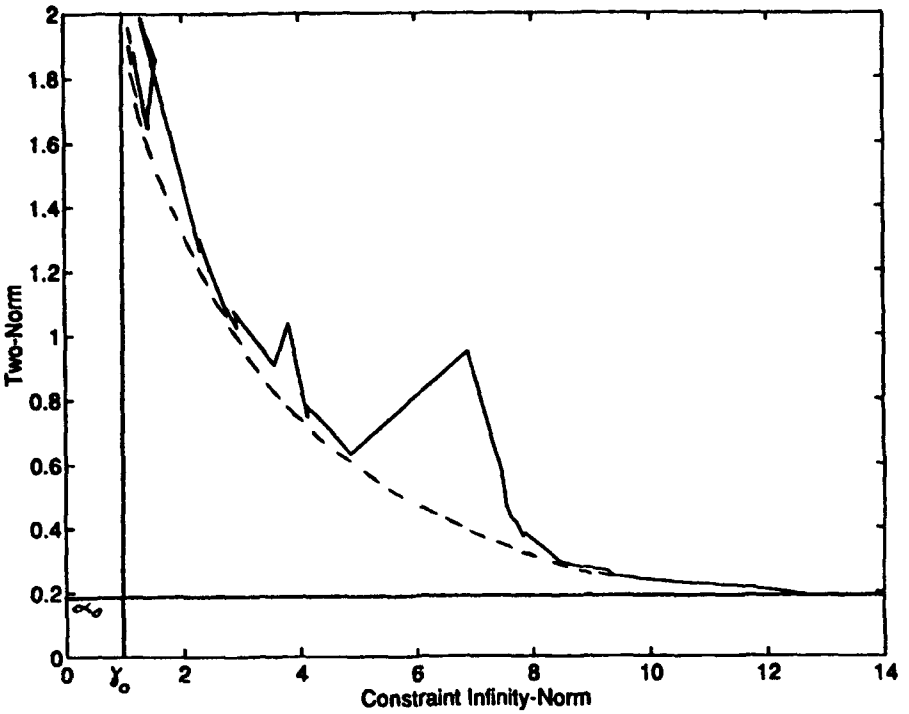


Figure 5.1. Two-Norm vs. Infinity Norm; Tracking

The starting point of the curve (H_2 optimal) is α=0.193 and γ=12.65. The absolute lowest possible infinity-norm of the constraint, given the nature of the pitch rate response and weighting, is 1. One can see that the plot does resemble Figure 2.6, although there are some deviations. Considering that this is a 60 parameter (number of elements of the controller) numerical search, this type of numerical jumping is not surprising. The main thing is that the basic curve shape can be obtained from the figure, which is shown by the dashed curve in Figure 5.1. The final point of the plot, which is used as the reduced order controller, has α=1.88 and γ=1.20.

The program attempts to reduce the infinity-norm of the constraint. Figure 5.2 shows the maximum singular value plots for the constraint at each iteration.

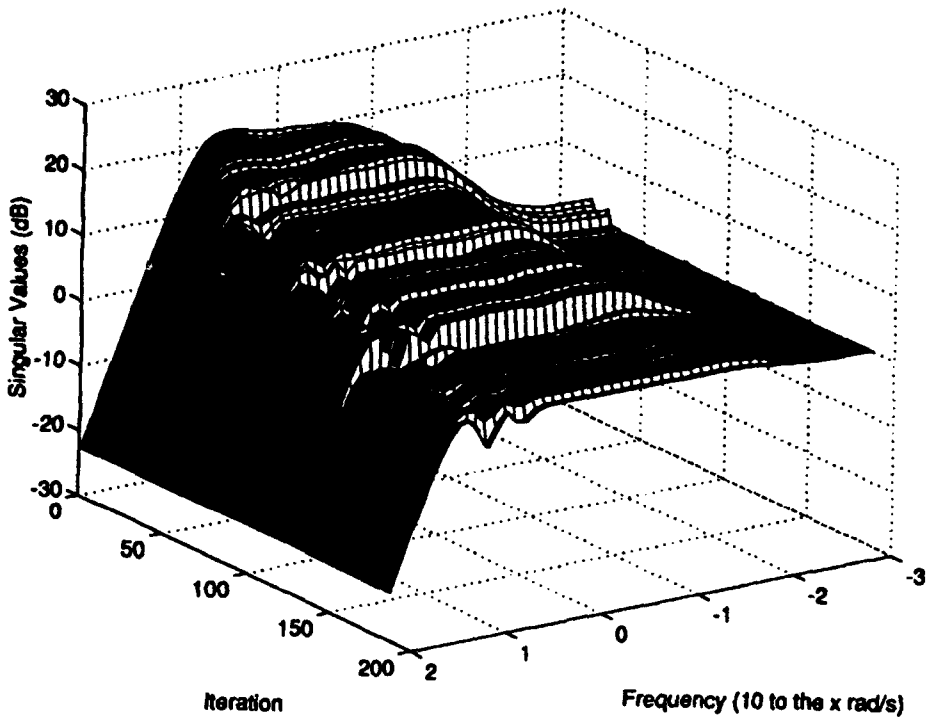


Figure 5.2. Singular Value Plots at Each Step for Tracking

One can definitely see how the weighting decreases the norm over the frequency of interest. It is also interesting to note that the movement of the optimization is not always smooth. Often the program gets stuck at a certain point and then jumps down a bit. One should also notice the waves that are produced at around 1 to 10 rad/sec at the higher iterations. This is often seen in numerical solvers, but tends to be a bit of a problem for this program. The analytical method for finding the derivative of the infinity-norm is based on the peak singular value and its frequency. When there are multiple peaks of the same magnitude at different frequencies, the effectiveness of the program decreases.

5.1.1 Closed Loop Frequency Response

The purpose of this constraint was to provide the reduced order mixed controller with model following characteristics. From the full order analysis, we

found that the tracking problem does indeed cause the output to follow the model. Figure 5.3 shows the frequency response to stick command.

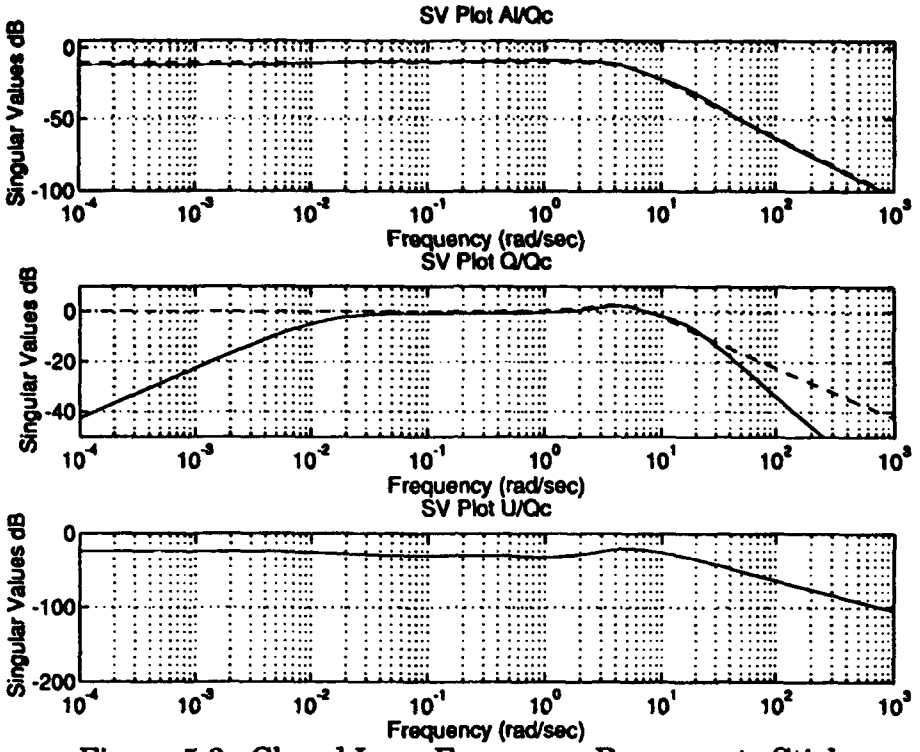


Figure 5.3. Closed Loop Frequency Response to Stick

Considering that this controller is a reduction in order of 9 over the full order controller, it does very well. The alpha response is very close to the ideal model and velocity is decoupled very well. The weakest aspect of this seems to be the pitch rate response. It is quite close to the ideal from 0.1 to 1 rad/sec, the curve is just below 0 dB here and well below for frequencies less than 0.1 rad/sec. This means that the steady state error will be nonzero but small. The deviations below 0.01 and above 30 rad/sec are due to the lack of weighting in those areas (see Figure 3.7). In future work it may prove to be helpful to extend the weighting down further in frequency to help improve the tracking performance.

Figure 5.4 shows the closed loop frequency response to throttle inputs.

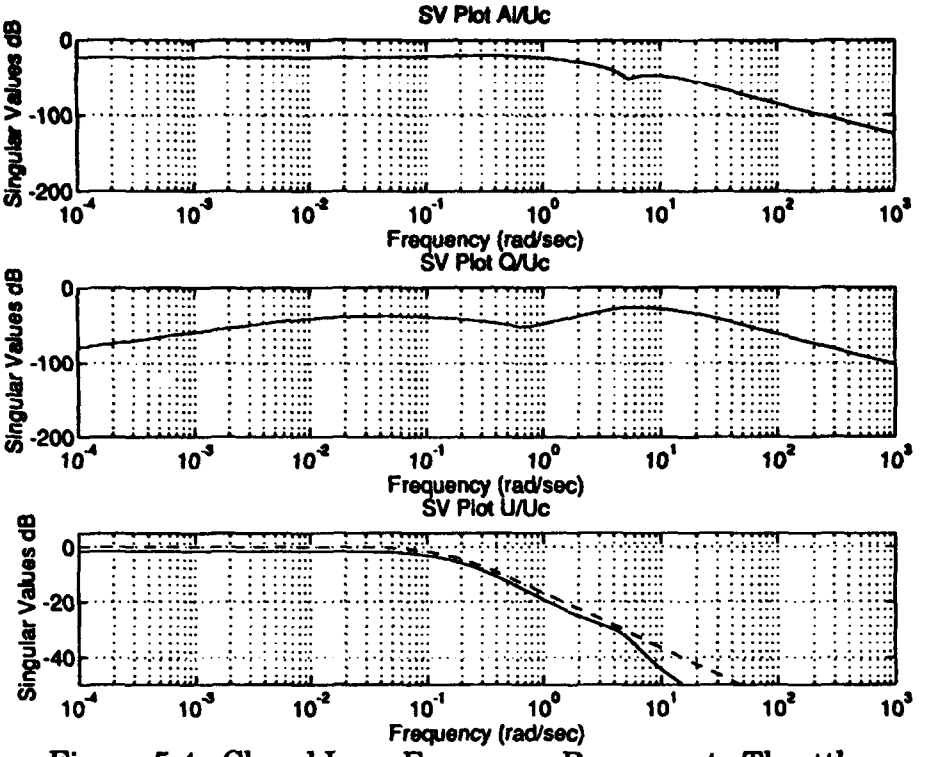


Figure 5.4. Closed Loop Frequency Response to Throttle

Again, this controller does very well considering the large drop in order. Actually, comparing this to Figure 4.18 shows that the decoupling of pitch rate and alpha is better than achieved by the full order controller. The velocity response is slightly worse than the full order but is still quite close. Comparing this controller to the full order tracking controller shows that not a lot of performance is lost when the order is reduced and rejection properties are addressed.

5.1.2 Open Loop Analysis

As mentioned earlier, the purpose of the H_2 part of the problem is to reduce the order and provide rejection properties. To get an idea of how well this is done, the open loop singular value plots are examined. These can be seen in Figure 5.5. The open loop bandwidth is around 10 rad/sec. This should

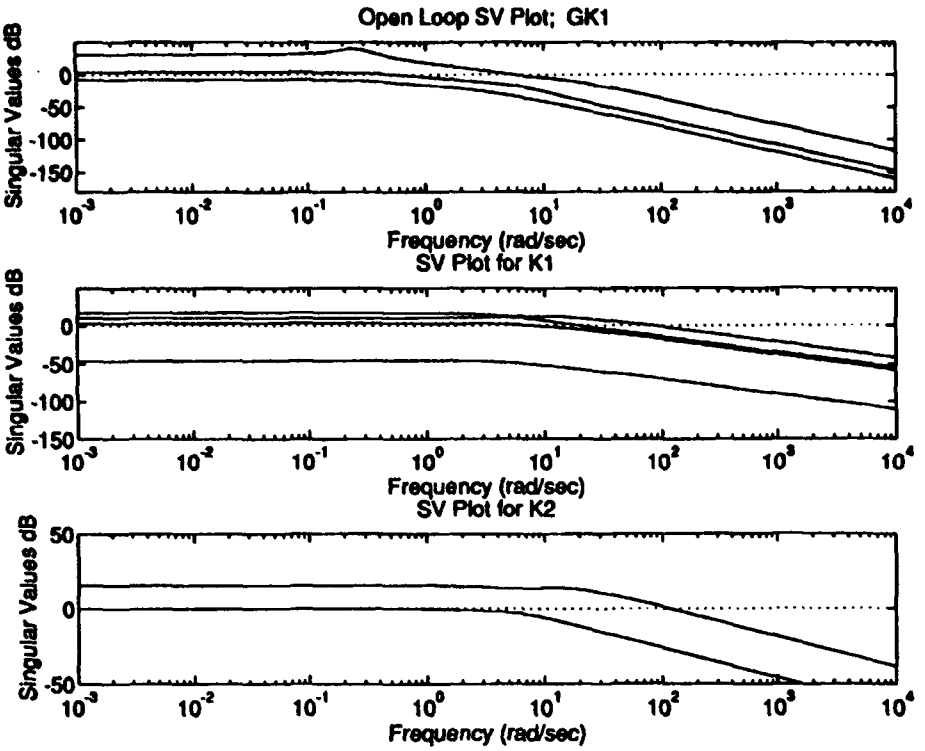


Figure 5.5. Singular Value Plot for GK1, K1, and K2

produce adequate rejection. The bandwidths of the controllers are fairly high at around 100 rad/sec, but this was also seen in Chapter 4. Comparing this figure to Figure 4.9, one can see that the open loops are somewhat similar but the mixed problem is using a lot more of the information. One can also see that K2 resembles the K2 produced by the full order tracking controller, but has a much smaller bandwidth. There is truly a mixing of the characteristics of the different problems.

From the previous two sections, it appears that the synthesis has produced a direct reduced order controller that provides model following and has good rejection properties. The next section will test these observations in the time domain.

5.1.3 Step Responses

The time responses were produced by a 1 deg/sec step pitch rate command using the Simulink model discussed previously. The noiseless responses are shown in Figure 5.6. The loss in performance is more easily seen

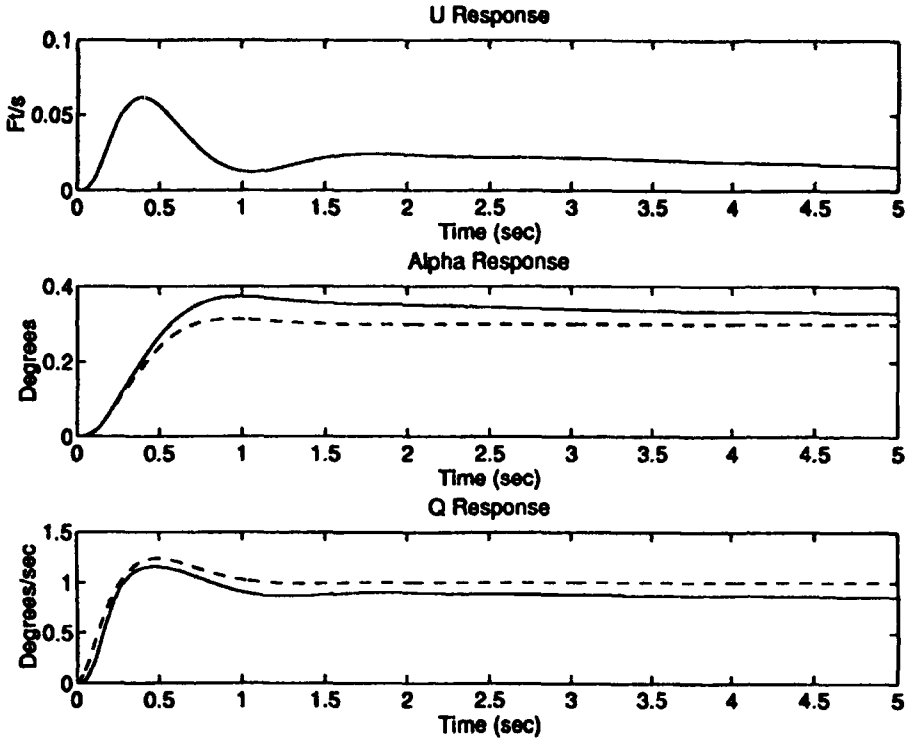


Figure 5.6. Response to 1 deg/sec Pitch Rate Command (no Disturbance or Noise)

here than in the frequency responses. The biggest loss is in the pitch rate response, which is what was expected. Even though there is a loss in performance, it is still quite good. The steady state error is only around 10%. The deviations in performance in the alpha and velocity responses are totally acceptable. The change in velocity is essentially unnoticeable, and the alpha response deviation is less important, given feedback from test pilots.

The actuator deflections are shown in Figure 5.7. These deflections are

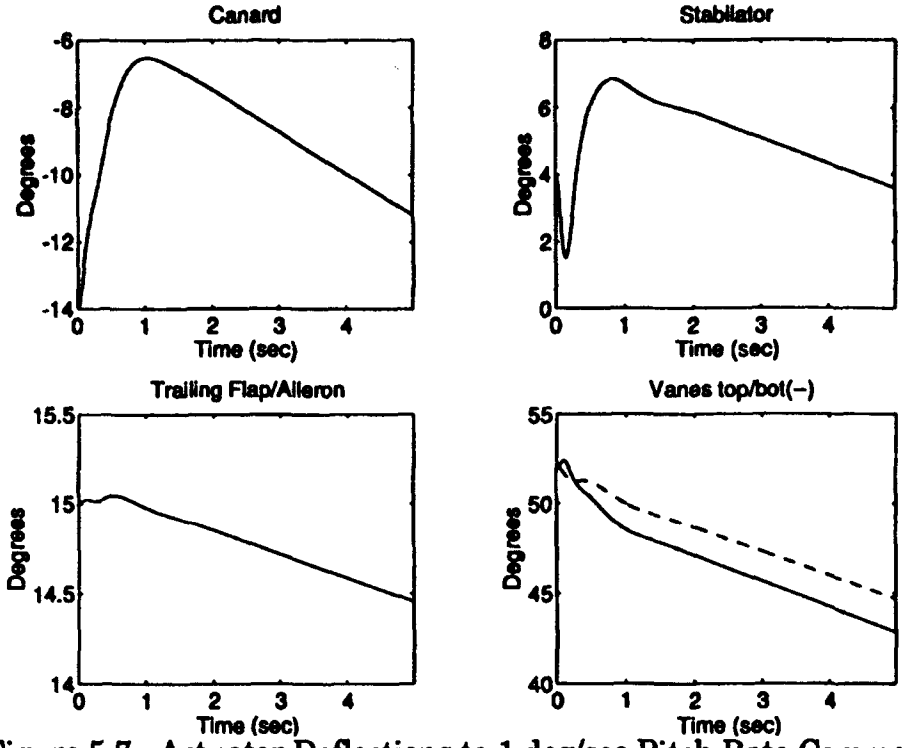


Figure 5.7. Actuator Deflections to 1 deg/sec Pitch Rate Command (no Disturbance or Noise)

very similar to the deflections shown in Figure 4.21. The main difference is that the flap/aileron is used less and the stabilator is used more. There also appears to be less pitching moment produced by the vanes.

The big question that remains is: does the controller have good rejection properties? The answer to this is in Figure 5.8. This compares very well to the

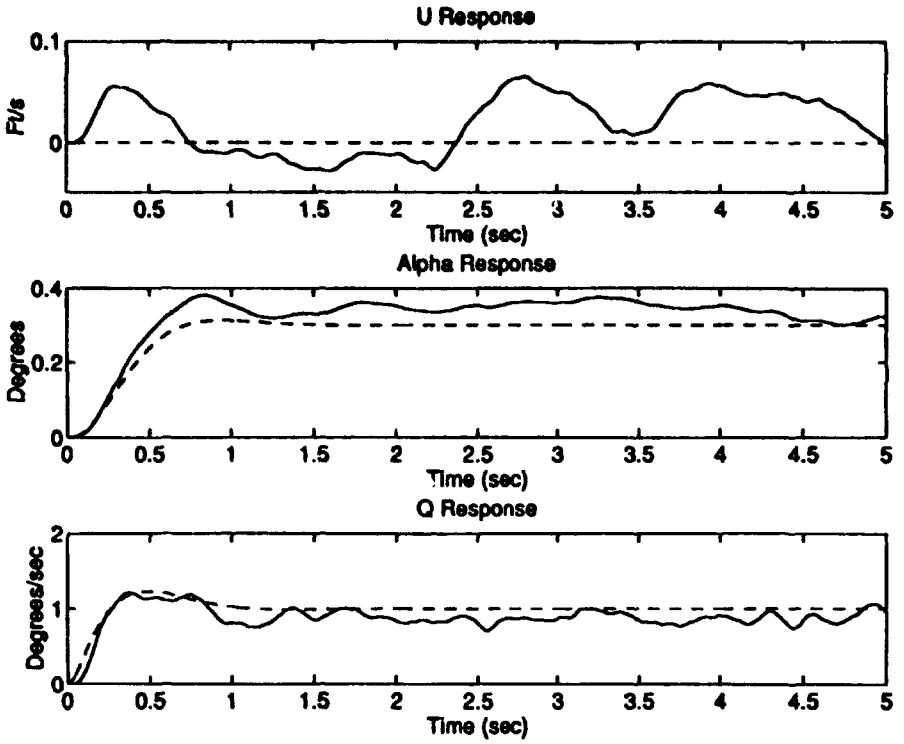


Figure 5.8. Response to 1 deg/sec Pitch Rate Command (Wind Disturbance and Noise)

response of the full order H_2 controller shown in Figure 4.6. This response could be considered better in that the high frequency jitter in Figure 4.6 is not present here. This helps make a case for separating the different constraints into their specific tasks. This helps eliminate the conflicts and cross terms that are present in an all-in-one design and appears to improve the design. The reduced amount of noise can also be seen in the actuator deflections in Figure 5.9.

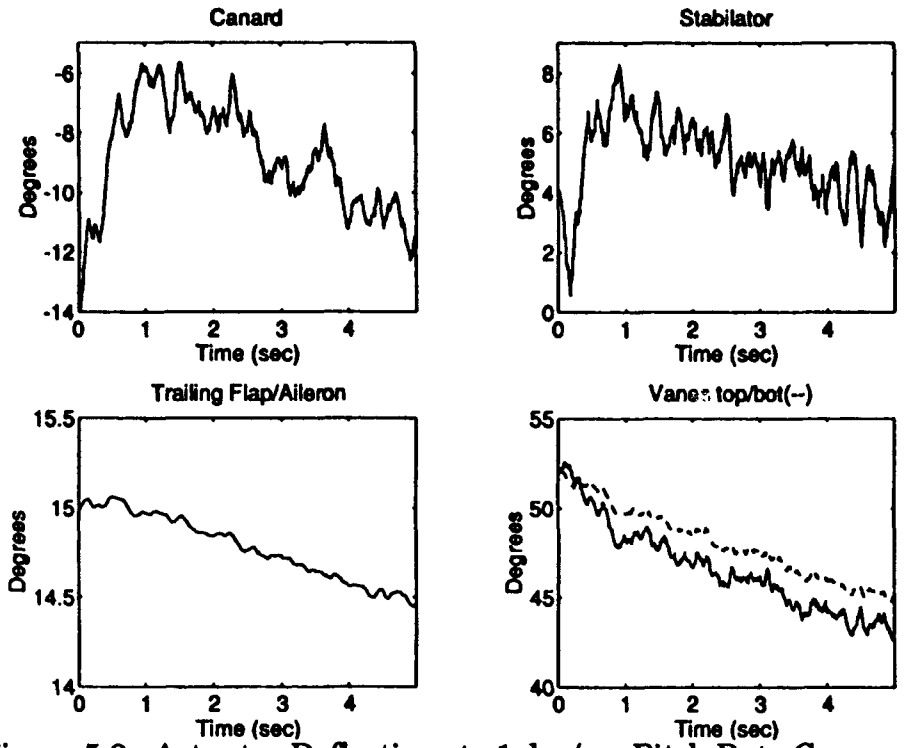


Figure 5.9. Actuator Deflections to 1 deg/sec Pitch Rate Command (with Wind Disturbance and Sensor Noise)

5.1.4 Complementary Sensitivity and Margins

The last aspect of this design to be examined is the margins. Figure 5.10 shows the complementary sensitivity. When looking back at the results in

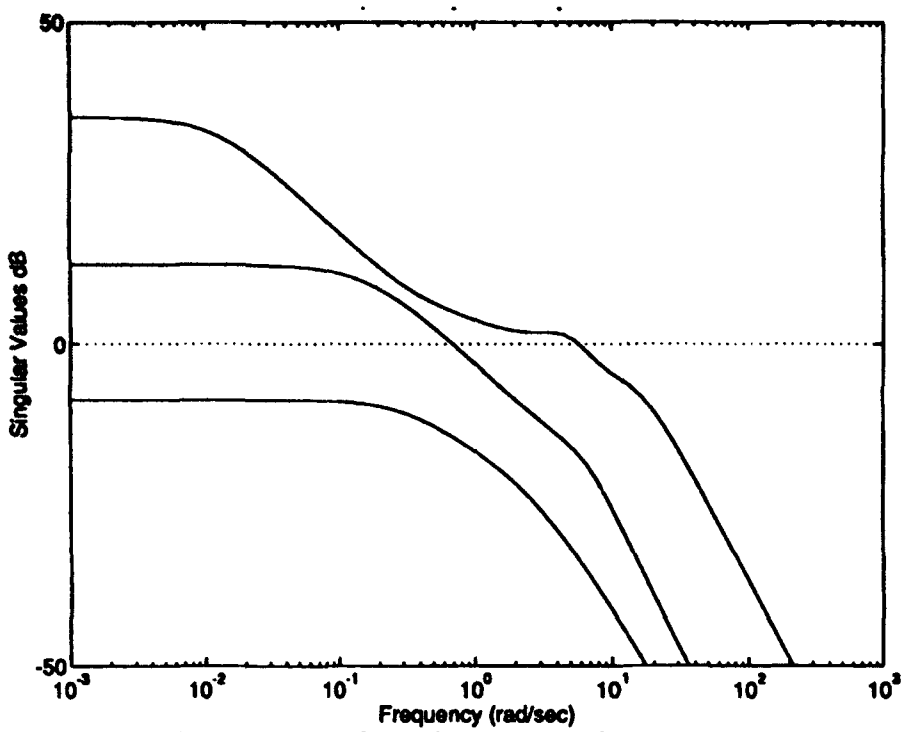


Figure 5.10. Complementary Sensitivity

Chapter 4, one can see that this is also a mixing of the characteristics of the components. The most notable is that it appears that the infinity norm of the complementary sensitivity is lowered and should therefore produce better margins than just the tracking problem alone. The margins for various points in the optimization are shown in Table 5.1.

Table 5.1
Mixed Margins; 4th order Tracking

Step #	α	γ	Lower Gain Margin (dB)	Upper Gain Margin (dB)	Phase Margin (deg)
1	0.193	12.65	-3.51	2.49	19.11
10	0.238	9.94	-3.49	2.48	19.04
20	0.418	7.71	-0.64	0.60	4.08
30	0.632	4.88	-1.05	0.94	6.56
100	1.078	2.75	-0.39	0.38	2.54
187	1.880	1.20	-0.15	0.15	0.98

In general it can be seen that as the solution approaches the optimal value of the tracking constraint, the margins approach the tracking problem margins. The final margins are better than those in the full order tracker, but they are

still very bad. It is interesting that the margins jumped up at an infinity-norm of 4.88. Looking back at Figure 5.1, notice that this point is where the two-norm jumps back down toward optimal, which could explain the increase.

These margins are clearly unacceptable for any type of real application. The next section will look at the results of trying to increase the margins.

5.2 Results of 4th Order Mixed Controller with Margin Constraint, Case 1

As seen in Chapter 4, the margins of this problem can be increased with a constraint on the complementary sensitivity. The mixed problem was run in a similar manner to the mixed problem with tracking constraint, but there was one major difference. Normally, the program is run from the optimal H_2 solution, but in this case it had to be run from a controller produced in the mixed problem with tracking constraint. The reason for this is that the optimal H_2 solution had an infinity norm of around 6000. The program would not start from this point. Using a controller that was produced by the mixed tracking problem, the initial infinity norm was around 600. The routine ran very well after that until it got down to around 100. It was run down to around 89.3, which is close to the full order optimal of 82. The solution path is shown in Figure 5.11.

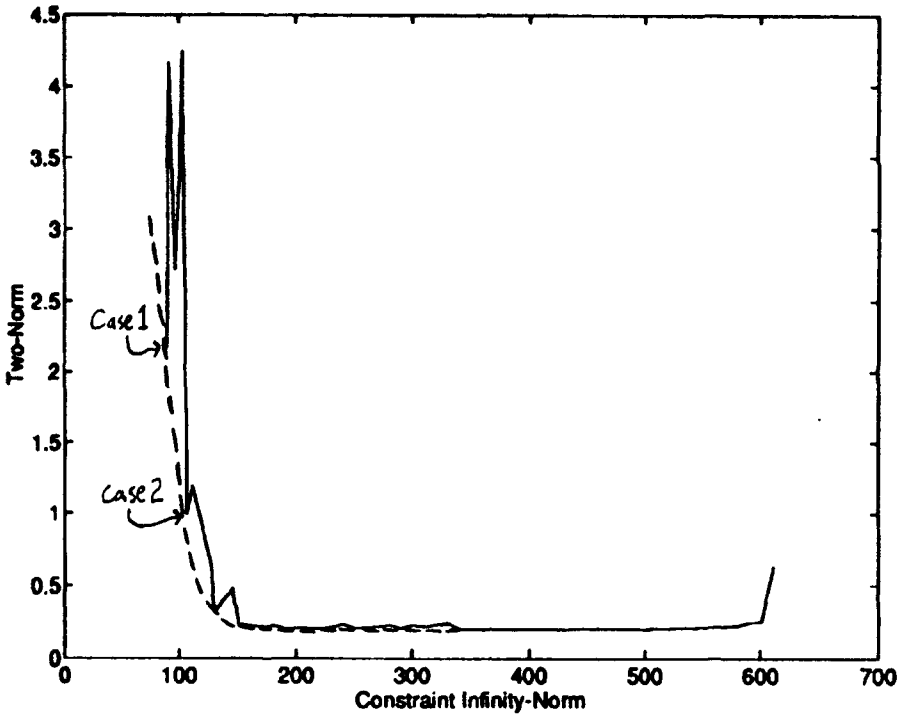


Figure 5.11. Two-Norm vs. Infinity-Norm; Margin Constraint

Initially, there is a large drop in two-norm down to the curve described in Figure 2.6. It then ran smoothly until it got to around 100. Figure 5.12 shows the maximum singular value plots for each iteration. This figure shows much

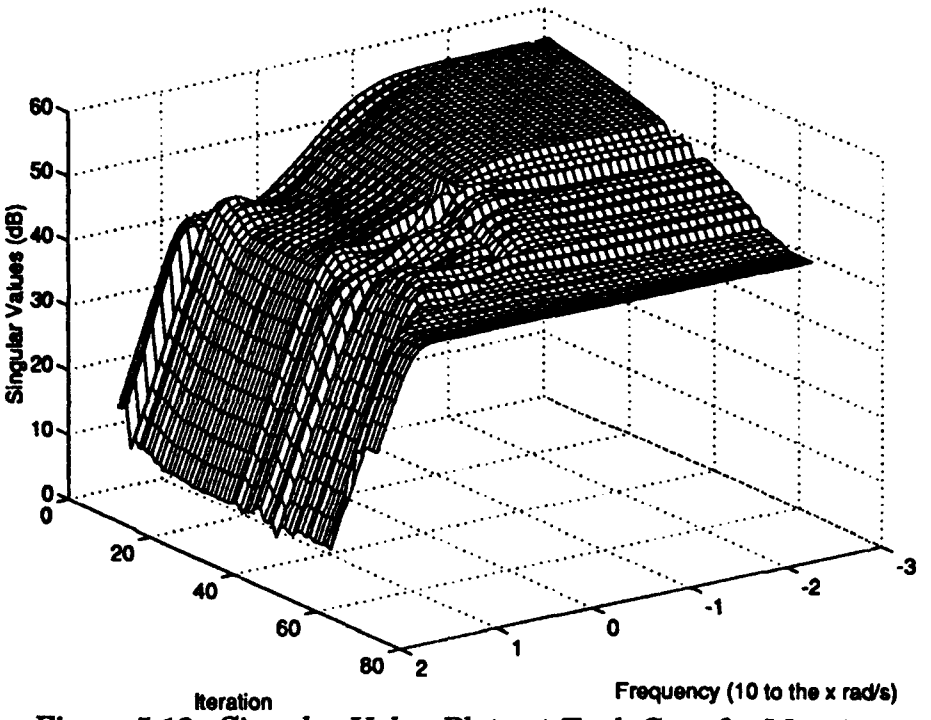


Figure 5.12. Singular Value Plots at Each Step for Margins

of the same information as Figure 5.11, but here one can see where the maximum errors are occurring. It was shown previously that the poor margins were due to the high gains in complementary sensitivity at low frequency. This figure shows that the synthesis for this design does the most work at low frequency. Therefore, the margins should be much better.

The analysis of this design consists of looking at two of the solutions. The first is the final solution which has a γ of 89.3 and an α of 2.18. The second, which is the next point along the boundary curve, is shown on Figure 5.11 which has a γ of 107 and an α of 0.99.

5.2.1 Open Loop Analysis, Case 1

The constraint should help improve the margins of the system, but it should not prevent the controller from having good regulator properties. Therefore, the open loop plots should resemble those of the regulator. These can be seen in Figure 5.13.

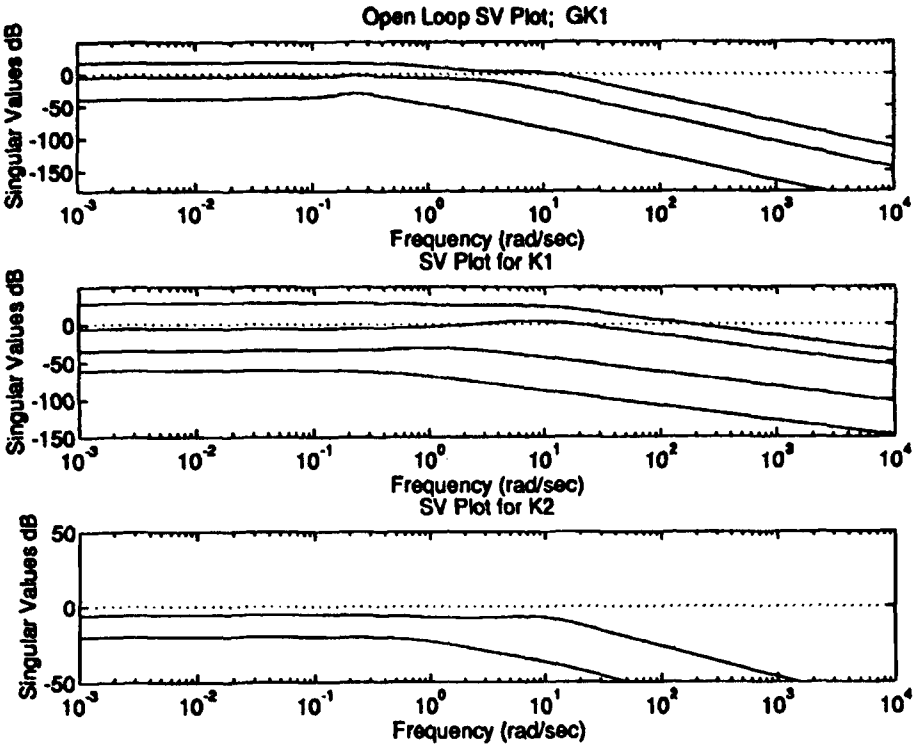


Figure 5.13. Singular Value Plots for GK1, K1, and K2

These plots do resemble the regulator but use more information. This is what was seen in the mixed tracking problem. The plot for K2 in the margin design was zero (Figure 4.24). This would be the same here if run from H_2 optimal, but since the initial controller was from the tracking problem, there are non-zero singular values for K2 which the optimization ignores (this is a 1 DOF problem). Another interesting thing is that the very high bandwidths from the constraint are not present, but their influence can be seen in the relatively high bandwidths of the open loop systems.

From this, one could expect potentially degraded regulator properties from this controller. This will be examined in the next section.

5.2.2 Time Responses, Case 1

As with the full order margin design analysis, the system was given a 1 deg alpha perturbation. The result of this can be seen in Figure 5.14. The

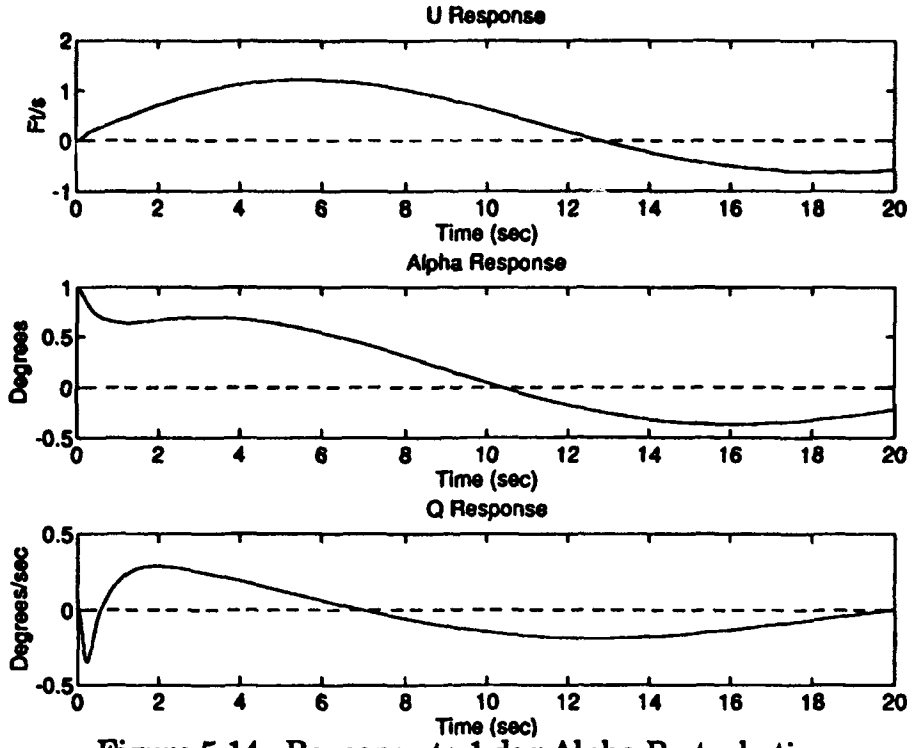


Figure 5.14. Response to 1 deg Alpha Perturbation
(no Wind Disturbance or Noise)

figure shows that the controller does indeed regulate the outputs, but again not well. The response resembles Figure 4.25 which is from the full order margin design controller. This response, however, has a better initial response and less deviation. This deviation is again produced by the nonlinear actuator dynamics. This is shown better in Figure 5.15. As with Figure 4.26, the initial

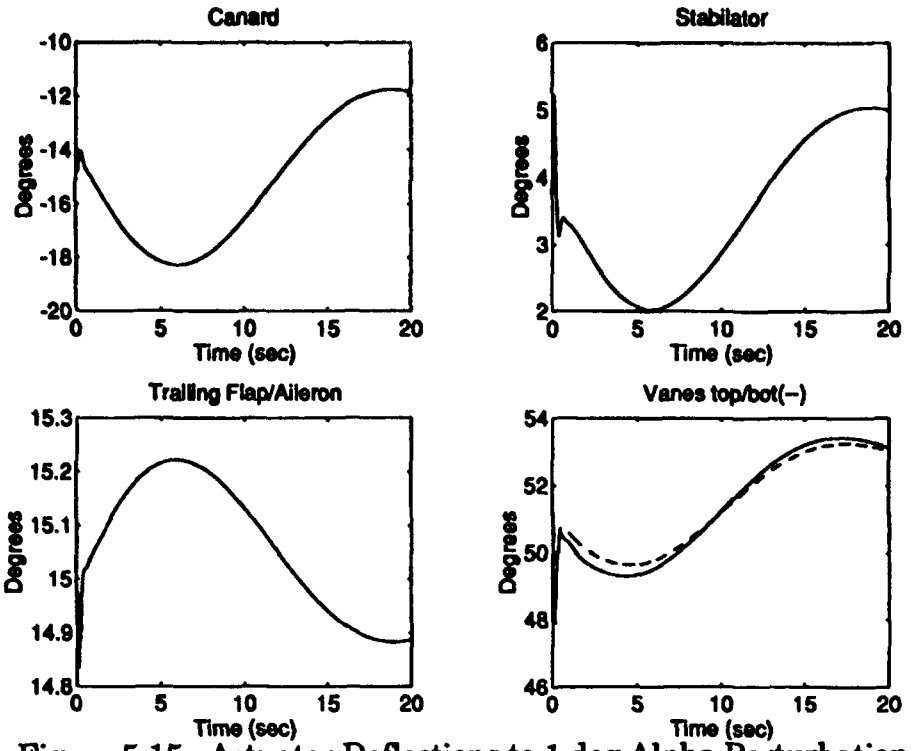


Figure 5.15. Actuator Deflections to 1 deg Alpha Perturbation (no Wind Disturbance or Noise)

deflections are at the maximum rates. This causes the deviation. It would appear that even though the frequency response analysis shows that the controller is more like the regulator design, time responses show that it acts in a similar manner to the full order margin design controller.

This was also evident when noises were introduced. Noises tended to destabilize the system, but not as badly as what was seen in the full order margin design. Like that simulation, time step size made a big difference in the response of the system, so the accuracy of these responses are questionable. For that reason, no plots of the response are included, and it is simply noted that this design does not have good noise rejection properties. This is likely due to the higher bandwidths produced by pushing down the complementary sensitivity. It should also be noted that the final two-norm of the design was around 10 times that of optimal. From this point of view, it makes sense that

the noise rejection would be bad. Looking at Figure 5.11, one could choose a controller with a higher infinity-norm (potentially worse margins) but lower two-norm (better noise rejection). This will be done is Section 5.3. The purpose of showing this mixed design was to see how good the margins could be, which is discussed in the next section.

5.2.3 Complementary Sensitivity and Margins, Case 1

The full order margin design showed that the margins [illegible] be substantially increased. This will show to be true for the mixed controller.

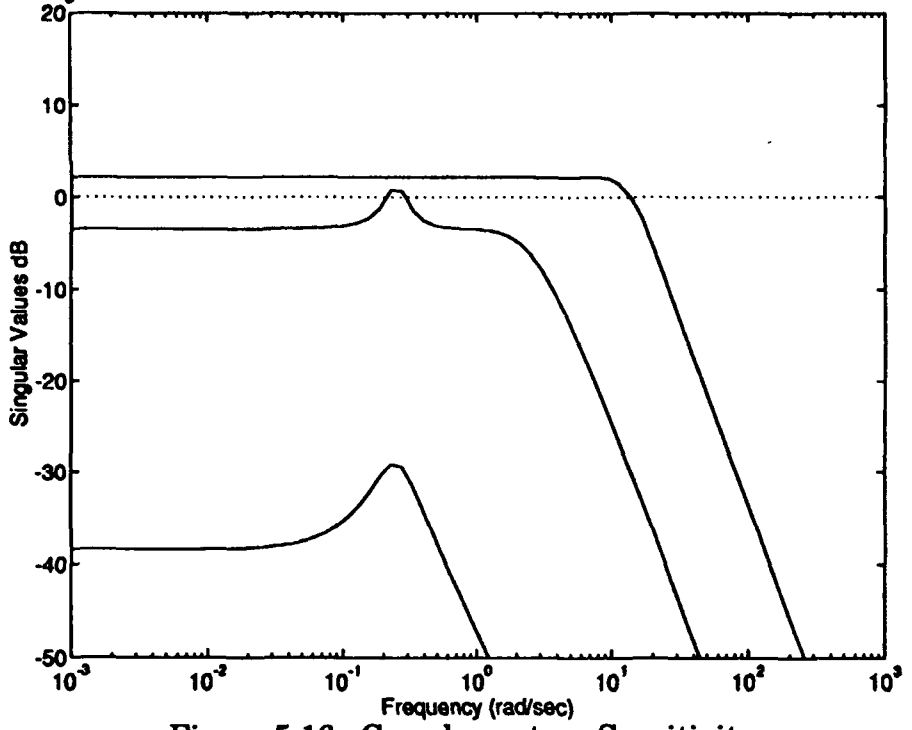


Figure 5.16. Complementary Sensitivity

Figure 5.16 shows that the margins are going to improved since the infinity-norm of the complementary sensitivity is clearly reduced. These plots are both similar to the plots for the regulator and the full order margin design but use more information, which was seen previously. The actual margins produced are shown in Table 5.2. The final margins are quite close to the full order

Table 5.2
Mixed Margins; 4th order Margin

Step #	α	γ	Lower Gain Margin (dB)	Upper Gain Margin (dB)	Phase Margin (deg)
1	0.63	610.8	-1.06	0.94	6.56
12	0.20	500.8	-1.31	1.14	8.03
22	0.20	400.8	-1.67	1.40	10.04
32	0.23	300.8	-2.30	1.82	13.37
42	0.21	200.8	-3.72	2.60	20.07
55	3.36	100.7	-10.32	4.59	40.67
63	2.18	89.3	-13.31	5.03	46.14

margin design and are reasonably good. This table also points out the potential for finding a controller with acceptable margins and good noise rejection somewhere between γ=200 and γ=100. This is examined in the next section.

5.3 Results of 4th Order Mixed Controller with Margin Constraint, Case 2

The point chosen for this analysis is located at the point γ=107.0, α=.99. The purpose of looking at this point is to show that there can be a controller with both good margins and regulator properties.

At this infinity-norm value the margins should be decent. The complementary sensitivity is shown in Figure 5.17. There is a noticeable

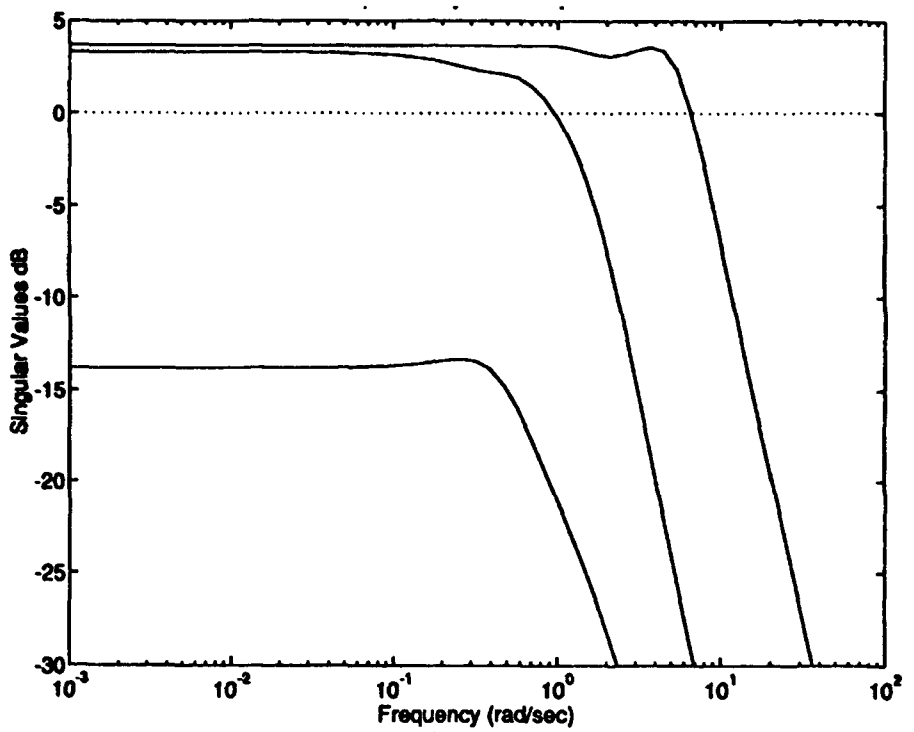


Figure 5.17. Complementary Sensitivity

difference between this plot and that shown in Figure 5.16. It is hard to see from the figures, but the highest value of complementary sensitivity has increased. Therefore, the margins should be lower. This was found to be true and are shown below

Gain Margin=[-9.23,4.37] dB

Phase Margin= ±38.18 degrees

The lowering of the margins was expected, but did the regulator performance get better? The open loop plots are shown in Figure 5.18. The

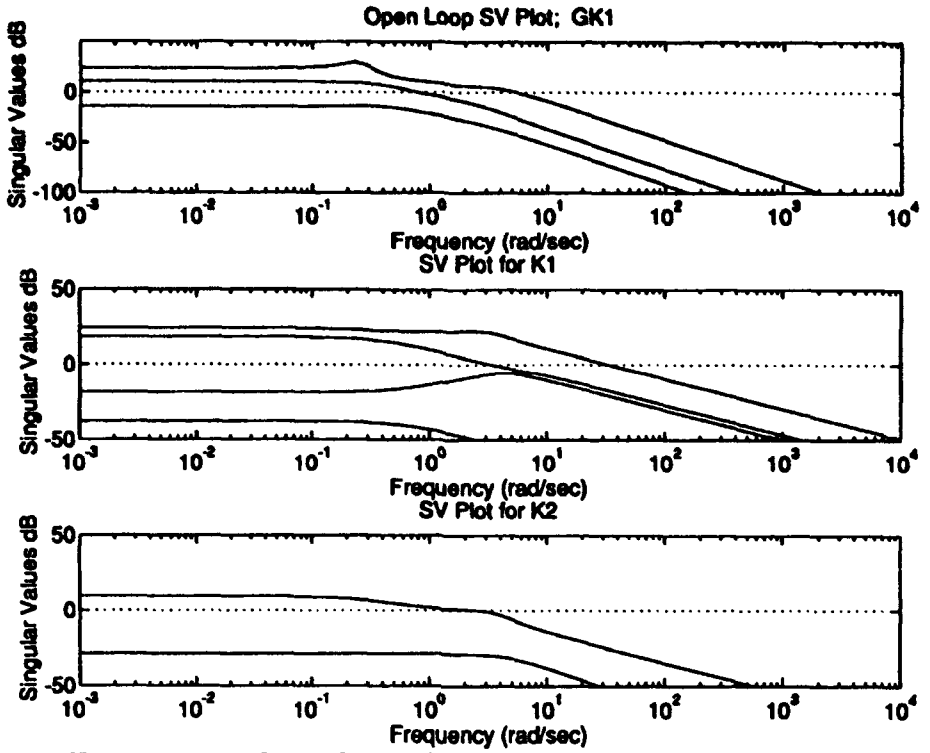


Figure 5.18. Singular Value Plots for GK1, K1, and K2

bandwidths of GK1 and K1 are smaller than those of Case 1. This should provide better noise rejection which will be seen in the time responses. The noiseless time responses in Figure 5.19 show that this controller is a much better regulator than Case 1.

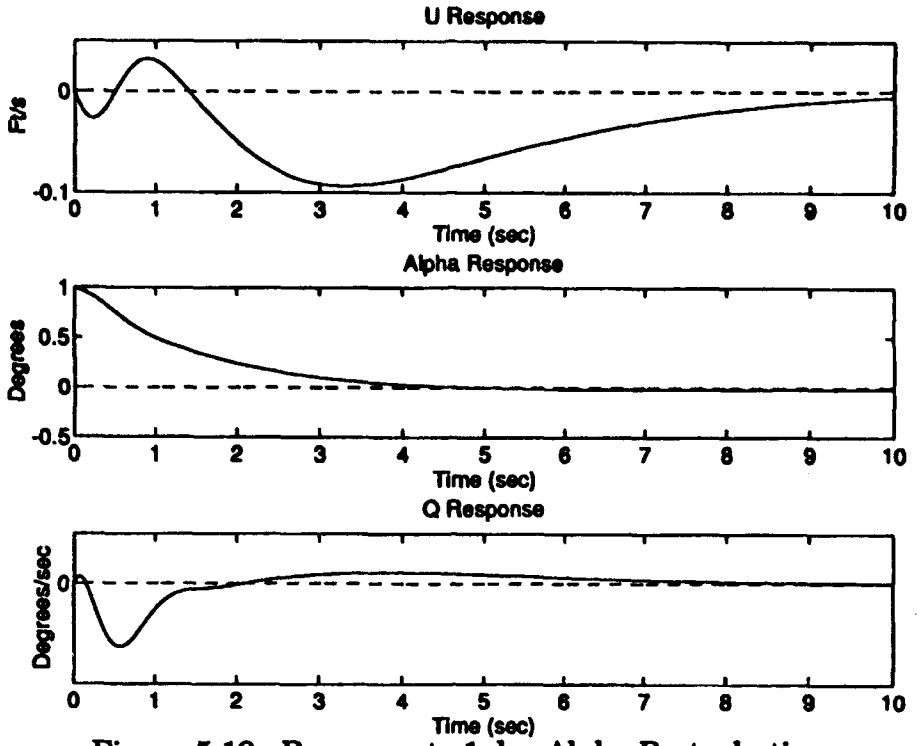


Figure 5.19. Response to 1 deg Alpha Perturbation
(no Wind Disturbance or Noise)

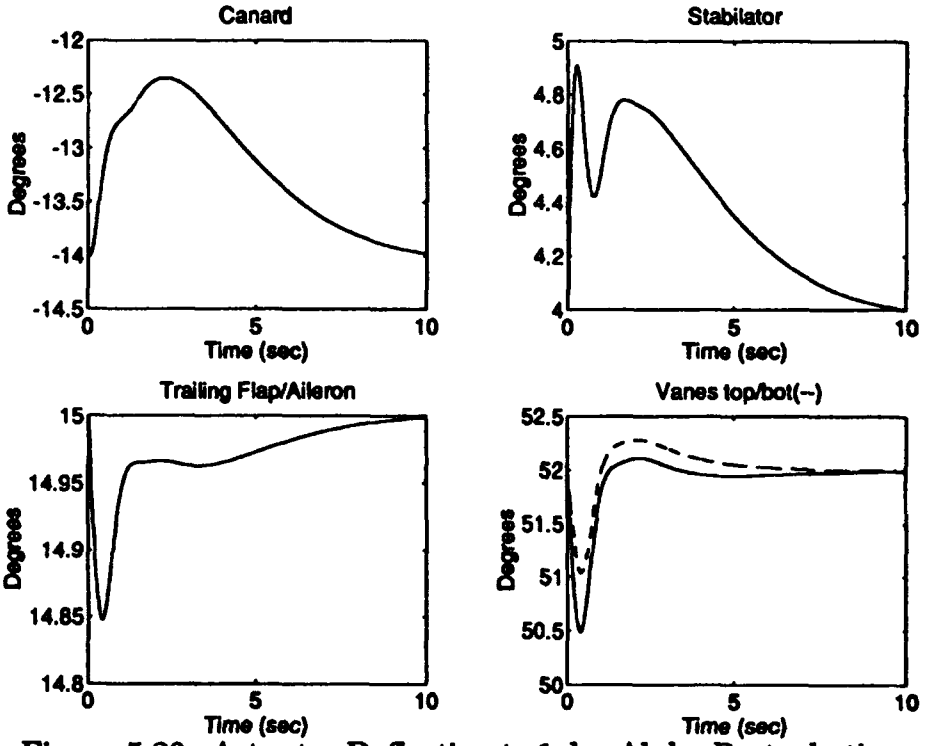


Figure 5.20. Actuator Deflection to 1 deg Alpha Perturbation
(no Wind Disturbance or Noise)

The actuator deflections show that this controller does not have the same problems that arose from nonlinearities as Case 1. This is obviously an improvement over the Case 1.

The noise rejection properties can be seen in Figure 5.21. This is a vast

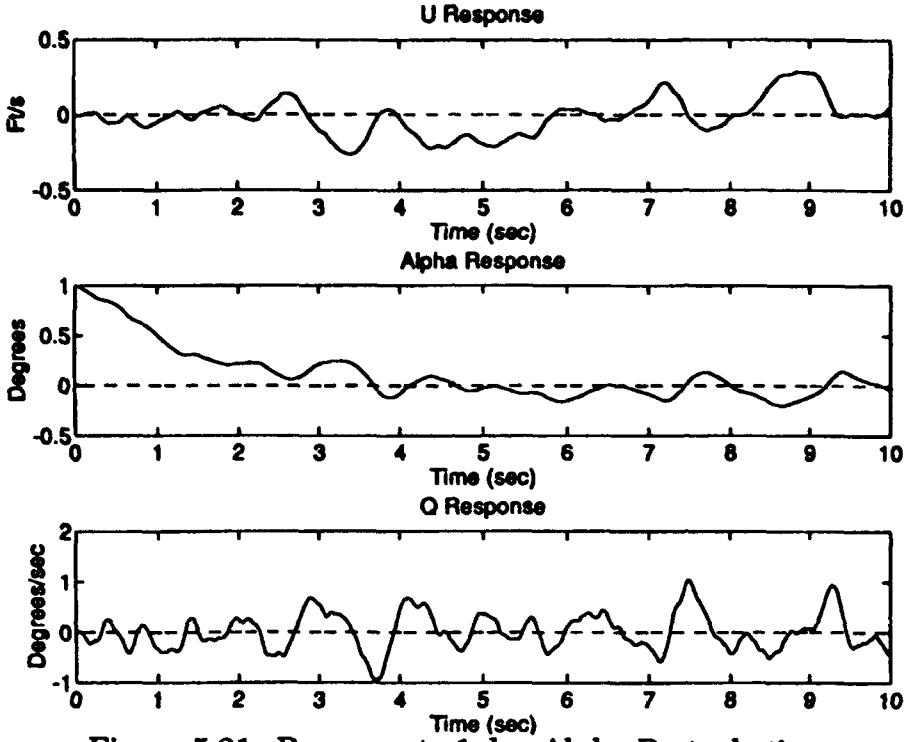


Figure 5.21. Response to 1 deg Alpha Perturbation (with Wind Disturbance and Noise)

improvement over Case 1, which shows that a controller with good margins and regulator properties can be found by choosing a controller that is not so far up on the solution curve. If the designer is willing to give up more of the margins, additional improvements in regulator performance can be attained.

5.4 Results of 3rd Order Mixed Controller with Tracking Constraint

One advantage of using this numerical approach is that any order controller can be designed, so long as a stabilizing controller exists at the chosen order. The order of the controller is determined by the order of the initial guess. In the first case, the initial guess was the H_2 optimal controller, which was of order four. Below fourth order, the theory cannot guarantee that the

solution will have certain properties. Little has been done to examine this problem, so a run of order three was conducted to see if any interesting results could be obtained.

The initial guess was obtained from one of the fourth order controllers by using SCHMR.M in Matlab. This produces a reduced order model of the given controller. Care was taken to make sure that the controller was stabilizing. The program was run in the same way as the 4th order mixed tracker and the solution path is shown in Figure 5.22. Surprisingly, this ran very well all the

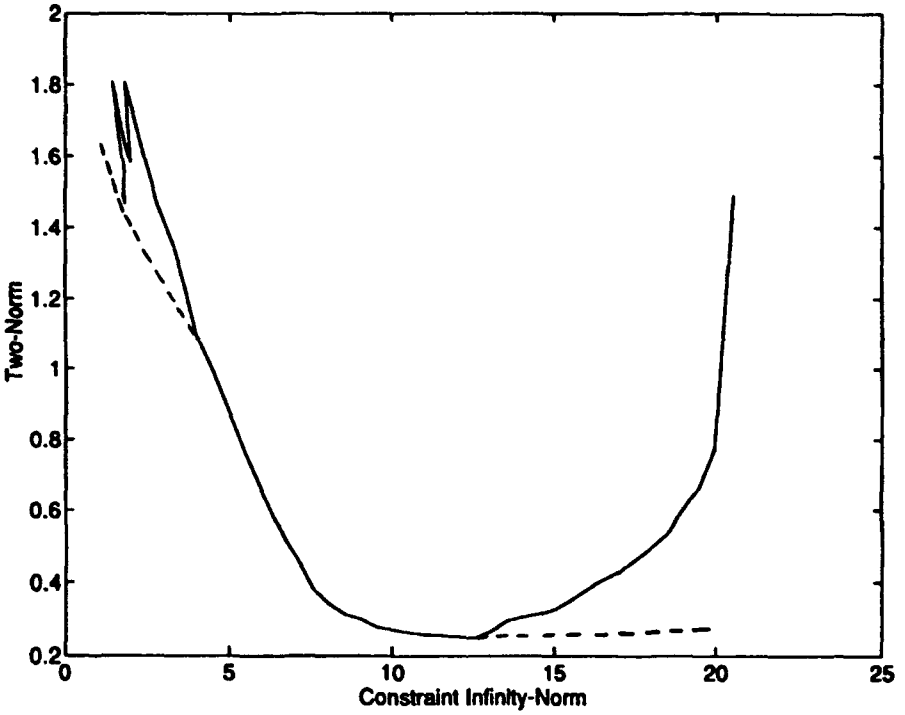


Figure 5.22. Two-Norm vs Infinity-Norm; Tracking 3rd Order

way down to an infinity-norm of about 1.7. As can be seen in the figure, after this point the numerics were not as stable. A solution was found with an infinity-norm of 1.4, which was as low as the solution would go. This is the point where the solution has the best performance. The only drawback was that it had a two-norm of around 1.8. This was found to be poor for noise rejection, so the solution was run further to attempt to reduce the two-norm. This was accomplished, but at the cost of increasing the infinity-norm (reduced

performance). The final solution used in this analysis has an infinity-norm of 1.74 and a two-norm of 1.49. It was found that the loss in performance was small in comparison to the improvement in noise rejection.

The maximum singular value of the tracking constraint problem is shown in Figure 5.23. This figure shows the same trends as Figure 5.22. One can also see the waves that were present in the fourth order solution.

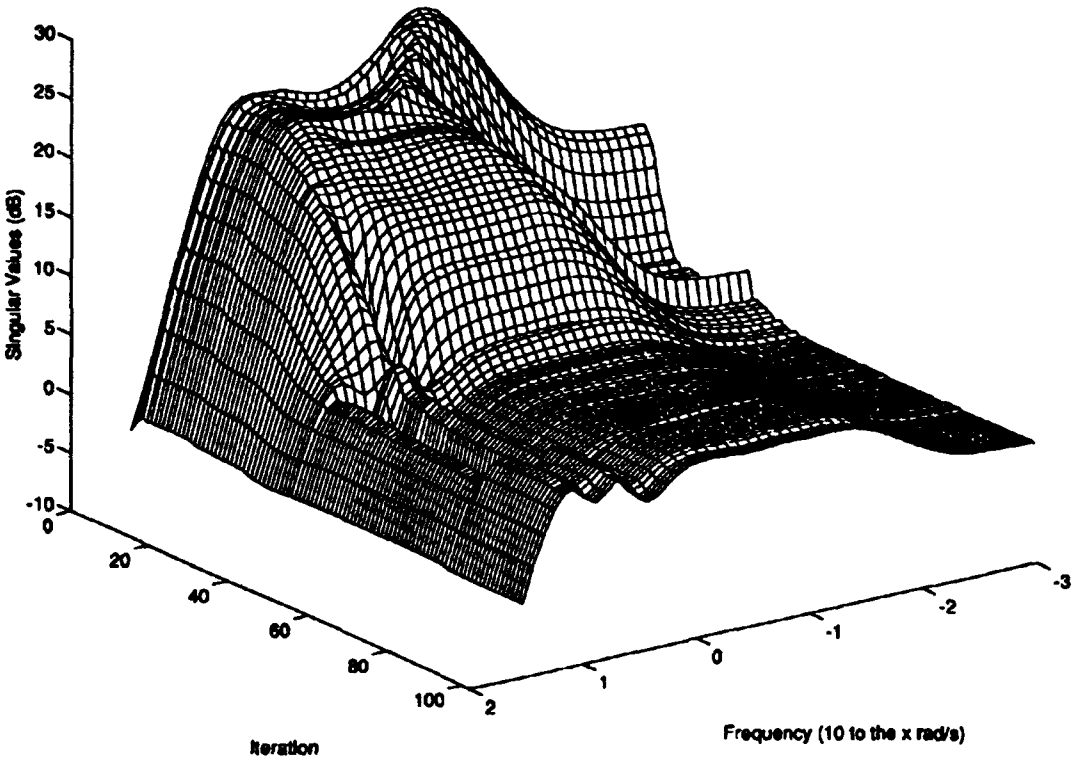


Figure 5.23. Singular Value Plots at Each Step for 3rd Order Tracking

From the infinity-norm of the solution, it is expected that the closed loop performance will not be as good as the fourth order solution. This is examined in the next section.

5.4.1 Closed Loop Frequency Response

The closed loop frequency response to stick commands is shown in Figure 5.24. The alpha and velocity responses are still quite good, but it is

obvious that there is loss of performance in the pitch rate response. There should be a larger steady state error in the time response since the flat portion of the response does not reach 0 dB. Similar trends are seen in Figure 5.25.

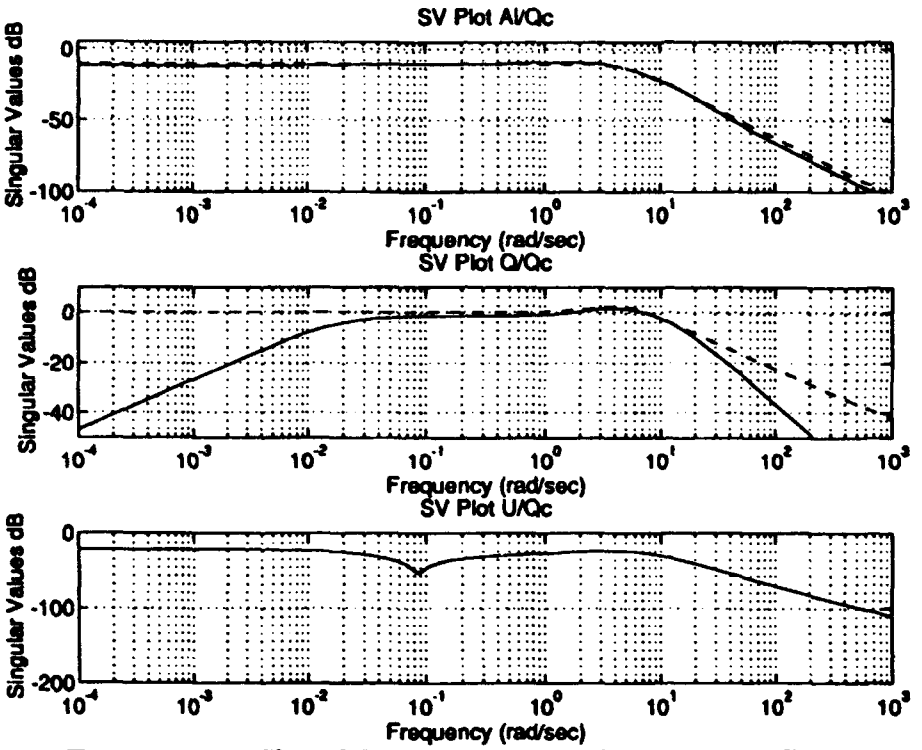


Figure 5.24. Closed Loop Frequency Response to Stick

The alpha and pitch rate responses are about the same as the fourth order, but the performance of the velocity response is slightly less than the fourth order solution. As mentioned earlier, these losses in performance could be reduced by picking a different controller, but it would be at the expense of noise rejection.

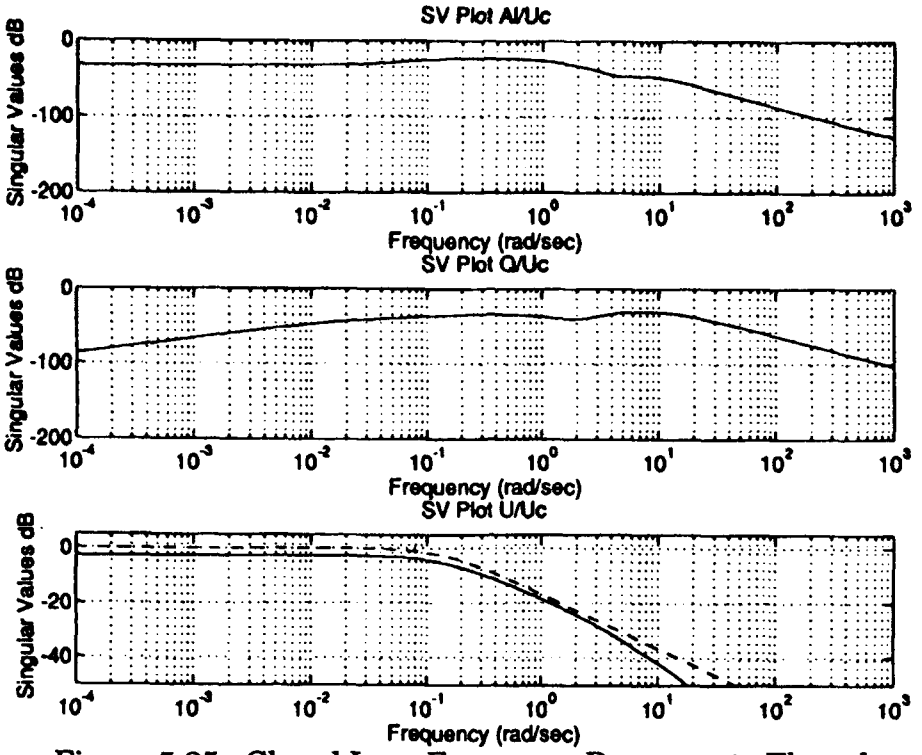


Figure 5.25. Closed Loop Frequency Response to Throttle

5.4.2 Open Loop Analysis

The open loop plots can be seen in Figure 5.26. Little difference can be found when comparing these plots to the plots for the fourth order solution. Since the fourth order solution had decent noise rejection properties, it is safe to conclude that the third order solution will also have decent noise rejection properties. As we will see in a later section, this is indeed the case.

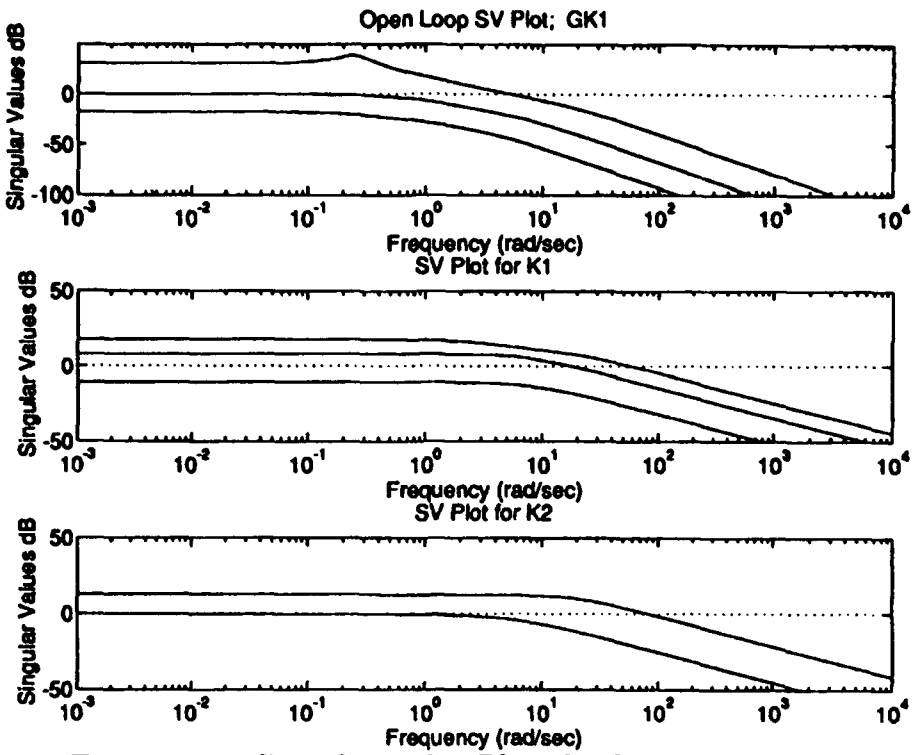


Figure 5.26. Singular Value Plots for GK1, K1, and K2

5.4.3 Step Responses

The step responses played an important role in the choice of the final solution. Performance is often a driving force in the design of a controller, so possibly the solution with the lowest infinity-norm would be the obvious choice. In this case, this seemed to be the logical choice since the two-norm never got above 1.83, which is still lower than the two-norm of the final fourth order solution. However, this was found to be a poor choice through the use of the step responses. Unsatisfactory noise rejection was produced by the lowest infinity-norm solution. Therefore, a different solution with a γ of 1.74 and an α of 1.49 was used. This solution does have reduced performance, but it is only 5% below the smallest infinity-norm solution in terms of steady state performance. Also, this transient response looks better. The responses and actuator deflections are shown in Figures 5.27 and 5.28.

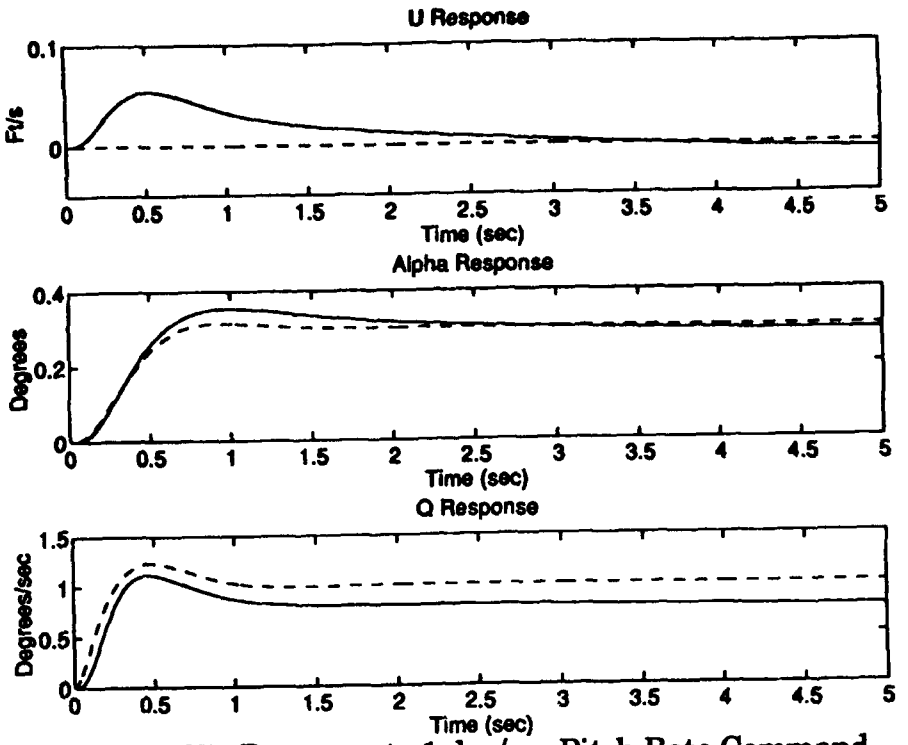


Figure 5.27. Response to 1 deg/sec Pitch Rate Command (no Wind Disturbance or Noise)

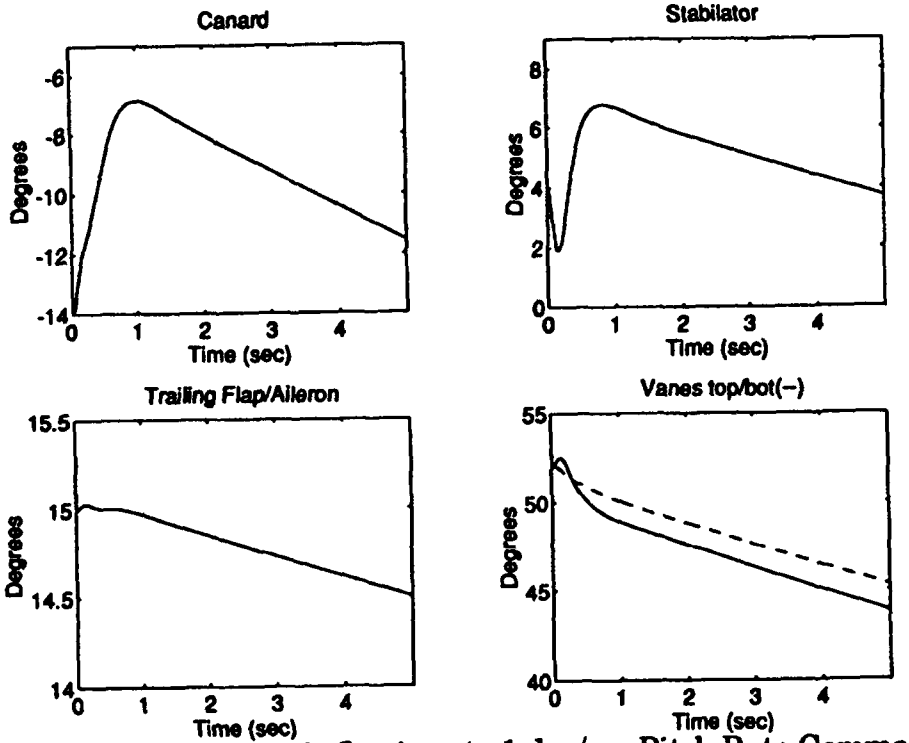


Figure 5.28. Actuator Deflections to 1 deg/sec Pitch Rate Command (no Wind Disturbance or Noise)

As was expected from the closed loop frequency analysis, the velocity and alpha responses are still very good. The pitch rate response is not as good as the fourth order solution but the degradation is fairly small. The steady state error is around 20%. Depending on the importance of have a third order controller, this could still be acceptable. The actuator deflections are almost identical to the fourth order solution.

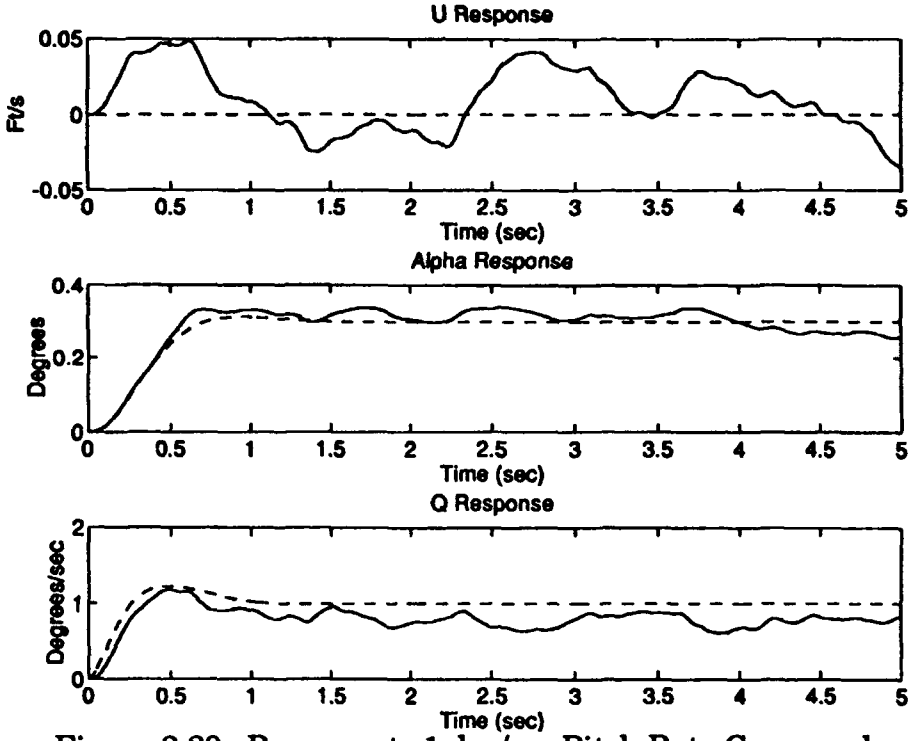


Figure 2.29. Response to 1 deg/sec Pitch Rate Command (with Wind Disturbance and Noise)

As mentioned earlier, the noise responses made a big difference in finding a solution. The responses are shown in Figure 5.29. Comparing this figure to Figure 5.8 shows that the third order noise rejection is just as good as that of the fourth order. The performance of the third order minimum infinity-norm solution did not reach the same level as the fourth order, so it was decided that a bit of a loss in performance was well worth attaining the same noise rejection properties. Figure 5.30 shows that the actuator deflections are also very similar to the fourth order deflections.

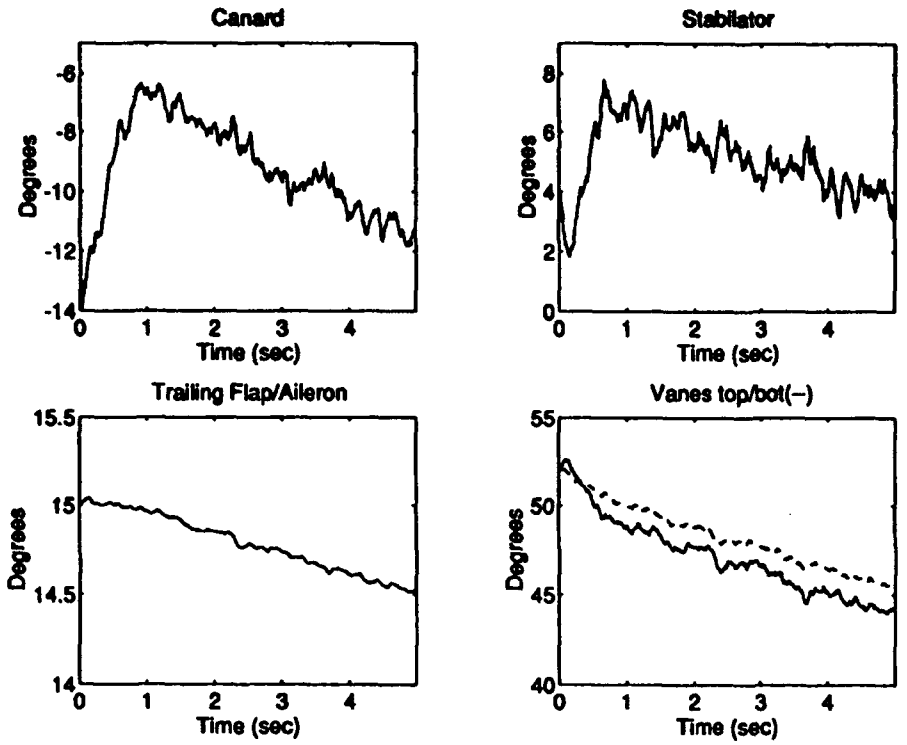


Figure 5.30. Actuator Deflections to 1 deg/sec Pitch Rate Command (with Wind Disturbance and Noise)

5.4.4 Complementary Sensitivity and Margins

Generally, when one reduces the order of a controller, there is a loss somewhere. We have already seen that the performance has been lowered (to achieve the same type of noise rejection). It is interesting to see what happens to the margins.

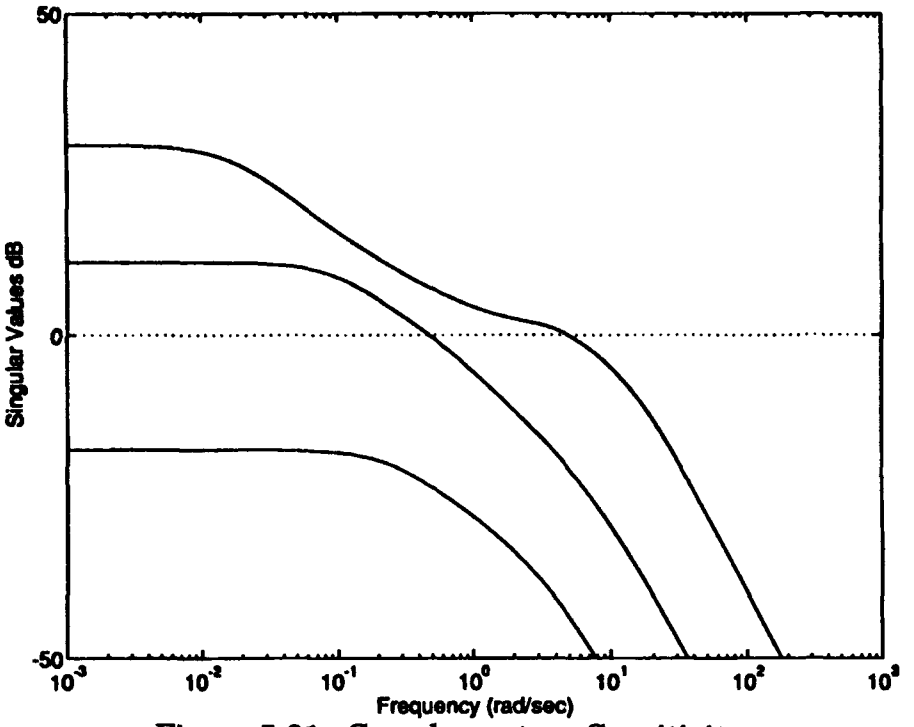


Figure 5.31. Complementary Sensitivity

The complementary sensitivity shown in Figure 5.26 is very similar to that of the fourth order plots. Table 5.3 shows the margins at various points in the solution. The points were chosen to have the same infinity-norm values as shown in Table 5.1. A couple of general conclusions can be made from comparing this table to Table 5.1. The first is that, for a given infinity-norm, the two-norm of the third order solution will be higher than that of the fourth. This conclusion is exactly what is expected by the theory for the lower order

Table 5.3
Mixed Margins; 3rd order Tracking

Step #	α	γ	Lower Gain Margin (dB)	Upper Gain Margin (dB)	Phase Margin (deg)
1	1.49	20.5	-1.32	1.15	8.13
17	0.25	12.5	-2.60	2.00	14.8
22	0.27	10.0	-2.59	2.00	14.8
26	0.34	8.0	-1.72	1.43	10.29
33	0.88	4.9	-0.21	0.20	1.37
38	1.46	2.77	-0.06	0.06	0.42
80	1.79	1.44	-0.08	0.08	0.53
101	1.49	1.74	-0.29	0.29	1.91

solutions. The second is that the margins are indeed less than those of the fourth order solution, which agrees with what common sense would say.

This chapter went through the results of the mixed synthesis. It was found that the low order controllers can be found with the mixed method. The next chapter discusses some conclusions based on the previous results and also suggests some recommendations for further study.

Chapter 6. Conclusions and Recommendations

6.1 Conclusions

The main conclusion that can be drawn from this research is that the mixed H_2 / H_∞ theory and synthesis program are valid, and can be applied to a complicated problem such as the SMTD. One can address a number of different objectives using this method. Three of them were shown in this research. The first is direct reduced order controllers. This work found fourth and third order solutions to a problem that had full order solutions of 11th and 13th order (full order H_2 and full order H_∞, respectively). The second is performance. This problem had very specific response requirements relating to ideal handling qualities. This was provided through a singular H_∞ model following constraint. Although there was some loss in performance compared to the higher order solutions, the responses were quite good. The third thing is stability robustness. The robustness was improved dramatically by using a singular H_∞ margin constraint on the complementary sensitivity.

Although the margins were improved by the margin constraint, there are still some issues that need to be addressed. It was found that on all of the designs that had good tracking performance, the margins were very bad. Even though vector margins are conservative and robustness was not a specified goal of those designs, the margins should have been better. When the one loop-at-a-time margins were examined, they were much higher than the vector margins. Looking at the results and the specifics of the problem, one can conclude that this should be looked into further. Having both pitch rate and theta in the feedback loop obviously caused some problems. It was found that the frequency response for pitch rate command to pitch rate dropped off at low frequency. This corresponds to a zero at the origin. This is difficult for the

design to handle since one would need infinite control power to track pitch rate at zero rad/sec. Additionally, it is natural that a pitch rate input to the complementary sensitivity would cause a large theta output when that input is at low frequency. This does not necessarily mean that the system is less stable. Yet, the resulting large infinity-norm of the complementary sensitivity produces very low vector margins. Therefore, it can be concluded that care must be taken when looking at stability robustness if some quantity and its rate are both used as feedback.

6.2 Recommendations

The problem has been mapped out with respect to a tracking constraint and a margin constraint. The tracking problem was found to improve performance but had horrible margins. The margin problem improved robustness but had poor tracking. The next logical step is to combine these two problems to produce solutions with both good performance and stability robustness. There are two ways that this can be accomplished. The first is to combine the two constraints into one single H_∞ constraint. The second is to run the program with two singular H_∞ constraints. The advantage of the second over the first is that it avoids conflicts due to the cross coupling terms that would arise from joining the two problems. This should produce a solution that has good performance and good margins.

An advantage of the method is that the trade-offs between the two are fairly easily controlled. This research has found where the two problems provide good performance and good margins, so the target area for the multiple constraint problem is fairly well defined, and it would be very interesting to see the results of attempting to reach this area.

Another interesting problem would be to find a mixed solution at full order. This would show exactly how much is being lost by running at reduced orders, or what exactly is possible from the design method. The only problem in doing this is that the program takes a long time to run. It must search over every element of the controller. In the fourth order examples, there were 60 elements of the controller to search over, and typical runs lasted up to 24 hours. Considering that the same problem with a tenth order controller would have 210 elements, this will be a time consuming process, but still worth doing.

As mentioned in the conclusions, it was found that the margins of the tracking systems were small but that there was some question as to whether the systems were truly not robust. Further study of this is necessary in order to answer these questions. Two possible ways to attack this are: one, to look into alternative ways of assessing the robustness, and two, to make slight (but valid) adjustments to the original system so that there is a zero close to the origin but not at the origin.

Appendix A: Controller and Closed Loop Eigenvalues

This appendix contains the controller and closed loop eigenvalues of the examples discussed in Chapter 4 and Chapter 5.

H_2 Full Design

Controller Eigenvalues

real	imaginary	frequency	damping
1.0670e-02	0.0000e+00	1.0670e-02	-1.0000e+00
1.1091e-01	0.0000e+00	1.1091e-01	-1.0000e+00
-1.5000e-01	0.0000e+00	1.5000e-01	1.0000e+00
-6.5298e-01	-6.2858e-01	9.0636e-01	7.2044e-01
-6.5298e-01	6.2858e-01	9.0636e-01	7.2044e-01
-3.1399e+00	-1.8611e+00	3.6500e+00	8.6024e-01
-3.1399e+00	1.8611e+00	3.6500e+00	8.6024e-01
-3.5000e+00	-3.5707e+00	5.0000e+00	7.0000e-01
-3.5000e+00	3.5707e+00	5.0000e+00	7.0000e-01
-6.6924e+00	0.0000e+00	6.6924e+00	1.0000e+00
-1.3965e+02	0.0000e+00	1.3965e+02	1.0000e+00

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-4.7375e-06	0.0000e+00	4.7375e-06	1.0000e+00
-1.5000e-01	0.0000e+00	1.5000e-01	1.0000e+00
-1.0088e-01	-2.0762e-01	2.3083e-01	4.3703e-01
-1.0088e-01	2.0762e-01	2.3083e-01	4.3703e-01
-6.0541e-01	0.0000e+00	6.0541e-01	1.0000e+00
-7.0140e-01	-4.6061e-01	8.3912e-01	8.3587e-01
-7.0140e-01	4.6061e-01	8.3912e-01	8.3587e-01
-1.5170e+00	0.0000e+00	1.5170e+00	1.0000e+00
-2.3145e+00	-1.9315e+00	3.0146e+00	7.6777e-01
-2.3145e+00	1.9315e+00	3.0146e+00	7.6777e-01
-3.5000e+00	-3.5707e+00	5.0000e+00	7.0000e-01
-3.5000e+00	3.5707e+00	5.0000e+00	7.0000e-01
-5.0033e+00	0.0000e+00	5.0033e+00	1.0000e+00
-6.6421e+00	0.0000e+00	6.6421e+00	1.0000e+00
-1.3468e+02	0.0000e+00	1.3468e+02	1.0000e+00

Regulator Design

Controller Eigenvalues

real	imaginary	frequency	damping
-2.0913e-01	0.0000e+00	2.0913e-01	1.0000e+00
-9.4252e-01	-6.9114e-01	1.1688e+00	8.0642e-01
-9.4252e-01	6.9114e-01	1.1688e+00	8.0642e-01
-6.1948e+00	0.0000e+00	6.1948e+00	1.0000e+00

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-2.1198e-01	0.0000e+00	2.1198e-01	1.0000e+00
-5.7209e-02	-2.4423e-01	2.5084e-01	2.2807e-01
-5.7209e-02	2.4423e-01	2.5084e-01	2.2807e-01
-5.3449e-01	0.0000e+00	5.3449e-01	1.0000e+00
-1.2872e+00	0.0000e+00	1.2872e+00	1.0000e+00
-2.7559e+00	0.0000e+00	2.7559e+00	1.0000e+00
-2.1260e+00	-2.0315e+00	2.9406e+00	7.2300e-01
-2.1260e+00	2.0315e+00	2.9406e+00	7.2300e-01

Tracking Design

Controller Eigenvalues

real	imaginary	frequency	damping
-1.0000e-02	0.0000e+00	1.0000e-02	1.0000e+00
-1.0000e-02	0.0000e+00	1.0000e-02	1.0000e+00
-7.9589e-02	0.0000e+00	7.9589e-02	1.0000e+00
-1.5000e-01	0.0000e+00	1.5000e-01	1.0000e+00
-4.1636e-01	0.0000e+00	4.1636e-01	1.0000e+00
-3.5000e+00	-3.5707e+00	5.0000e+00	7.0000e-01
-3.5000e+00	3.5707e+00	5.0000e+00	7.0000e-01
-6.6840e+02	-6.5637e+02	9.3679e+02	7.1350e-01
-6.6840e+02	6.5637e+02	9.3679e+02	7.1350e-01
-1.0708e+03	-1.0365e+03	1.4903e+03	7.1853e-01
-1.0708e+03	1.0365e+03	1.4903e+03	7.1853e-01
-3.1453e+03	-3.1191e+03	4.4296e+03	7.1006e-01
-3.1453e+03	3.1191e+03	4.4296e+03	7.1006e-01

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-1.0296e-03	0.0000e+00	1.0296e-03	1.0000e+00

-1.0000e-02	0.0000e+00	1.0000e-02	1.0000e+00
-1.0000e-02	0.0000e+00	1.0000e-02	1.0000e+00
-1.0000e-02	0.0000e+00	1.0000e-02	1.0000e+00
-1.5000e-01	0.0000e+00	1.5000e-01	1.0000e+00
-5.7209e-02	-2.4423e-01	2.5084e-01	2.2807e-01
-5.7209e-02	2.4423e-01	2.5084e-01	2.2807e-01
-5.3449e-01	0.0000e+00	5.3449e-01	1.0000e+00
-1.2872e+00	0.0000e+00	1.2872e+00	1.0000e+00
-3.5000e+00	-3.5707e+00	5.0000e+00	7.0000e-01
-3.5000e+00	3.5707e+00	5.0000e+00	7.0000e-01
-6.6784e+02	-6.6780e+02	9.4444e+02	7.0713e-01
-6.6784e+02	6.6780e+02	9.4444e+02	7.0713e-01
-1.0710e+03	-1.0710e+03	1.5146e+03	7.0711e-01
-1.0710e+03	1.0710e+03	1.5146e+03	7.0711e-01
-3.1454e+03	-3.1454e+03	4.4483e+03	7.0711e-01
-3.1454e+03	3.1454e+03	4.4483e+03	7.0711e-01

Margin Design

Controller Eigenvalues

real	imaginary	frequency	damping
-6.8603e+06	0.0000e+00	6.8603e+06	1.0000e+00
-1.1552e+06	0.0000e+00	1.1552e+06	1.0000e+00
-2.4534e+05	0.0000e+00	2.4534e+05	1.0000e+00
-5.4545e+04	0.0000e+00	5.4545e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-5.7209e-02	-2.4423e-01	2.5084e-01	2.2807e-01
-5.7209e-02	2.4423e-01	2.5084e-01	2.2807e-01
-1.0000e+00	0.0000e+00	1.0000e+00	1.0000e+00
-1.2872e+00	0.0000e+00	1.2872e+00	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.0000e+04	0.0000e+00	1.0000e+04	1.0000e+00
-1.6357e+05	0.0000e+00	1.6357e+05	1.0000e+00
-7.4981e+05	0.0000e+00	7.4981e+05	1.0000e+00
-7.8455e+05	0.0000e+00	7.8455e+05	1.0000e+00
-6.6174e+06	0.0000e+00	6.6174e+06	1.0000e+00

Fourth Order Mixed Tracking Design

Controller Eigenvalues

real	imaginary	frequency	damping
-3.0902e+00	0.0000e+00	3.0902e+00	1.0000e+00
-6.6292e+00	-5.1217e+00	8.3773e+00	7.9134e-01
-6.6292e+00	5.1217e+00	8.3773e+00	7.9134e-01
-1.5099e+01	0.0000e+00	1.5099e+01	1.0000e+00

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-1.2420e-02	0.0000e+00	1.2420e-02	1.0000e+00
-1.5828e-01	0.0000e+00	1.5828e-01	1.0000e+00
-8.4832e-01	0.0000e+00	8.4832e-01	1.0000e+00
-3.2055e+00	0.0000e+00	3.2055e+00	1.0000e+00
-2.4838e+00	-4.5461e+00	5.1804e+00	4.7945e-01
-2.4838e+00	4.5461e+00	5.1804e+00	4.7945e-01
-1.1561e+01	-1.1089e+01	1.6019e+01	7.2171e-01
-1.1561e+01	1.1089e+01	1.6019e+01	7.2171e-01

Fourth Order Mixed Margin Design, Case 1

Controller Eigenvalues

real	imaginary	frequency	damping
-7.1243e-01	0.0000e+00	7.1243e-01	1.0000e+00
-7.9908e+00	0.0000e+00	7.9908e+00	1.0000e+00
-6.3679e+00	-6.3524e+00	8.9947e+00	7.0797e-01
-6.3679e+00	6.3524e+00	8.9947e+00	7.0797e-01

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-5.0123e-02	-2.4269e-01	2.4781e-01	2.0226e-01
-5.0123e-02	2.4269e-01	2.4781e-01	2.0226e-01
-7.4577e-01	0.0000e+00	7.4577e-01	1.0000e+00
-1.9172e+00	-1.3291e+00	2.3329e+00	8.2183e-01
-1.9172e+00	1.3291e+00	2.3329e+00	8.2183e-01
-2.4904e+00	0.0000e+00	2.4904e+00	1.0000e+00
-7.5677e+00	-1.0965e+01	1.3323e+01	5.6802e-01
-7.5677e+00	1.0965e+01	1.3323e+01	5.6802e-01

Fourth Order Mixed Margin Design, Case 2

Controller Eigenvalues

real	imaginary	frequency	damping
-2.9600e-01	0.0000e+00	2.9600e-01	1.0000e+00
-1.9233e+00	-2.7513e+00	3.3569e+00	5.7294e-01
-1.9233e+00	2.7513e+00	3.3569e+00	5.7294e-01
-5.9641e+00	0.0000e+00	5.9641e+00	1.0000e+00

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-3.9961e-01	-3.4214e-01	5.2607e-01	7.5962e-01
-3.9961e-01	3.4214e-01	5.2607e-01	7.5962e-01
-6.6756e-01	0.0000e+00	6.6756e-01	1.0000e+00
-1.4981e+00	0.0000e+00	1.4981e+00	1.0000e+00
-1.2086e+00	-9.8636e-01	1.5600e+00	7.7475e-01
-1.2086e+00	9.8636e-01	1.5600e+00	7.7475e-01
-2.7958e+00	-4.5544e+00	5.3440e+00	5.2317e-01
-2.7958e+00	4.5544e+00	5.3440e+00	5.2317e-01

Third Order Mixed Tracking Design

Controller Eigenvalues

real	imaginary	frequency	damping
-2.9005e+00	0.0000e+00	2.9005e+00	1.0000e+00
-1.3119e+01	-1.7487e+00	1.3235e+01	9.9123e-01
-1.3119e+01	1.7487e+00	1.3235e+01	9.9123e-01

Closed Loop Eigenvalues

real	imaginary	frequency	damping
-1.4756e-02	0.0000e+00	1.4756e-02	1.0000e+00
-1.6005e-01	0.0000e+00	1.6005e-01	1.0000e+00
-6.8798e-01	0.0000e+00	6.8798e-01	1.0000e+00
-3.2686e+00	-1.8322e+00	3.7471e+00	8.7230e-01
-3.2686e+00	1.8322e+00	3.7471e+00	8.7230e-01
-1.1302e+01	-1.0152e+01	1.5193e+01	7.4394e-01
-1.1302e+01	1.0152e+01	1.5193e+01	7.4394e-01

Appendix B: Computer Files Developed

Over the course of this research, many programs were developed. This appendix contains final versions of some of those programs.

The following two programs were used to set up the full design model.

ACDF2.M

```
%*********************************************************************
%
%      This file holds the aircraft data to be used in pff.m.  Any
%     changes in the design weightings are made here.
%
%          by William C. Reigelsperger Jr.
%
%      STOL/MTD Landing Flight Condition 1
%
%      This program used ideal model for alpha,q,U
%      controller has 6 inputs(u,theta,alpha,q,st,th)
%        5 outputs(cnrd,stb,flp,tvn,bvn)
%      This program is used with pff.m to form packed sys matrix
%
%*********************************************************************

clear cm dm dp cp


% Aircraft data for given flight condition

ap=[-.1729 -30.62 -12.49 0 0;
    0 0 0 1 0;
 -.00264 -.0583 -.3129 1 0;
  .001067 0 .7453 -.3813 0;
  0 168.7 -168.7 0 0];

bp=[ .003912 -.08876 -.042 -.042 -.05352 -.09201 -.05352 -.09201;
  0 0 0 0 0 0 0 0;
  -.000837 -.001365 -.0002601 -.0002601 .0005339 -.0001733 .0005339
-.0001733;
  .0117 -.02452 .0008338 .0008338 .007877 -.007061 .007877 -.007061;
 0 0 0 0 0 0 0 0];

cp=[.0148 0 1.681 0 0;
.01538 0 2.067 -.1975 0;
1 0 0 0 0;
0 57.3 0 0 0;
```

```
0 0 57.3 0 0;
0 0 0 57.3 0;
0 0 0 0 1];

dp=[.004142 .00766 .001699 .001699 -.002154 .001743 -.002154 .001743;
.0102 -.00504 .002131 .002131 .001926 -.001915 .001926 -.001915;
zeros(5,8)];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

%  Remove any unwanted states or outputs for design model
%  Remove h state and nz output,combine flap/aleiron and lft/rt vains

am=ap(1:4,1:4);
bm(:,1:2)=bp(1:4,1:2);bm(:,3)=bp(1:4,3)+bp(1:4,4);bm(:,4)=bp(1:4,5)+bp(1:4,7);
bm(:,5)=bp(1:4,6)+bp(1:4,8);
cm=cp(2:6,1:4); cm=ms*cm;
dm(:,1:2)=dp(2:6,1:2);dm(:,3)=dp(2:6,3)+dp(2:6,4);dm(:,4)=dp(2:6,5)+dp(2:6,7);
dm(:,5)=dp(2:6,6)+dp(2:6,8);
dm=ms*dm;

%  The desired response(see pg 29 thesis notebook1)

%idealdat
gam=.7;
w=5;
a=.15;

deni=conv([1 gam*w*2 w^2],[1 a]);
n11=[0 0 7.5 7.5*a];
ntmp=[0 7.5/w^2 1];
ntmp=w^2*ntmp;
n21=conv(ntmp,[1 a]);
n31=[0 0 0 0];
n12=[0 0 0 0];
n22=[0 0 0 0];
n32=[0 a a*gam*w*2 a*w^2];

numi=[n11;n21;n31;n12;n22;n32];

[ati,bti,cti,dti]=tfm2ss(numi,deni,2,3);
[ai,bi,ci,di]=minreal(ati,bti,cti,dti);


% Disturbance dynamics

dend=[1 6.7];
```

```
numd=[.0187];

nsp=length(ap);
[ad,bd,cd,dd]=tfm2ss(numd,dend,1,1);


% Actuator weighting

%wcnd=.032;wstb=.0175;wtef=.1;wvrt=.0175;wvrb=.0175;
wcnd=.037;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;
%wcnd=1.5;wstb=.7;wtef=1;wvrt=.5;wvrb=.5;
wcnd=.04;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;

aa=[];ba=[];ca=[];
da=[wcnd 0 0 0 0 ;0 wstb 0 0 0 ;0 0 wtef 0 0 ;0 0 0 wvrt 0 ;0 0 0 0 wvrb];

da=ro*da;
%da=ro*eye(5);

% Noise

%denn=1;
%numn=[1;1;1;1;1;1];

%[an,bn,cn,dn]=tfm2ss(numn,denn,1,6);
bn=[];cn=[];an=[];
dn=[  1 0 0 0 ; 0 sqrt(.05) 0 0 ; 0 0 sqrt(.1) 0 ; 0 0 0 sqrt(.02) ];
dn=dn;

% Weighting on ideal minus actual signal

ntmp=[1 5];dtmp=[1 .0005];

wq=10;
walpha=5;
wwu=.1;
z=[0 0];

numI=[walpha*ntmp;z;z;z;wq*ntmp;z ;z;z;wwu*ntmp];
denI=dtmp;

[at,bt,ct,dt]=tfm2ss(numI,denI,3,3);
%[aI,bI,cI,dI]=minreal(at,bt,ct,dt);
aI=at;bI=bt;cI=ct;dI=dt;

% Disturbance matrix gam

gam=am(:,3);
```

```
% Actual selective matrix mI (alpha,q,u)

mI=[ 0 0 1 0 ; 0 0 0 1 ; 1 0 0 0 ];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];
```

PFF.M

```
%*************************************************************************
%
%       This program forms the packed P matrix using data from acdf2.m
%
%       for full design.
%
%       by William C. Reigelsperger Jr.
%
%*************************************************************************

nsp=length(am);

% Get the lengths of each state space, and other important lengths

nsi=length(ai);ii=length(di(1,:));oi=length(di(:,1));
nsd=length(ad);id=length(dd(1,:));od=length(dd(:,1));
nsI=length(aI);iI=length(dI(1,:));oI=length(dI(:,1));

ip=length(dm(1,:));op=length(dm(:,1));
ia=length(da(1,:));oa=length(da(:,1));
in=length(dn(1,:));on=length(dn(:,1));

% Now form A,B1,B2,C1,C2,D11,D12,D21,D22

A=[am zeros(nsp,nsi) gam*cd zeros(nsp,nsI) ;
  zeros(nsi,nsp) ai zeros(nsi,nsd+nsI);
  zeros(nsd,nsp+nsi) ad zeros(nsd,nsI);
 -bI*mI*cm bI*ci zeros(nsI,nsd) aI ];

B1=[ zeros(nsp,ii) gam*dd zeros(nsp,in);
    bi zeros(nsi,id+in);
    zeros(nsd,ii) bd zeros(nsd,in);
  bI*di zeros(nsI,id+in)];

B2=[bm;zeros(nsi,ip);zeros(nsd,ip);-bI*mI*dm];

B=[B1 B2];

C1=[-dI*mI*cm dI*ci zeros(oI,nsd) cI;
```

```
  zeros(oa,nsp+nsi+nsd+nsI) ];

C2=[cm zeros(on,nsi+nsd+nsI) ;
  zeros(ii,nsp+nsi+nsd+nsI) ];

C=[C1;C2];


D11=[dI*di zeros(oI,id+in);
   zeros(oa,ii+id+in)];

D12=[-dI*mI*dm;
     da];

D21=[zeros(on,ii+id) dn;
   eye(ii) zeros(ii,id+in)];

D22=[dm;zeros(ii,ip)];

D=[D11 D12;
  D21 D22];

sys=pck(A,B,C,D);

length(A)
```

The following two programs were used to set up the regulator design model that was examined in Chapter 4 and in the mixed design.

ACDFH2.M

```
%*****************************************************************
%
%      This file holds the aircraft data to be used to set up problem.  Any
%      changes in the design weightings are made here.
%
%      by William C. Reigelsperger Jr.
%
%      STOL/MTD Landing Flight Condition 1
%
%      This program used ideal model for alpha,q,U
%      controller has 6 inputs(u,theta,alpha,q,st,th)
%        5 outputs(cnrd,stb,flp,tvn,bvn)
%      This program is used with pfh2.m to form packed sys matrix
%
%*****************************************************************

clear cm dm dp cp
```

```
% Aircraft data for given flight condition

ap=[-.1729 -30.62 -12.49 0 0;
    0 0 0 1 0;
 -.00264 -.0583 -.3129 1 0;
  .001067 0 .7453 -.3813 0;
  0 168.7 -168.7 0 0];

bp=[ .003912 -.08876 -.042 -.042 -.05352 -.09201 -.05352 -.09201;
  0 0 0 0 0 0 0 0;
  -.000837 -.001365 -.0002601 -.0002601 .0005339 -.0001733 .0005339
-.0001733;
  .0117 -.02452 .0008338 .0008338 .007877 -.007061 .007877 -.007061;
 0 0 0 0 0 0 0 0];

cp=[.0148 0 1.681 0 0;
.01538 0 2.067 -.1975 0;
1 0 0 0 0;
0 57.3 0 0 0;
0 0 57.3 0 0;
0 0 0 57.3 0;
0 0 0 0 1];

dp=[.004142 .00766 .001699 .001699 -.002154 .001743 -.002154 .001743;
.0102 -.0050[4 .002131 .002131 .001926 -.001915 .001926 -.001915;
zeros(5,8)];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

% Remove any unwanted states or outputs for design model
% Remove h state and nz output,combine flap/aleiron and lft/rt vains

am=ap(1:4,1:4);
bm(:,1:2)=bp(1:4,1:2);bm(:,3)=bp(1:4,3)+bp(1:4,4);bm(:,4)=bp(1:4,5)+bp(1:4,7);
bm(:,5)=bp(1:4,6)+bp(1:4,8);
cm=cp(2:6,1:4); cm=ms*cm;
dm(:,1:2)=dp(2:6,1:2);dm(:,3)=dp(2:6,3)+dp(2:6,4);dm(:,4)=dp(2:6,5)+dp(2:6,7);
dm(:,5)=dp(2:6,6)+dp(2:6,8);
dm=ms*dm;


% Disturbance dynamics

dend=[1 6.7];
numd=[.0187];
```

```
nsp=length(ap);
%[ad,bd,cd,dd]=tfm2ss(numd,dend,1,1);
ad=[];bd=[];cd=[];dd=.0187/6.7;


% Actuator weighting

wcnd=.01;wstb=.01;wtef=.05;wvrt=.02;wvrb=wvrt;
%wcnd=.037;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;
%wcnd=1.5;wstb=.7;wtef=1;wvrt=.5;wvrb=.5;
%wcnd=.04;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;

aa=[];ba=[];ca=[];
da=[wcnd 0 0 0 0 ;0 wstb 0 0 0 ;0 0 wtef 0 0 ;0 0 0 wvrt 0 ;0 0 0 0 wvrb];
ro=5
da=ro*da;

% Noise

bn=[];cn=[];an=[];
dn=[  1 0 0 0 ; 0 sqrt(.05) 0 0 ; 0 0 sqrt(.1) 0 ; 0 0 0 sqrt(.02) ];
dn=dn;



% Disturbance matrix gam

gam=am(:,3);

% Actual selective matrix mI (alpha,q,u)

mI=[ 0 0 1 0 ;  0 0 0 1 ;  1 0 0 0 ];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[   0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

%  State weight h.
wuu=1;wth=1;wal=1;wqq=1;

h=.18*ro*[wuu 0 0 0;0 wth 0 0;0 0 wal 0;0 0 0 wqq];
```

PFH2.M

```
%*********************************************************************
%
%       This program forms the packed P matrix using data from acdfh2.m
%
%       for the h2/lqg part of the mixed problem
%
%*********************************************************************

nsp=length(am);

% Get the lengths of each state space, and other important lengths


nsi=length(ai);ii=length(di(1,:));oi=length(di(:,1));
nsd=length(ad);id=length(dd(1,:));od=length(dd(:,1));
nsI=length(aI);iI=length(dI(1,:));oI=length(dI(:,1));

ip=length(dm(1,:));op=length(dm(:,1));
ia=length(da(1,:));oa=length(da(:,1));
in=length(dn(1,:));on=length(dn(:,1));

% Now form A,B1,B2,C1,C2,D11,D12,D21,D22

A2=[am  ];

Bw=[ zeros(nsp,id +in+ii)];

Bu=[bm];

B=[Bw Bu];

Cz=[h ;zeros(5,nsp)];

Cy=[cm  ;
  zeros(ii,nsp) ];

C=[Cz;Cy];


Dzw=zeros(nsp+ip,id+op+ii);

Dzu=[zeros(nsp,ip);
     da];

Dyw=[zeros(on,id) dn zeros(on,ii);
   zeros(ii,id+on) eye(2)];
```

```
Dyu=[zeros(on+ii,ip)];

D=[Dzw Dzu;
  Dyw Dyu];

sys=pck(A2,B,C,D);
```

The tracking design was set up by the following two programs.

ACDFHI2.M

```
%*************************************************************************
%
%       This file holds the aircraft data to be used in seting up problem.
%       Any changes in the design weightings are made here.
%
%       by William C. Reigelsperger Jr.
%
%       STOL/MTD Landing Flight Condition 1
%
%       This program used ideal model for alpha,q,U
%       controller has 6 inputs(u,theta,alpha,q,st,th)
%         5 outputs(cnrd,stb,flp,tvn,bvn)
%       This program is used with pfhi2.m to form packed sys matrix
%
%*************************************************************************

clear cm dm dp cp


% Aircraft data for given flight condition

ap=[-.1729 -30.62 -12.49 0 0;
    0 0 0 1 0;
 -.00264 -.0583 -.3129 1 0;
  .001067 0 .7453 -.3813 0;
  0 168.7 -168.7 0 0];

bp=[ .003912 -.08876 -.042 -.042 -.05352 -.09201 -.05352 -.09201;
  0 0 0 0 0 0 0 0;
  -.000837 -.001365 -.0002601 -.0002601 .0005339 -.0001733 .0005339
-.0001733;
  .0117 -.02452 .0008338 .0008338 .007877 -.007061 .007877 -.007061;
 0 0 0 0 0 0 0 0];

cp=[.0148 0 1.681 0 0;
.01538 0 2.067 -.1975 0;
1 0 0 0 0;
```

```
0 57.3 0 0 0;
0 0 57.3 0 0;
0 0 0 57.3 0;
0 0 0 0 1];

dp=[.004142 .00766 .001699 .001699 -.002154 .001743 -.002154 .001743;
.0102 -.00504 .002131 .002131 .001926 -.001915 .001926 -.001915;
zeros(5,8)];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

% Remove any unwanted states or outputs for design model
% Remove h state and nz output,combine flap/aleiron and lft/rt vains

am=ap(1:4,1:4);
bm(:,1:2)=bp(1:4,1:2);bm(:,3)=bp(1:4,3)+bp(1:4,4);bm(:,4)=bp(1:4,5)+bp(1:4,7);
bm(:,5)=bp(1:4,6)+bp(1:4,8);
cm=cp(2:6,1:4); cm=ms*cm;
dm(:,1:2)=dp(2:6,1:2);dm(:,3)=dp(2:6,3)+dp(2:6,4);dm(:,4)=dp(2:6,5)+dp(2:6,7);
dm(:,5)=dp(2:6,6)+dp(2:6,8);
dm=ms*dm;

% The desired response(see pg 29 thesis notebook1)

%idealdat
gam=.7;
w=5;
a=.15;

deni=conv([1 gam*w*2 w^2],[1 a]);
n11=[0 0 7.5 7.5*a];
ntmp=[0 7.5/w^2 1];
ntmp=w^2*ntmp;
n21=conv(ntmp,[1 a]);
n31=[0 0 0 0];
n12=[0 0 0 0];
n22=[0 0 0 0];
n32=[0 0 0 a];
n32=[0 a a*gam*w*2 a*w^2];

numi=[n11;n21;n31;n12;n22;n32];

[ati,bti,cti,dti]=tfm2ss(numi,deni,2,3);
[ai,bi,ci,di]=minreal(ati,bti,cti,dti);


% Disturbance dynamics
```

```
dend=[1 6.7];
numd=[.0187];

nsp=length(ap);
[ad,bd,cd,dd]=tfm2ss(numd,dend,1,1);


% Actuator weighting

%wcnd=.032;wstb=.0175;wtef=.1;wvrt=.0175;wvrb=.0175;
wcnd=.037;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;
%wcnd=1.5;wstb=.7;wtef=1;wvrt=.5;wvrb=.5;
wcnd=.04;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;

aa=[];ba=[];ca=[];
da=[wcnd 0 0 0 0 ;0 wstb 0 0 0 ;0 0 wtef 0 0 ;0 0 0 wvrt 0 ;0 0 0 0 wvrb];

%da=ro*da;
%da=ro*eye(5);
da=1e-5*eye(5);

% Noise

%denn=1;
%numn=[1;1;1;1;1;1];

%[an,bn,cn,dn]=tfm2ss(numn,denn,1,6);
bn=[];cn=[];an=[];
dn=[  1 0 0 0 ; 0 sqrt(.05) 0 0 ; 0 0 sqrt(.1) 0 ; 0 0 0 sqrt(.02) ];
dn=dn;

% Weighting on ideal minus actual signal


ntmp=100*[ 1 1e-2]; dtmp=conv([1 .1],[1 10]);

wq=1;
walpha=1;
wwu=1;
z=[ 0 0];

numI=[walpha*ntmp;z;z;z;wq*ntmp;z ;z;z;wwu*ntmp];
denI=dtmp;

[at,bt,ct,dt]=tfm2ss(numI,denI,3,3);
%[aI,bI,cI,dI]=minreal(at,bt,ct,dt);
aI=at;bI=bt;cI=ct;dI=dt;
```

```
% Disturbance matrix gam

gam=am(:,3);

% Actual selective matrix mI (alpha,q,u)

mI=[ 0 0 1 0 ; 0 0 0 1 ; 1 0 0 0 ];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];
```

PFHI2.M

```
%************************************************************************
%
%       This program forms the packed P matrix using data from acdfhi2.m
%
%       for the tracking contraint in the mixed problem
%
%
%************************************************************************

nsp=length(am);

% Get the lengths of each state space, and other important lengths


nsi=length(ai);ii=length(di(1,:));oi=length(di(:,1));
nsd=length(ad);id=length(dd(1,:));od=length(dd(:,1));
nsI=length(aI);iI=length(dI(1,:));oI=length(dI(:,1));

ip=length(dm(1,:));op=length(dm(:,1));
ia=length(da(1,:));oa=length(da(:,1));
in=length(dn(1,:));on=length(dn(:,1));

% Now form A,B1,B2,C1,C2,D11,D12,D21,D22

A=[am zeros(nsp,nsi+nsI) ;
   zeros(nsi,nsp) ai zeros(nsi,nsI);
  -bI*mI*cm bI*ci aI ];

B1=[ zeros(4,2);bi;zeros(nsI,2)];

B2=[bm;zeros(nsi+nsI,5)];

B=[B1 B2];
```

```
C1=[-dI*mI*cm dI*ci cI ];

C2=[cm zeros(on,nsi+nsI) ;
  zeros(ii,nsp+nsi+nsI) ];

C=[C1;C2];


D11=zeros(3,2);

D12=[zeros(3,5)];

D21=[zeros(on,ii) ;
   eye(ii) ];

D22=[dm;zeros(ii,ip)];

D=[D11 D12;
  D21 D22];

sys=pck(A,B,C,D);
```

The margin design was set up by the following programs.

ACDFHIM.M

```
%*********************************************************************
%
%       This file holds the aircraft data to be used in problem.  Any
%       changes in the design weightings are made here.
%
%       by William C. Reigelsperger Jr.
%
%       STOL/MTD Landing Flight Condition 1
%
%       This program used ideal model for alpha,q,U
%       controller has 6 inputs(u,theta,alpha,q,st,th)
%         5 outputs(cnrd,stb,flp,tvn,bvn)
%       This program is used with pfhim.m to form packed sys matrix
%
%*********************************************************************

clear cm dm dp cp


% Aircraft data for given flight condition

ap=[-.1729 -30.62 -12.49 0 0;
    0 0 0 1 0;
```

```
 -.00264 -.0583 -.3129 1 0;
  .001067 0 .7453 -.3813 0;
  0 168.7 -168.7 0 0];

bp=[ .003912 -.08876 -.042 -.042 -.05352 -.09201 -.05352 -.09201;
  0 0 0 0 0 0 0 0;
  -.000837 -.001365 -.0002601 -.0002601 .0005339 -.0001733 .0005339
-.0001733;
  .0117 -.02452 .0008338 .0008338 .007877 -.007061 .007877 -.007061;
 0 0 0 0 0 0 0 0];

cp=[.0148 0 1.681 0 0;
.01538 0 2.067 -.1975 0;
1 0 0 0 0;
0 57.3 0 0 0;
0 0 57.3 0 0;
0 0 0 57.3 0;
0 0 0 0 1];

dp=[.004142 .00766 .001699 .001699 -.002154 .001743 -.002154 .001743;
.0102 -.00504 .002131 .002131 .001926 -.001915 .001926 -.001915;
zeros(5,8)];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[  0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

% Remove any unwanted states or outputs for design model
% Remove h state and nz output,combine flap/aleiron and lft/rt vains

am=ap(1:4,1:4);
bm(:,1:2)=bp(1:4,1:2);bm(:,3)=bp(1:4,3)+bp(1:4,4);bm(:,4)=bp(1:4,5)+bp(1:4,7);
bm(:,5)=bp(1:4,6)+bp(1:4,8);
cm=cp(2:6,1:4); cm=ms*cm;
dm(:,1:2)=dp(2:6,1:2);dm(:,3)=dp(2:6,3)+dp(2:6,4);dm(:,4)=dp(2:6,5)+dp(2:6,7);
dm(:,5)=dp(2:6,6)+dp(2:6,8);
dm=ms*dm;



% Disturbance dynamics

dend=[1 6.7];
numd=[.0187];

nsp=length(ap);
[ad,bd,cd,dd]=tfm2ss(numd,dend,1,1);
```

```
% Actuator weighting

%wcnd=.032;wstb=.0175;wtef=.1;wvrt=.0175;wvrb=.0175;
wcnd=.037;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;
%wcnd=1.5;wstb=.7;wtef=1;wvrt=.5;wvrb=.5;
wcnd=.04;wstb=.02;wtef=.05;wvrt=.02;wvrb=.02;

aa=[];ba=[];ca=[];
da=[wcnd 0 0 0 0 ;0 wstb 0 0 0 ;0 0 wtef 0 0 ;0 0 0 wvrt 0 ;0 0 0 0 wvrb];
ro=1;
da=ro*da;
%da=ro*eye(5);

% Noise

%denn=1;
%numn=[1;1;1;1;1;1];

%[an,bn,cn,dn]=tfm2ss(numn,denn,1,6);
bn=[];cn=[];an=[];
dn=[  1 0 0 0 ; 0 sqrt(.05) 0 0 ; 0 0 sqrt(.1) 0 ; 0 0 0 sqrt(.02) ];
dn=dn;

% Disturbance matrix gam

gam=am(:,3);

% Actual selective matrix mI (alpha,q,u)

mI=[ 0 0 1 0 ;  0 0 0 1 ;  1 0 0 0 ];

% Sensed selective matrix ms (u,pitch att,alpha,q)

ms=[   0 1 0 0 0 ; 0 0 1 0 0 ; 0 0 0 1 0 ; 0 0 0 0 1];

% set up output weighting on e1


ntmp=70*[1 100];dtmp=[1 10000];
num1=[ntmp;z;z;z; z;ntmp;z;z; z;z;ntmp;z; z;z;z;ntmp];
den1=dtmp;
[a1,b1,c1,d1]=tfm2ss(num1,den1,4,4);

% set up input weighting on d2

a2=[];b2=[];c2=[];d2=eye(4);
```

PFHIM.M

```
%*********************************************************************
%       This program forms the packed P matrix using data from acdfhim.m
%       for the margin constraint of the mixed problem
%*********************************************************************

nsp=length(am);

% Get the lengths of each state space, and other important lengths


nsi=length(ai);ii=length(di(1,:));oi=length(di(:,1));
nsd=length(ad);id=length(dd(1,:));od=length(dd(:,1));
nsI=length(aI);iI=length(dI(1,:));oI=length(dI(:,1));
ns1=length(a1);i1=length(d1(1,:));o1=length(d1(:,1));
ns2=length(a2);i2=length(d2(1,:));o2=length(d2(:,1));

ip=length(dm(1,:));op=length(dm(:,1));
ia=length(da(1,:));oa=length(da(:,1));
in=length(dn(1,:));on=length(dn(:,1));

% Now form A,B1,B2,C1,C2,D11,D12,D21,D22

A=[am zeros(nsp,ns1+ns2) ;
  b1*cm a1 zeros(ns1,ns2);
   zeros(ns2,nsp+ns1) a2 ];

B1=[ zeros(nsp+ns1,i2);b2];

B2=[bm;zeros(ns1+ns2,5)];

B=[B1 B2];

C1=[d1*cm zeros(o1,ns1+ns2) ];

C2=[cm zeros(i2,ns1) c2;
    zeros(2,nsp+ns1+ns2) ];

C=[C1;C2];


D11=zeros(o1,i2);

D12=[zeros(o1,5)];

D21=[ d2;
    zeros(ii,in)];
```

```
D22=[zeros(ii+i2,ip)];

D=[D11 D12;
  D21 D22];

sys=pck(A,B,C,D);
```

After a controller was found, the following program set up the plant and controller to be used in the evaluation model.

SIMSET2.M

```
%*************************************************************
%       This program sets up the controller state space and the
%       aircraft evaluation state space.
%
%*************************************************************

atm=ap;ctm=cp;

dtm(:,1:2)=dp(:,1:2);dtm(:,3)=dp(:,3)+dp(:,4);dtm(:,4)=dp(:,5)+dp(:,7);
dtm(:,5)=dp(:,6)+dp(:,8);

dtm(:,6)=zeros(7,1);

btm(:,1:2)=bp(:,1:2);btm(:,3)=bp(:,3)+bp(:,4);btm(:,4)=bp(:,5)+bp(:,7);
btm(:,5)=bp(:,6)+bp(:,8);

btm(:,6)=ap(:,3);
```

The following programs made the response and actuator plots once the Simulink model was run.

RESP.M

```
%*************************************************************
%
%       This program takes the data sent to the workspace by
%       the evaluation model and plots out the ideal and
%       actual velocity, angle of attack, and pitch rate
%       responses.
%
%*************************************************************


subplot(3,1,1),plot(t,y(:,3),t,yi(:,3),'--')
xlabel('Time (sec)'),ylabel('Ft/s'),title('U Response')

subplot(3,1,2),plot(t,y(:,5),t,yi(:,1),'--')
```

```
xlabel('Time (sec)'),ylabel('Degrees'),title('Alpha Response')

subplot(3,1,3),plot(t,y(:,6),t,yi(:,2),'--')
xlabel('Time (sec)'),ylabel('Degrees/sec'),title('Q Response')
```

ACT2.M

```
%*************************************************************
%
%       This program plots the actuator deflections produced by
%       the evaluation model.
%
%*************************************************************

subplot(2,2,1),plot(t,u(:,1))
xlabel('Time (sec)'),ylabel('Degrees')
title('Canard')

subplot(2,2,2),plot(t,u(:,2))
xlabel('Time (sec)'),ylabel('Degrees')
title('Stabilator')

subplot(2,2,3),plot(t,u(:,3))
xlabel('Time (sec)'),ylabel('Degrees')
title('Trailing Flap/Aileron')


subplot(2,2,4),plot(t,u(:,4),t,u(:,5),'--')
xlabel('Time (sec)'),ylabel('Degrees')
title('Vanes top/bot(--)')
```

The closed loop system was found using the following program.

CLP3.M

```
%*************************************************************
%
%       This program uses the data produced by ACDF*.M with the
%       controller to calculate the closed loop state space.
%
%*************************************************************
a=am;

b1=zeros(4,2);
b2=bm;

c1=cm;
c2=[cm;zeros(2,4)];

d11=zeros(5,2);
```

```
d12=dm;

d21=[zeros(4,2);eye(2)];
d22=[dm;zeros(2,5)];

s=inv(eye(5)-dk*d22);

acl=[a+b2*s*dk*c2 b2*s*ck;bk*c2+bk*d22*s*dk*c2 ak+bk*d22*s*ck];

bcl=[b1+b2*s*dk*d21;bk*d21+bk*d22*s*dk*d21];

ccl=[c1+d12*s*dk*c2 d12*s*ck];

dcl=[d11+d12*s*dk*d21];
```

The open loop state spaces and complementary sensitivity state space were set up using the following program.

MARG2.M

```
%*********************************************************************
%
%       This program finds K1, K2, open loop GK1, and T state space
%
%*********************************************************************

% form the plant

msel=[zeros(4,2) eye(4) zeros(4,1)];
cpy=msel*cp;dpy=msel*dp;
%ms=eye(4);

% Break up the controller into k1,k2

clear ak1 ,clear bk1 ,clear ck1 ,clear dk1 ,clear ak2 ,clear bk2 ,
clear ck2 ,clear dk2 ,clear tmp

ak1=ak;ak2=ak;
nsk=length(ak);

bk2(1:nsk,1:2)=bk(1:nsk,5:6);
tmp(1:nsk,1:4)=bk(1:nsk,1:4);
bk1=tmp;

ck2(:,1:nsk)=ck(1:5,1:nsk);
ck1(:,1:nsk)=ck(1:5,1:nsk);

dk2(:,1:2)=dk(1:5,5:6);
dk1(:,1:4)=dk(1:5,1:4);
```

```
dk1=dk1;

% Now form Lo = GK1

[alo,blo,clo,dlo]=series(ak1,bk1,ck1,dk1,am,bm,cm,dm);

[at,bt,ct,dt]=cloop(alo,blo,clo,dlo,1);
```

The mixed H_2 / H_∞ routine used a large number of programs that were developed by David Walker and Brett Ridgely. The following lists those that were developed for this particular problem.

The mixed tracking problems were set up by the following program.

SETM3.M

```
%*************************************************************************
%
%       This program sets up h2/hi tracking problem.
%
%*************************************************************************
global A2 Ai Bd Bw Bu2 Bui Ce Cz Cy2 Cyi Ded Deu Dzu Dyd Dyw
global ns2 nsi ndd nww nuu nee nzz nyy ncc

disp('forming H-infinity problem')
acdfhi2
pfhi2;
disp('forming H2 problem')
acdfh2
pfh2;

A2=A2;  Ai=A;

Bd=B1;  Bw=Bw;  Bu2=Bu;  Bui=B2;

Ce=C1;  Cz=Cz;  Cy2=Cy;  Cyi=C2;

Deu=D12;  Dzu=Dzu;  Dyd=D21;  Dyw=Dyw;  Dyu=Dyu;  Ded=D11;


[ns2,ns]=size(A2);
[nsi,ns]=size(Ai);
[nd,ndd]=size(Bd);
[nw,nww]=size(Bw);
nuu=length(Bu(1,:));
nee=length(Ce(:,1));
nzz=length(Cz(:,1));
nyy=length(Cyi(:,1));
```

```
Dzw=zeros(nzz,nww); Dyu2=Dyu; Dyui=Dyu;
```

The mixed margin problem was set up by the following program.

SETM5.M

```
%*************************************************************************
%
%       This program sets up h2/hi margin problem.
%
%*************************************************************************
global A2 Ai Bd Bw Bu2 Bui Ce Cz Cy2 Cyi Ded Deu Dzu Dyd Dyw
global ns2 nsi ndd nww nuu nee nzz nyy ncc



disp('forming H-infinity problem')
acdfhim
pfhim;
disp('forming H2 problem')
acdfh2
pfh2;

A2=A2; Ai=A;

Bd=B1; Bw=Bw; Bu2=Bu; Bui=B2;

Ce=C1; Cz=Cz; Cy2=Cy; Cyi=C2;

Deu=D12; Dzu=Dzu; Dyd=D21; Dyw=Dyw; Dyu=Dyu; Ded=D11;


[ns2,ns]=size(A2);
[nsi,ns]=size(Ai);
[nd,ndd]=size(Bd);
[nw,nww]=size(Bw);
nuu=length(Bu(1,:));
nee=length(Ce(:,1));
nzz=length(Cz(:,1));
nyy=length(Cyi(:,1));

Dzw=zeros(nzz,nww); Dyu2=Dyu; Dyui=Dyu;
```

The actual program was run by two different programs. One starts the run and the other restarts. A specific starting program was used for each case but could be restarted by the same program. The following are used to start the runs.

RUN3RD.M

```
%*****************************************************************
% This script is the driver for the H /H   optimization
%                                   2 inf
%
% routine.  It sets up the problem then iterates over
%
% The routine is entered with a gamma and a close X.
%
% for the problem.  This is for the third order tracking problem.
%*****************************************************************
global lambda gradtype iterd ncc
iterd=10%iterd=input('Number of DFP iterations:' );
gradtype=1;
format long e
lambda=1%lambda=input('Enter lambda: ')
setm3;
load third;
ncc=3;
[Ac,Bc,Cc,Dc]=unpck(k);
X=setupx(Ac,Bc,Cc);
X=X';
[Tzw Ted]=fun2(X)
gam1=Ted
gam=gam1;
[n,nn]=size(X);
normvec=[Tzw Ted Ted 0 0];
xvec=[X'];
del=input('Enter the absolute step size for gamma:  ')
steps=input('Enter the number of steps desired:  ')
for ii=1:steps
 gam=gam-del
 [Xf,iter,err]=dfp('h2inf',X,gam);
 [Tzw,Ted]=fun2(Xf)
 normvec=[normvec;Tzw Ted gam iter err];
 xvec=[xvec;Xf'];
 X=Xf;
 save out3rd normvec xvec;
 plot(normvec(:,2),normvec(:,1));grid;
end
```

RUNM3.M

```
%****************************************************************
% This script is the driver for the H /H    optimization
%                                    2 inf
%
% routine.  It sets up the problem then iterates over
%
% The routine is entered with a gamma and a close X.
%
% for the problem.  This is for the fourth order tracking problem.
%****************************************************************
global lambda gradtype iterd
iterd=10%iterd=input('Number of DFP iterations:' );
gradtype=1;
format long e
lambda=1%lambda=input('Enter lambda: ')
setm3;
[k,g]=h2syn(sys,6,5);
[Ac,Bc,Cc,Dd]=unpck(k);
X=setupx(Ac,Bc,Cc);
X=X';
[Tzw Ted]=fun2(X)
gam1=Ted
gam=gam1;
[n,nn]=size(X);
normvec=[Tzw Ted Ted 0 0];
xvec=[X'];
del=input('Enter the absolute step size for gamma:  ')
steps=input('Enter the number of steps desired:  ')
for ii=1:steps
 gam=gam-del
 [Xf,iter,err]=dfp('h2inf',X,gam);
 [Tzw,Ted]=fun2(Xf)
 normvec=[normvec;Tzw Ted gam iter err];
 xvec=[xvec;Xf'];
 X=Xf;
 save outm3 normvec xvec;
 plot(normvec(:,2),normvec(:,1));grid;
end
```

RUNM5.M

```
%************************************************************
% This script is the driver for the H /H    optimization
%                                   2  inf
%
% routine.  It sets up the problem then iterates over
%
% The routine is entered with a gamma and a close X.
%
% for the problem.  This is for the fourth order margin problem.
%************************************************************
global lambda gradtype iterd
iterd=10%iterd=input('Number of DFP iterations:' );
gradtype=1;
format long e
lambda=1%lambda=input('Enter lambda: ')
setm5;
load fourth; Ac=ak;Bc=bk;Cc=ck;
X=setupx(Ac,Bc,Cc);
X=X';
[Tzw Ted]=fun2(X)
gam1=Ted
gam=gam1;
[n,nn]=size(X);
normvec=[Tzw Ted Ted 0 0];
xvec=[X'];
del=input('Enter the absolute step size for gamma:  ')
steps=input('Enter the number of steps desired:  ')
for ii=1:steps
 gam=gam-del
 [Xf,iter,err]=dfp('h2inf',X,gam);
 [Tzw,Ted]=fun2(Xf)
 normvec=[normvec;Tzw Ted gam iter err];
 xvec=[xvec;Xf'];
 X=Xf;
 save outm5 normvec xvec;
 plot(normvec(:,2),normvec(:,1));grid;
end
```

The two programs used to restart the runs are as follows. One is for fourth order and the other is for third.

RROPT.M

```
%*************************************************************
% This script is the driver for the H /H   optimization
%                                   2 inf
%
% routine.  It sets up the problem then iterates over
%
% The routine is entered with a gamma and a close X.
%
% for the problem.  Restarts fourth order runs.
%*************************************************************
global lambda gradtype iterd ncc
iterd=input('Number of DFP iterations:' );
gradtype=1;
format long e
lambda=input('Enter lambda: ');
setup=input('Name of Setup File:  ','s');
eval(setup);
datainput=input('From what datafile is the original normplot drawn?  ','s');
rerun=['rr'];
outvec=[rerun datainput];
eval(['load ',datainput])
size(normvec)
disp('Beyond which point are we going?  ');
point1=input('')
X=xvec(point1,:);
X=X';
ncc=length(A2)%input('input order of controller','s')
[Tzw Ted]=fun2(X)
gam1=Ted
gam=gam1;
[n,nn]=size(X);
normvec=[Tzw Ted Ted 0 0];
xvec=[X'];
del=input('Enter the absolute step size for gamma:  ')
steps=input('Enter the number of steps desired:  ')
for ii=1:steps
 gam=gam-del
 [Xf,iter,err]=dfp('h2inf',X,gam);
 [Tzw,Ted]=fun2(Xf)
 normvec=[normvec;Tzw Ted gam iter err];
 xvec=[xvec;Xf];
 X=Xf;
eval(['save ',outvec,' normvec xvec']);
 %save outm1 normvec xvec;
 plot(normvec(:,2),normvec(:,1));grid;
end
```

RROPT3.M

```
%****************************************************************
% This script is the driver for the H /H   optimization
%                                    2 inf
%
% routine.  It sets up the problem then iterates over
%
% The routine is entered with a gamma and a close X.
%
% for the problem.  This restarted a third order run.
%****************************************************************
global lambda gradtype iterd ncc
iterd=input('Number of DFP iterations:' );
gradtype=1;
format long e
lambda=input('Enter lambda: ');
setup=input('Name of Setup File:  ','s');
eval(setup);
datainput=input('From what datafile is the original normplot drawn?  ','s');
rerun=['rr'];
outvec=[rerun datainput];
eval(['load ',datainput])
size(normvec)
disp('Beyond which point are we going?  ');
point1=input('')
X=xvec(point1,:);
X=X';
ncc=3%input('input order of controller','s')
[Tzw Ted]=fun2(X)
gam1=Ted
gam=gam1;
[n,nn]=size(X);
normvec=[Tzw Ted Ted 0 0];
xvec=[X'];
del=input('Enter the absolute step size for gamma:  ')
steps=input('Enter the number of steps desired:  ')
for ii=1:steps
 gam=gam-del
 [Xf,iter,err]=dfp('h2inf',X,gam);
 [Tzw,Ted]=fun2(Xf)
 normvec=[normvec;Tzw Ted gam iter err];
 xvec=[xvec;Xf'];
 X=Xf;
eval(['save ',outvec,' normvec xvec']);
 %save outm1 normvec xvec;
 plot(normvec(:,2),normvec(:,1));grid;
end
```

Once a solution was found, the controller was extracted from the solution vector with the following function.

GETK.M

```
function [ac,bc,cc,dd]=getk(x,ncc)
%*************************************************************************
% This function produces the controller state space from the mixed
% solution vector
%*************************************************************************

%global A2 Ai Bu2 Bui Cy2 Cyi Bd Dyd Bw Dyw Ce Deu Cz Dzu Ded
%global ncc ndd nee nuu nyy

nc=ncc;
nyy=6;nuu=5;

for i=1:nc
 for j=1:nc
  ac(i,j) = x((i-1)*nc+j);
 end
 for j=1:nyy
  bc(i,j) = x(nc*nc+(i-1)*nyy+j);
 end
 for j=1:nuu
  cc(j,i) = x((nc+nyy)*nc+(i-1)*nuu+j);
 end
end
dd=zeros(nuu,nyy);
```

The margins found in Chapter 4 and Chapter 5 were found by the following function which used information from MARG2.M.

IGM.M

```
function [gm2,pm2]=igm(a,b,c,d,mI)
l=length(d);
%*************************************************************************
%       This function calculates the vector margins given the
%       complementary senstivity state space
%*************************************************************************

w=logspace(-4,3,700);
% calculate infnorm of T

eig(a)
[sv,w]=sigma(a,b,c,d);

a0=1/max(max(sv))
```

```
g1=1-a0;g2=1+a0;
gm2=[20*log10(g1) 20*log10(g2)];
pm2=2*asin(a0/2)*180*7/22;
```

Bibliography

[1] Moorhouse, David J., Kevin D. Citurs. "The Control System Design Methodology of the STOL & Maneuver Technology Demonstrator," International Journal of Control, to be published in early 1994.

[2] Moorhouse, David J. "Lessons Learned from the S/MTD Program for the Flying Qualities Specification," AIAA Paper No. 90-2849-CP.

[3] Moorhouse, David J., David B. Leggett, and Kenneth A. Feeser. "Flying Qualities Criteria for Precise Landing of a STOL Fighter," AIAA Atmospheric Flight Mechanics Conference, Paper No. 89-3390. Washington DC: American Institute of Aeronautics and Astronautics, August 1989.

[4] WRDC-TR-89-3050, Volume IV,(USAF), "STOL and Maneuver Technology Demonstrator (S/MTD) Program," May 1989.

[5] Moorhouse, David J., Kevin D. Citurs, Richard W. Thomas, and Mark R. Crawford. "The Handling Qualities of the STOL and Maneuver Technology Demonstrator from Specification to Flight Test," AGARD, Flying Qualities 9, (see N91-23108 15-05), February 1991.

[6] Iwasaki, T., and R. E. Skelton. "All Low Order H_∞ Controllers: Observer-Based Structure and Covariance Bounds," submitted to IEEE Transactions on Automatic Control, April 1993.

[7] Sweriduk, Gregory D., and Anthony J. Calise. "Robust Fixed Order Dynamic Compensation: A Differential Game Approach," for the Conference on Aerospace Control Systems, West Lake Village, CA, May 1993.

[8] Hsu, Chin S., Xianggang Yu, Hsi-Han Yeh, and Siva S. Banda. "H_∞ Compensator Design with Minimal Order Observers," submitted to American Control Conference, San Francisco, CA, June 1993.

[9] Osborne, Robert C., Richard J. Adams, Chin S. Hsu, and Siva S. Banda. "Reduced-Order H_∞ Compensator Design for an Aircraft Control Problem," Proceedings of The Second IEEE Conference on Control Application, 2: 853-858 (September 1993).

[10] DeShetler, W. Bruce, and D. Brett Ridgely. "H_∞ Reduced Order Optimization With Zero and Non-Zero D_C Terms," submitted to the 32nd Conference on Decision and Control, December 1993.

[11] Walker, David E., and D. Brett Ridgely. "Reduced Order Mixed H_2 / H_∞ Optimization with a Singular H_∞ Constraint," submitted to 1994 American Control Conference and the International Journal of Control.

[12] Ridgely, D. B. "A Nonconservative Solution to the General Mixed H_2 / H_∞ Optimization Problem". PhD dissertation. Massachusetts Institute of Technology, Cambridge MA, 1991.

[13] R.L. Dailey, "Lecture Notes for the Workshop on H_∞ and μ Methods for Robust Control", in conjunction with American Control Conference, San Diego, CA, 21-22 May 1990.

[14] J.C. Doyle, K. Glover, P.P. Khargonekar and B.A. Francis, "State-Space Solutions to Standard H_2 and H_∞ Control Problems", IEEE Transactions on Automatic Control, Vol. 34, No. 8, Aug. 1989, pp. 831-847.

[15] Giesy, Daniel P., and Kyong B. Lim. "H_∞ Norm Sensitivity Formula with Control System Design Applications," AIAA Journal of Guidence Control, and Dynamics, Vol. 16, No. 6 November-December 1993.

[16] Matlab/Simulink, The MathWorks, Inc., Cochituate Place, 24 Prime Park Way, Natick, Massachusetts 01760.

[17] Moorhouse, David J., SMTD Chief Engineer, WL/FIMS, Wright Patterson AFB OH. Personal interview. 1 September 1993.

[18] Stephens, M.J. and D. Brett Ridgely, "Autopilot Design for a Tail-Controlled Missile Using Two-Degree-of-Freedom Controllers", submitted to the Symposium on Automatic Control in Aerospace, Palo Alto, CA, September 1994.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	March 1994	Master's Thesis

4. TITLE AND SUBTITLE
DIRECT REDUCED ORDER MIXED H-TWO / H-INFINITY CONTROL FOR THE SHORT TAKE-OFF AND LANDING/MANEUVER TECHNOLOGY DEMONSTRATOR (STOL/MTD)

5. FUNDING NUMBERS

6. AUTHOR(S)
William C. Reigelsperger Jr., Second Lieutenant, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology, WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GAE/ENY/94M-3

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
David Moorhouse, WL/FIMS
2645 5th St Suite 16
WPAFB OH 45433-7922

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release: distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

One of the conclusions from the STOL/MTD program was the need for a multivariable method of designing controllers of low order. This research investigated that problem by studying reduced order mixed H-two/H-infinity control theory applied to the STOL Landing configuration which employs both thrust vectoring and the use of a canard. Model matching techniques were used to obtain responses that met handling qualities criteria and reduced pilot workload by decoupling pitch rate and velocity commands. The time responses were found through nonlinear simulation and showed that the full order designs did match the ideal models very well and had good noise and wind rejection. Singular value analysis showed that the commands were decoupled very well. The reduced order method was mixed H-two/H-infinity optimization. A fourth order controller that had good performance was found by using a performance constraint, and a fourth order controller that provided good margins was found using a robustness constraint. A third order controller was also found with a performance constraint. Recommendations for finding a low order controller, with good performance and robustness are given

14. SUBJECT TERMS	15. NUMBER OF PAGES
Mixed H-two/H-infinity Optimization, STOL/MTD, Direct Reduced Order Optimization	157
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL