An Experimental Comparison of Controllers for a Vectored Thrust, Ducted Fan Engine

Michael Kantner  Bobby Bodenheimer
Pascale Bendotti  Richard M. Murray

Division of Engineering and Applied Science
California Institute of Technology
Pasadena, CA 91125

Abstract
Experimental comparisons between four different control design methodologies are applied to a small vectored thrust engine. Each controller is applied to three trajectories of varying aggressiveness. The control strategies considered are LQR, \( H_\infty \), gain scheduling, and feedback linearization. The experiments show that gain scheduling is essential to achieving good performance. The strengths and weaknesses of each methodology are also examined.

1 Introduction
This paper is concerned with the application of linear and nonlinear control design methods to high-performance jet aircraft performing aggressive maneuvers. Very few design methods are available for building robust, nonlinear control laws for this class of systems. We are currently concentrating on the control of a ducted fan engine. By focusing on a specific nonlinear system, we hope to generate new ideas and techniques for applying the tools of robust linear control theory to systems with strongly nonlinear behavior.

There is a large literature on vectored propulsion systems and they are gaining in popularity 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 [5] (see also the survey article [4]). 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 improving 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 [6].

A picture of our experimental system is shown in Figure 1. The ducted fan consists of a high-efficiency electric motor with a 6-inch diameter propeller, 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. It is mounted on a three degree of freedom stand which allows horizontal and vertical translation as well as unrestricted pitch angle. A detailed description of the fan, including models for the thrust as a function of flap angle and motor current is given in [3].

We present experimental comparisons between several different methodologies which are used to design controllers: traditional operating point linearization, feedback linearization, and linearization about multiple operating points. These experimental comparisons comprise the first step towards our eventual goal of synthesizing robust nonlinear controllers for high-performance operation over the full flight envelope. While our experimental system is considerably simpler than a real vectored thrust aircraft, it has many of the characteristics of existing systems—strongly nonlinear dynamics, limited actuator bandwidth, and input saturation, to name a few.

One of our main goals in presenting these comparisons is to indicate how well currently available design tools work on a specific physical system and to gain insight into the types of problems which limit the applicability of those tools. In this sense, the ducted fan experiment provides a testbed for nonlinear controller designs which can be used to evaluate and verify both different underlying control theories and the different underlying control software. We hope our experience will help motivate future development of theoretical and software tools which can be used for designing real-time controllers for physical systems.

This paper is organized as follows: In Section 2 we introduce the experimental setup and briefly describe the nonlinear model. We describe the implementation of the controllers in Section 3 and give a detailed evaluation of their performance on a set of three different trajectories in Section 4. Section 5 summarizes our main conclusions and indicates a number of directions for future work. More details can be obtained electronically at http://avalon.caltech.edu/~dfan.

2 Experimental Setup

2.1 Hardware
The design philosophy was to have a relatively simple ducted fan which could provide two dimensional vectored and reverse thrust. The ducted fan is mounted on a rotating arm, as shown in Figure 2, which limits the ducted fan's 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. An optical encoder with an angular resolution of \( \pi/1000 \) radians is mounted on each axis. The system is described in detail in [3].

The experiment is interfaced to an 80486 computer running an MS-DOS-based real-time kernel called Sparrow [7]. Custom hardware is used to read in joint angles via encoders and generate PWM signals for the R/C servos. A voltage-to-current amplifier powers the fan engine. Joint angles are read in at 200 Hz and their rates are estimated using a four state filter. The PWM signals are output at 50 Hz, which is the standard update rate for R/C servos.

Controllers are designed and simulated using Matlab on Sun workstations. Sparrow loads linear controllers directly from Matlab data files. Gain scheduled controllers, which are im-
implemented as a set of linear controllers, are also loaded from Matlab data files. Nonlinear controllers implemented as Matlab functions require a small amount of revision before they are linked to Sparrow.

### 2.2 Modeling

The dynamics of the system are modeled using standard rigid body mechanics with the angles, \( \alpha \), and their velocities, \( \dot{\alpha} \), as the states. The equations of motion for the system have the functional form

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

where \( M(\alpha) \) is the generalized inertia matrix, \( C(\alpha, \dot{\alpha}) \) is the Coriolis matrix, \( N(\alpha, \dot{\alpha}) \) is the vector of gravity term and drag forces, and \( T(\alpha, F) \) is the vector of applied joint torques [3].

A simplified model for the system which accounts for many of the important nonlinearities is given by

\[
\begin{align*}
(\eta - \delta s^2)\ddot{\alpha}_1 &= 2\delta s_1 c_2 \alpha_1 \dot{\alpha}_2 + f_1 \dot{\alpha}_1 - f_2 \dot{\alpha}_1 \\
\beta \dot{\alpha}_2 &= -\beta s_2 c_1 - \delta s_2 c_1 \alpha_1^2 + f_1 \dot{\alpha}_1 + f_2 \dot{\alpha}_1 \\
J \dot{\alpha}_3 &= -m g s_3 + r f_1
\end{align*}
\]

where \( \eta, \beta, \delta \) are moments of inertia for the stand plus fan, \( \gamma \) is the effective gravitational constant (taking the counterweight into account), \( m \) is the mass of the fan, \( J \) the inertia of the fan about the \( \alpha_3 \) axis, \( r \) is the distance from the flap assembly to the \( \alpha_3 \) axis, and \( l \) is the offset of the center of mass from the \( \alpha_3 \) axis. The terms \( \dot{\alpha}_1 \) and \( \dot{\alpha}_2 \) are shorthand for \( \cos \alpha_1 \) and \( \sin \alpha_1 \), respectively. The forces \( f_1 \) and \( f_2 \) are the forces perpendicular and parallel to the axis of the fan. Specific numerical values for the parameters in this model can be derived from the values given in [3].

Identification experiments have shown that this model is correct near hover and successful control designs also support this conclusion. For the control designs in this paper, we ignore all actuator dynamics, sensor limitations, friction, and aerodynamic effects. Static friction about the \( \alpha_1 \) axis is significant. Aerodynamic effects have been observed in the lab during forward flight, but are of little importance during hover and at slower speeds. The model also omits the gyroscopic terms that result from the angular momentum of the propeller. This term can be significant at moderate pitch (\( \alpha_1 \)) and high \( \dot{\alpha}_1 \).

There are several additional sources of known uncertainty in this model. A key simplifying assumption is 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 by as much as 20%. Furthermore, the forces generated by the flaps are not directly commanded. Instead, an experimentally determined lookup table maps desired forces to motor current and flap angle. This table is computed during fan calibration and assumed to be constant, even though the actual mapping between motor current and flap angle varies as a function of operating conditions.

The strongly nonlinear features of the model include the effect of the pitch \( \alpha_1 \) on the input forces and also the centrifugal forces (\( \alpha_1^2 \) terms in \( \dot{\alpha}_1 \) equation). In particular, the latter forces can be quite high when the fan is flying quickly, and they tend to push the altitude, \( \dot{\alpha}_3 \), towards zero.

### 3 Control Schemes

Four different control schemes were examined, ranging from static state feedback to gain scheduled control. Initial controller designs were performed on the model described in [3]. While the controllers' performances are a function not only of the method, but also the designers' skills, meaningful comparisons can still be made.

Two of the control designs are based solely upon the plant's linearization about hover.

\[
\begin{bmatrix}
\alpha_1(s) \\
\alpha_2(s) \\
\alpha_3(s)
\end{bmatrix} =
\begin{bmatrix}
-0.2102s^2 + 0.8314 & 0 & 0 \\
0 & \frac{0.0416}{s^2 + 0.5637} & 0 \\
-5.7095 & 0 & 2.4 \times 10^{-5}
\end{bmatrix}
\begin{bmatrix}
f_1 \\
f_2
\end{bmatrix}
\]

At hover, the \( \alpha_1 \) and \( \alpha_2 \) axes are decoupled and the \( \alpha_1(s)/f_1(s) \) transfer function is non-minimum phase. However, when \( \alpha_3 \neq 0 \), the system is not decoupled.

The control objectives were to track both \( \alpha_1 \) and \( \alpha_2 \) trajectories. As minimal requirements on the controllers, the closed loop transfer function from \( \alpha_1 \) reference to \( \alpha_1 \) error must be below -20 dB at frequencies below 0.05 radians/second and the closed loop transfer function from \( \alpha_2 \) reference to \( \alpha_2 \) error must be below -20 dB at frequencies below 0.1 radians/second.

#### 3.1 Linear Quadratic Regulator

The linear quadratic regulator (LQR) design is performed on the error system of plant linearized about hover. The control authority provided by the amplifier and servos leads to unity input weighting. State weightings are chosen to allow moderate \( \alpha_2 \) swings while zeroing the \( \alpha_1 \) error because a large penalty on \( \alpha_3 \) drastically slows the \( \alpha_1 \) response. Some penalty is placed on \( \alpha_3 \) since large pitch swings lead to large changes in the plant's dynamics. A large weighting is also placed on the \( \alpha_3 \) error. The state derivatives have zero weights.

This initial controller gives unacceptable performance. The system naturally has little gain from \( f_2 \) to \( \alpha_3 \), as can be seen in (2). To meet the performance requirement, a constant gain of 30 is needed. Due to unmodeled dynamics, a gain this large drives the system unstable.

To achieve the necessary low frequency gain, an integrator is added to the \( \alpha_3 \) axis and a second LQR design is performed on the augmented plant. The weights are chosen using similar guidelines as before, and the \( \alpha_3 \) weight is chosen to be seven times greater than the weight on its integral.

#### 3.2 H∞ Control

\( H_\infty \) synthesis is a disturbance rejection methodology. To track trajectories, the performance goal is to reject low frequency components of the error signal. The tracking becomes faster as higher frequencies are rejected.

As compared with the LQR design, the \( H_\infty \) structure is almost identical. A constant but large performance weight is placed on \( \alpha_1 \). Integral action is needed on \( \alpha_2 \) since the natural gain of the channel is not large enough to track satisfactorily. Finally, a weight on input uncertainty is added to ensure controller roll off at high frequencies. The performance weight on \( \alpha_2 \) seeks to make the dynamics of the response of \( \alpha_2 \) much faster than the dynamics on \( \alpha_1 \). This was done in an effort to reduce the \( \alpha_2 \) error induced when the fan is tilted at an angle. Sensitive noise is explicitly considered in all measurements. The resulting controller has nine states.
3.3 Gain Scheduled Control

Since linear design techniques are well developed, one common practice is to linearize a plant about several different operating points, design a controller for each operating point, and switch between them. From (1), it is clear that the ducted fan's linearization is sensitive to changes in \( \alpha \).

Three operating points potentially encountered during the trajectories of interest were chosen. They correspond to hover, acceleration, and deceleration for flight in the positive \( \alpha \) direction. At each operating point, an LQR controller was designed. The hover controller is identical to the previously described LQR controller.

Since the controllers have no states, switching transients due to improper initial conditions are avoided. Many types of switching were examined, including switching based upon nominal state, switching based upon actual state, and blending of control outputs (smooth switching).

3.4 Feedback Linearization Based Control

It is tedious but straightforward to check that the dynamics for the ducted fan are not full state feedback linearizable using static state feedback. However, the system is LQR linearizable with respect to any pair of outputs (\( \alpha_1, \alpha_2 \)). It is thus possible to feedback linearize the dynamics and decouple the control of the pitch (\( \alpha_1 \)) and altitude (\( \alpha_2 \)) axes. In this case, the zero dynamics for the fan are the \( \alpha_1 \) dynamics.

Using the same notation as the simplified model (1), applying the feedback linearizing law

\[
\begin{bmatrix}
\dot{f}_2 \\
\dot{f}_1
\end{bmatrix} = \begin{bmatrix}
\frac{\partial \psi}{\partial \alpha_2} \\
\frac{\partial \psi}{\partial \alpha_1}
\end{bmatrix} + \begin{bmatrix}
\frac{\partial \psi}{\partial \alpha_2} \\
\frac{\partial \psi}{\partial \alpha_1}
\end{bmatrix} \begin{bmatrix}
\alpha_1 \\
\alpha_2
\end{bmatrix} - \frac{m g l c_2}{r} \begin{bmatrix}
\tan \alpha_2 \\
\tan \alpha_1
\end{bmatrix} \begin{bmatrix}
\nu_1 \\
\nu_2
\end{bmatrix},
\]

(3)

where \( \psi = \beta \gamma c_2 \delta \tan \alpha_1 \), \( \delta = \frac{m g l c_2}{r} \), yields the output response

\[
\begin{bmatrix}
\dot{\alpha}_1 \\
\dot{\alpha}_2
\end{bmatrix} = \begin{bmatrix}
\frac{\tan \alpha_2}{n - \beta} \\
\frac{\tan \alpha_1}{n - \beta}
\end{bmatrix} \begin{bmatrix}
\nu_1 \\
\nu_2
\end{bmatrix} + \begin{bmatrix}
\frac{\tan \alpha_2}{n - \beta} \\
\frac{\tan \alpha_1}{n - \beta}
\end{bmatrix} \begin{bmatrix}
\nu_1 \\
\nu_2
\end{bmatrix} - \frac{m g l c_2}{r} \begin{bmatrix}
\tan \alpha_2 \\
\tan \alpha_1
\end{bmatrix} \begin{bmatrix}
\nu_1 \\
\nu_2
\end{bmatrix},
\]

(4)

where \( \gamma = 2 \delta c_2 \alpha_1 \alpha_2 \tan \alpha_1 + \frac{m g l c_2}{r} \tan \alpha_2 - \psi \).

Standard I/O linearization based control design uses pole placement to stabilize the system. This design method was not successful for the ducted fan since I/O linearization is essentially a plant inversion and due to unmodeled dynamics, the inversion is not exact. The system behaves as two poles near the origin whose position changes as the plant's state varies. To stabilize each axis, a second order lead filter was used to add phase margin at the desired crossover frequency.

The \( \alpha_1 \) control loop is designed around the previous closed loop system using a separation of time scales argument and some simplifying assumptions. First, it is assumed that \( \alpha_1 \) can be commanded directly and used as input to control \( \alpha_1 \). The second assumption, \( \alpha_2 = 0 \), removes many of the stand's effects from the equations of motion. Third, it is assumed that the ducted fan is operating at constant \( \alpha_2 \) and \( \alpha_3 \), so \( \nu_1, \nu_2 \), and \( \alpha_2 = 0 \). With these assumptions, the \( \alpha_1 \) equation from (4) becomes

\[
\dot{\alpha}_1 = \frac{1}{n} \left( \frac{m g l}{r} - \beta \right) \tan \alpha_3.
\]

(5)

Finally, using the approximation \( \tan \alpha_3 = \alpha_3 \), which is valid for small \( \alpha_3 \), the simplified system is

\[
\begin{bmatrix}
\dot{\alpha}_1 \\
\dot{\alpha}_2
\end{bmatrix} = \begin{bmatrix}
k \alpha_3 \\
\nu_1
\end{bmatrix}, \quad \kappa = \frac{1}{n} \left( \frac{m g l}{r} - \beta \right)
\]

(6)

For this experiment, \( \kappa = 0.52 \).

The \( \alpha_1 \) controller generates an \( \alpha_3 \) command based upon the \( \alpha_1 \) error and \( \dot{\alpha}_1 \) error. To ensure that \( \tan \alpha_3 \approx \alpha_3 \), a saturation element at \( \pm 1.2 \) radians is placed on the command. The saturation element limits performance, but improves stability. For a large step disturbance, a linear controller would generate a large command. However, commands larger than \( \pi/2 \) are obviously not desired. Furthermore, the decoupler is singular when \( \alpha_3 = \pm \pi/2 \).

### Table 1: Comparison of controllers for a ramp in \( \alpha_1 \). The difference between \( \alpha_1 \) commanded and \( \alpha_1 \) measured is denoted by \( \epsilon_1 \).

<table>
<thead>
<tr>
<th>Criteria</th>
<th>LQR</th>
<th>( \alpha_1 ) rise time</th>
<th>( \alpha_1 ) 90% delay</th>
<th>( \alpha_1 ) % overshoot</th>
<th>steady state ( \epsilon_1 )</th>
<th>( \epsilon_1 ) rise time</th>
<th>( \epsilon_1 ) 90% delay</th>
<th>steady state ( \epsilon_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \alpha_1 )</td>
<td>3.34</td>
<td>4.24</td>
<td>3.24</td>
<td>3.68</td>
<td>0.03</td>
<td>0.26</td>
<td>0.34</td>
<td>0.25</td>
</tr>
<tr>
<td>( \alpha_2 )</td>
<td>2.33</td>
<td>3.96</td>
<td>2.35</td>
<td>6.09</td>
<td>0.05</td>
<td>0.05</td>
<td>0.07</td>
<td>0.03</td>
</tr>
<tr>
<td>( \alpha_3 )</td>
<td>53.59</td>
<td>73.51</td>
<td>48.91</td>
<td>136.64</td>
<td>0.51</td>
<td>0.44</td>
<td>0.68</td>
<td>0.37</td>
</tr>
<tr>
<td>( \alpha_1 ) rise time</td>
<td>10.42</td>
<td>8.11</td>
<td>14.87</td>
<td>8.79</td>
<td>1.28</td>
<td>1.12</td>
<td>1.20</td>
<td>0.80</td>
</tr>
<tr>
<td>( \epsilon_1 ) rise time</td>
<td>4.14</td>
<td>2.29</td>
<td>3.35</td>
<td>2.05</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4 Experimental Comparisons

The controllers were tested on three trajectories. Two of the trajectories are simple, commanding changes to one axis at a time. The third trajectory commands rapid changes to both the \( \alpha_1 \) and \( \alpha_2 \) axes.

The first trajectory is a one radian change on the \( \alpha_1 \) axis over 5 seconds. The second is a 0.1 radian step change on \( \alpha_2 \) over 5 seconds. While these trajectories are not challenging, they demonstrate the controllers' abilities to track each axis independently.

The third trajectory is more complex, commanding the fan to fly rapidly in the positive \( \alpha_1 \) direction while \( \alpha_2 \) is varied. During forward flight, the fan achieves \( \alpha_1 \) of 0.628 radians per second, more than three times greater than during the first trajectory.

The results for each of the controllers are presented in a figure with 4 plots. The upper left plot shows the \( \alpha_1 \) trajectory (called "X" in the plot) and the system's response. The lower right plot shows the \( \alpha_2 \) trajectory (called "Y" in the plot) and the system's response. The upper right plot shows the \( \alpha_3 \) behavior. For the decoupled controller results, Figures 6, 10, and 14, this plot also contains the \( \alpha_3 \) command. The final plot shows the commanded forces. The axial force, \( f_2 \), has a DC component of 2.65 Newtons to maintain hover.

Quantitative results are summarized in three tables, one for each trajectory. The controllers' rankings with respect to the performance measures are indicated by superscript numbers in each table entry. The 10% rise time is a standard figure of merit for step responses. For the \( \alpha_1 \) ramp, it provides a measure of 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 are self-explanatory. For all measures, \( \epsilon_1 \) is defined as the difference between the \( \alpha_1 \) commanded and \( \alpha_1 \) measured.

4.1 Ramp in \( \alpha_1 \) (horizontal position)

This trajectory consisted only of a slow change to one axis, and all controllers performed well. As shown in Table 1, the responses are similar. The most notable differences are in the delay to 90% of the final value and the \( \alpha_3 \) measures.

Since the LQR controller has no filtering states, any step changes at the inputs are passed directly to the plant. In Figure 3, the ramp corresponds to a step change on \( \alpha_1 \) and abrupt changes are seen in the commanded forces. This causes \( \alpha_3 \) to have larger variations than the other controllers.
Figure 3: Closed loop response of LQR controller for $\alpha_1$ ramp

Figure 4: Closed loop response of $H_\infty$ controller for $\alpha_1$ ramp

Figure 5: Closed loop response of gain scheduled controller for $\alpha_1$ ramp

The response of the $H_\infty$ controller for the ramp on $\alpha_1$ is shown in Figure 4. The $H_\infty$ controller has a much larger delay than the LQR controller, but is otherwise identical.

Various switching techniques were examined on the gain scheduled controller. The differences between smooth blending of output and abrupt switching were small. For all results in this paper, switching occurred at $|\alpha_1| > 0.4$. The gain scheduled controller’s performance, shown in Figure 5, is similar to the LQR controller. This is expected, since the gain scheduled controller at hover is identical to that of the LQR controller.

The I/O decoupling controller tracks the ramp slowly, but with a very smooth $\alpha_3$ response, as shown in Figure 6.

4.2 Step in $\alpha_2$ (altitude)

All controllers were able to track this trajectory well. The design requirement ensured steady state errors would be small, and this was the case. Table 2 summarizes the results. Due to encoder quantization, the minimum detectable error for this trajectory is a 3.1% error. Values less than this should be considered zero.

With the addition of an integrator on $\alpha_2$, the LQR controller tracks the step with essentially zero steady state error, as shown in Figure 7. The performance is not drastically different from that of the other controllers. The response of the $H_\infty$ controller to the step in $\alpha_2$ is shown in Figure 8 and the

Table 2: Comparison of controllers for a step in $\alpha_2$. The difference between $\alpha_1$ commanded and $\alpha_1$ measured is denoted by $\epsilon_1$.

Table: [Table content]

Figure 6: Closed loop response of I/O decoupled controller for $\alpha_1$ ramp

Table: [Table content]

Figure 7: Closed loop response of LQR controller for $\alpha_2$ step
Figure 8: Closed loop response of $H_\infty$ controller for $\alpha_2$ step

Figure 9: Closed loop response of gain scheduled controller for $\alpha_2$ step

Figure 10: Closed loop response of I/O decoupled controller for $\alpha_2$ step

Figure 11: Closed loop response of augmented LQR controller for complex trajectory

Figure 12: Closed loop response of $H_\infty$ controller for the complex trajectory.

Figure 13: Closed loop response of gain scheduled controller for complex trajectory

(2), any error in $\alpha_2$ is controlled by changing only the motor current. However, when $\alpha_3 \neq 0$, this induces both an $\alpha_3$ and an $\alpha_1$ change. At large $\alpha_3$, the motor current change has little effect on $\alpha_2$. In addition, the integrator on the $\alpha_2$ error winds up, and the commanded forces quickly saturate the actuators.

The response of the $H_\infty$ controller, is shown in Figure 12. Due to different weight selection, the controller performs better on $\alpha_2$, but worse on $\alpha_3$.

The gain scheduled controller performs extremely well on all measures. Once the initial transients die, it has the best performance of all the controllers.

One notable item in Figure 14 is the relatively smooth $\alpha_3$ response. Due to the structure of the controller, $\alpha_3$ does not have large oscillations. The saturation element also ensures that excessive $\alpha_3$ values are not achieved. This type of behavior is desirable, since it corresponds to a smoother set of commanded forces.

4.4 Results Summary

All controllers are able to follow the ramp in $\alpha_1$. The gain scheduled controller performs the best of all the controllers.
The \( H_m \) and LQR designs have similar performance, with the \( H_m \) design performing slightly better than the LQR design at the expense of nine states. The I/O decoupling controller performs the worst overall, although its \( \alpha_2 \) performance over the trajectory is better than the other controllers.

All controllers do a good job on the step in \( \alpha_2 \) as well. The I/O decoupling controller performs best. The gain scheduled controller has the second best performance, followed quite closely by the LQR controller. The \( H_m \) controller performs the worst. It is interesting that the I/O decoupler performs the best, since no decoupling is needed for a step on \( \alpha_2 \). The fact that the gain scheduled and LQR designs perform quite similarly is not surprising either, as no scheduling occurs on this trajectory.

Three controllers perform reasonably well on the complex trajectory. The exception is the LQR controller, which has an integrator wind-up problem, and thus is unable to follow the \( \alpha_2 \) changes in the trajectory. This is clearly evidenced by the Figure 14. The best controller overall is the gain scheduled controller, which has the best \( \alpha_2 \) tracking performance while maintaining good \( \alpha_2 \) performance. The \( H_m \) controller tracks the \( \alpha_2 \) portion of the trajectory very well, but suffers on the \( \alpha_2 \) response. The I/O decoupling controller achieves consistently good results overall rather than being best at one particular performance criterion.

## 5 Summary and Conclusions

The objective of this project is to study the application of different methodologies for flight control to a relatively simple nonlinear system with nonlinearities similar to those found on vectored thrust aircraft. We draw the following conclusions from the work described in this paper:

1. **Scheduled designs achieve better tracking performance than linear designs.** For aggressive trajectories over a wide operating envelope, the gain scheduled controller performs better than linear controllers. This, combined with other experience on the fan [2], suggests that pure linear techniques are not sufficient for high performance on this test case.

2. **Good tracking performance is achievable using (good) first principles models.** The controllers designed in this study are based on first principles models of the dynamics. The use of system identification techniques is limited in this context because of the need for (nonlinear) models which hold over the entire operating envelope. When poor models are available, there is little difference in overall performance between the different linear and nonlinear techniques.

3. **Gain scheduling performs as well as or better than feedback linearization.** Simple gain scheduled controllers built from three pointwise linearized models performed better than much more complicated nonlinear controllers on most of the trajectories tested. This shows that local linearizations can often capture most of the nonlinear effects which appear in flight control systems. More advanced scheduling techniques using linear parameter-varying synthesis (LPV) perform even better than the simple gain scheduled controllers studied here (see [2]), supporting this claim.

4. **Nonlinear techniques need to be more closely coupled with good linear control design.** Many nonlinear control techniques implicitly rely on the use of pole placement as part of the overall control law synthesis. Nonlinear designs based on pole placement do not perform well on the ducted fan and more advanced nonlinear techniques must take advantage of the wealth of good linear design techniques which are available for achieving small-signal performance around an operating point.

Future work on the ducted fan will improve upon the controllers presented here and continue to address the practical issues involved in good nonlinear control design. In particular, we will focus on the modeling of the system by including the gyroscopic effects of the blade, aerodynamic drag terms, and results from identification experiments in the model. With improved models, we expect to design controllers capable of achieving exceptional performance over aggressive trajectories.

Two specific tasks currently underway are the application of LPV synthesis techniques [2] and real-time trajectory generation with two-degree of freedom nonlinear controller design. These are complementary techniques for building nonlinear controllers which exploit the known nonlinearities of the system while giving guaranteed performance over a large range of operating conditions. The ducted fan also serves as an experimental testbed for new theoretical developments. This work includes the application of nonlinear robustness analysis [3, 8] and model reduction techniques for parameter-varying models [1].

## References


