Linear Parameter-Varying Control of a Ducted Fan Engine

Bobby Bodenheimer, Pascale Bendotti, and Michael Kantner
Mail Code 116-81
Dept. of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125

Submitted to *The International Journal of Robust and Nonlinear Control*

January 16, 1995

Abstract

Parameter-dependent control techniques are applied to a vectored thrust, ducted fan engine. The synthesis technique is based on the solution of Linear Matrix Inequalities and produces a controller which achieves specified performance against the worst-case time variation of measurable parameters entering the plant in a linear fractional manner. Thus the plant can have widely varying dynamics over the operating range. The controller designed performs extremely well, and is compared to an $\mathcal{H}_\infty$ controller.

1 Introduction

Recent advances in optimal control theory use linear matrix inequalities (LMIs) extensively in an attempt to develop theoretical and computational machinery for gain scheduling.[13] In this paper, we present the first application of this technique to an actual physical system, a ducted fan engine. By applying this technique to a specific nonlinear system, we hope to gain insight into its utility and, more generally, generate new ideas and insights into the problem of nonlinear robust control.

A picture of the experimental system, a vectored thrust ducted fan engine, is shown in Figure 1. It consists of a high-efficiency electric motor with a 6-inch diameter blade, capable of generating up to 9 Newtons of thrust. Flaps on the fan allow the thrust to be vectored from side to side and even reversed. The engine is mounted on a three degree of freedom stand which allows horizontal and vertical translation as well as unrestricted pitch angle. In Choi *et al.*,[6] a detailed description of the performance of the fan was given, including models for the thrust as a function of flap angle and fan speed, as well as some discussion of ground effects.

In designing controllers for plants which operate over a wide dynamic range, a common technique is to schedule various fixed-point designs. Unfortunately, there are no known methods for scheduling such controllers which provide an *a priori* guarantee on the resulting

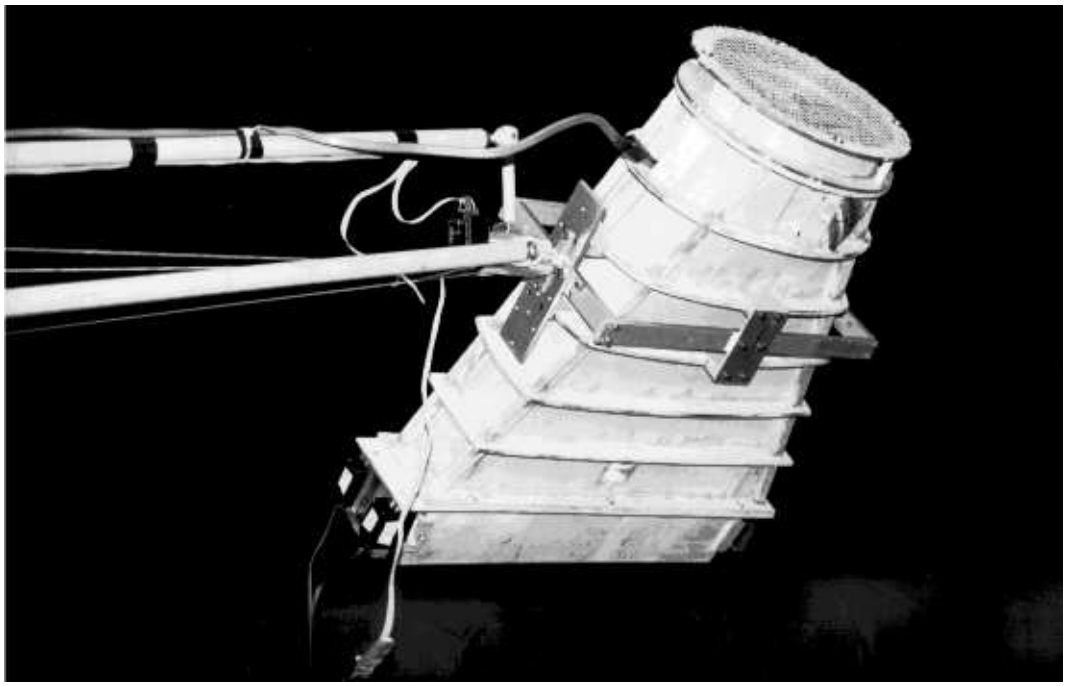

Figure 1: Ducted fan apparatus

performance or stability of the closed-loop system. Additionally, large and often unacceptable transients can occur when switching between controllers. Recent advances in optimal control theory provide a design technique which avoids these difficulties by producing an optimal parameter-dependent controller; i.e., the controller is already scheduled depending on parameter values which are not known beforehand.[1,13] The controller is optimized to provide performance against the worst-case time variation of the parameters. Such a controller is called a linear parameter-varying (LPV) controller. The controller is synthesized via an iteration which involves solving three LMIs at each step.

A possible approach to control the ducted fan is to design a parameter-dependent controller with various measurements, such as velocity and pitch, as the parameters. One advantage such a controller would have over a standard gain-scheduled controller is that performance and stability could be guaranteed over the operating range of the plant, and large transients in switching are avoided. An additional advantage of LPV synthesis is that the controller is designed in one step, rather than designing several controllers and then scheduling them. The potential drawback of LPV synthesis is that an LPV optimal controller is optimized against a worst-case time variation of parameters, and this time-variation may be so unrealistic that the controller has no performance. One of our goals in this paper is to determine if LPV synthesis can produce controllers which have reasonable performance, or better performance than linear controllers.

There is a large literature on vectored propulsion systems and they are gaining popu-

larity as a method of improving the performance capabilities of modern jet aircraft. The fundamental concepts in vectored propulsion are described in the book by Gal-Or[9] (see also the survey article[8]). Most of the existing literature and experiments concentrate on control of full-scale jet engines and are primarily concerned with extending the flight envelope by extending existing (linear) control methodologies. Our goal is to explore the nonlinear nature of flight control systems in a laboratory setting. A similar experiment has been constructed by Hauser at the University of Colorado, Boulder.[11]

Although we are aware of LPV control being applied to examples and simulations,[2–5] we believe this is the first application of these techniques to a real physical example. Our controller performance can be compared with other linear, nonlinear, and gain scheduled controllers previously designed for the ducted fan.[10]

In Section 2, we describe the configuration of the ducted fan and discuss its dynamics. In Section 3, the derivation of the parameter-varying models is presented. Section 4 provides a theoretical background on the parameter-varying synthesis. The experimental results of implementing the parameter-varying controllers are presented with analysis in Section 5. We conclude with a discussion of our findings and present avenues for future work.

2 Description of the Fan Engine

2.1 Hardware

The overall experimental setup consists of a ducted fan attached to a three degree of freedom stand, as shown in Figure 2. The different thrust modes available are shown in Figure 3. The intent of the design was to have a simple ducted fan aircraft which could provide two dimensional vectored and reverse thrust. The aircraft is bolted to a rotating arm, which limits its motion to three degrees of freedom: one rotational and two translational, approximately on the surface of a sphere defined by the arm. With this geometry, the ducted fan is completely controllable with just the vectored thrust. A detailed discussion of the components is available elsewhere.[6]

The aircraft is composed of a variable speed electric motor which drives a four-blade propellor. The motor and propellor assembly are mounted inside a wooden duct which has two flaps attached at the end. The flaps provide vectored and even reverse thrust. The pitch stability of the fan is configurable, and can be changed from stable to unstable. For these experiments, the ducted fan was in a stable configuration. An optical encoder with an angular resolution of $\pi/1024$ radians is mounted on each axis.

2.2 Software interface

The experiment is interfaced to an 80486 computer running an MS-DOS-based real-time kernel called Sparrow.[12] Custom hardware is used to read in joint angles via the encoders and generate PWM signals for the R/C servos and a speed controller at a software selectable rate. Currently, the joint angles are read in at 200Hz and the PWM signals are output at 50Hz, the standard update rate for R/C servos. The R/C servos control the flap angles and

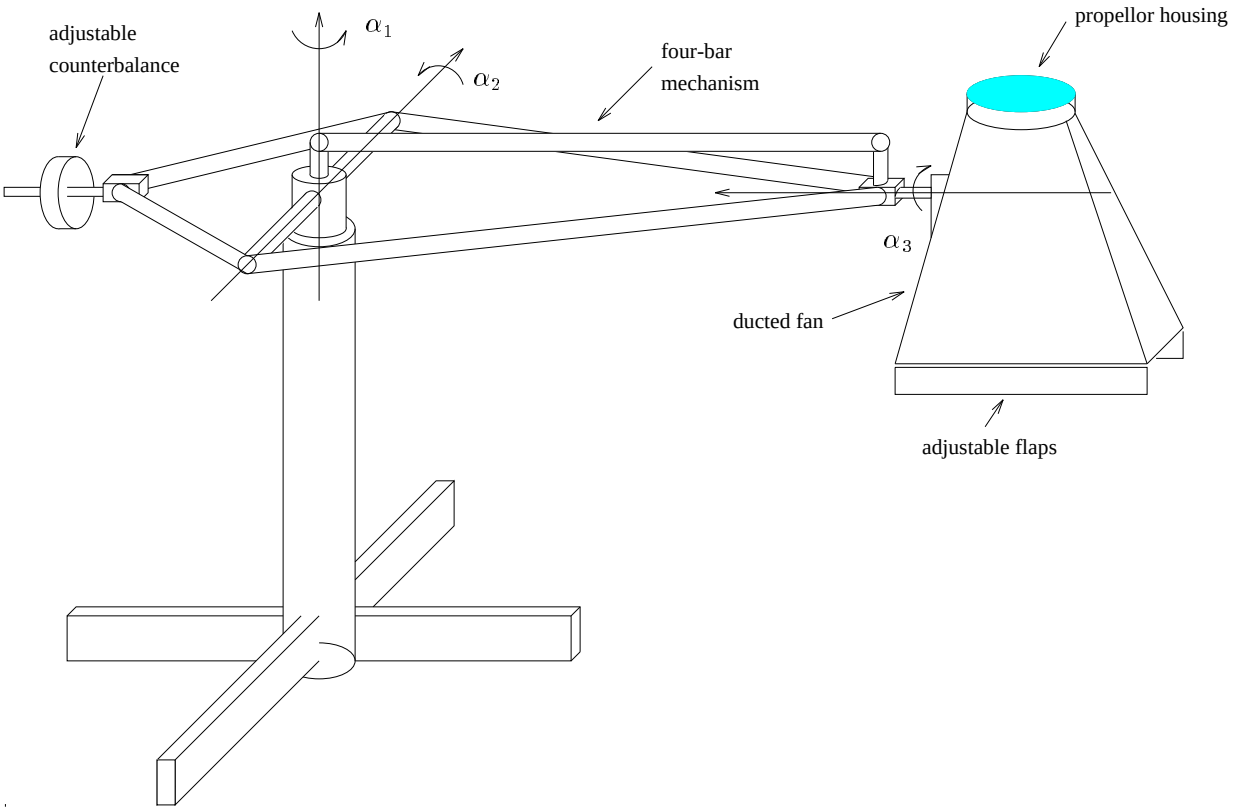

Figure 2: Ducted fan attached to stand.

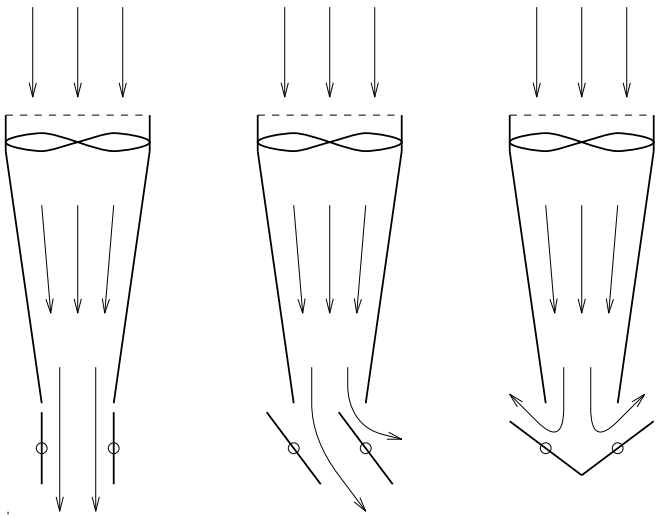

Figure 3: Different thrust modes for the ducted fan.

the speed controller controls the motor speed, and consequently the fan force. The finite angular resolution of the encoders and unavoidable delay in observing states of the system make it desirable to run the controller as fast as possible.

Controllers are designed and simulated using MATLAB on Sun workstations. Sparrow loads linear, LPV, and some types of gain-scheduled controllers directly from MATLAB data files. Once a controller is designed, it can be tested immediately. Nonlinear controllers, implemented as MATLAB S-functions, require a small amount of revision before they are linked to Sparrow. Further discussion on the implementation of LPV controllers is found in section 5.

3 Modelling

All controllers are designed using a first principles model of the ducted fan based on standard rigid body mechanics. A six state model, (α_1, α_2, α_3, $\dot{\alpha}_1$, $\dot{\alpha}_2$, $\dot{\alpha}_3$), was selected for the control designs. The equations of motion for the system, derived from Lagrange's equations, have the functional form

$$M(\alpha)\ddot{\alpha} + C(\alpha, \dot{\alpha})\dot{\alpha} + N(\alpha) = T(\alpha, F),$$

where $M(\alpha)$ is the generalized inertia matrix, $C(\alpha, \dot{\alpha})$ is the Coriolis matrix, $N(\alpha)$ is the matrix of gravity terms and $T(\alpha, F)$ is the matrix of applied joint torques.[6]

The model is accurate enough for control design, although it does have limitations. Initial step responses on single axes compared nicely with the experiment , and a PID control test gave expected results.[6] A decoupling controller, essentially a plant inversion, worked well.[10] Nonetheless, the model omits many effects; improving the model by identifying and including them is an ongoing effort. Effects which are omitted include all actuator dynamics, sensor limitations, friction, and aerodynamic effects. The speed controller has about a 0.3 second lag, while the servo loop controlling flap position is much faster. Static friction about the α_1 axis is significant. Aerodynamic effects have been observed in the lab during forward flight.

The model does not include the gyroscopic terms that result from the angular momentum of the whirling fan blade. However, a four bar mechanism constrains the fan to rotate about an axis that is always parallel to the ground, so the gyroscopic forces generated by rotation of the fan blade coupled with pitch motion are completely taken up by the stand and do not enter into the dynamics. Additional gyroscopic forces are present due to rotation about α_1 coupled with the fan blade rotation.

The model also assumes that the commanded forces act at a fixed point on the fan. Experiments have shown that the distance from the fan's center of mass to the point at which the force acts, r, varies as the flap angle changes. Furthermore, motor speed and flap angle, not forces, are commanded. An experimentally determined lookup table maps desired forces to motor speed and flap angle.

An examination of the nonlinear model reveals that the most significant variations in parameters occur as a function of α_3 and $\dot{\alpha}_1$. The fan is strictly proper and thus the D matrix of the state space model is zero. Moreover, although the rates are not measured, an

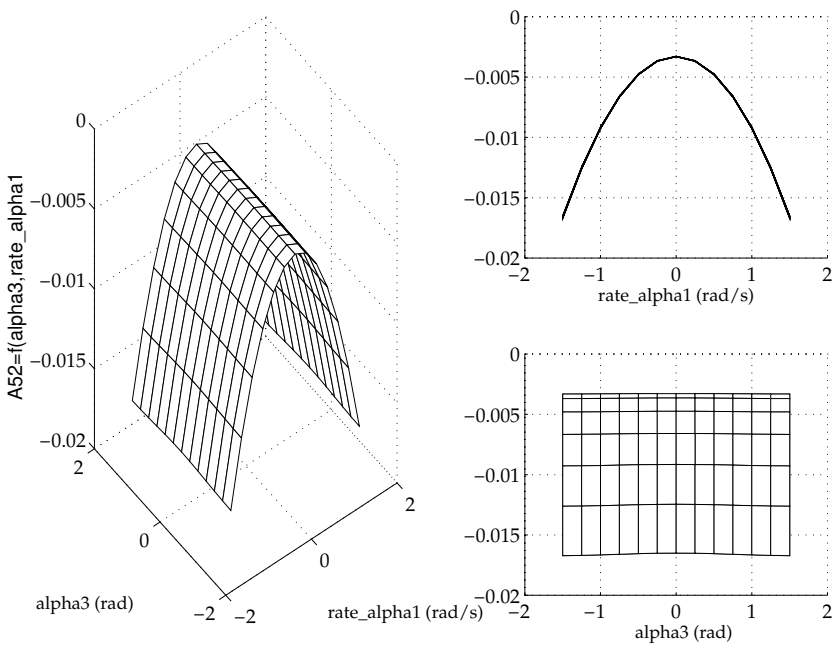

Figure 4: Dependence of a_{52} upon α_3 and $\dot{\alpha}_1$.

inner loop of the software control estimates them; hence in our models $C = I$. The A and B matrices are the only matrices which have parameter variations. Their structure is

$$\left[\begin{array}{c|c} A & B \end{array}\right] = \left[\begin{array}{cccccc|cc} 1 & 0 & 0 & T & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 & 0 \\ 0 & a_{42}(\alpha_3,\dot{\alpha}_1) & a_{43}(\alpha_3) & 1 & 0 & 0 & b_{41}(\alpha_3) & b_{42}(\alpha_3) \\ 0 & a_{52}(\dot{\alpha}_1) & a_{53}(\alpha_3) & 0 & 1 & 0 & b_{51}(\alpha_3) & b_{52}(\alpha_3) \\ 0 & a_{62}(\alpha_3,\dot{\alpha}_1) & a_{63}(\alpha_3) & 0 & 0 & 1 & b_{61}(\alpha_3) & b_{62}(\alpha_3) \end{array}\right] \tag{1}$$

where T is the sampling rate. Note, since one of the goals of this paper is to investigate the applicability general LPV techniques, we will not exploit the state feedback nature of the problem.

Figures 4 and 5 show the parameter dependence of one term in the A matrix and all terms in the B matrix, important for the simple control model we will be employing. The remaining variations are shown in the Appendix, in Figures 25 through 29. The dependence is obtained by linearizing the nonlinear model at various equilibrium operating points for different values of α_3 and $\dot{\alpha}_1$. Note the dependence of a_{42} and a_{62} on both α_3 and $\dot{\alpha}_1$, while a_{52} depends only on $\dot{\alpha}_1$; a_{43}, a_{53}, a_{63} and all terms of the B matrix depend only on α_3. In Figure 5, the actual values are shown as asterisks, and the least-squares fit described below is the solid line.

To construct a model of the ducted fan suitable for LPV control, we assume the system has the general time-varying structure shown in Figure 6, where $x(k), e(k), y(k), d(k)$, and $u(k)$

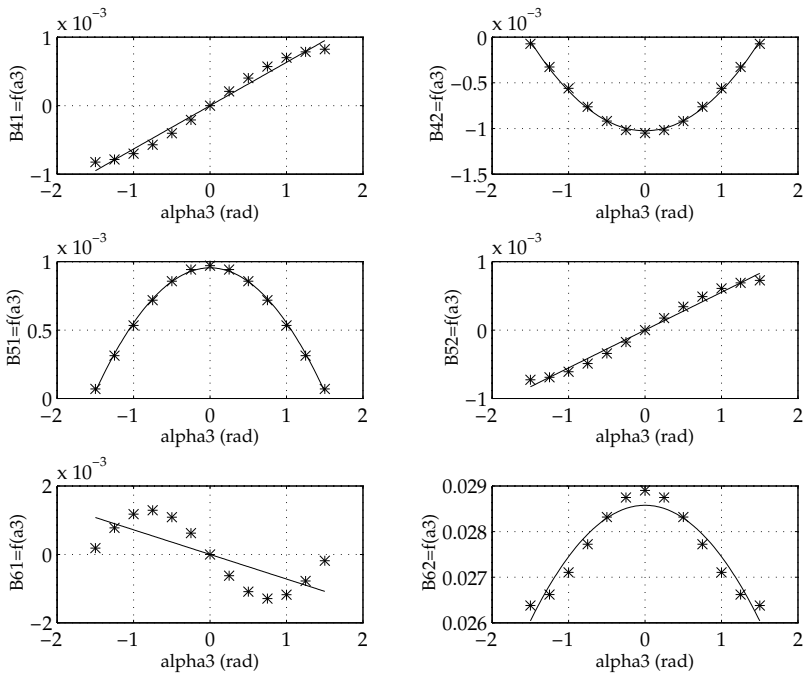

Figure 5: Dependence of the entries of the B matrix upon α_3.

are the state, error, measurement, disturbance and input vectors, respectively. We assume the time-variation of the plant can be represented as a linear-fractional transformation (LFT) of a parameter and a constant matrix. Thus $P(k)$ is given by

$$P(k) = P_{22} + P_{21}\Delta(k)\left(I - P_{11}\Delta(k)\right)^{-1} P_{12} \tag{2}$$

where

$$\Delta(k) = \begin{bmatrix} \delta_1(k) I_{n_1} & & \\ & \ddots & \\ & & \delta_m(k) I_{n_m} \end{bmatrix} \tag{3}$$

with $|\delta_i(k)| \leq 1$ for all k. The δ_i are assumed to be measurable. Any rational time-varying system can be represented in this framework, and many others can be arbitrarily closely approximated. This type of system is known as a parameter-dependent LFT system.

To derive the parameter dependence and fit it into this framework, each of the parameters mentioned above is fit with a rational function of first or second order using a least-squares technique. Immediately some approximations are made. The dependence of a_{42} and a_{62} on $\dot{\alpha}_1$ was neglected, making them depend only on α_3. The parameters a_{42}, a_{62}, b_{41}, b_{52}, and b_{61} were approximated as lines, i.e., first-order LFTs. The rest were approximated as second order LFTs. Assume in the following that $\delta_1 = z^{-1}$, $\delta_2 = \dot{\alpha}_1$, and $\delta_3 = \alpha_3$, and let

$$\Delta \ = \{diag\ [\delta_2 I_{n_2}, \delta_3 I_{n_3}]\}\,.$$

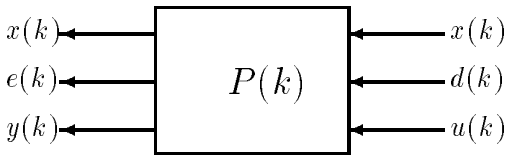

Figure 6: Time-Varying System.

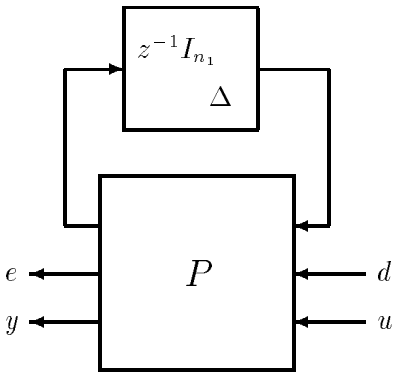

Figure 7: Parameter-Dependent Plant. The $z^{-1}I_{n_1}$ term represents the states of P, and the Δ represents the time variation of Equation 3.

The resulting model with δ-dependence, $P(\delta)$, is shown in Figure 7, where n_1 is the size of the block corresponding to δ_1.

A simplified model of the ducted fan is considered in this paper, for which the range of α_3 is assumed to be from 0 radians to 1.5 radians, and the range of $\dot{\alpha}_1$ is assumed to be from 0 rad/s to 1.5 rad/s. The software for implementing LPV controllers holds any δ at 0 if it becomes negative, and at 1.5 if it exceeds this value. The model considers only variations in the cross-coupling terms of the B matrix, i.e., b_{41}, b_{52}, and b_{61}, and the variation of a_{52}. The other terms are assumed to be constant with the values they take at hover. Thus $n_1 = 6$, $n_2 = 2$, and $n_3 = 2$ ($n_3 \neq 3$ because the fact that all three are lines can be exploited to reduce the size). We refer to this model as the "simplified pitch-velocity model", and it is the model used in our control designs.

4 Controller Synthesis

This section gives an introduction to the theoretical background for the synthesis methodology. All the results are due to Packard,[13] so the discussion is of a brief and general nature. We then discuss the synthesis for the ducted fan in detail.

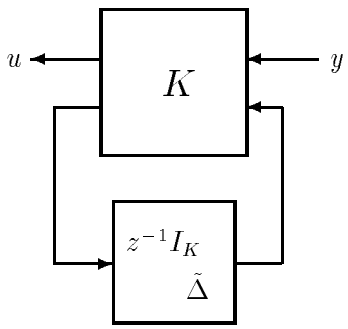

Figure 8: Parameter-Dependent Controller; $z^{-1}I_K$ represents the states of the controller and $\tilde{\Delta}$ the time variations.

4.1 Theoretical Background

Recall that the plant has the structure given in Figure 7. The controller we will design for this plant will also be parameter-dependent, depending on the same measurable δ_i's as the plant; it will thus have the form shown in Figure 8. P can be augmented to collect all the time-varying parameters and states together; K can then be treated as a simple matrix. This is depicted in Figure 9, where R is the augmented form of P, and K is a matrix. The problem thus appears as a robust control problem with a special structure on the plant and parameters. This special structure allows the problem to be cast as LMIs, which can be solved using convex optimization techniques. The design objective is to find a controller K such that the interconnection is stable and the $\ell_2 \to \ell_2$ induced norm from d to e is small for all allowable parameter variations $\Delta(k)$ (see Equation 3). This is simply a small-gain condition. Since the small-gain theorem can be quite conservative, we can reduce the conservatism by introducing scaling matrices from a set $\mathcal{D}$ which commutes with the set of parameter variations.

The resulting condition is then the state-space upper bound (SSUB).[14] Introducing the notation $\mathcal{F}_l(R,Q) = R_{11} + R_{12}Q(I - R_{22}Q)^{-1}R_{21}$ for a block partitioned 2×2 matrix R with $\det(I - R_{22}Q) \neq 0$, this condition becomes (compare Lemma 3.1 in Packard[13] and Theorem 10.4 in Packard and Doyle[14]):

Theorem 1 *Let R be determined as above, along with an uncertainty structure $\mathbf{\Delta}$. If there is a $D \in \mathcal{D}$ and a stabilizing, finite-dimensional, time-invariant K such that*

$$\left\| \begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix} \mathcal{F}_l(R,K) \begin{bmatrix} D^{-1} & 0 \\ 0 & I \end{bmatrix} \right\|_\infty < 1 \tag{4}$$

then there is a γ, $0 \leq \gamma < 1$, such that for all parameter sequences $\delta_i(k)$ with $\|\delta_i\|_\infty \leq 1$, the system in Figure 9 is internally exponentially stable, and for zero initial conditions, if $d \in \ell_2$, then $\|e\|_2 \leq \gamma \|d\|_2$.

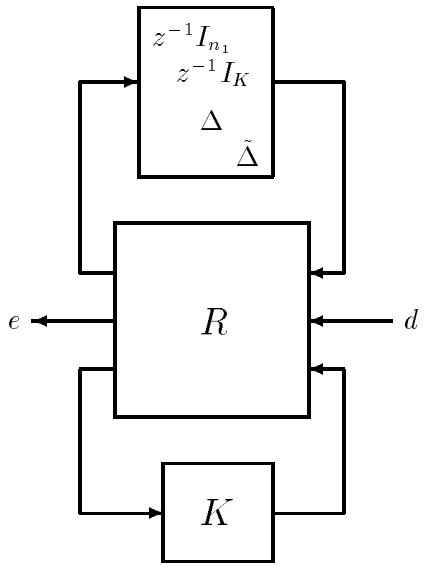

Figure 9: Parameter-Dependent Closed-Loop System.

Pictorially, this theorem is shown in Figure 10. The important fact about Theorem 1 is that the synthesis of D and K to meet the objective can be cast as a computationally tractable convex optimization problem involving 3 LMIs. These LMIs have the following form:

$$U_\perp^T \left(E \begin{bmatrix} X & 0 \\ 0 & I \end{bmatrix} E^T - \begin{bmatrix} X & 0 \\ 0 & I \end{bmatrix} \right) U_\perp < 0$$

$$V \left(E^T \begin{bmatrix} Y & 0 \\ 0 & I \end{bmatrix} E - \begin{bmatrix} Y & 0 \\ 0 & I \end{bmatrix} \right) V^T < 0$$

$$\begin{bmatrix} X & I \\ I & Y \end{bmatrix} \geq 0$$

where $U_\perp$, $V_\perp$, and E are obtained from the system realization, and X and Y are structured positive definite matrices. Interested readers may find the exact LMIs in Theorem 6.3 of Packard.[13] The synthesis procedure is a γ-iteration, as $\mathcal{H}_\infty$ is.

A few points are important in understanding the ramifications of employing the SSUB. Most importantly, this technique designs a controller optimal with respect to a time-varying perturbation with memory (the sequence $\Delta(k)$ of Equation 3). The relationship between such an operator and a parameter useful in gain-scheduling is tenuous, at best. Depending on the problem, this technique could conceivably yield controllers so conservative as to have no performance. Nonetheless, if a controller with acceptable performance can be designed with this technique, then it will achieve performance for all variations of the operating point. As a

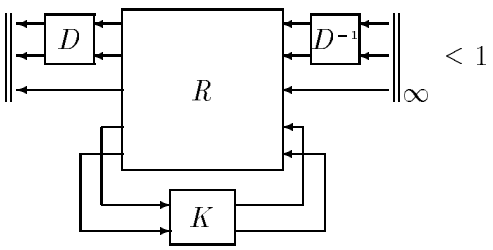

Figure 10: Diagram of Theorem 1.

corollary of this, a time-varying operator with memory in general does not have a spectrum, so there is no way to "filter" it to achieve a closer relationship to an operating parameter. Moreover, it is interesting to contrast this technique with μ-synthesis, where instead of the SSUB, the frequency-domain upper bound is usually employed; this difference reflects the different assumptions about the type of perturbations.

If Δ is a constant value, and is "wrapped into" the plant, the resulting model becomes a linear model around the operating point of the Δ. Similarly, we can do this for controllers, and the LPV controller becomes a linear controller. We will refer to the linear controller obtained by holding Δ at a constant value as the LPV controller *locked* at the value of Δ, e.g., the LPV controller locked at hover would correspond to the Δ value with α_3 and $\dot{\alpha}_1$ equal to 0. We are interested in looking at controllers locked in various positions because by comparing them with the full LPV controller, we hope to gain better insight into the nature of LPV control.

4.2 Weight Selection

The design process parallels classical $\mathcal{H}_\infty$ design and LTI weights were used. Using weights depending on parts of the Δ-block can enhance performance,[5] but was not needed here. Both LPV and $\mathcal{H}_\infty$ synthesis produce controllers which reject disturbances. A tracking problem, such as the ducted fan, can be cast in this framework by rejecting the low frequency components of the error between the plant output and the reference. The tracking will become faster as higher frequencies are rejected. The synthesis structure used is shown in Figure 11, with uncertainty and performance weights included; u, y, r, and n are the controls, the measurements, the reference signals the controller must track, and sensor noises, respectively. W_u is a penalty on the control, and W_a is multiplicative actuator uncertainty. W_p is a diagonal performance weight on the signals α_1, α_2, α_3, and $\dot{\alpha}_1$. The initial values of the weights were chosen identically to a previously designed $\mathcal{H}_\infty$ controller, and refined from there.

The basic performance criteria for a controller is to track a ramp on α_1, and a step on α_2. There is no explicit objective on α_3. The weights used in the design are shown in Figure 12. The shaded lines in this figure are a singular value Bode plot of the model linearized at hover. The integral action with a mode is the response of α_1 and the response of α_3 shares the same

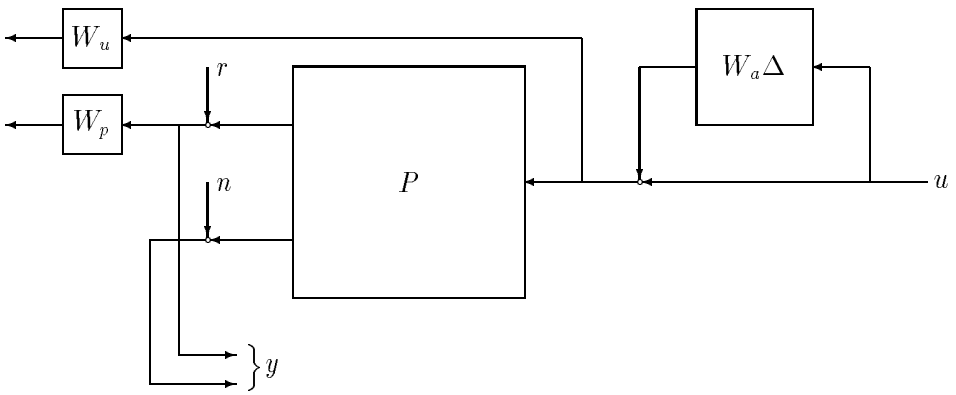

Figure 11: Synthesis Structure used for designing the LPV controller.

frequency. This mode is the "rocking mode" of the fan as it rotates about the α_3 axis. The slower mode is the response of α_2. There are constant weights on α_1, $\dot{\alpha}_1$, and α_3; they are shown in dashed lines (α_1 and $\dot{\alpha}_1$ are identical). Constant weights are sufficient on α_1 since it has high gain in the low frequency. Since the α_2 channel does not have very much gain, an integral-like weight is used on it (solid line). Finally, we bundle all the uncertainty of the model into multiplicative uncertainty at the input. Due to the limitations of the model discussed in Section 3, using more precise uncertainty descriptions with this model are not profitable.

5 Experimental Results

The LPV controllers were implemented using Sparrow, as mentioned previously. The LPV package supports arbitrary scaling and offsets of the δ parameters, and arbitrary Δ size. Thus, LPV controllers can be designed for a variety of plant parameter ranges and model parameterizations without requiring any software modifications. Since an LPV controller is an LFT on a Δ of measured parameters, implementing it requires a real-time matrix inversion. To reduce computations, though, the LFT can be eliminated using past values of Δ. This approximation allows faster sampling rates. As long as the δ-values do not change much between samples, the approximation is very accurate. All experiments in this paper were run using this approximation.

The nominal offset forces, required to maintain equilibrium over all values of Δ, were computed *a priori* and stored in a lookup table. These forces are then fed directly into the plant, in addition to any control outputs.

The results for implementing the LPV controller based on the simplified pitch-velocity model are presented in Section 5.2. For comparison, we also present results obtained by

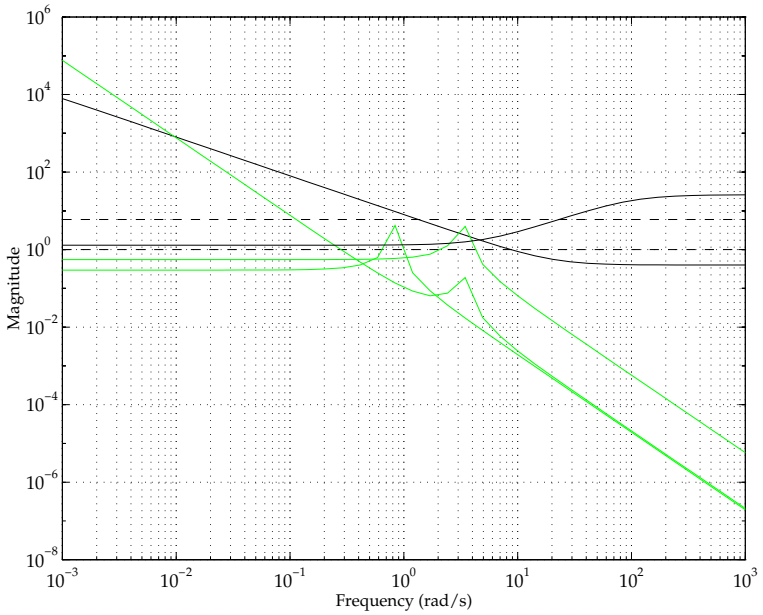

Figure 12: Bode plot of ducted fan model linearized about hover (shaded lines) with weights for LPV synthesis (solid and dashed lines)

locking the LPV controller at hover and in forward flight, as well as for a standard $\mathcal{H}_\infty$ controller designed previously.[10] The results shown are quite representative of the behavior of the ducted fan, but each one represents the "best" result of a series of three to five tests. The results are repeatable.

5.1 Trajectory Description

The controllers were tested on three trajectories. Two of the trajectories are simple and command changes on only one axis. The third trajectory is demanding and commands rapid changes to the α_1 and α_2 axes simultaneously.

The first trajectory is a one radian change on the α_1 axis over 5 seconds. The second is a 0.1 radian step change on α_2. While these trajectories are not challenging, they demonstrate the controllers' abilities to track each axis independently.

The third trajectory is more complex, and satisfies the equations of motion for the full nonlinear model.[6] It commands the fan to fly rapidly in the positive α_1 direction. During forward flight, the fan achieves rates larger than 1 radian per second, over five times greater than during the first trajectory. While in forward flight, two α_2 changes are commanded. The first is a smooth rise to 0.1 radians followed by a smooth drop back to 0 radians. The second is a smooth drop to -0.5 radians followed by a rise back to 0 radians.

While not immediately evident from the plots, the plant's non-minimum phase behavior is shown in the nominal complex trajectory. The nominal trajectory has a small initial undershoot. While one would not normally command a controller to follow this effect, it was not removed since it satisfies the equations of motion.

5.2 Results

The results upon implementing each of the controllers are presented in Figures 13 through 24, each consisting of 6 plots. Going from left to right, the first plot presents the α_1 trajectory (called "X" in the plot) in shaded lines with the system's response in solid lines. The second plot shows the same data, but the fan's position has vectors corresponding to α_3 (pitch) superimposed on it, to better illustrate how the fan moves. The third plot shows the α_2 trajectory (called "Y") with the system's response, and the fourth plot is the same as the third, with α_3 vectors superimposed. The fifth plot shows the α_3 variations. The final plot shows the commanded forces. The perpendicular force, u_1 (shaded), always has a DC component since experiments are conducted starting at hover.

The performance on the α_1 ramp for the $\mathcal{H}_\infty$ controller is shown in Figure 13. The controller performs rather slowly and with a steady state error. Notice though, that the deviation from 0 on α_2 is almost non-existent. This controller was designed for a model at hover and the decoupling is evident in the commanded forces. The performance on the step in α_2 is shown in Figure 14 and is similar, with a slow response on α_2. The decoupling is not as effective here, and there is some ripple on α_1 as the fan rocks. The response to the complex trajectory is shown in Figure 15. The controller is reasonably good at tracking the

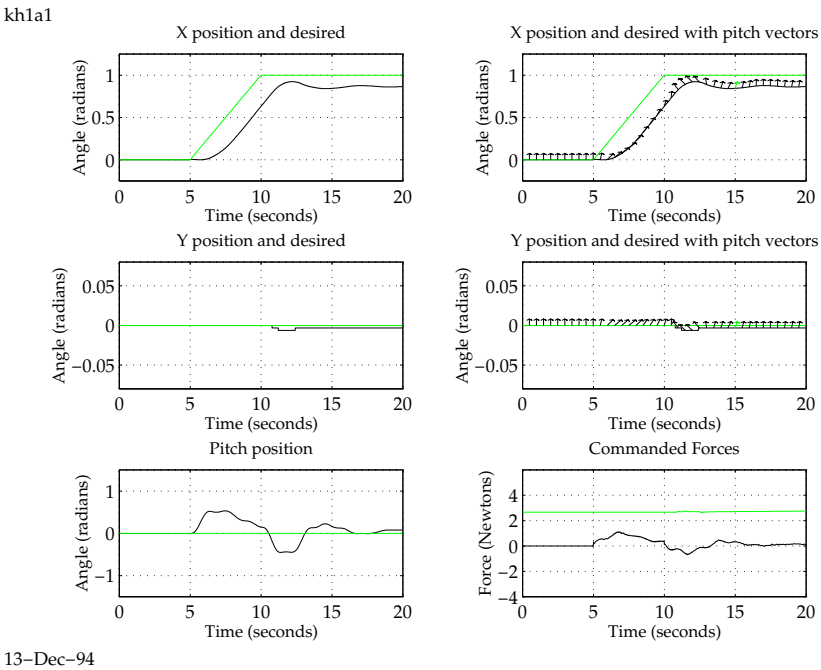

Figure 13: Closed loop response of the H_∞ controller for an α_1 ramp.

α_1 reference, although again, it is slow. However, it cannot follow the changes in α_2 very well at all, particularly the downward motion.

The performance of the LPV controller locked at hover is shown next, in Figures 16, 17, and 18. This controller is faster than the $\mathcal{H}_\infty$ controller, and performs very well in tracking α_1. The controller does not hold its position on α_2 as well as the $\mathcal{H}_\infty$ controller. The response to the step in α_2 is also faster, but the controller does not hold its position in α_1 as well. On the complex trajectory, the controller tracks α_1 better than the $\mathcal{H}_\infty$ controller but there are much wilder oscillations in pitch and on α_2 than in the previous case. The oscillations in the commanded forces are particularly alarming.

The performance of the LPV controller locked for forward flight is shown in Figures 19, 20, and 21. Overall, the performance is very similar to the LPV controller locked at hover. Noticeable differences are that this controller is more oscillatory on the α_1 ramp, while being less oscillatory, and performing much better, on the complex trajectory. Note, the non-minimum phase nature of the plant is evident here in the downward dip of the response in Figure 19.

The responses of the full LPV controller are shown in Figures 22, 23, and 24. The performance is different from either of the locked controllers. The LPV controller is much faster at tracking the ramp in α_1, at the expense of overshoot and ringing. Its maximum deviation from 0 on α_2 is less than the maximum for either of the locked controllers, but it doesn't stay as close to the reference. The response of the LPV controller to the step in

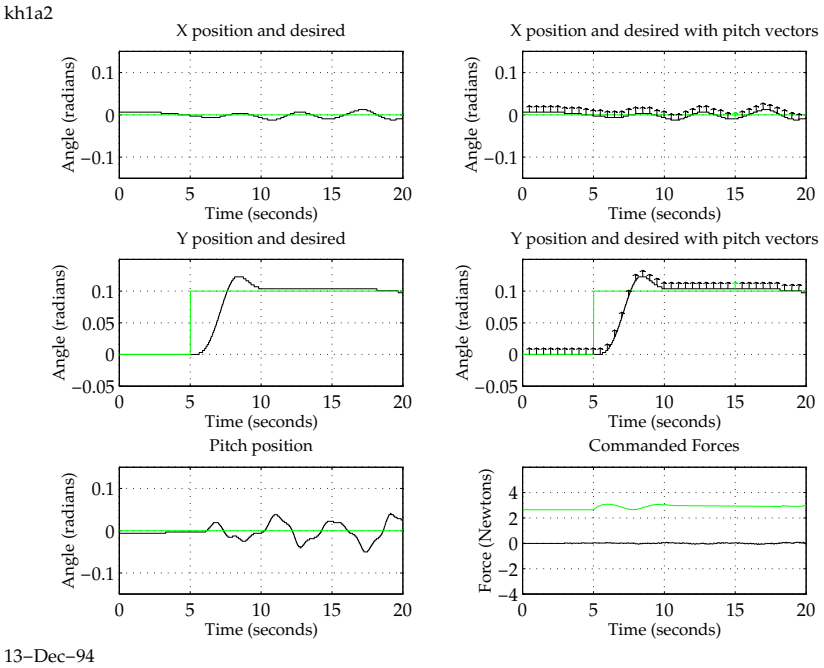

Figure 14: Closed loop response of the H_∞ controller for an α_2 step.

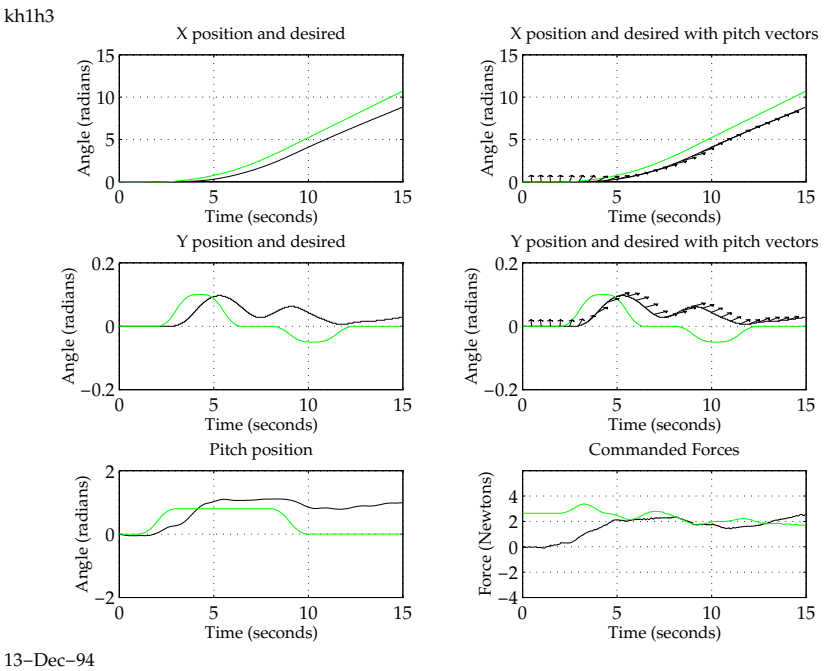

Figure 15: Closed loop response of the H_∞ controller for the complex trajectory.

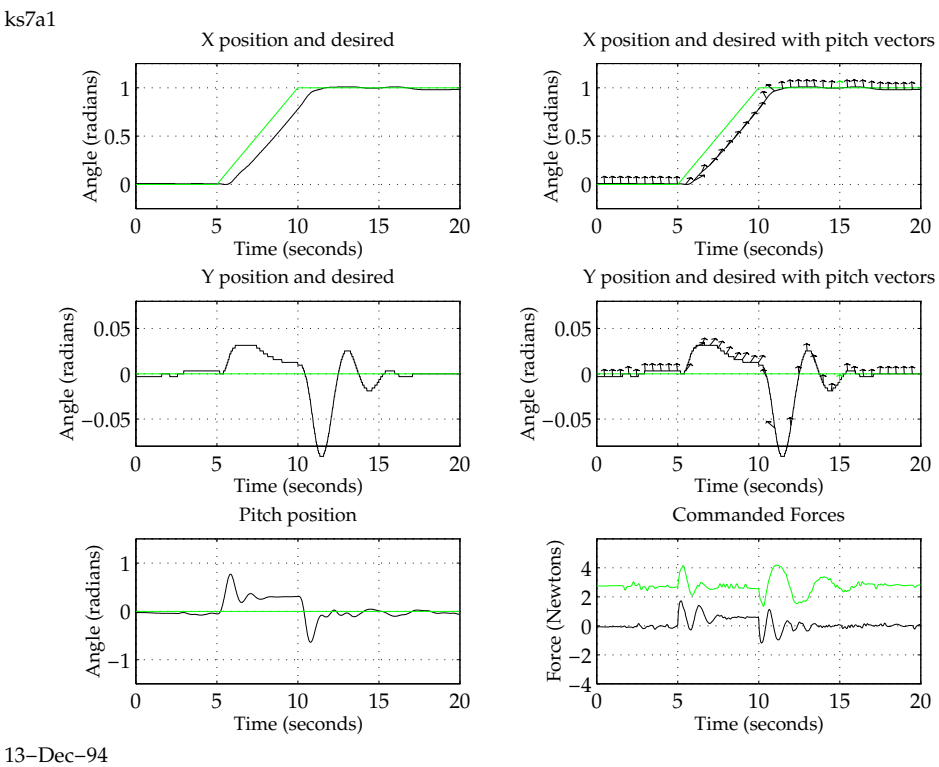

Figure 16: Closed loop response of the LPV controller locked at hover for an α_1 ramp.

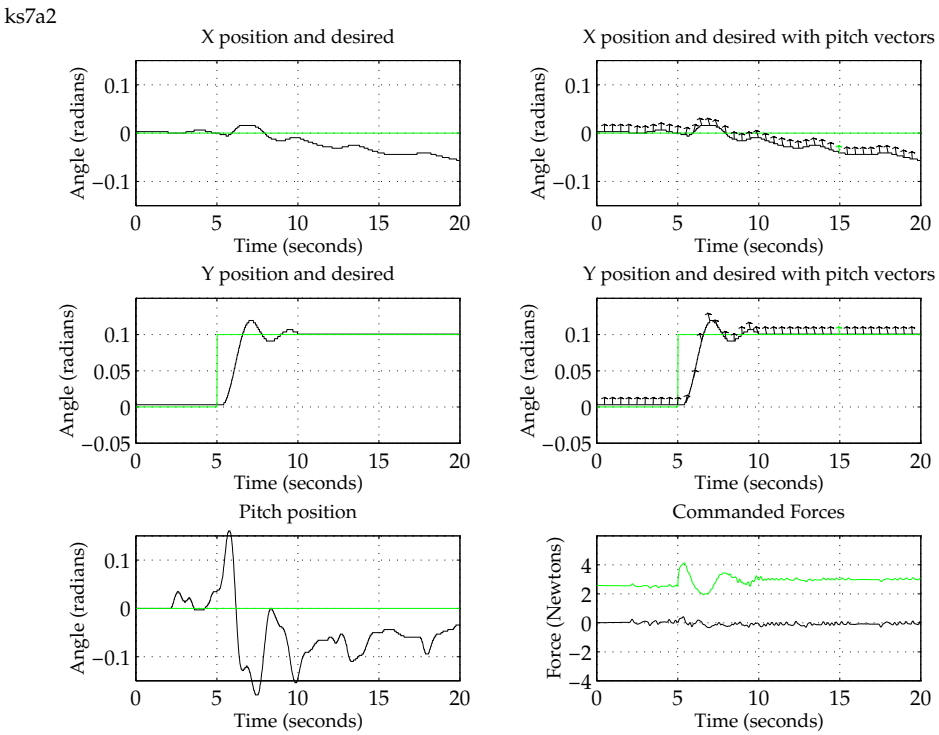

Figure 17: Closed loop response of the LPV controller locked at hover for an α_2 step.

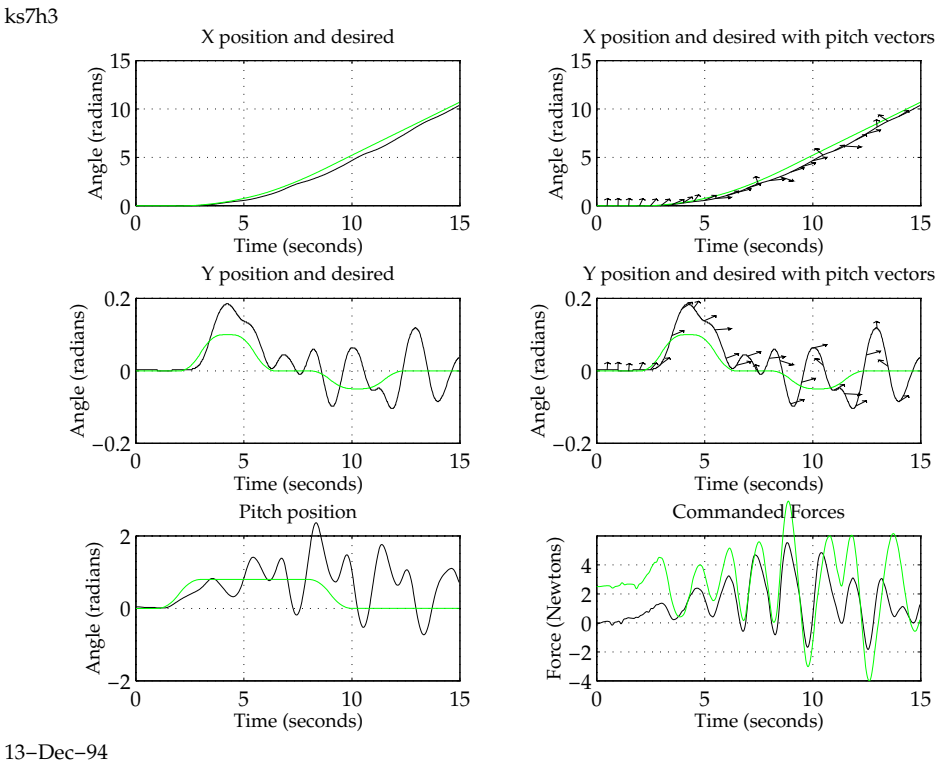

Figure 18: Closed loop response of the LPV controller locked at hover for the complex trajectory.

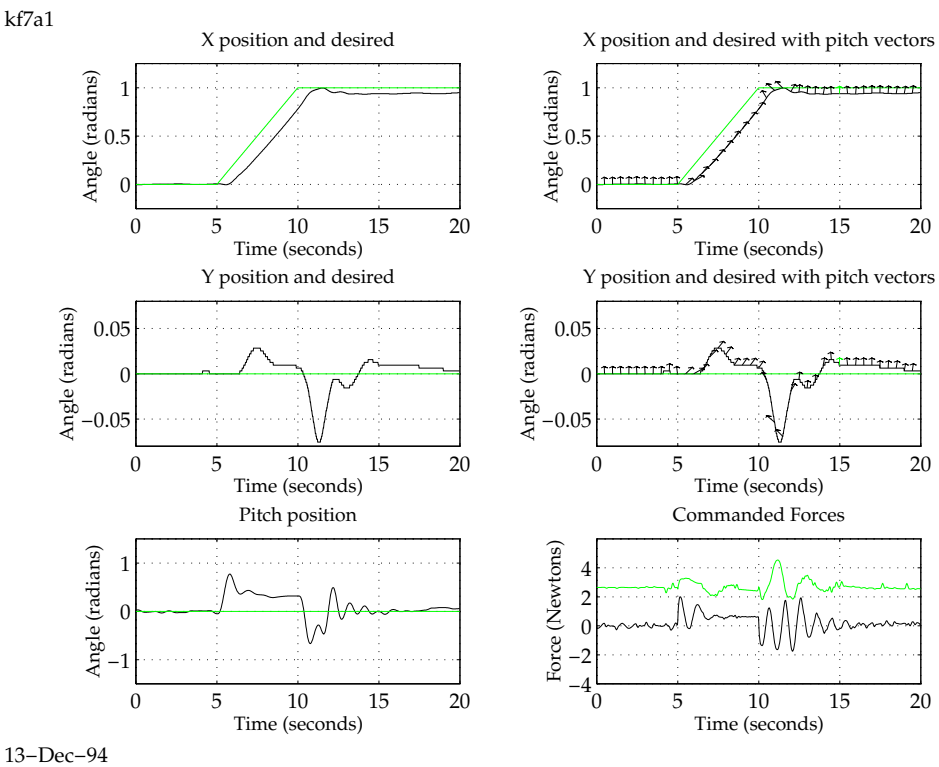

Figure 19: Closed loop response of the LPV controller locked in forward flight for an α_1 ramp.

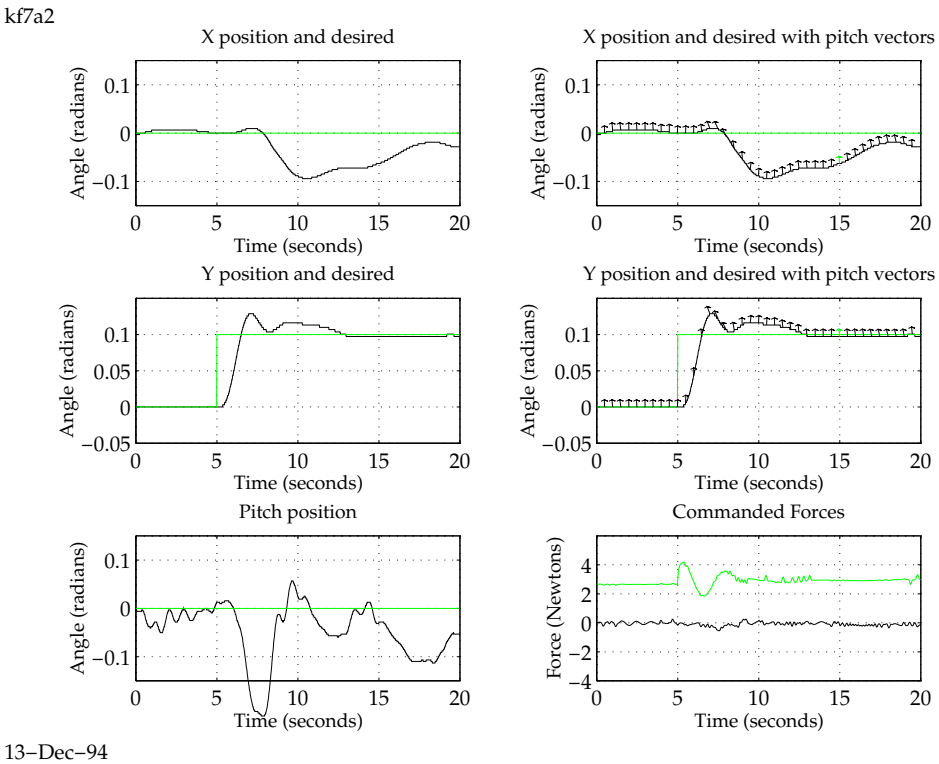

Figure 20: Closed loop response of the LPV controller locked in forward flight for an α_2 step.

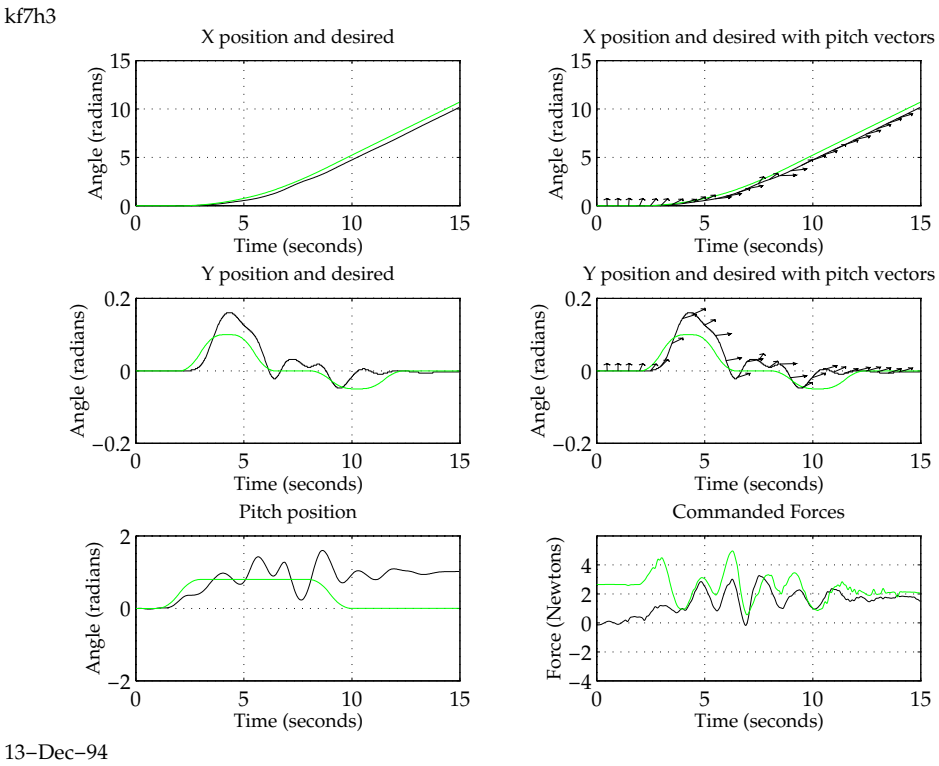

Figure 21: Closed loop response of the LPV controller locked in forward flight for the complex trajectory.

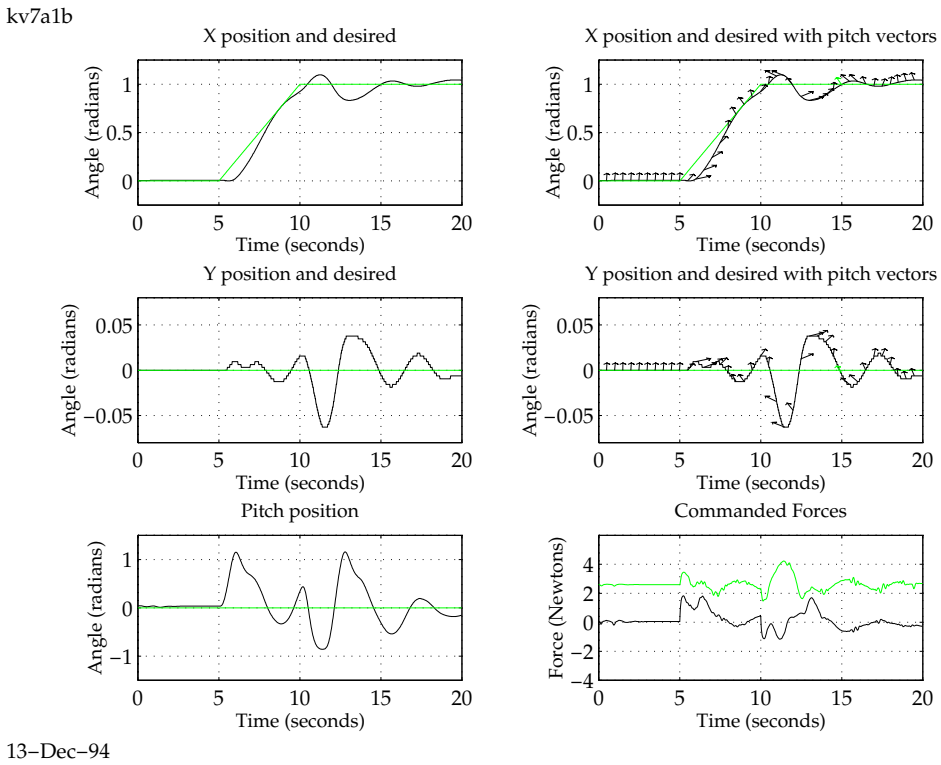

Figure 22: Closed loop response of the LPV controller for an α_1 ramp.

α_2 is not as good as either of the locked controllers. The response rings more, and there is an unexplained steady state error in α_1 position. We conjecture this is caused by an error in the force-torque table, and are currently investigating it. The LPV controller performs noticeably better than all previous controllers on the complex trajectory. It is able to track the α_1 reference more closely, and provides excellent tracking on α_2. Moreover, there is much less oscillation in the commanded forces for the LPV controller.

Finally, the controllers are quantitatively compared based on several figures of merit, which have been used for other controller comparisons on the ducted fan.[10] Table 1 shows the figures of merit for the ramp in α_1, Table 2 shows the results for the step in α_2, and Table 3 shows the results for the complex trajectory. The LPV controller locked at hover is called "hover" in the tables and when locked in forward flight it is called "forward."

The 10-90% rise time shown is a standard figure of merit for step responses. For the ramp in α_1 it provides a measure how closely the ramp follows the signal. The 90% delay factor is computed by measuring the difference, in seconds, between when the trajectory reached 90% of its final value, and when the system reached this same value. Steady state error is computed by averaging the absolute value of the error over the last four seconds of the trajectory. The remaining figures of merit (% overshoot, maximum drift in the constant channel, maximum pitch angle, and RMS errors in the α_1 and α_2 channels) are self-explanatory. Note that the

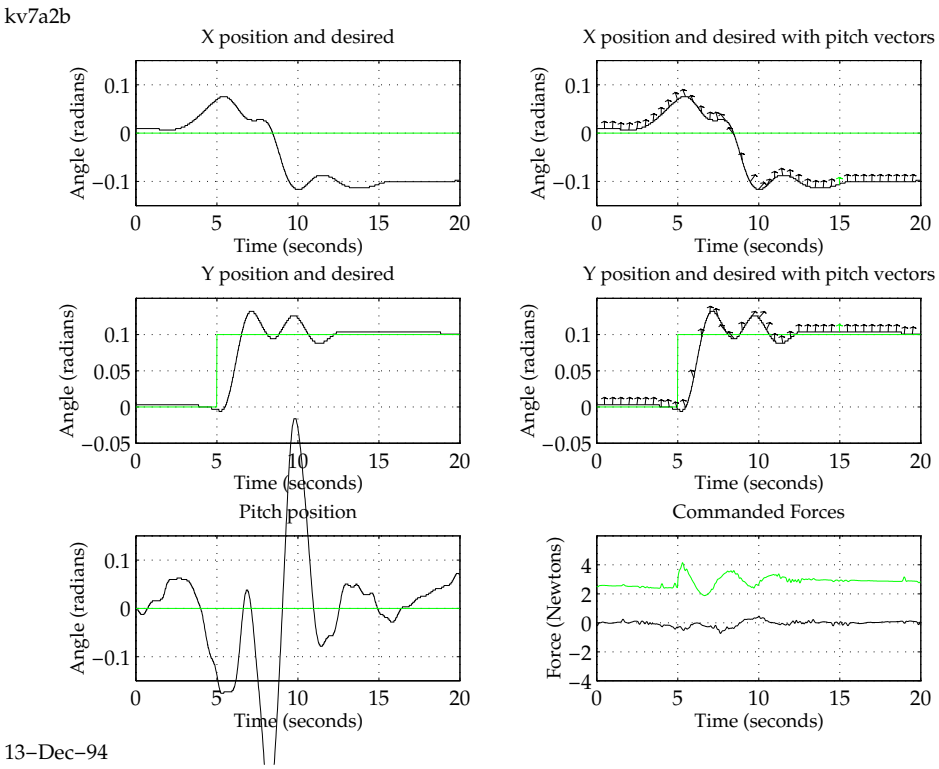

Figure 23: Closed loop response of the LPV controller for an α_2 step.

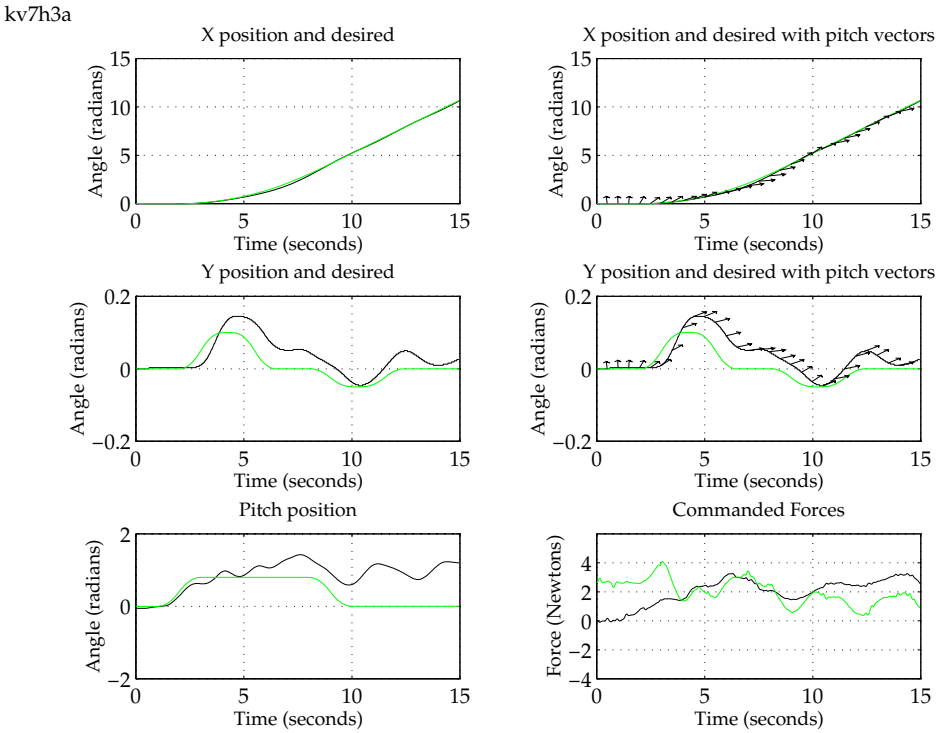

Figure 24: Closed loop response of the LPV controller for the complex trajectory.

Criteria	Controllers			
	$\mathcal{H}_\infty$	Hover	Forward	LPV
10-90% rise time (s)	4.36	4.20	4.10	3.34
Delay to 90% (s)	2.08	1.04	1.00	0.30
% Overshoot	0.00	0.95	0.00	9.75
Steady-state error (%)	13.12	1.41	5.61	2.20
Maximum drift in α_2 (rad)	0.0063	0.0911	0.0754	0.0628
Maximum α_3 (rad)	0.5341	0.7665	0.7700	1.1561
RMS α_1 error (rad)	0.1900	0.0995	0.1076	0.0744
RMS α_2 error (rad)	0.0025	0.0232	0.0169	0.0183

Table 1: Comparison of Controllers for a ramp in α_1.

maximum pitch figure gives an indication of the gain of the controller.

The figures of merit are generally consistent with the qualitative observations done earlier. The LPV controller has the fastest response on all trajectories, and also the most overshoot. Except for the $\mathcal{H}_\infty$ controller on the ramp in α_1, all controllers have small steady-state error. The $\mathcal{H}_\infty$ controller is much better at maintaining a constant reference on one axis while the other axis is changed. An interesting observation from Table 3 is that the pitch angle exceeds $\pi/2$ for both locked controllers, although not by much for the controller locked forward. A linear controller cannot maintain pitch angles greater than $\pi/2$ and still have good performance, since a sign change in the force direction occurs there.

The figures of merit do not measure many important properties of controllers; nonetheless they provide some basis for comparison. From them it seems clear that the LPV controller is overall the best controller designed.

6 Conclusions

In this paper we have applied an LMI based control technique to a practical application, a ducted fan engine. An LPV controller was synthesized which successfully controlled the ducted fan with adequate performance on all test trajectories and excellent performance on one particularly difficult trajectory. The LPV controller is the best controller yet designed for the complex trajectory. An advantage of LPV designs over conventional methods of gain-scheduling is they design a controller of fixed order that works reasonably well over the operating regime of the plant.

Experimentally, our goals are oriented toward further study of nonlinear robust control using this fan or a successor. Currently, a new ducted fan with a more aerodynamic shape is being designed and modelled using a wind tunnel at Caltech. This fan will be much more powerful and maneuverable than the rather heavy model used currently. Also, there are plans

Criteria	Controllers			
	$\mathcal{H}_\infty$	Hover	Forward	LPV
10-90% rise time (s)	1.32	0.84	0.76	0.72
Delay to 90% (s)	2.44	1.44	1.40	1.42
% Overshoot	22.52	19.38	28.81	31.95
Steady-state error (%)	2.4	0.0	2.4	2.7
Maximum drift in α_1 (rad)	0.0126	0.0565	0.0942	0.1100
RMS α_1 error (rad)	0.0065	0.0272	0.0472	0.0798
RMS α_2 error (rad)	0.0281	0.0206	0.0222	0.0235

Table 2: Comparison of Controllers for a step in α_2.

Criteria	Controllers			
	$\mathcal{H}_\infty$	Hover	Forward	LPV
Difference in position at 15s in α_1 (rad)	1.8618	0.3099	0.5282	0.0463
Maximum α_3 (rad)	1.1090	2.3656	1.5991	1.4231
RMS α_1 error (rad)	0.9923	0.3784	0.3805	0.0907
RMS α_2 error (rad)	0.0473	0.0517	0.0256	0.0376
Maximum α_2 error (rad)	0.1017	0.1194	0.0602	0.0851

Table 3: Comparison of Controllers for the complex trajectory.

to add a wing to the ducted fan, so that aerodynamic effects become more significant.

Strictly from the point of view of applying the LPV techniques to the ducted fan, there are many more designs and experiments to do. Theoretically, we are investigating ways to improve the performance of the LPV synthesis. We would like to extend the parameter range in our designs, with the idea of eventually testing trajectories similar to Herbst maneuvers in aircraft. The robustness properties of the LPV controllers could be further investigated, and we would like to design controllers using more of the parameter variations than the reasonably simple model used in this paper.

Since we have explored only full LPV synthesis in this paper, we have not exploited any special properties of the ducted fan. In particular, the state is available for measurement, and the parameter variations are all states. When this is the case, there is an equivalence between the general synthesis problem and a full-information version of it. Solving the full information problem would result in a nonlinear constant-gain controller with the same performance level as the full LPV controller. Further investigation of this is currently in progress.

Acknowledgements

The authors have enormously benefited from the guidance and advice of Richard Murray, whom we thank here. Carolyn Beck helped with many useful comments. Gary Fay constructed the force lookup tables used in the software, and has assisted throughout in hardware maintenance. The second author gratefully acknowledges financial support provided by Ministère de la Recherche et de l'Espace (MRE), France. We further thank the authors of the *LMI Control Toolbox*[7] for providing an early release which was used to solve the synthesis LMIs.

Appendix: Other Parameter Variations

In this appendix, the remaining parameter variations of the model of Equation 1 are shown in Figures 25 through 29.

References

[1] Apkarian, P., and P. Gahinet, 'A Convex Characterization of Parameter Dependent $\mathcal{H}_\infty$ Controllers', to appear in *IEEE Trans. on Automatic Control*, (1995).

[2] Apkarian, P., P. Gahinet, and J. Biannic, 'Self-Scheduled $\mathcal{H}_\infty$ Control of a Missile via LMIs', *Proc. 33rd Conf. on Decision and Control*, 3312–3317, Florida (1994).

[3] Apkarian, P., P. Gahinet, and G. Becker, '$\mathcal{H}_\infty$ Control of Linear Parameter-Varying Systems: A Design Example', submitted to *Automatica*.

[4] Balas, G. J., and A. Packard, 'Design of Robust, Time-Varying Controllers for Missile Autopilots,' *Proc. IEEE Conf. on Control Appl.*, Dayton, Ohio (1992).

[5] Bodenheimer, B., and P. Bendotti, 'Optimal Linear Parameter-Varying Control Design for a Pressurized Water Reactor', California Institute of Technology Technical Report, CDS 95-001, (1995).

[6] Choi, H., P. Sturdza, and R. M. Murray, 'Design and Construction of a Small Ducted Fan Engine for Nonlinear Control Experiments,' *Proc. 1994 American Control Conf.*, 2618–2622, Baltimore (1994).

[7] Gahinet, P., A. Nemirovskii, A. J. Laub, and M. Chilali, 'The LMI Control Toolbox,' *Proc. 33rd Conf. on Decision and Control*, 2038–2041, Florida (1994).

[8] Gal-Or, B., 'Fundamental Concepts of Vectored Propulsion,' *Journal of Propulsion*, **6**(6), 747–757 (1990).

[9] Gal-Or, B., *Vectored Propulsion, Supermaneuverability, and Robot Aircraft*, Springer-Verlag, New York (1990).

[10] Kantner, M., B. Bodenheimer, P. Bendotti, and R. M. Murray, 'An Experimental Comparison of Controllers for a Vectored Thrust, Ducted Fan Engine,' submitted to *American Control Conf.* (1995).

[11] Lemon, C., and J. Hauser, 'Design and Initial Flight Test of the Champagne Flyer', *Proc. 33rd IEEE Conf. on Decision and Control*, 3852–3853, Florida (1994).

[12] Murray, R. M., and E. L. Wemhoff, *Sparrow 2.0 Reference Manual*, California Institute of Technology (1994). Available electronically from `http://avalon.caltech.edu/murray/sparrow.html`.

[13] Packard, A., 'Gain Scheduling via Linear Fractional Transformations,' *System and Control Letters*, **22**, 79–92 (1994).

[14] Packard, A., and J. Doyle, 'The Complex Structured Singular Value', *Automatica*, **29**(1), 71–109 (1993).

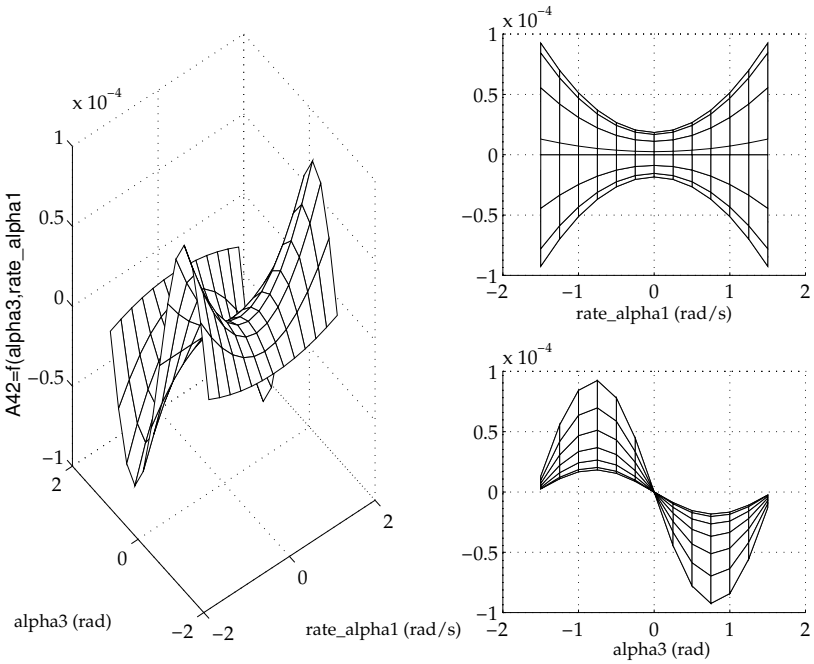

Figure 25: Dependence of a_{42} upon α_3 and $\dot{\alpha}_1$.

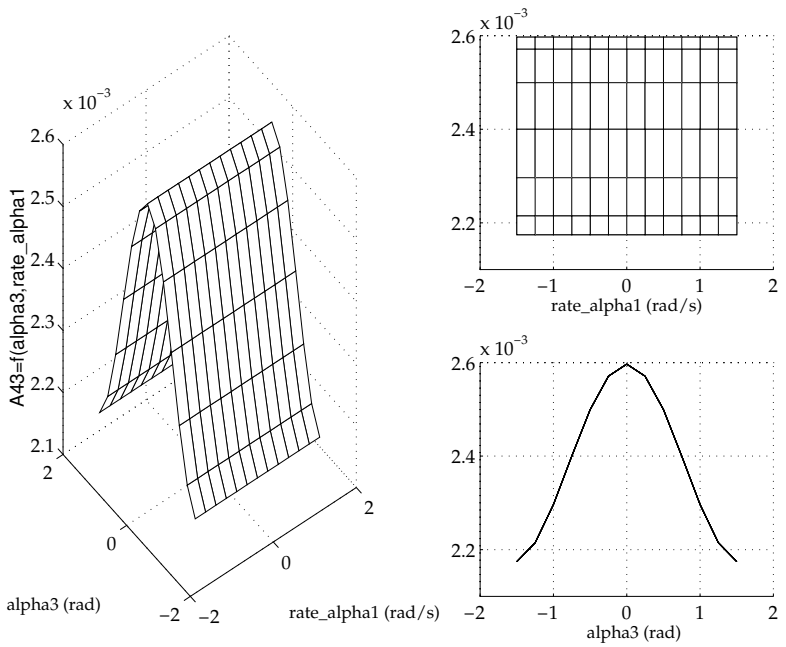

Figure 26: Dependence of a_{43} upon α_3 and $\dot{\alpha}_1$.

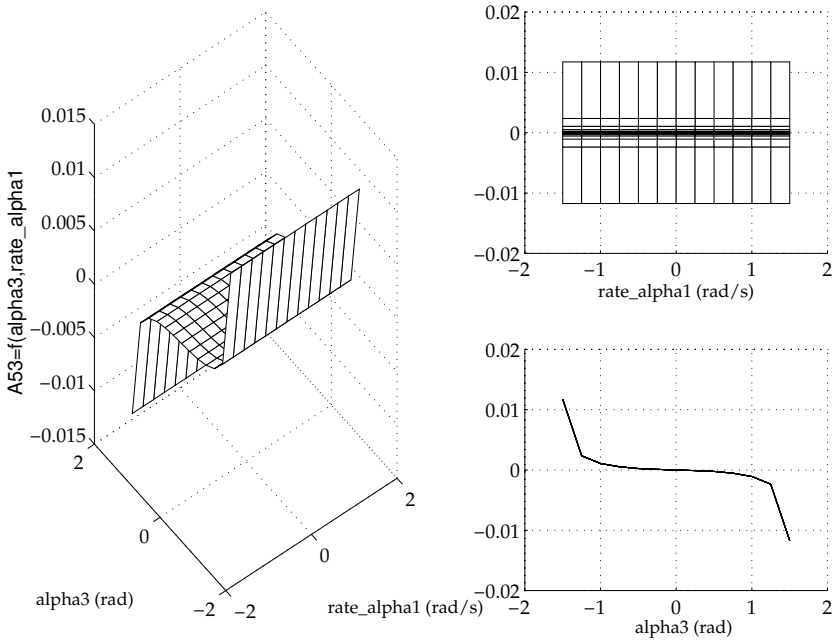

Figure 27: Dependence of a_{53} upon α_3 and $\dot{\alpha}_1$.

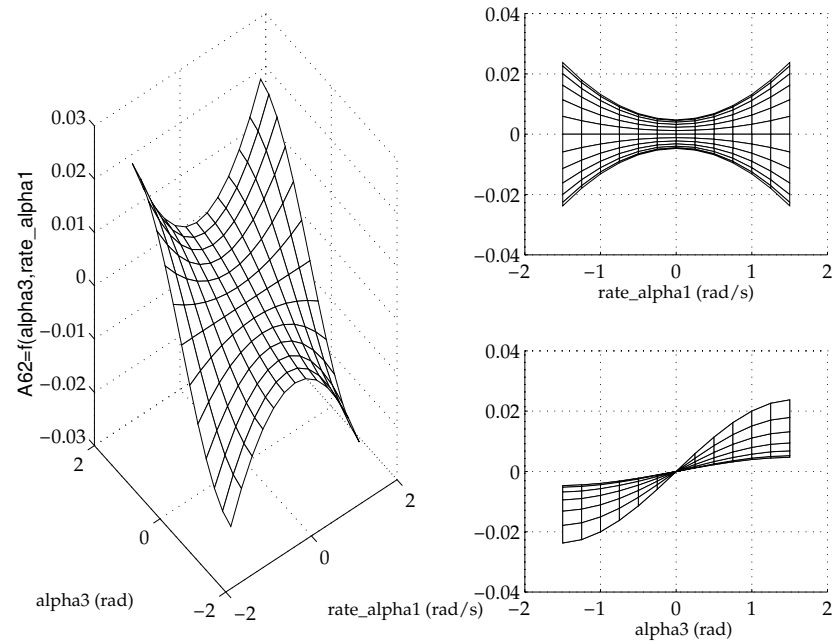

Figure 28: Dependence of a_{62} upon α_3 and $\dot{\alpha}_1$.

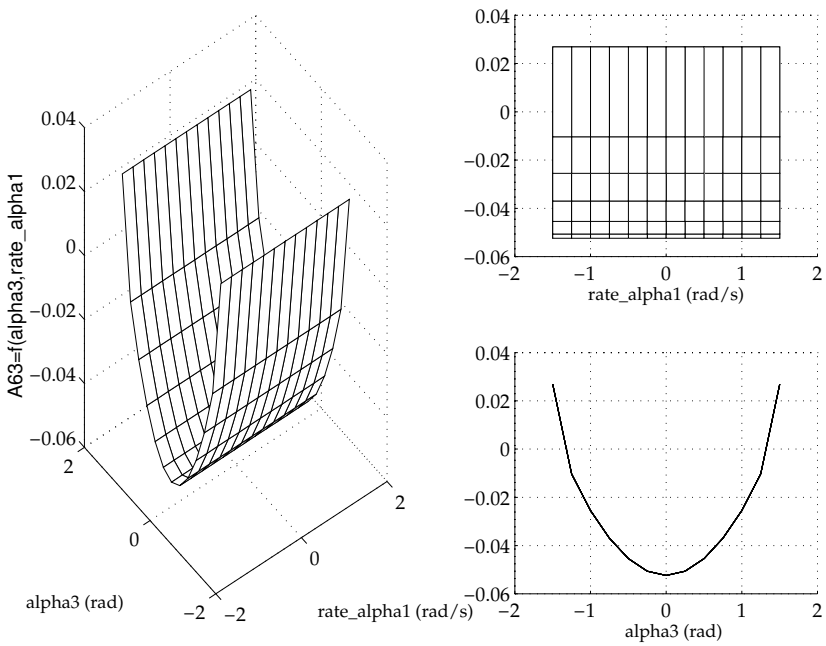

Figure 29: Dependence of a_{63} upon α_3 and $\dot{\alpha}_1$.