Convex Modeling Techniques for Aircraft Control

by

Abhishek Kumar

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Masters of Science
in
Aerospace Engineering

Dr. Mark Anderson, chair
Dr. Frederick Lutze
Dr. Christopher D. Hall

June 7, 2000
Blacksburg, Virginia

Keywords: LPV modeling, LMI, Convex hull, Probability of instability

Convex Modeling Techniques for Aircraft Control

Abhishek Kumar

(ABSTRACT)

The need to design a controller that self-schedules itself during the flight of an aircraft has been an active area of research. New methods have been developed beyond the traditional gain-scheduling approach. One such design method leads to a linear parameter varying (LPV) controller that changes based on the real-time variation of system dynamics. Before such a controller can be designed, the system has to also be represented as an LPV system. The current effort proposes a LPV modeling technique that is inspired by an affine LPV modeling techniques found in recent research. The properties of the proposed modeling method are investigated and compared to the affine modeling technique. It is shown that the proposed modeling technique represents the actual system behavior more closely than the existing affine modeling technique.

To study the effect of the two LPV modeling techniques on controller design, a linear quadratic regulator (LQR) controller using linear matrix inequality (LMI) formulation is designed. This control design method provides a measure of conservatism that is used to compare the controllers based on the different modeling techniques. An F-16 short-period model is used to implement the modeling techniques and design the controllers. It was found that the controller based on the proposed LPV modeling method is less conservative than the controller based on the existing LPV method. Interesting features of LMI formulation for multiple plant models were also discovered during the exercise.

A stability robustness analysis was also conducted as an additional comparison of the performance of the controllers designed using the two modeling methods. A scalar measure, called the probability of instability, is used as a measure of robustness. It was found that the controller based on the proposed modeling technique has the necessary robustness properties even though it is less conservative than the controller designed based on the existing modeling approach.

Acknowledgments

I would like to thank my parents who have been constant source of inspiration for me. They have taken the pain of sending me for higher studies to United States of America. It is through their love and support that I have been able to achieve whatever I have aimed for. Thanks to my brother Rohit who is a bundle of fun and quips and who has been such a joy for my parents all these two years while I got to make the best of my stay in Virginia Tech.

I cannot thank less my advisor Dr. Mark Anderson for guiding me through my research. I feel lucky to be under his tutelage for the past two years. With the depth of practical experience and knowledge that he had shared with me, I have learned a lot. Among the many things that I am going to take from my research experience, I consider this worth mentioning. Learning is complete only when you can share it with and make a difference for someone else. Thanks to Dr. Frederick Lutze and Dr. Christopher Hall for being in my committee. I also thank Dr. Durham for letting me work in the Flight Simulator laboratory; it has been a wonderful experience to be a part of this laboratory. Special thanks to Major Mike Philips and Iossif Mugtussidis for all the help and support regarding my research. I had a great time with them during our Tuesday/Thursday meetings with our Boss.

Finally I thank fellow students with whom I had shared so much of my academic time. Roger Beck, Mark Nelson, Bill Oetjens, thanks for the good company in the laboratory. There are few more friends I would like to thank for making the overall experience in Virginia Tech a very memorable one. Thanks to Gagan Batra, Arijit Roy, Akhilesh Jha, the Stone Family and finally thanks to all my undergrad friends (the FTOP) for all the fun in past two years.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Designing a controller for systems with widely varying nonlinear and/or parameter-dependent dynamics is a major area of research in control theory. Gain-scheduling is a technique widely used to control such systems in a variety of engineering applications. The "gains" of the gain-scheduled controllers are typically chosen using linear control design techniques and is a two step process. First, several operating points are selected to cover the range of system dynamics. At each of these points, the designer makes a linear time-invariant (LTI) approximation to the plant and designs a linear compensator for each linearized plant. This process gives a set of linear feedback control laws that perform satisfactorily when the closed-loop system is operated near the respective operating points. A global nonlinear controller for the nonlinear system is then obtained by interpolating, or 'scheduling', the 'gains' from the local operating point designs. Note that even though the local point designs may have good feedback properties, the global gain-scheduled design is not guaranteed to have any of these properties. That is, any performance and robustness guarantees in the individual operating regions are not applicable in the transition region between point controllers. Thus, the designer typically cannot assess a priori the stability, robustness, and performance properties

of gain-scheduled controller designs.

The above method represents the "classical" gain-scheduling method and has immediate application in flight control. Modern aircraft operate over a wide range of flight conditions and the dynamic response characteristics vary significantly during a typical mission. Gain-scheduling has been used to design controllers that would vary during the operation; but the design process lacks any theoretical framework for addressing stability and robustness properties of the global scheduled controller. Such properties have to be inferred from extensive simulations. Clearly there is a need for a design approach that would explicitly take into account the relationship between real-time parameter variations and controller characteristics. This approach would enable controllers to be designed for the whole range of operating conditions, with theoretical guarantee of performance and robustness throughout the region. Many new linear parameter varying (LPV) control approaches have surfaced in the last few years that address such needs.

1.2 LPV Models and Controllers

Linear parameter varying (LPV) systems have recently become popular as they provide a systematic means of computing gain-scheduled controllers. LPV systems, first introduced by Shamma and Athans [1], are linear time-varying plant models whose state-space description is a fixed function of some parameter vector θ. In state-space form, a LPV system is described by

$$\dot{x} = A(\theta)x + B(\theta)u \tag{1.1}$$

$$y = C(\theta)x + D(\theta)u \tag{1.2}$$

where θ is a vector of varying parameters on which the system characteristics are assumed to depend. The parameter vector θ is not uncertain and can often be measured in real-time during system operation. However, it is generally assumed that the parameters in θ vary

slowly in comparison to the dynamics given by equation (1.1). Consequently, the control strategy for LPV systems is to exploit the available measurements of θ for better controller performance.

LPV based gain-scheduling approaches are replacing ad-hoc techniques and are becoming widely used in control design. Shamma [1], Sun et al [2], and Apkarian [3] present some design examples and applications. Once an LPV model is built, several control synthesis techniques can be adopted for controller design. For example, we have LQG control design for LPV system by Fen Wu [5], H_∞ control for LPV systems by Apkarian [3], and μ synthesis design by Shamma and Cloutier [6].

Many of the control system design techniques using LPV models can be cast or recast as convex problems that involve linear matrix inequalities (LMI). Significant progress has been made recently in the use of LMI and H_∞ optimization in gain-scheduled control (e.g. Sun [2]). One such control design technique, described by Apkarian [3], is the lyapunov function/quadratic H_∞ approach wherein a single lyapunov function is sought to bound the performance of the LPV system. Such a framework generally has a strong form of robust stability with respect to time-varying parameters. However, due to the continuous variation of scheduling parameters, such a synthesis approach is generally associated with a convex feasibility problem with infinite constraints imposed on the LMI formulation. This problem can be addressed by using affine LPV modeling that reduces the infinite constraints imposed on the LMI formation to a finite number. Such a modeling approach has been used to solve design problems by Sun [2] and Apkarian [3]. The current research addresses this particular class of LPV systems wherein the parameter dependence is affine.

1.3 Objectives and Approach

Modeling of system dynamics such that the resulting model adequately represents the system behavior throughout a range of operating conditions is a key step in designing LPV based gain-scheduled controllers. The present research has been inspired by Sun [2] and Apkar-

ian [3], where affine LPV modeling has been used before a controller design is attempted. Although they realized that the LPV modeling approach can produce a conservative design for controllers, the question of how conservative the resulting design will be has not yet been addressed. The present research attempts to quantify and give some measure of conservatism to the controller design. An alternate LPV modeling technique is proposed and the resulting controller design is compared with the one obtained using the existing modeling approach. The alternate modeling technique described in the current work is an extension of the affine modeling approach described in [3].

To compare the controllers obtained from different LPV modeling techniques, a controller design framework is required that can be implemented universally and that produces some measure of conservatism. Boyd et al. [7] have proposed a systematic procedure to transform the convex programs resulting from H_2 related problems into optimization over LMIs. The LPV models obtained from the modeling techniques can be treated as an optimization problem over several LMIs [7]. Balakrishnan, Boyd et al. [8] suggest an LMI formulation for solving multiple plant models simultaneously. Such a controller design has been adopted for the present work. The performance of the resulting H_2 controller designs for different LPV models can then be compared directly using the magnitude of the optimization function. The analysis also gives an insight into plant model characteristics when comparing one plant model to another.

Once the controllers for different LPV models are checked for conservatism, a robustness analysis for the controllers is conducted. The stochastic robustness method described in [12], [13], and [14] is used to measure the stability robustness for the controllers. This method was chosen because it is one of the few methods available to yield a measure of stability over the entire parameter sg

The new LPV modeling technique developed in this work is a better representation of the varying aircraft dynamics and produces a less conservative design for controller than that produced by other existing LPV modeling techniques. Chapter 2 introduces the LPV model development techniques in detail. The LMI control design technique adopted to compare

the controllers is presented in Chapter 3. In Chapter 4, the control design method is implemented on a design example of F-16 short-period model and the subsequent results are discussed. Stability robustness analysis for the controllers obtained for different LPV models is conducted in Chapter 5 and Chapter 6 is the summary and conclusion of the entire thesis work.

Chapter 2

Model Development

Linear parameter varying systems are reviewed and discussed in detail in this chapter. The LPV modeling technique adopted by Sun [2] and Apkarian [3] is explained and the new LPV modeling procedure is presented. Simple examples are used to illustrate the properties of both modeling techniques.

2.1 Overview of Linear Parameter Varying Systems

LPV systems can be considered in light of two other important classes of linear systems: linear time-invariant (LTI) systems and Linear time-varying (LTV) systems. LTI is the most common class of system. It is described in state-space form as

$$\dot{x} = Ax + Bu \tag{2.1}$$

$$y = Cx + Du \tag{2.2}$$

LTV systems represent systems where the state-space description is completely defined by the functional time dependence of the state-space data A(t), B(t), C(t), D(t). The associated

differential equations are

$$\dot{x} = A(t)x + B(t)u \tag{2.3}$$

$$y = C(t)x + D(t)u \tag{2.4}$$

Linear parameter varying (LPV) systems, first introduced by Shamma & Athans [1], are linear systems where the state-space description is an explicit function of a parameter vector θ. In state-space form, a LPV system is described by

$$\dot{x} = A(\theta)x + B(\theta)u \tag{2.5}$$

$$y = C(\theta)x + D(\theta)u \tag{2.6}$$

where θ is a time-varying vector that can consist of system outputs, exogenous inputs, or combinations of them. Note that a LPV system can be reduced to a LTV system for a given trajectory, $\theta = \theta(t)$ and to a LTI system for a constant trajectory, $\theta = \theta_0$.

An important difference among the three systems is their 'off-line' or 'on-line' nature. Specifically, LTI and LTV systems are off-line systems in the sense that the state-space data A, B, C, D or $A(t)$, $B(t)$, $C(t)$, $D(t)$ must be known beforehand. The LPV systems are on-line systems because they are completely known only when the trajectory $\theta = \theta(t)$ is known; that is, when the system operates and experiences a particular trajectory in its domain. The term 'domain' refers to the operating domain, which is the parameter range of the system. The study of LPV systems requires us to anticipate their future behavior. A LPV system is well defined whenever its parameter dependence and its operating domain are fixed. No a priori information is assumed other then the range of variation of each of the parameters. Thus, LPV modeling of systems is of immediate advantage for many practical applications.

Given a LPV model, the structure of the corresponding controller should be varying with the same parameter dependence as that of the model. That is, the controller should be of

LPV form:

$$\dot{x_k} = A_K(\theta)x_k + B_K(\theta)y \tag{2.7}$$

$$u = C_K(\theta)x_k + D_K(\theta)y \tag{2.8}$$

Since the controller is a function of parameter vector θ, it adjusts to the variations in the system dynamics, thus maintaining satisfactory performance along all possible trajectories of θ. Such a controller is called 'self-scheduled' as it automatically gain-schedules with respect to the vector θ. Once a LPV model is formed, several design techniques can be used for controller design.

The current research addresses the class of LPV systems where 1) the state-space matrices depend affinely on the time-varying parameter vector θ and 2) the time-varying parameter vector θ varies within a polytope of vertices. These concepts are explained in more detail in the next section.

2.2 LPV Modeling: The Bounding Box Approach

This section presents the affine LPV modeling technique adopted by Sun [2] and Apkarian [3]. It also attempts to give a definition of the term 'matrix polytope' which is an essential ingredient in understanding the modeling process.

The affine LPV modeling method is based on the concept of convex hulls and polytopes (Appendix B). We start with the basic assumption of LPV systems that measurements of θ are available in real-time and the range of θ is known a priori. That is, given a lth-dimensional parameter vector $\theta = (\theta_1 \dots \theta_j \dots \theta_l)$, upper and lower bounds are known for each element such that

$$\underline{\theta_j} \leq \theta_j \leq \overline{\theta_j} \tag{2.9}$$

where $\underline{\theta_j}$ and $\overline{\theta_j}$ are the lower and upper bounds of θ_j respectively. Vectors can be formed by taking each possible permutation of upper and lower bounds of elements in θ. Let us represent these vectors using ω_i. For example, one of these vectors will look like:

$$\omega_i = (\underline{\theta_1}, \ldots, \overline{\theta_i}, \ldots, \overline{\theta_l}) \tag{2.10}$$

There will be $m = 2^l$ vectors ω_i and they can be put into polytopic form

$$\theta \in Co\{\omega_1, \omega_2, \ldots, \omega_m\} := \left\{ \sum_{i=1}^{m} \alpha_i \omega_i : \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i = 1 \right\} \tag{2.11}$$

to define the parameter vector θ. Equation (2.11) states that vector θ belongs to the convex hull formed by the vertices ω_i. It also means that any vector θ can be represented by a linear combination of vertices ω_i such that the coefficients α_i are greater than or equal to zero and sum of the coefficients is equal to one. This construction clearly assumes that the minimum and the maximum value of each parameter θ_j is known.

Consider a two dimensional example, i.e. $l = 2$. The parameter vector will be $\theta = (\theta_1, \theta_2)$. Using a priori knowledge of the bounds on θ_1 and θ_2, the number of vertices formed by all permutations of lower and upper bounds of θ_1 and θ_2 will be $m = 2^2$. These vertices will form a convex hull with four vertices given by:

$$\omega_1 = (\underline{\theta_1}, \underline{\theta_2})$$

$$\omega_2 = (\underline{\theta_1}, \overline{\theta_2})$$

$$\omega_3 = (\overline{\theta_1}, \overline{\theta_2})$$

$$\omega_4 = (\overline{\theta_1}, \underline{\theta_2})$$

Thus a two dimensional parameter vector θ can be written in the following form:

$$\theta \in Co\{\omega_1, \omega_2, \omega_3, \omega_4\} \tag{2.12}$$

which states that a vector $\theta = (\theta_1, \theta_2)$ lies within the convex hull defined by the vertices $\omega_1, \ldots, \omega_4$.

The convex hull generated by such a modeling process will always be of the form of a bounding box. In the preceding example, the vertices ω_i, $i = 1, \ldots, 4$, form a rectangular box as shown in Figure 2.1. All variations in (θ_1, θ_2) will lie within the box. In higher dimensions, the convex hull will be a hyper box. We will call this modeling approach the bounding box method because the convex hull formed by this approach will always be the shape of a rectangular bounding box.

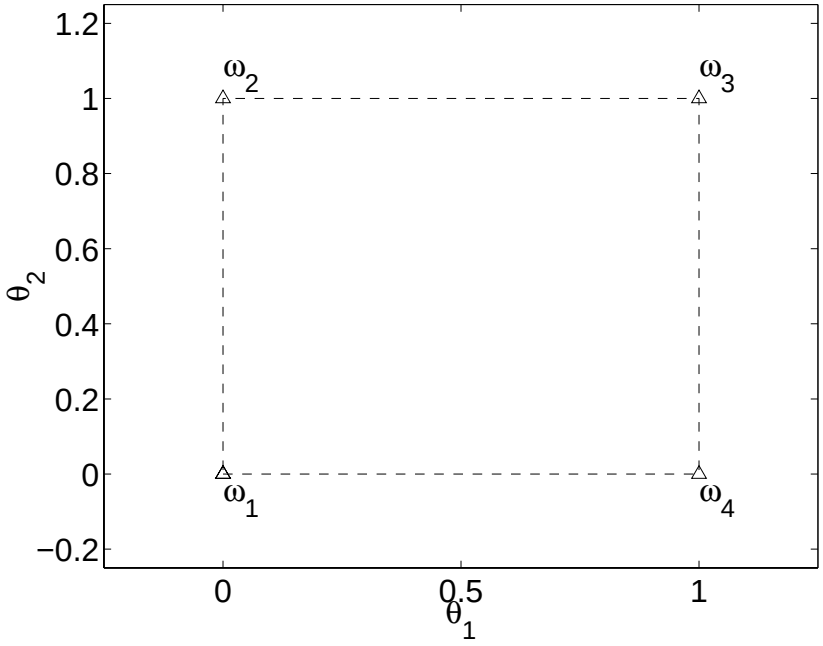

Figure 2.1: Bounding Box in Two Dimensions

If an affine function $f(\theta)$ is applied to the vector θ, the result can also be put into a polytopic form:

$$f(\theta) \in Co\{\Omega_1, \Omega_2, \ldots, \Omega_m\} := \left\{ \sum_{i=1}^{m} \alpha_i \Omega_i : \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i = 1 \right\} \tag{2.13}$$

Equation (2.13) states that the function $f(\theta)$ lies within the convex hull of vertices $\Omega_1, \ldots, \Omega_m$, where $\Omega_i = f(\omega_i)$. Note that the vertices ω_i are the same vertices that define the hull for

the varying parameter vector θ and m is the number of vertices. Thus, if the state-space matrices $A(\theta)$, $B(\theta)$, $C(\theta)$ and $D(\theta)$ depend affinely on θ, they will range in a polytope of matrices whose vertices are the images of the vertices, ω_i. Apkarian [3] describes a *matrix polytope* as the convex hull of a finite number of matrices N_i with the same dimensions. That is,

$$Co\{N_i, i = 1, \dots, m\} := \left\{ \sum_{i=1}^{m} \alpha_i N_i : \alpha_i \geq 0, \sum_{i=1}^{m} \alpha_i = 1 \right\} \tag{2.14}$$

Thus, for the matrices that make up a state-space LPV model:

$$\begin{pmatrix} A(\theta) & B(\theta) \\ C(\theta) & D(\theta) \end{pmatrix} \in Co \left\{ \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix} := \begin{pmatrix} A(\omega_i) & B(\omega_i) \\ C(\omega_i) & D(\omega_i) \end{pmatrix}, i = 1, \dots, m \right\} \tag{2.15}$$

Equation (2.15) states that the matrices A, B, C, and D lie within the convex hull formed by the matrices A_i, B_i, C_i, and D_i. The subscript i in the state-space matrices imply that the matrices are evaluated at vertex ω_i. Now we will explore what is meant by the term *matrix polytope*. Recalling the definition of a convex hull, equation (2.14) and (2.15) implies:

$$\begin{pmatrix} A(\theta) & B(\theta) \\ C(\theta) & D(\theta) \end{pmatrix} = \sum_{i=1}^{m} \alpha_i \begin{pmatrix} A_i & B_i \\ C_i & D_i \end{pmatrix} \tag{2.16}$$

with $\alpha_i \geq 0$ and $\sum_{i=1}^{m} \alpha_i = 1$.

To simplify the discussion, we start with a 2×2 matrix A. Thus, equation (2.16) applied to the matrix A will be:

$$A(\theta) = \sum_{i=1}^{m} \alpha_i (A_i) \tag{2.17}$$

Applying equation (2.17) to the elements of the A matrix yields:

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} = \sum_{i=1}^{m} \alpha_i \begin{pmatrix} a_{11_i} & a_{12_i} \\ a_{21_i} & a_{22_i} \end{pmatrix} \tag{2.18}$$

This equation implies that each element $\{a_{jk}\}$ of A is a linear combination of corresponding points $\{a_{jk}\}_i$ with the same coefficient α_i. This relationship is possible only when we define a vector ρ as the vector of matrix elements such that

$$\rho = (a_{11}, a_{12}, a_{21}, a_{22})^T \tag{2.19}$$

and form a convex hull using the vertices

$$\rho_i = (a_{11_i}, a_{12_i}, a_{21_i}, a_{22_i})^T \tag{2.20}$$

Thus, the vector ρ belongs to the convex hull formed by vertices ρ_i given by

$$\rho \in Co\{\rho_1, \ldots, \rho_i, \ldots, \rho_m\} \tag{2.21}$$

which in turn can be expanded to:

$$\begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \end{pmatrix} = \sum_{i=1}^{m} \alpha_i \begin{pmatrix} a_{11_i} \\ a_{12_i} \\ a_{21_i} \\ a_{22_i} \end{pmatrix} \tag{2.22}$$

with $\alpha_i \geq 0$ and $\sum_{i=1}^{m} \alpha_i = 1$.

Again, if only the range of elements in ρ is known to us, we can use the permutations of lower and upper bounds to form $m = 2^4$ vertices. These sixteen vertices will form the corners of a four dimensional box. Note that if we consider all of the elements of $A^{n\times n}$, $B^{n\times p}$, $C^{k\times n}$

and $D^{k\times p}$, the vector ρ will be a $(n^2 + np + kn + kp)$ dimensional vector. In such a case, the bounding box method will form a hyper box of appropriate dimension. This can pose a numerically challenging problem for control design. Choice of elements to be included in the vector ρ can be of important consideration in such a case. Also note that a point in a space given by the vector ρ actually represents a linear time-invariant plant model which corresponds to a unique point in the parameter space given by the vector θ.

2.3 LPV Modeling: The Small Hull Approach

The bounding box modeling approach is based upon a priori knowledge of the range of the varying parameters. An alternative approach proposed herein relies on the fact that not only is the range of varying parameters known, but also intelligent choices can be made about the intermediate values that the varying parameter θ can take. Note that we still do not consider the variation of θ in real-time. Thus, the proposed modeling technique is based on both range and the intermediate values of the varying parameter. With the knowledge of flight dynamics and experience gained in solving flight control problems, we can pick appropriate operating points lying in the flight envelope of an aircraft. These operating points give us the definition of parameter vector θ. Linearizing about these operating points yields the state-space matrices A, B, C, and D. The elements of the matrices define a stability derivative vector ρ. Enumerating the above two steps:

1. Choose n points within the flight envelope of an aircraft. This choice defines the operating conditions and thus the varying parameter θ. Say, we choose l parameters. The vectors θ_i, $i = 1, \ldots, n$, will be of dimension l.

2. The n parameter vectors θ_i in turn will form n stability derivative vectors ρ_i. The dimension of ρ will be given by the number of elements in the state-space matrix description and not necessarily by the dimension of θ. Lets say the vector ρ is of dimension d.

Given d-dimensional vectors ρ_i, $i = 1, \ldots, n$, the bounding box approach finds a convex box that will contain all of the ρ vectors. This bounding box will be defined by $m = 2^d$ corners. Note that these corners are formed by taking permutations of lower and upper bounds of d elements in the vector ρ, and are not obtained from the actual operating points, i.e. they do not necessarily occur in practice.

Alternatively, we can use a computational algorithm to find the points, from among the given set of points (ρ_i), that would form the vertices of a convex hull. They would be the "extreme" points in the d dimensional space. The proposed modeling approach uses the Qhull [10] algorithm for such a purpose. The points that form the vertices of the hull come from among the points that are obtained using the actual operating conditions.

To illustrate the two modeling approaches, a simple two dimensional stability derivative vector example is presented. This example does not represent any real system. It has been fabricated to highlight some important features of the proposed modeling process and to compare it with the bounding box approach. Let $d = 2$, which implies $\rho = (a, b)$, where a and b are two elements of vector ρ. The 'o' symbols in Figure 2.2 represent typical points of the system in (a, b) space. The '*' symbols overlapping the 'o' symbols are the points that form the vertices of the convex hull that would result from an algorithm like Qhull. The solid line gives the shape of the hull. The bounding box method produces $m = 2^2$ vertices by taking the permutations of lower and upper bounds of a and b. The rectangular box in dashed lines with the vertices given by '$\triangle$' symbols is the bounding convex box.

The figure also shows two possible trajectories that the system can take. Notice that the system is more likely to follow "path B" than it is to follow "path A", because the former goes through the typical points of the system. While "path A" lies inside the bounding box, it passes outside the smaller convex hull. The objective is to form a model that best represents the true behavior of the system. Clearly a model that uses the points that form the vertices of the smaller convex hull will be more representative of the system because those points are part of the typical operating points of the system. Modeling based on the points that form the vertices of the bounding box will not be as close a representation of the

system behavior because these points are clearly not part of the typical operating points that the system goes through. It is also possible that some of these vertices may be unrealistic. For instance, the system in this example might never realize a $(\overline{a}, \underline{b})$ configuration. This example illustrates that a better modeling of the system can be attained using the proposed new model. Note that both modeling approaches produce convex hulls, but they will be termed as the 'bounding box' and the 'small hull' to distinguish one hull from the other. The proposed modeling technique will be termed as the *small hull approach*.

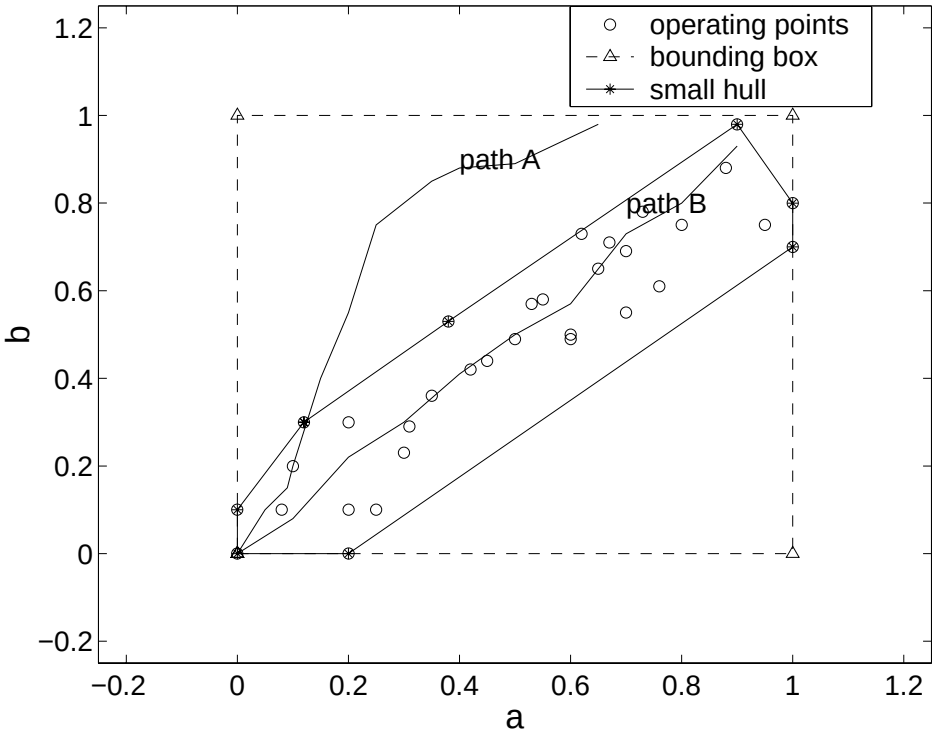

Figure 2.2: Convex Hull for the Two Modeling Techniques

The convex hull formed using the proposed approach is a smaller volume than the convex bounding box and is therefore contained inside it, hence the name 'small hull'. The circles are representative of the stability derivative vector ρ obtained from the operating points. The '*' symbols are the vertices that define the small hull, and all points lying inside the hull are defined by the vertices forming the small hull. Corners of the bounding box represent the vertices formed by the bounding box approach and they do not necessarily reflect the

actual operating conditions.

In this hypothetical example, eight vertices formed the small hull while the bounding box is formed using four vertices. This result suggests that more plant models are required for LPV modeling using the small hull approach vis-a-vis the bounding box approach. Such a case is not always true. The number of vertices that form a convex hull depends on the position of the points in the d dimension space. For example, in two dimensions, three points can form a convex hull, while we need four points to form the corners of the bounding box. Similarly, in three dimensions, only four points in space can form a convex hull (with four faces) while eight points are required to form the corners for the bounding convex box. Figure 2.3 shows a tetrahedral (with four vertices and four faces) hull vis-a-vis an eight cornered bounding box.

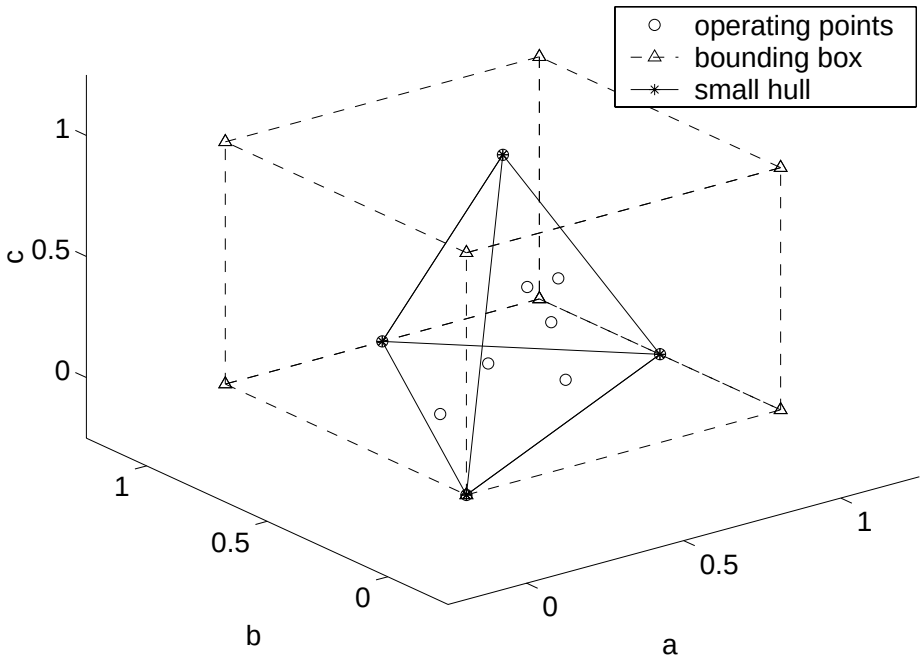

Figure 2.3: Convex Hulls in Three Dimensions

The small hull more closely represents the actual operating conditions and should result in a better control design. One might argue that such a control design will fail for vectors lying

outside the small hull. However, this concern can be addressed by choosing the operating points such that they adequately capture the behavior of the system. A design based on the bounding box will produce a conservative design that may work for extreme conditions, but the resulting control law may not perform well at more typical operating points. It has been shown that the number of vertices needed for design do not necessarily increase in the small hull technique. Thus, with appropriate choice of operating conditions, the number of vertices needed for controller design can be potentially reduced while still leading to a less conservative design. But, in cases where modeling of a system based on the small hull approach requires more plant models, we have to weigh the advantages of the resulting LPV model against the numerical complexity involved. There is a trade off that clearly has to be done. More insight can be gained once a controller based on the two LPV modeling methods is designed and some measure of conservatism for the subsequent controller designs is formulated.

Chapter 3

Controller Design

3.1 LQR Problem for Multiple Models

To achieve the desired objective, a control design algorithm has to be chosen such that the resulting designs for the two modeling approaches can be adequately compared. The aim is not to design a LPV controller, but to have a representative control design so that the closed-loop performance can be compared. A state-feedback controller design using the linear quadratic regulator (LQR) method is adopted to serve such a purpose. Each LPV modeling approach yields a set of plant models. We intended to obtain one state-feedback gain matrix for each set that would work for not only all the plant models in the set but also in between the operating points. This chapter describes the control design problem formulation and the method used to solve the problem.

Consider the linear time-invariant system described by the state equations:

$$\dot{x} = Ax + Bu + w \tag{3.1}$$

$$z = \begin{bmatrix} R^{\frac{1}{2}} & 0 \\ 0 & Q^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} u \\ x \end{bmatrix} \tag{3.2}$$

where u is the control input and z is the output signal of interest. The LQR problem is to design a feedback controller from the state x to the control input u which minimizes the H_2 norm between w and z. It is known that the optimal feedback law is a constant state-feedback $u = -Kx$ and optimizes the following problem (see Appendix A):

$$min \left\{ \mathbf{Tr}(QP) + \mathbf{Tr}(R^{\frac{1}{2}} K P K^T R^{\frac{1}{2}}) \right\} \tag{3.3}$$

subject to

$$(A - BK)P + P(A - BK)^T + I < 0 \tag{3.4}$$

and

$$P = P^T > 0 \tag{3.5}$$

By defining a new quantity $Y = KP$, the above problem can be written as

$$min \left\{ \mathbf{Tr}(QP) + \mathbf{Tr}(R^{\frac{1}{2}} Y P^{-1} Y^T R^{\frac{1}{2}}) \right\} \tag{3.6}$$

subject to

$$AP + PA^T - BY - Y^T B^T + I < 0 \tag{3.7}$$

and

$$P = P^T > 0 \tag{3.8}$$

which is a convex program. Balakrishnan et.al. [7] have shown that the optimization problem

can also be written as

$$min\,\{\mathbf{Tr}(QP) + \mathbf{Tr}(X)\} \tag{3.9}$$

subject to

$$AP + PA^T - BY - Y^T B^T + I < 0 \tag{3.10}$$

and

$$\begin{bmatrix} X & R^{\frac{1}{2}}Y \\ Y^T R^{\frac{1}{2}} & P \end{bmatrix} > 0 \tag{3.11}$$

The constraints in the above problem are linear matrix inequalities (LMIs) and the unknowns in the LMIs are X, Y, and P.

An extension of this method is suggested by Boyd [8], where one can solve more than one plant model simultaneously and find one feasible controller that works for all of the plant models. In such a case, the LMI constraint in equation (3.10) will look like:

(3.12)

where m is the number of plant models being solved. Once Y and P are determined, the state feedback gain matrix for all m plant models is given by

$$K = YP^{-1} \tag{3.13}$$

The gain matrix K will be the controller that satisfies the LMI constraints and works for all the plant models. This control design strategy will be called the LMI formulation for solving multiple plant models. The next section describes some of the features of this control design strategy.

3.2 Properties of the LMI Formulation

Before using this control design strategy for aircraft models, some simple examples were developed to test and validate the LMI formulation. Many other characteristics were identified, over a period of time, as the strategy was applied to different models. This section presents the features of the control design method using simple second-order design examples.

3.2.1 LMI Example 1

A linearized second-order plant model given by equation (3.14) is used to test the LMI formulation for designing a LQR controller.

$$\dot{x} = \begin{pmatrix} 1 & 0 \\ 0 & 2 \end{pmatrix} x + \begin{pmatrix} 1 \\ 1 \end{pmatrix} u \tag{3.14}$$

$$R = 1 \tag{3.15}$$

$$Q = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{3.16}$$

A standard LQR solution method (the *lqr* command in $MATLAB^{TM}$) is used to obtain a gain matrix K_{lqr}. It is then compared to the gain matrix K_{lmi} obtained using the LMI setup. The *LMI toolbox*^{TM} in $MATLAB^{TM}$ is used to solve the convex optimization problem. The H_2 norm is calculated in the first case and is compared with the value of minimization function obtained by solving the LMIs. The weighting matrix for the control input, R, and the weighting matrix for the state vector, Q, is given by equation (3.15) and (3.16) respectively.

Table 3.1 illustrates that the value of the feedback gains obtained using both LQR and LMI methods are approximately same. The value of the H_2 norm and value of the minimization function obtained by solving the LMIs are also very close. Thus, the LMI setup of the LQR problem gives the same results as the standard LQR algorithm and has been verified for our

use in the design example in the next chapter.

Table 3.1: Testing the LMI Setup (Example 1)

Method	Gain Matrix	H_2 norm
Standard LQR	$K_{lqr} = [-7.61 \quad 14.21]$	8.869
LMI Setup	$K_{lmi} = [-7.59 \quad 14.19]$	8.876

3.2.2 LMI Example 2

Now that the LMI setup for LQR controller has been validated for a single plant model, another example is used to study the LMI control design formulation for multiple plant models. Consider Table 3.2, where the state-space matrices for four plant models are given. All of the plant models represent unstable second-order systems with one unstable pole. The state-space matrices are in modal form, i.e. the poles are shown along the diagonal. This state transformation is used to keep the example simple and easy to interpret. The H_2 norm and the gain matrix K_{lqr} obtained for the individual plant models are also shown in the table. It should be emphasized that the gain matrix K_{lqr} in Table 3.2 is found for the associated plant model without considering the other models. The weighing matrix Q is an identity matrix and the weighting matrix R is unity. Both Q and R are held the same for all plant models so that the resulting H_2 norm can be compared directly. Plant model 2 has the largest H_2 norm and is the 'worst' among all plant models.

The LMI formulation for multiple plant models (Section 3.1) is implemented to obtain a single controller for all four plant models. The LMI constraints given by equation (3.12) will be

$$A_i P + P A_i^T - B_i Y - Y^T B_i^T + I < 0, \quad i = 1, \ldots, 4 \tag{3.17}$$

The four LMIs should all be negative definite or, equivalently, the eigenvalues of the LMIs

Table 3.2: Plant Models Used to Test LMI Formulation (Example 2)

Plant Model	A	B	H_2 norm	K_{lqr}
1	$\begin{pmatrix} -1 & 0 \\ 0 & 5 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 3 \end{pmatrix}$	1.35	[-0.252, 3.80]
2	$\begin{pmatrix} -1 & 0 \\ 0 & 3 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$	2.81	[-0.202, 6.75]
3	$\begin{pmatrix} -1 & 0 \\ 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$	2.42	[-0.131, 4.74]
4	$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 \\ 1 \end{pmatrix}$	1.93	[-0.00, 2.73]

should be less than zero. Since the constraint is to have all the LMIs with negative eigenvalues, it is intuitive to hypothesize that the LMI corresponding to the 'worst' plant model will have the least negative eigenvalue. Having a negative definite LMI for the 'worst' plant model should automatically guarantee negative definiteness for other LMIs since only one controller is to be found. Thus, by observing the eigenvalues of the LMIs we should be able to identify the 'worst' plant model from a given set of plant models.

Table 3.3 shows the eigenvalues resulting from the four LMIs. As expected, the LMI of the second plant model has one of its eigenvalues as the least negative among all the eigenvalues of other LMIs. This result implies that the second plant model is closest to violating its constraint equation. Note that the second plant model also has the largest H_2 norm (Table 3.2).

The results obtained from LMI formulation are as follows. The single controller design is

$$K_{lmi} = [-0.24, 7.01] \tag{3.18}$$

Table 3.3: Eigenvalues of Inequalities

Plant Model	Eigenvalues
1	(-0.0121 -3.8232)
2	(-0.8160 -0.0012)
3	(-0.8041 -0.2589)
4	(-0.8006 -0.5082)

and the value of the optimization function is

$$(\mathbf{Tr}(QP) + \mathbf{Tr}(X)) = 2.88 \tag{3.19}$$

The gain matrix K_{lmi} is similar to the gain matrix K_{lqr} obtained for the second plant model in Table 3.2. The value of the optimization function in the LMI formulation is also close to the H_2 norm of the second plant model. Note that in Table 3.2, plant model 1 has a right-half-plane pole ($s = +5$) that is more unstable than the right-half-plane pole ($s = +3$) of plant model 2, but plant model 1 is not the 'worst' plant model because it has more control power than plant model 2. Thus, the 'worst' plant model is determined by not only the location of its poles, but also how controllable the plant model is. This example illustrates that the controller design using LMI formulation is most affected by the 'worst' plant model.

Thus, the LMI set-up for designing LQR controller has been verified and validated. Important characteristics of LMI formulation for solving multiple plant models were identified. While working with this control strategy to find a controller design for multiple short-period plant models of the F-16 design example, a curious, not previously known behavior of the LMI formulation was discovered. LMI Example 3 highlights this particular feature of the control design strategy.

3.2.3 LMI Example 3

The following exercise was conducted to study an interesting behavior of the LMI formulation and to understand its implications to controller design. Six cases of multiple plant models were designed. Each case (C) had two plant models (P1 and P2). For each plant model P, in each case C, the eigenvalues, gain matrix K_{lqr}, and the H_2 norm was calculated. Table 3.4 shows the state matrix A and the eigenvalues of all the plant models. The input matrix B for all the plant models is constant:

$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{3.20}$$

The LMI formulation to solve multiple plant models is applied to obtain a single controller for each case. The resulting controller K_{lmi} and the value of the minimization function for each case are tabulated in Table 3.5 against the values of K_{lqr} and H_2 norm for each plant model. Note that plant model 1 (P1) is the same in all the cases and it is only the $A(1,1)$ element of plant model 2 (P2) that changes in each case.

Table 3.5 reveals some interesting features of LMI behavior. Consider case 1 (C1). Although the magnitude of the elements in the gain matrix K_{lqr} is larger for P1, the H_2 norm is lower than that for P2. The LMI solution for C1 identifies P2 as the 'worst' plant model and has the gain matrix K_{lmi} and the value of minimization function close to the gain matrix K_{lqr} and H_2 norm of P2 respectively. But such a case is not true for cases C2-C5, where the LMI solutions picks a gain matrix, which lies in between the gain matrices of the individual plant models in each case. Note that as the H_2 norm for P2 steadily decreased from C1 to C6, the solution of the LMI, i.e., the gain matrix K_{lmi} changes from being close to P2 to being close to P1. Clearly there is some gradual change in strategy followed by the LMI formulation.

Further efforts were made to better understand this behavior of LMIs. A root-locus plot was made for the state feedback design in each case. The following was used for the root-locus

construction:

$$det[sI - A + kBK_{lmi}] = 0 \tag{3.21}$$

for varying k = 0 to ∞.

The single controller obtained from the LMIs in each case is used to study the root-locus. This analysis was intended to see if the location of poles and zeros had any bearing on how the LMI behaves. Figures 3.1 and 3.2 show the root-locus of plant models in each case. In each of the root-locus plots in the two figures the 'D' symbols overlapping the 'x' symbols show that location of poles for a value of $k = 1$. Thus, from equation (3.21) these pole locations are the closed-loop poles of the plant models for the corresponding K_{lmi} in each case. For C1, the gain matrix K_{lmi} is such that the zero cancels the stable pole in P2. The other root is given by the 'D' symbols overlapping the 'x' symbol and is close to the mirror image of the unstable open-loop plane. In P1, the stable pole is not canceled, but the zero from the LMI controller is placed to the left of the pole which leads to complex conjugate roots.

As we look at cases C2-C5 we see that the controller K_{lmi} is such that stable pole of neither P1 nor P2 is canceled. But a gradual transition in LMI behavior is easy to identify. The zero for P1 gradually moves from its position in C1 to cancel the stable pole for P1 in C6. In case of C6, the LMI controller is such that the zero cancels the stable pole for P1 and not for P2. That the LMI formulation demonstrates a consistent shift in control strategy depending on the plant model characteristics is evident from the figures. This is a very interesting observation and can have important implications for control design of multiple plant models.

3.2.4 Summary of Controller Properties

Following is the summary of characteristics for solving multiple plant models using LMI formulation:

Table 3.4: Plant Models Used in Example 3

Case/Plant Model	State Matrix	Eigenvalues
C1/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C1/P2	$\begin{pmatrix} -0.4 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(2.589 , -7.989)
C2/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C2/P2	$\begin{pmatrix} -0.8 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(2.305 , -8.105)
C3/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C3/P2	$\begin{pmatrix} -1.0 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(2.166 , -8.166)
C4/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C4/P2	$\begin{pmatrix} -1.2 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(2.028 , -8.228)
C5/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C5/P2	$\begin{pmatrix} -1.4 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(1.892 , -8.292)
C6/P1	$\begin{pmatrix} -0.057 & 1.0 \\ 22.68 & -0.237 \end{pmatrix}$	(4.617 , -4.911)
C6/P2	$\begin{pmatrix} -1.8 & 1.0 \\ 22.68 & -5.0 \end{pmatrix}$	(1.624 , -8.424)

Table 3.5: Results for Example 3

Case/Plant Model	H_2 norm	Gain Matrix (K_{lqr})	Gain Matrix (K_{lmi})	H_2 norm
C1/P1	4761.79	[44.81 9.23]	[39.66 5.25]	5513.15
C1/P2	5508.20	[39.29 5.17]		
C2/P1	4761.79	[44.81 9.23]	[36.28 5.42]	5031.86
C2/P2	5006.46	[33.68 4.61]		
C3/P1	4761.79	[44.81 9.23]	[38.55 6.46]	4889.75
C3/P2	4761.61	[31.04 4.33]		
C4/P1	4761.79	[44.81 9.23]	[40.96 7.55]	4807.50
C4/P2	4520.58	[28.50 4.05]		
C5/P1	4761.79	[44.81 9.23]	[42.95 8.55]	4771.82
C5/P2	4283.19	[26.07 3.78]		
C6/P1	4761.79	[44.81 9.23]	[45.02 9.31]	4768.17
C6/P2	3818.47	[21.52 3.24]		

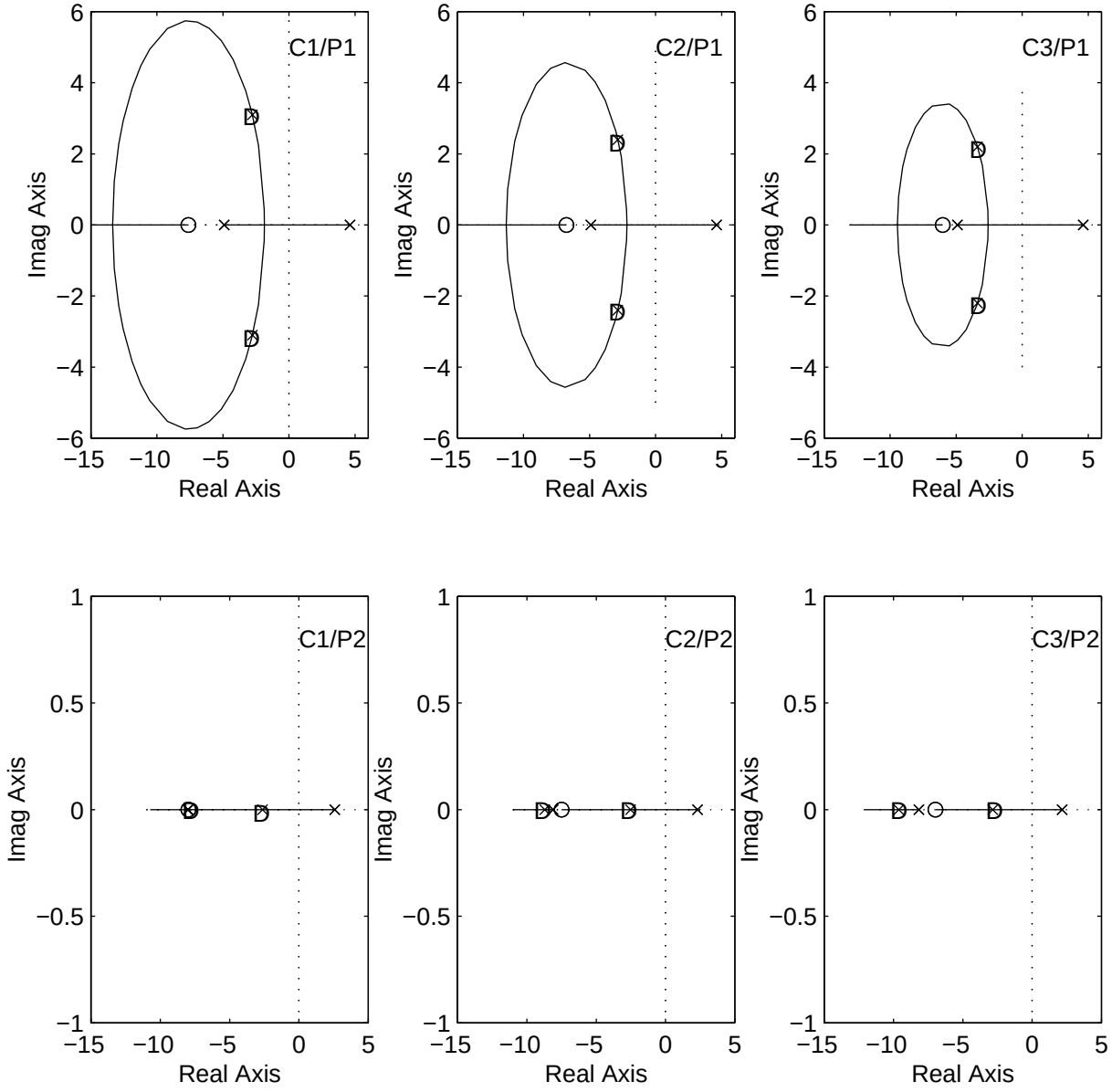

Figure 3.1: The Root-Locus of Plant Models (C1-C3)

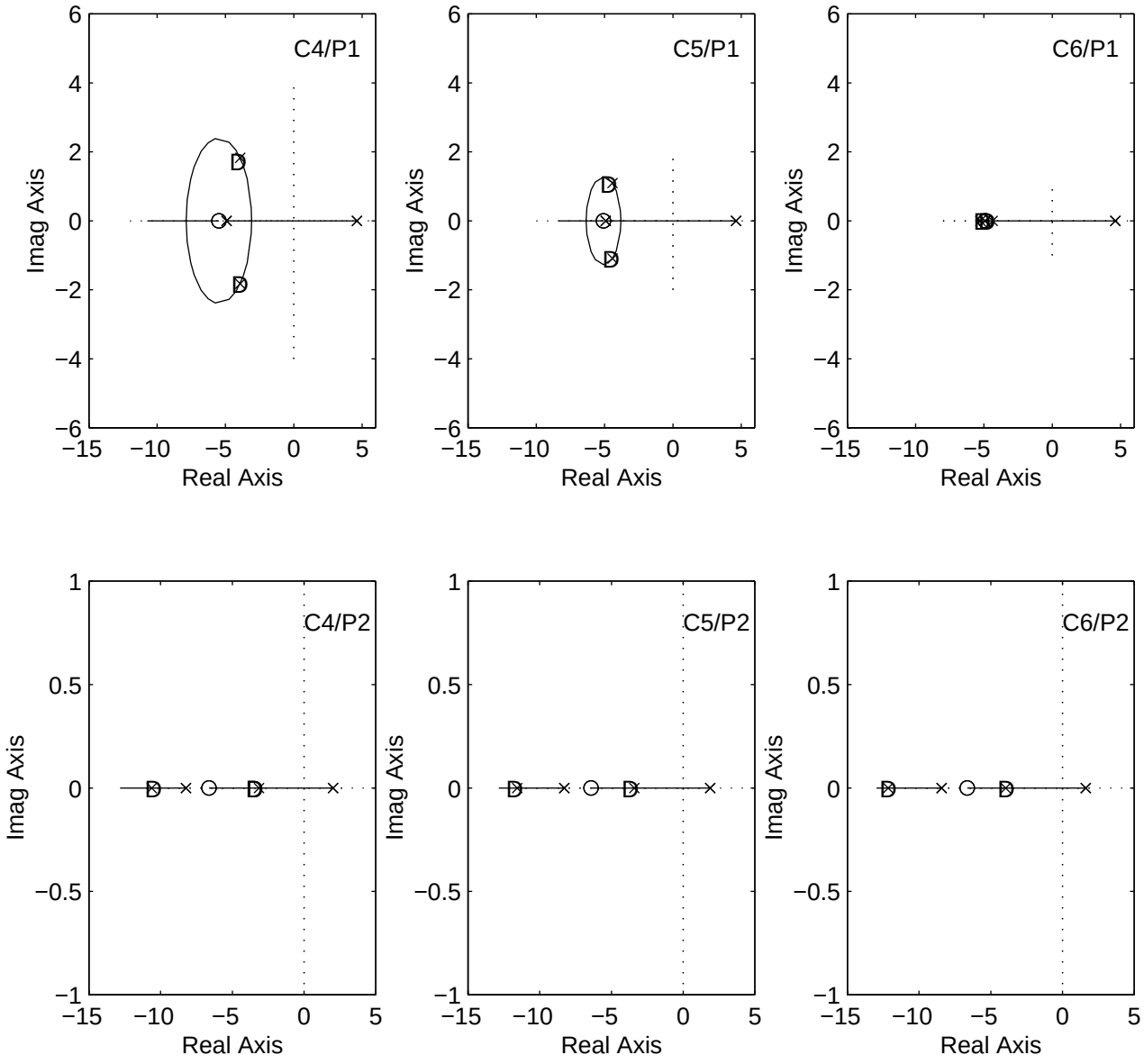

Figure 3.2: The Root-Locus of Plant Models (C4-C6)

1. The single controller design for multiple plant models obtained using LMI formulation is most effected by the 'worst' plant model. The 'worst' plant model is determined by not only how unstable the poles are, but also how controllable the plant model is (Section: 3.2.1).

2. The eigenvalues of the LMI inequality for each plant model can be used to determine the 'worst' plant model from a given set of plant models. The plant model that has the least negative eigenvalue (closest to becoming positive definite) for its inequality will be the 'worst' plant model (Section: 3.2.2).

3. We cannot just look at the open-loop poles and the control gains to suggest if one plant model is easier to control than another. In C1 of the LMI Example 3, the control gains for P1 are higher in magnitude than that of P2 and the right-half-plane pole for P1 is more unstable than the right-half-plane pole for P2. But the H_2 norm for P1 is less than that obtained for P2 and the LMI formulation chooses a controller gain matrix K_{lmi} which is very close to gain matrix obtained by considering P2 alone.

4. The LMI formulation does not always design a controller based solely on the 'worst' plant model. In LMI Example 3, C4 is a case where the H_2 norm for P1 and P2 are very close. We expected the LMI to pick either of the plant model as 'worst' and design a controller based on that condition. However, the controller design appears to be a blend of LQR controllers for the individual plant models. The root-locus analysis showed that the pole/zero constellation determines the appropriate control strategy. The 'worst' plant model results from a combination of pole location, zero locations and existing control power. It may not even be possible to identify a single plant model as the 'worst' plant model.

The LMI control design has been studied in detail and some of its important features have been identified. In the next chapter, an F-16 design example is used to implement the two LPV modeling techniques. Controllers are designed using the LMI set-up and the subsequent results are discussed.

Chapter 4

Aircraft Modeling

The proposed modeling technique and a control design framework using LMIs has already been introduced. This chapter presents the aircraft model that is used to study the LPV modeling techniques in a more practical application environment. LQR controllers are designed using the LMI formulation and the results are compared.

4.1 F-16 Model

An F-16 longitudinal model is used to form the design example. The linearized state-space models will be given by:

$$\dot{x} = Ax + Bu \tag{4.1}$$

$$y = x \tag{4.2}$$

The state vector x is

$$x = \left\{ \begin{array}{c} \alpha \\ q \end{array} \right\} \tag{4.3}$$

where α is the angle of attack (in radians) and q is the pitch rate (in radians per second). Thus, the differential equations represent the longitudinal, short-period motion of the aircraft. The state-space matrices A and B are defined by

$$A = \begin{bmatrix} Z_\alpha & 1 \\ M_\alpha & M_q \end{bmatrix}, B = \begin{bmatrix} 0 \\ M_{\delta_e} \end{bmatrix}$$

where Z_α, M_α, M_q, M_{δ_e} are the dimensional stability derivatives given by

$$Z_\alpha = -Cl_\alpha \frac{\bar{q}S}{mV} \tag{4.4}$$

$$M_\alpha = \left(Cm_\alpha - Cl_\alpha(\bar{x}_{cg_{ref}} - \bar{x}_{cg})\right) \frac{\bar{q}S\bar{c}}{I_{yy}} \tag{4.5}$$

$$M_{\delta_e} = Cm_{\delta_e} \frac{\bar{q}S\bar{c}}{I_{yy}} \tag{4.6}$$

$$M_q = Cm_q \frac{\bar{q}S\bar{c}}{I_{yy}} \frac{\bar{c}}{2V} \tag{4.7}$$

The non-dimensional derivatives are Cl_α (lift curve slope), Cm_α (pitch moment curve slope), Cm_{δ_e} (pitch moment due to elevator deflection), and Cm_q (pitch moment due to pitch rate). The aircraft physical constants are I_{yy} (moment of inertia about the y-axis), $\bar{c}$ (mean aerodynamic chord), m (mass), S (wing area), and $\bar{x}_{cg_{ref}}$ (reference longitudinal center of gravity location). Air relative variables are V (velocity) and $\bar{q}$ (dynamic pressure). The aerodynamic coefficients are given as polynomials of α and δ_e in the work by E.A.Morelli [9]. The aircraft constants and the polynomials for aerodynamic coefficients are given in Appendix C.

4.2 LPV Modeling of F-16 Model

The first step in forming a LPV model is to define the parameter vector θ. The operating condition parameters that can be varied are the flight variables: velocity(vel), altitude (alt), angle of attack (α), elevator deflection (δ_e), and center of gravity location ($\bar{x}_{cg}$). The elements

in the state-space matrices that are functions of the flight variables will form ρ, the vector of stability derivatives. It is important to ensure that the numerical values in the vector ρ form a realistic representation of an aircraft in flight.

As already mentioned, the aerodynamic coefficients are given as polynomials of α and δ_e. Figure 4.1 shows the aerodynamic lift coefficient C_L as a function of α and δ_e. We can find the maximum C_L (C_{Lmax}) from the graph and use the steady-state equation (4.8) for lift,

$$\bar{q}SC_L \approx nW \tag{4.8}$$

to find the minimum dynamic pressure ($\bar{q}_{min}$). In equation (4.8), S (wing area) and W (weight) are assumed known quantities. We choose load factor equal to unity to find the minimum dynamic pressure of $\bar{q}_{min} = 28\ psf$. Maximum dynamic pressure is selected as $\bar{q}_{max} = 1000\ psf$.

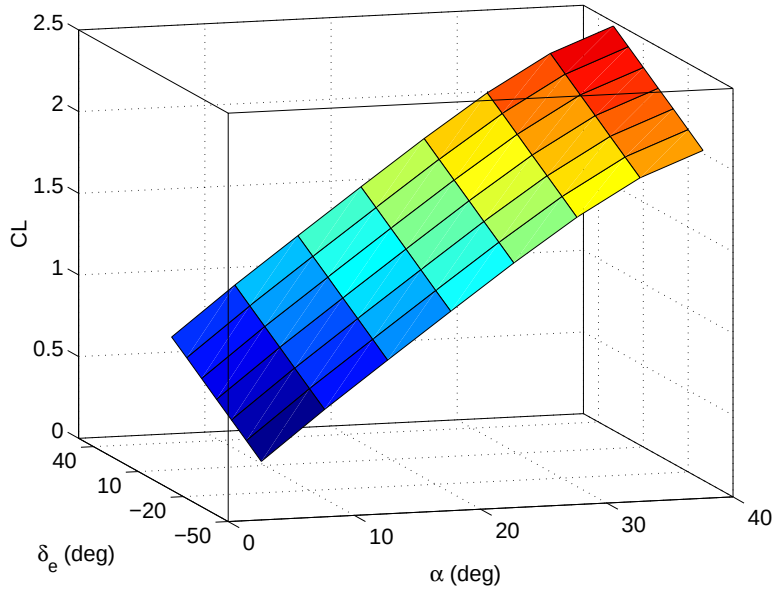

Figure 4.1: Lift Coefficient as a Function of Alpha and Elevator Deflection

Given the minimum and the maximum dynamic pressures, combinations of velocity and altitude can be found to define the flight envelope. An altitude of 30,000 ft is chosen for the service ceiling. Figure 4.2 shows the flight envelope for F-16 based on the above approxima-

tions.

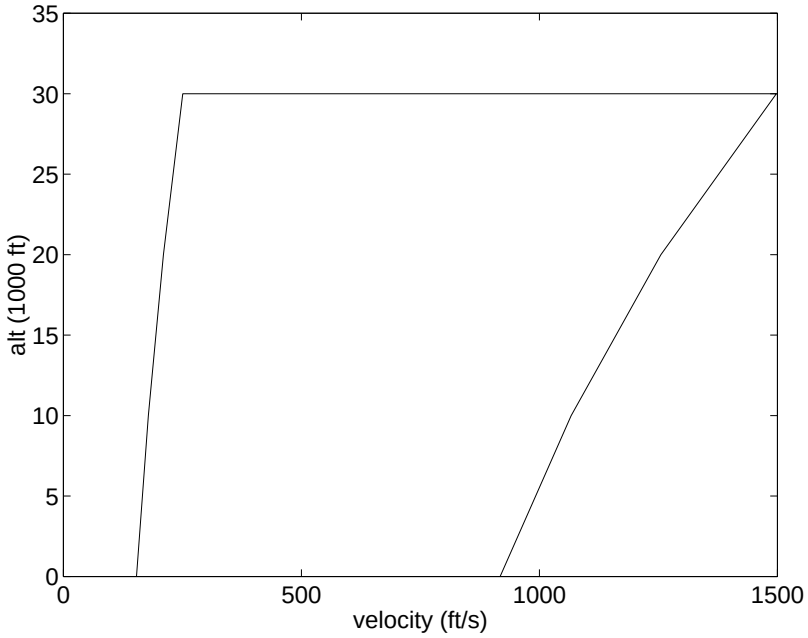

Figure 4.2: Flight Envelope

The limits of load factor for an F-16 are assumed as,

$$-3 \leq n \leq 9 \tag{4.9}$$

Now we have to find the combinations of α and δ_e such that the load factor is within the constraints given by equation (4.9). The steps needed to find the feasible α and δ_e range for different dynamic pressures is given below.

1. Fix a dynamic pressure, $\bar{q}$.
2. Vary α from -10 *deg* to 40 *deg*.
3. For each α, vary δ_e from -25 *deg* to 25 *deg*.
4. Find C_L using the polynomial equation from Appendix C.
5. Find n using equation (4.8).

6. Check if n is within the limits (equation (4.9)). If it is within the limits then combination of α, δ_e, and $\bar{q}$ is feasible and can be used to define a vector θ.

7. The parameter vector θ is then used to find the stability derivative vector ρ.

Three values of dynamic pressure are used to find the feasible operating points. Table 4.1 shows the range of feasible operating points at different dynamic pressures. At maximum dynamic pressure, the full range of α is not available because the load factor exceeds its maximum limit at high angles-of-attack.

Table 4.1: Feasible Range of Alpha and Elevator Deflection

$\bar{q}$ (psf)	α (deg)	δ_e (deg)
28	$-10 \leq \alpha \leq 40$	$-25 \leq \delta_e \leq 25$
200	$-10 \leq \alpha \leq 40$	$-25 \leq \delta_e \leq 25$
1000	$-2 \leq \alpha \leq 8$	$-25 \leq \delta_e \leq 25$

The center of gravity location is required to find the stability derivatives which, in turn, defines the vector ρ. Thus, we can choose $\bar{q}$, α, δ_e, and $\bar{x}_{cg}$ as independent variables to define a region of feasible operating conditions. Figure 4.3 shows the operating points in stability derivative space for $\bar{x}_{cg} = 0.3$. There are a total of fifty points distributed between the minimum and the maximum dynamic pressures. That is, the vector of stability derivatives ρ is defined as

$$\rho_i = \begin{bmatrix} Z_\alpha(i) \\ M_\alpha(i) \\ M_q(i) \end{bmatrix} \qquad i = 1, \ldots, 50. \tag{4.10}$$

Note that the fourth element M_{δ_e} of the state-space model has not been included in the stability derivative vector ρ. This choice was made because a three dimensional stability derivative vector and the corresponding convex hulls can be illustrated readily in pictures.

A four dimensional case would be difficult to visualize. Also, shapes of the convex hull obtained using the two modeling methods are easier to compare if ρ is three dimensional. In addition, a four dimensional vector has a tendency to become computationally infeasible when designing the controller using the current implementation of the *LMI toolbox*^{TM} . Thus, a three dimensional case facilitates easy understanding, visualization, and interpretation of the results. The variations in the derivative M_{δ_e} are no longer considered and the matrix B is normalized such that,

$$B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \tag{4.11}$$

for all operating points. A constant B matrix for all operating points ensures that all plant models are controllable.

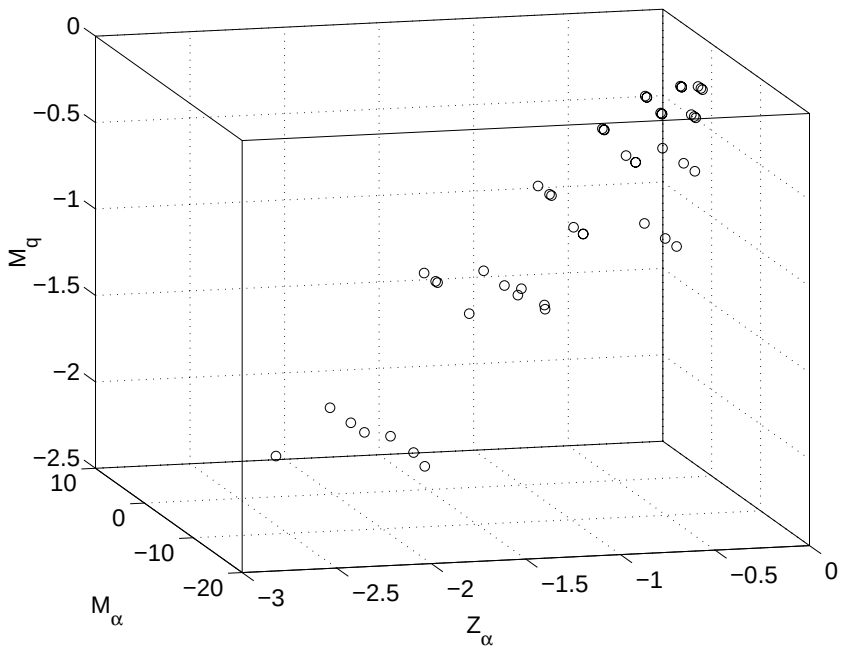

Figure 4.3: Fifty Operating Points in Stability Derivative Space (fixed c.g.)

We now have fifty three-dimensional stability derivative vectors, each of which represents an operating point. The objective is to apply the two modeling methods, form LPV models, design controllers, and compare the designs. The fifty vertices in Figure 4.3 are used to

form the convex hulls using the two modeling approaches. Recall that in the bounding box method, we find the lower and upper bounds of the three elements of ρ, and use the $m = 2^3$ permutations to find the vertices that will form the corners of the bounding box. The small hull method identifies a list of eighteen points in the stability derivative space that form the vertices of a smallest volume convex hull. Note that these points are a subset of the fifty points obtained from the assumed operating conditions. Figure 4.4 shows the faces of the small convex hull encompassed by the bounding box. It is evident from the figure that the corners of the bounding box are far away from the vertices of the small hull. Therefore, it seems intuitive that a design based on the vertices of the small hull will be less conservative than a design based on the bounding box.

At this point we have obtained two LPV design models with the following properties.

1. *LPV Model* 1 (Bounding box model) : 8 vertices forming a convex hull of rectangular shape. Each vertex is formed by taking a permutation of lower and upper bounds of each element in the vector ρ. The plant models formed by these vertices do not necessarily represent actual operating conditions. All of the assumed fifty operating points lie inside the hull.

2. *LPV Model* 2 (Small hull model) : 18 vertices forming the small convex hull. Each vertex represents an actual operating point. All of the assumed fifty operating points lie inside the hull.

Note that a point in the stability derivative space (Figure 4.3) represents a linear time-invariant plant model and corresponds to a unique combination of operating conditions given by $\bar{q}$, α, δ_e and $\bar{x}_{cg}$. Henceforth, the points in the ρ space will be referred to as 'plant models'. Combinations of operating conditions forming a plant model can be feasible or infeasible. For instance, the fifty plant models in Figure 4.3 are formed by feasible operating conditions. But it is very likely that the eight plant models that form the vertices of the bounding box do not correspond to feasible operating conditions.

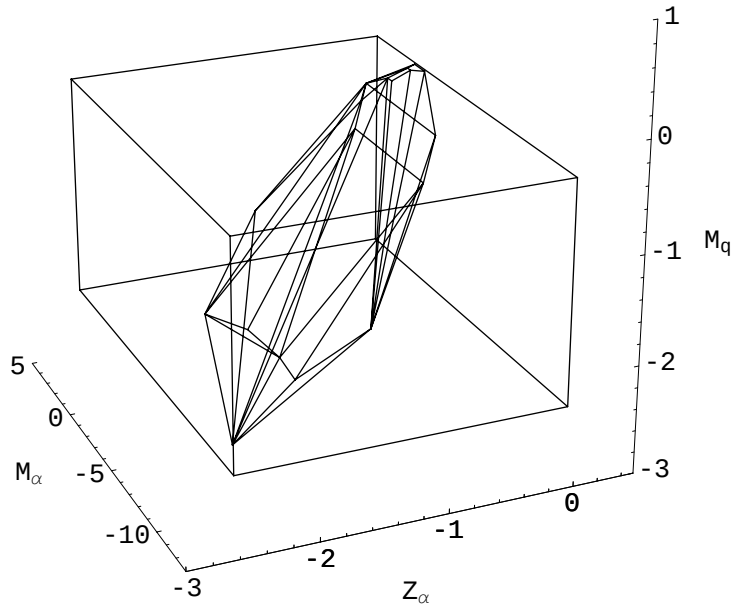

Figure 4.4: Small Hull and the Bounding Box (fixed c.g.)

4.3 Physical Significance of Points in Stability Derivative Space

In this section, we try to interpret the physical significance of points in the stability derivative vector (ρ) space and see if characteristics of a plant model can be identified by its location in ρ space. The characteristic equation for the second-order design model can be written in terms of the stability derivatives. The characteristic equation can then be solved to find the open-loop eigenvalue locations as a function of stability derivatives.

The determinant of the state transition matrix gives the characteristic equation for a plant model:

$$
\begin{aligned}
det(sI - A) &= \begin{vmatrix} s - Z_\alpha & -1 \\ -M_\alpha & s - M_q \end{vmatrix} && (4.12) \\
&= s^2 - (Z_\alpha + M_q)s + (M_q Z_\alpha - M_\alpha) = 0 && (4.13)
\end{aligned}
$$

The characteristic equation for a classical second-order system is given as:

$$s^2 + 2\zeta\omega_n s + {\omega_n}^2 = 0 \tag{4.14}$$

From equation (4.13) and (4.14), damping and natural frequency of the short-period mode dynamics can be written in terms of the stability derivatives:

$$
\begin{aligned}
\omega_n &= \sqrt{M_q Z_\alpha - M_\alpha} && (4.15) \\
\zeta &= \frac{-(Z_\alpha + M_q)}{2\omega_n} && (4.16)
\end{aligned}
$$

Given a point in the stability derivatives space, equation (4.15) and (4.16) can be used to predict the stability of the associated plant model. The most unstable plant model would occur where the ${\omega_n}^2$ term in equation (4.14) is most negative. For example, consider the point

$$(Z_{\alpha max}, M_{\alpha max}, M_{q_{max}}) = (-0.057, 4.49, -0.232) \tag{4.17}$$

which is one of the corners in the bounding box. It should represent the most unstable plant model, as the ${\omega_n}^2$ term will be most negative in this case. Similar observations can be made for the vertices that constitute the convex hull from the small hull method.

Another way of visualizing the characteristics of all the plant models is to plot their eigenvalues in the complex plane. Figure 4.5 shows the poles of all the fifty plant models ('x' symbols), the poles of plant models that form the eighteen vertices of the small hull ('o' symbols) and the poles of plant models that form the 8 corners of the bounding box ('□'

symbols). Note that the 'o' symbols overlap the 'x' symbols confirming that the eighteen vertices of the small hull are among the fifty plant models formed from actual operating conditions. In contrast, poles of the plant models from the bounding box method, shown by the '□' symbols, seem to lie at the periphery or at the edges of the space covered by the roots. In a sense, they seem to "encompass" all poles from other plant models. This result is consistent with Figure 4.4 where the rectangular bounding box encompasses the small hull from the proposed method. If we were to draw the poles of plant models obtained from the two modeling methods only, it becomes immediately evident that the poles from the eighteen vertices of the small hull are a much better representation of the true operating conditions. The poles of the vertices of the bounding box define a wider range than is necessary and thus controller designs based on this approach are inherently more conservative.

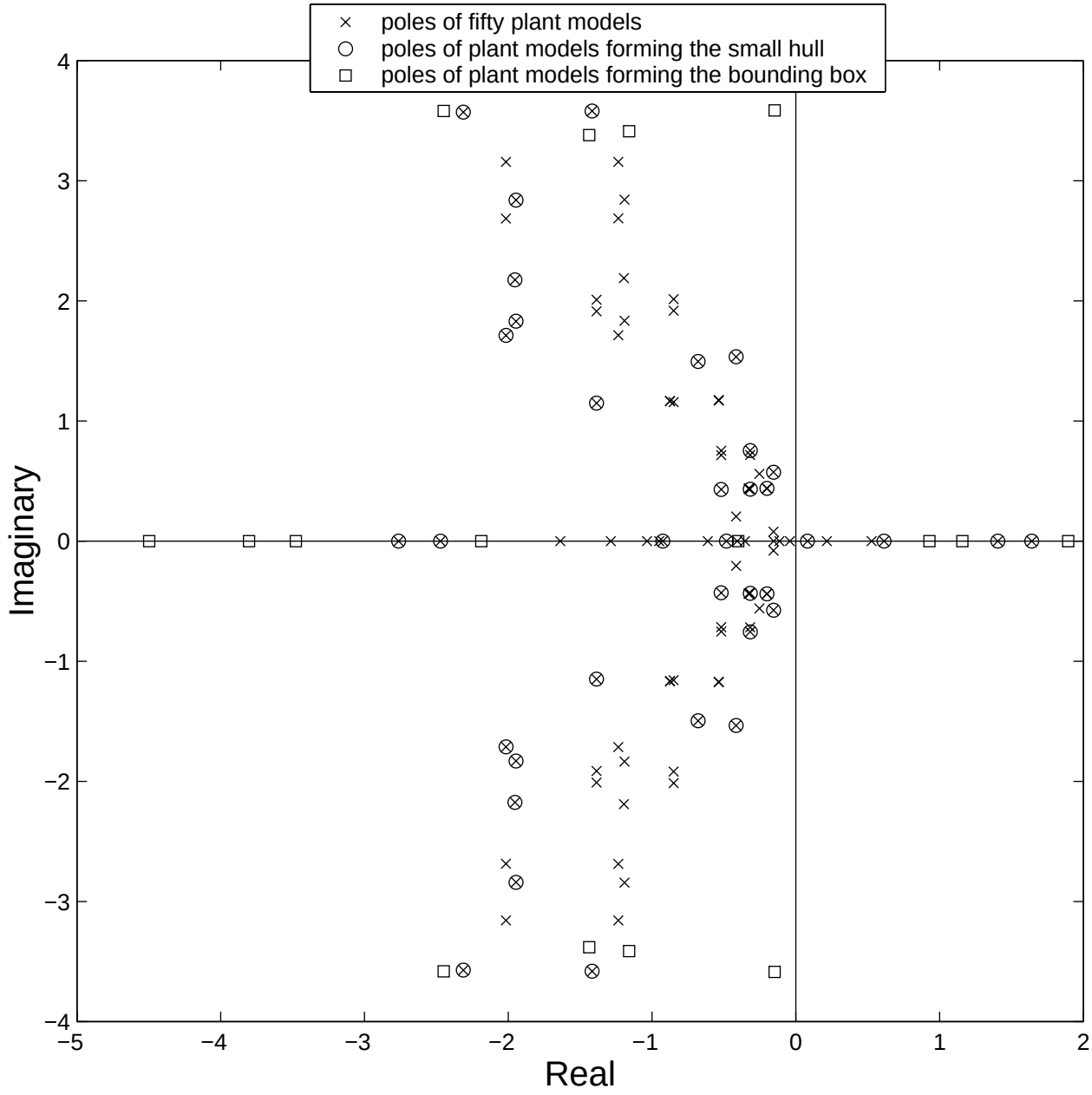

Figure 4.5: Pole Locations (fixed c.g.)

4.4 Controller Design

The LMI problem formulation (as described in Chapter 3) is used to solve for a single LQR controller suitable for the LPV models formed by the two modeling approaches. In each LPV model, the plant models represented by the vertices are solved simultaneously to find a single controller for the entire set. The weighting matrices on the states and the controls are selected as

$$Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{4.18}$$

$$R = 1000 \tag{4.19}$$

respectively. Table 4.2 shows the gain matrix obtained from LMI solutions for both modeling approaches. Although the number of vertices to solve is more in the small hull technique, it is observed that controller obtained from this model is less conservative than the controller obtained from the bounding box model. This result is evidenced by the larger magnitude gain values for the controller from the bounding box model.

Eigenvalues of the LMI inequality

$$A_i P + P A_i^T - B_i Y - Y^T B_i^T + I < 0$$

are used to identify the 'worst' plant model in both the LPV models. The state-space matrices of the 'worst' plant model and the corresponding eigenvalues in each case is shown in the Table 4.3. Note that even though the right-half-plane pole of the 'worst' plant model from LPV model 1 (bounding box method) is less positive than the right-half-plane pole for the 'worst' plant model from LPV model 2 (small hull method), the H_2 norm for the former is larger than that for the latter (Refer Table 4.2). This result confirms that the location of poles alone does not determine the 'worst' plant model. Since the weighting matrices remain the same for all plants, we can use the magnitude of H_2 norm as a measure of 'worstness' or difficulty in controlling a plant model. Going back to the discussion about how the

LMI method behaves, it is the H_2 norm of individual plant model that most influences the controller characteristics that result from LMI formulation.

Table 4.2: LMI Solution for the Two LPV Modeling Methods (fixed c.g.)

Modeling Method	Gain matrix (K_{lmi})	H_2 norm
bounding box method	[11.88 4.06]	177.58
small hull method	[9.30 3.51]	144.72

Table 4.3: Characteristics of Worst Plant Model for the Two Modeling Method (fixed c.g.)

Worst Plant Model from	State Matrix	Eigenvalues
bounding box method	$\begin{pmatrix} -0.057 & 1.000 \\ 4.159 & -2.262 \end{pmatrix}$	(1.159, -3.478)
small hull method	$\begin{pmatrix} -0.152 & 1.000 \\ 4.159 & -0.680 \end{pmatrix}$	(1.640, -2.472)

Recall that the preceding results are for a fixed center of gravity location. Another design example was formed where a range of three different center of gravity locations was used.

$$\bar{x}_{cg} = [\ 0.3 \quad 0.35 \quad 0.4\ ] \tag{4.20}$$

Defining fifty operating points for each center of gravity location yields 150 points in the stability derivative space. These are shown in Figure 4.6. The small hull method identifies twenty one plant models that form the vertices of the small hull shown in Figure 4.7. Comparing Figure 4.4 and Figure 4.7, we notice that inclusion of center of gravity location as one of the parameter variables had the most effect on M_α (see equation (4.5)). Also note that although the number of operating points was increased from 50 to 150, the number of vertices that formed the convex hull in the small hull method increased only marginally, from 18 to 21. It again confirms the argument that it is the location of points in the space

that defines the shape of convex hull. The number of vertices from the small hull approach may not always be more than the number of vertices from the bounding box approach.

Figure 4.8 shows the poles of all the 150 plant models ('x' symbols), the poles of plant models that form the vertices of small hull ('o' symbols) and the poles of plant models that form the vertices of bounding box ('□' symbols). Again the poles of plant models from the bounding box approach seem to be in the periphery of the root variations while the poles of plant models from the small hull approach appear more representative of the actual pole locations of the system.

The results of the control design for the case of varying center-of-gravity are shown in Table 4.4. Again, the gain matrix (K_{lmi}) obtained using the bounding box approach is much more conservative than that obtained from the small hull approach. The H_2 norm of the 'worst' plant model from the bounding box approach is higher than the H_2 norm of the 'worst' plant model from the small hull approach. This result also confirms that the controller design based on the bounding box approach is more conservative. The state-space matrices of the 'worst' plant model and the corresponding eigenvalues in each case is shown in the Table 4.5. Note that although the right-half-plane pole for the 'worst' plant model from both the modeling approaches are close to one another, the H_2 norm of the 'worst' plant model from the bounding box method is much higher than the H_2 norm of the 'worst' plant model from the small hull method. This observation again shows that there is more than just the pole locations of plant models that influence the control design obtained using LMI formulation.

These two design examples confirm the hypothesis that a better and less conservative controller design can be achieved if the LPV modeling is conducted based on the small hull approach.

Table 4.4: Gain Matrix from the Two Modeling Methods (varying c.g.)

Modeling Method	Gain matrix (K_{lmi})	H_2 norm
bounding box method	[43.36 7.97]	524.57
small hull method	[27.3 6.32]	404.86

Table 4.5: Characteristics of Worst Plants for the Two Modeling Method (varying c.g.)

Worst Plant Model from	State Matrix	Eigenvalues
bounding box method	$\begin{pmatrix} -0.057 & 1.000 \\ 22.684 & -2.262 \end{pmatrix}$	(3.73, -6.05)
small hull method	$\begin{pmatrix} -1.283 & 1.000 \\ 22.684 & -1.186 \end{pmatrix}$	(3.53, -5.99)

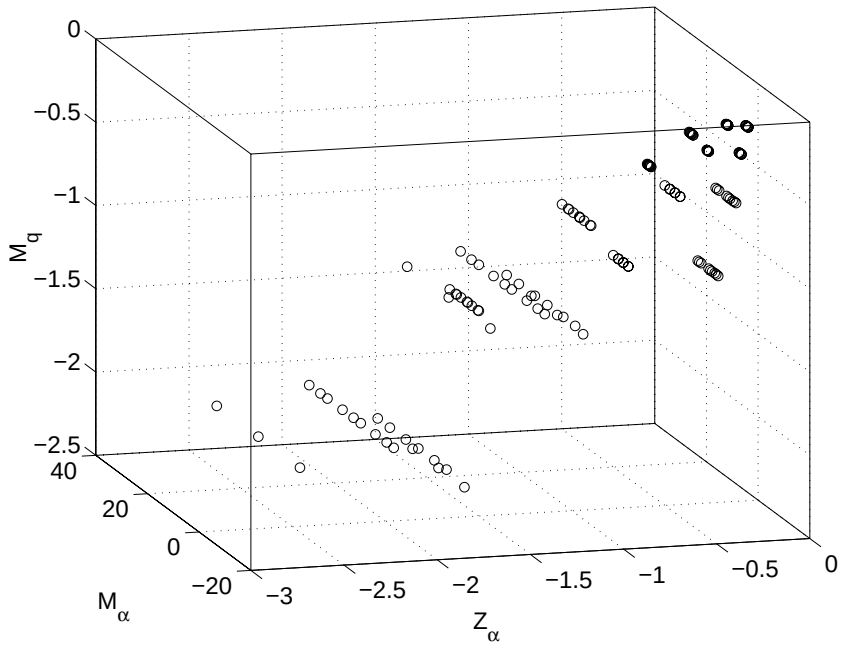

Figure 4.6: Operating Points in Stability Derivative Space (varying c.g.)

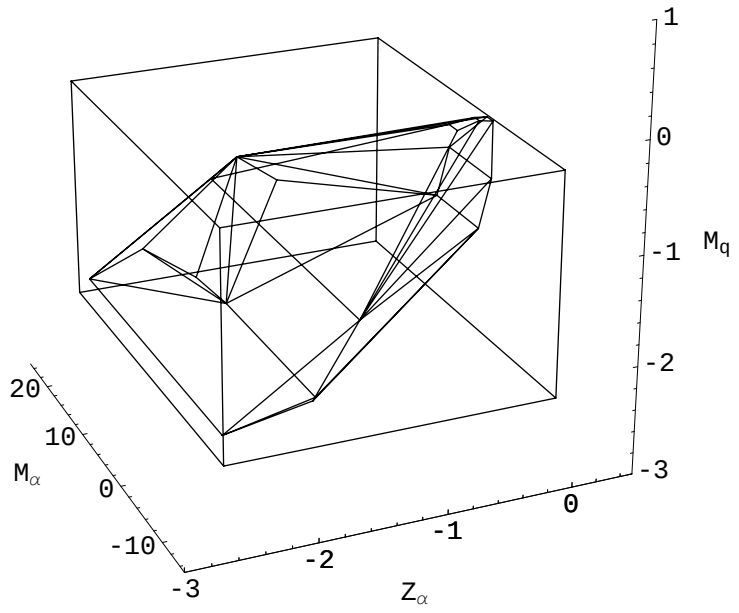

Figure 4.7: Small Hull and the Bounding Box (varying c.g.)

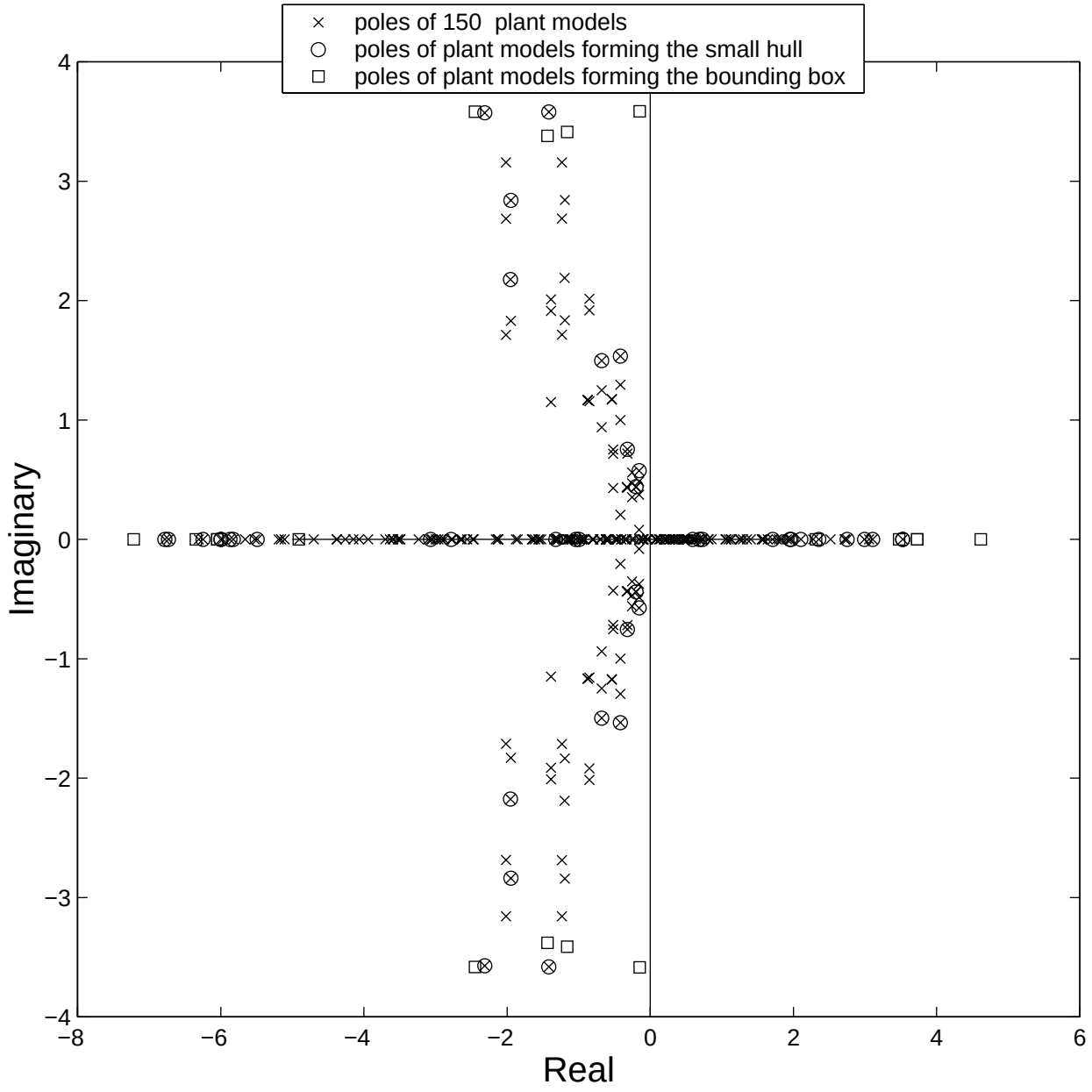

Figure 4.8: Pole Locations (varying c.g.)

Chapter 5

Robustness Analysis

Control system robustness is defined as the ability to maintain satisfactory stability or performance characteristics in the presence of system parameter variations and uncertainties. A robustness analysis will be conducted as an additional comparison of performance of the controllers designed using the two modeling methods. This analysis is also necessary to ensure that the controllers obtained through the LMI method will indeed work for all possible variations in the parameters. The next section explains in detail the robustness analysis technique used for our purpose.

5.1 Probability of Instability

There are standard measures that can be used for robustness analysis like static gain and phase margins or singular value analysis. But these methods are deterministic. Stengel has pointed out that deterministic metrics can be conservative and/or difficult to determine for systems with many uncertain parameters. He has established a scalar measure of robustness called the *probability of instability*, which will be applied for the current analysis. This concept was first introduced by Stengel [12], and is further refined in [13] and [14]. The method determines the stochastic robustness of a linear time-invariant system by finding

the probability distribution of closed-loop eigenvalues, given the statistics of the varying parameters in the plant dynamic model. The probability that all of the eigenvalues lie in the open left-half-plane is the scalar measure of robustness.

Consider a linear, time-invariant (LTI) system subject to state-feedback control:

$$\dot{x} = Ax + Bu \tag{5.1}$$

$$y = Cx \tag{5.2}$$

$$u = -Kx \tag{5.3}$$

The vectors x, y, and u are state, output, and control vectors of dimensions n, m, and r respectively. The state-space matrices A, B, and C could be functions of a parameter θ as in an LPV model. The n eigenvalues of the matrix $[A - BK]$ determine closed-loop stability of the system. The roots of the following matrix determinant

$$|sI - (A - BK)| = 0 \tag{5.4}$$

will be the eigenvalues or the characteristic roots of the closed-loop system.

While the explicit relationship between the parameter vector θ and the closed-loop eigenvalues is complicated, the probability of instability of the closed-loop system can be found from repeated eigenvalue calculation. Note that the probability of stability plus the probability of instability is equal to unity. The stochastic robustness of the feedback system is computed by Monte Carlo simulation. The probability of instability from Monte Carlo evaluation is given by

$$Pr(instability) = \lim_{J \to \infty} \frac{N}{J} \tag{5.5}$$

where J is the number of times the eigenvalues are evaluated. At each trial, the elements of the parameter vector θ are specified by a random-number generator shaped by the probability density function for θ. The larger the value of J, the more precise the estimate of the

probability of instability becomes. In equation (5.5), N is the number of cases for which at least one eigenvalue has a positive real part. Note however, that for $J < \infty$, the probability of instability resulting from Monte Carlo evaluation is only an estimate.

Stengel [12] presented a simple example to illustrate the concepts of stochastic robustness and the probability of instability. Consider a classical second order system with characteristic equation

$$s^2 + 2\zeta\omega_n s + {\omega_n}^2 = 0 \tag{5.6}$$

The mean or nominal values of damping ratio (ζ) and natural frequency (ω_n) are 0.707 and 1, respectively. It is further assumed that each is a Gaussian-distributed random variable with a standard deviation of 0.2. If both damping ratio and natural frequency are uncertain and uncorrelated, the parameter vector will be $\theta = [\zeta, \omega_n]^T$. A Monte Carlo simulation will assign Gaussian-distributed random values to the vector θ. The possible root locations for the simulation will be obtained from equation (5.6) and will form "clouds" surrounding the nominal roots ($s = -0.707 \pm j0.707$). Figure 5.1 is based on 5000 Monte Carlo simulations and shows the scatter plot of the eigenvalues. The probability of instability estimate for the simulated variations in θ was 0.0004 (i.e. $N = 2$ was found after $J = 5000$ trials).

Figure 5.2 shows the probability density of the roots above the root plane. This calculation is done by dividing the complex plane into subspaces (or "bins") and counting the number of roots in each bin as a sampled estimate of the root density. The density of the roots depicts the likelihood that the eigenvalues vary from their nominal value for variations in damping ratio and natural frequency. This idea can be further exploited to find the most probable eigenvalues for given variations in parameter vector, which in turn can be used to identify the most likely system configuration. This concept is further explored in subsequent sections.

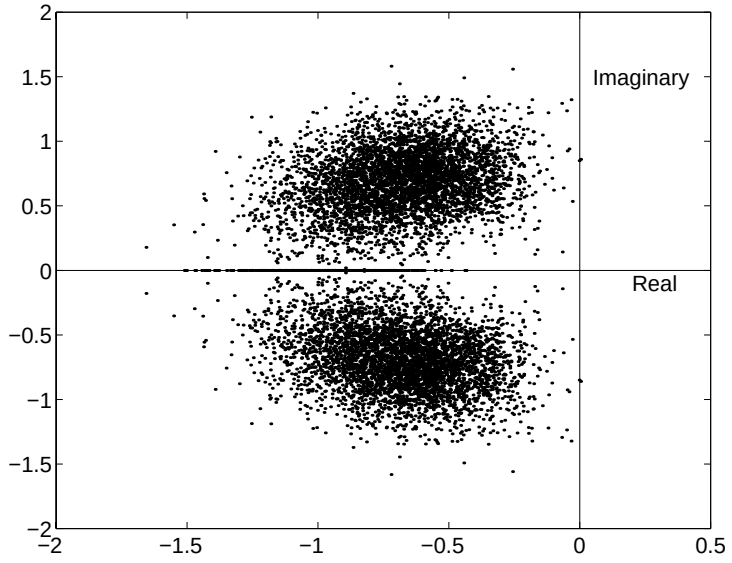

Figure 5.1: Closed-Loop Eigenvalues of a Second-Order Example

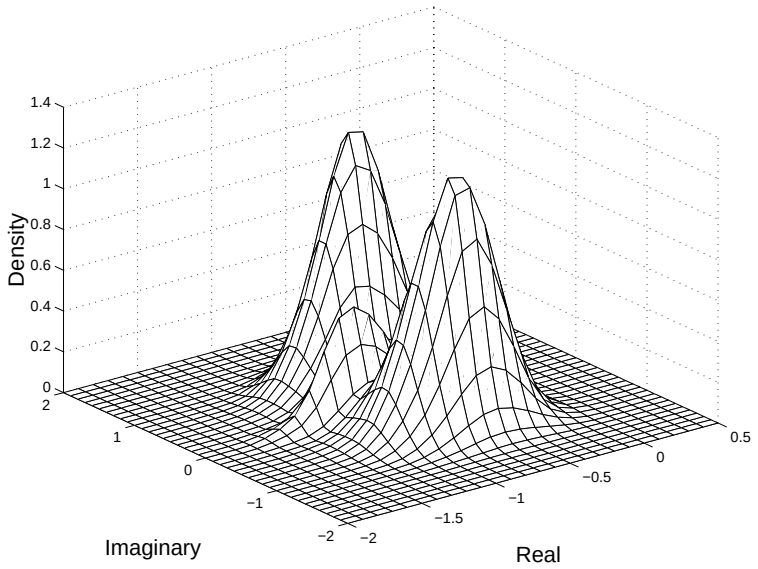

Figure 5.2: Root Density Representation of the Closed-Loop Eigenvalues

5.2 Stability Robustness for the Design Example

Robustness analysis for the F-16 model based upon the probability of instability concept is carried out in this section. The objective is to perform a robustness analysis for the controller designs obtained using the bounding box and the small hull approaches. First, the controller designs for fixed center of gravity location are considered. Recall that the parameter vector θ is

$$\theta = [vel, alt, \alpha, \delta_e] \tag{5.7}$$

where vel, alt, α and δ_e are assumed to have a uniform probability distribution. Only those combinations of vel, alt, α and δ_e are chosen which satisfy equations (5.8) and (5.9). Recall that these equations must be satisfied for feasibility condition.

$$\bar{q}SC_L \approx nW \tag{5.8}$$

$$-3 \leq n \leq 9 \tag{5.9}$$

Thus, the elements in vector θ are not necessarily independent. The center-of-gravity location is fixed to $\bar{x}_{cg} = 0.3$.

Figure 5.3 shows the uniform distribution of points within the flight envelope but, as stated earlier, only those points are chosen which satisfy the feasibility condition. This sampling gave feasible operating conditions for the Monte Carlo simulations. Starting with 3000 random combinations of points in the parameter vector space, 1750 points were found to be feasible. The parameter points are plotted in the stability derivative space (Figure 5.4) by finding the stability derivative vector ρ for each vector θ. Similarly, the pole locations for each plant model are shown in Figure 5.5. These figures cover the entire range of flight conditions and the corresponding range of short-period pole locations.

The probability density of the resulting eigenvalues is shown in Figure 5.6. The peaks in the figure reveal the regions of highest probability of occurrence of eigenvalues. These peaks can

be used to find the state-space configuration of the plant model that yields the most likely poles. The corresponding system plant model will be referred to as the maximum likelihood plant model. In the current example, the maximum likelihood plant model is found by identifying the most probable eigenvalue pair. An LQR controller is then designed using the same weighting matrix Q and R as used by the controller designs from the two modeling approaches. A robustness analysis is then carried for all the three controller designs.

The number of cases that resulted in right-half-plane poles after the state-feedback controller was applied, divided by the total number of trials is the probability of instability associated with the controller. Table 5.2 shows the results of the analysis for the three different design methods. The results conform to our expectation. The LMI controller designs are conservative and lead to stable systems for the entire range of flight conditions. This result is not the case for the controller designed for the maximum likelihood plant model. It has a probability of instability of 0.052.

Table 5.1 shows the state-matrix and the eigenvalues of the maximum likelihood plant model. It has stable open-loop poles and the resulting LQR controller is given in Table 5.2, which is small in magnitude as compared to the controllers from the LPV modeling method.

Similar robustness checks were carried out for LMI controller designs with varied locations of center of gravity, $\bar{x}_{cg}$. Monte Carlo simulations were carried out and the probability of instability was evaluated. Figure 5.7 shows the all the simulation points in stability derivative space. As discussed earlier, the inclusion of center-of-gravity as one of the varying parameters had the most significant effect on M_α (compare Figure 5.4 and Figure 5.7). Figure 5.8 shows the roots obtained from the simulation. Figure 5.9 shows the probability density of the poles. Note that the probability of real poles is higher than the probability of complex conjugate poles. This result occurs because the short-period roots become real as the c.g. location moves aft, $0.3 < \bar{x}_{cg} < 0.4$.

The results of stability robustness analysis are as expected and are shown in Table 5.4. The probability of instability for the controller from the small hull approach is zero even though the controller is far less conservative that the controller from the bounding box approach

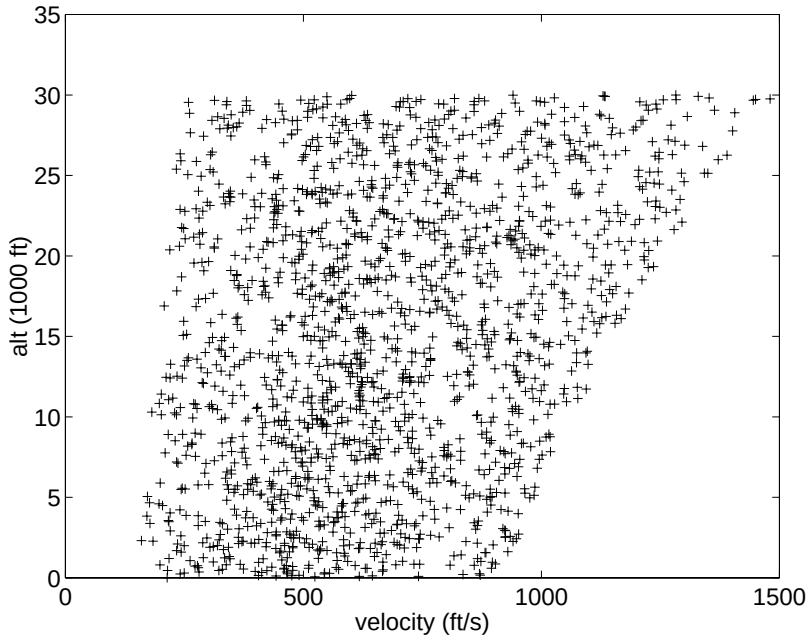

Figure 5.3: Uniform Distribution of Points in the Flight Envelope

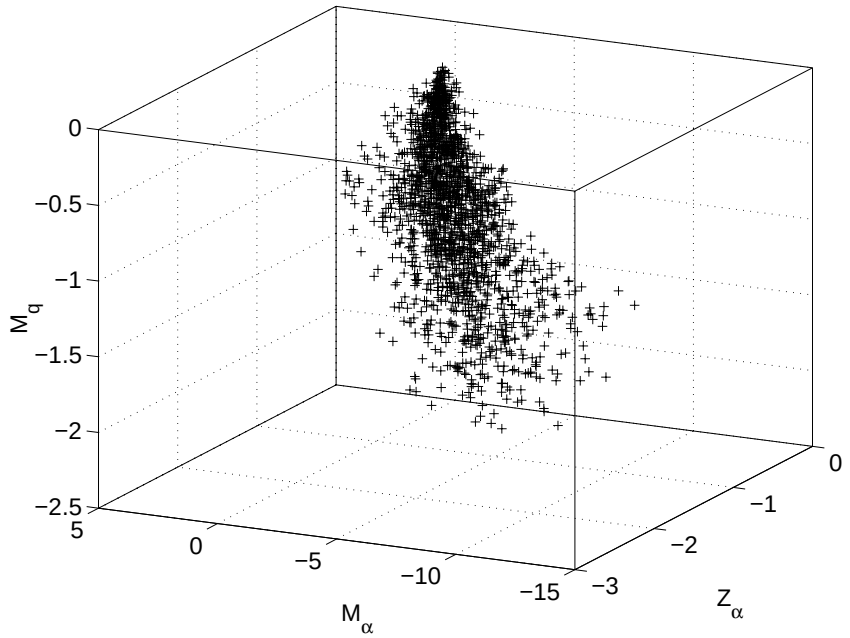

Figure 5.4: Points in the Stability Derivative Space (fixed c.g.)

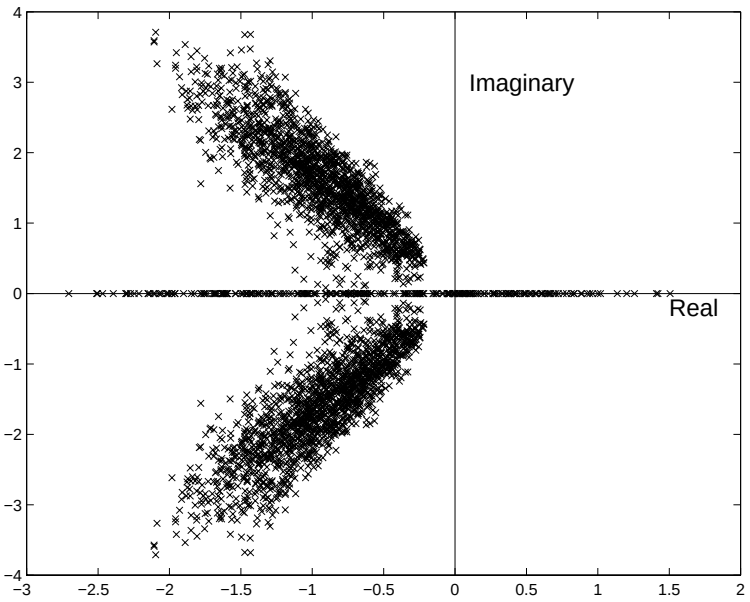

Figure 5.5: Open-Loop Pole Locations(fixed c.g.)

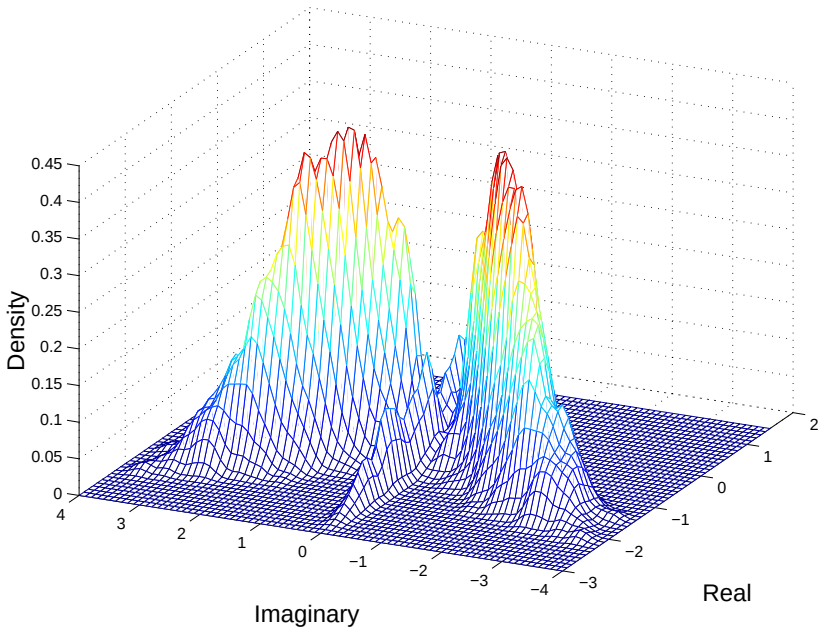

Figure 5.6: The Root Density for Open-Loop Model (fixed c.g.)

Table 5.1: Characteristics of Maximum Likelihood Plant Model (fixed c.g.)

State Matrix	Open-Loop Eigenvalues
$\begin{pmatrix} -0.432 & 1.0 \\ -1.797 & -0.957 \end{pmatrix}$	(-0.695±j1.315)

Table 5.2: Robustness Results (fixed c.g.)

Control Design	Gain Matrix	Probability of Instability
from bounding box approach	[11.88 4.06]	0
from convex hull approach	[9.30 3.51]	0
from maximum likelihood model	[0.0294 0.5528]$\times 10^{-3}$	0.052

(Refer Chapter 4). The probability of instability for the controller design based on the maximum likelihood plant model is close to 0.3.

Table 5.3 shows the state-matrix and the eigenvalues of the maximum likelihood plant model for the varying center of gravity case. It has stable open-loop poles and the resulting LQR controller is given in Table 5.4, which is small in magnitude as compared to the controllers from the LPV modeling method.

The robustness analysis conducted for the controllers in the two cases (center-of-gravity fixed and varying) reveal some interesting results. The controller designs based on the LPV modeling approaches (Chapter 4) produce zero probability of instability. This result was expected as the controllers had been specifically designed to achieve this objective. It is evident that the controller design based on the small hull approach produces a less conservative design and yet maintains a zero probability of instability for variation in the parameter vector. An important implication of comparing the robustness of the LMI controllers to the robustness

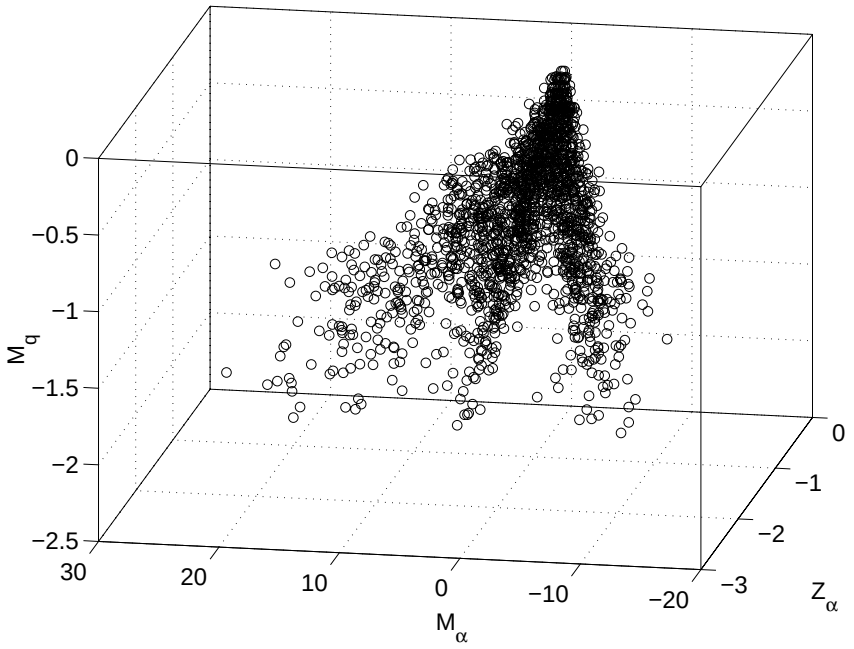

Figure 5.7: Points in the Stability Derivative Space (varying c.g.)

of controller based on the maximum likelihood plant model is that it is always the worst plant or configuration of a system that drives the controller design. Note that although the system has a high probability of being in a state given by the maximum likelihood plant model, yet the resulting controller for this plant model had a high probability of instability.

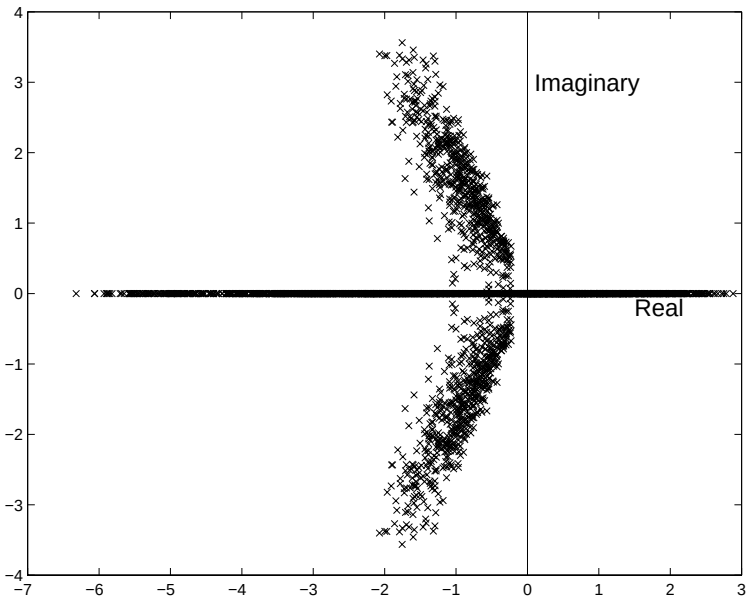

Figure 5.8: Open-Loop Pole Locations (varying c.g.)

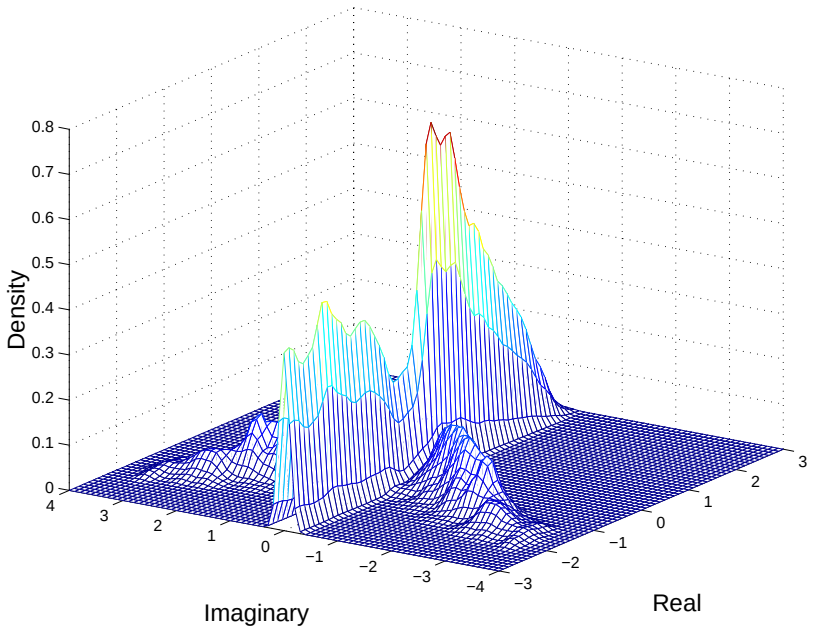

Figure 5.9: The Root Density of Open-Loop Model (varying c.g.)

Table 5.3: Characteristics of Maximum Likelihood Plant Model (varying c.g.)

State Matrix	Open-Loop Eigenvalue
$\begin{pmatrix} -1.118 & 1.0 \\ 2.514 & -1.257 \end{pmatrix}$	(0.4 -2.77)

Table 5.4: Robustness Results (varying c.g.)

Control Design	Gain Matrix	Probability of Instability
from bounding box approach	[43.36 7.97]	0
from convex hull approach	[27.3 6.32]	0
from maximum likelihood plant	[1.324 0.799]	0.293

Chapter 6

Summary and Conclusion

The objective of this research was to propose and investigate a new LPV modeling technique for improved controller design. The modeling technique proposed in this work is an extension of the existing affine LPV modeling technique called the 'bounding box' approach. The aim was to show that the new modeling process, called the small hull approach, will produce LPV models that more closely represent the actual dynamics of a system and will produce less conservative controllers. These ideas were illustrated by describing the two modeling processes in detail and comparing their properties using simple, intuitive examples. A design example using non-linear F-16 short-period dynamics was used to implement the two modeling techniques and compare the controller designs.

Model Development

Affine LPV modeling was discussed in detail in light of the concepts of multidimensional convex hulls. The bounding box approach and the small hull approach were studied and their properties were investigated using simple, intuitive examples. It was hypothesized that the LPV model using the small hull approach will be a closer representation of the system dynamics as the plant models used to construct the LPV model were obtained from feasible operating conditions. In contrast, the bounding box approach formed a LPV model

using plant models obtained by permutations of lower and upper bounds of elements in stability derivatives. They might not represent feasible operating conditions. Thus, it was claimed that a less conservative design could be formed using the LPV model from the small hull approach. Although the small hull LPV modeling might require more plant models to characterize a system, the benefit of less conservative design makes up for increased numerical complexity.

Controller Design Framework

In Chapter 4, the controller design for solving the LPV models was described. A LMI formulation was chosen to design LQR controllers. LMIs for plant models that formed a LPV model were solved simultaneously to obtain a single controller that would work for all the plant models. While completing this exercise, important characteristics of the control design method were discovered. It was found that the single controller design obtained from the LMI formulation was most influenced by the 'worst' case plant model in the LPV model. The criterion for the 'worstness' of a plant model was determined by the LMI formulation. It was originally hypothesized that the 'worst' plant model would be the one with the most unstable pole and that this plant model would be most difficult to control. However, it was found that there are more issues such as control power and not just the location of poles that determine the 'worst' plant model.

Aircraft Modeling and Design Implementation

An F-16 design example was used to implement the two techniques of LPV modeling and design controllers based on the LMI formulation. It was found that the controller design based on the LPV model using the small hull method is less conservative than the controller based on the LPV model using the bounding box approach, thus confirming our hypothesis.

Robustness Analysis

To check the stability robustness of the controller design for the two LPV modeling approaches, a scalar measure of stochastic robustness called the probability of instability was investigated. It was found that the controller obtained using the small hull modeling technique has a zero probability of instability. This is an encouraging result as the controller is less conservative and yet has good robustness qualities. A conservative controller design produced by the bounding box method might be robust enough to work for extreme conditions but it may not satisfy nominal performance criterion at the actual operating points. This unwanted characteristic is less likely to occur in a less conservative controller design.

A maximum likelihood plant model was found for two different examples. The robustness of a controller designed for this plant model was compared to the robustness of the controllers based on LPV modeling techniques. The controller based on the maximum likelihood plant model resulted in a high probability of instability and is definitely not the most effective plant model to design a controller with. In the LMI formulation for controller design it is the 'worst' plant model that drives the controller design; thus producing zero probability of instability.

Future Work

It has been shown that the proposed LPV modeling technique is a better representation of the true dynamics of a system and thus can produce a controller design that can yield better performance while maintaining acceptable robustness properties. The control design structure used in this work was intended to bring forth the properties of the LPV modeling. A promising area of future work would be to use the small hull LPV modeling technique to design LPV controllers and compare its performance to a LPV controller based on the bounding box LPV model.

The F-16 design example used to implement the LPV models and design controller was a second-order system to keep the analysis simple and numerically feasible. More complicated

and higher dimensional models need to be considered.

The LMI formulation was used to come up with a single controller that would work for all the plant models in an LPV model. This technique could develop into an entirely new area of research. Although important characteristics of the LMI formulations were discovered, it was not theoretically proven. More work can be done in this area to clearly find the reasons for the way the LMIs behave.

Appendix A

Steps for Finding the H_2 Norm

Given A and B for plant, define an output vector z such that such that

$$z = \begin{bmatrix} R^{\frac{1}{2}} & 0 \\ 0 & Q^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} u \\ x \end{bmatrix} \tag{A.1}$$

Therefore

$$z_1 = R^{\frac{1}{2}} u \tag{A.2}$$

$$z_2 = Q^{\frac{1}{2}} x \tag{A.3}$$

where R is a weighting matrix on the control vector, and Q is a weighting matrix on the state vector. Since we are solving a static feedback problem

$$u = -Kx \tag{A.4}$$

the closed-loop dynamics are give as

$$\dot{x} = (A - BK)x + w \tag{A.5}$$

Define X as the solution of the steady-state lyapunov equation implied by (A.5) driven by gaussian white noise w,

$$(A - BK)X + X(A - BK)^T + I = 0 \tag{A.6}$$

Note the similarity of equation (A.6) to equation (3.4). Substitute (A.4) into (A.2) to yield,

$$z = \begin{bmatrix} -R^{\frac{1}{2}}K \\ Q^{\frac{1}{2}} \end{bmatrix} x \tag{A.7}$$

The H_2 norm from w to z will be given by,

$$trace(zz^T) = trace\left(\begin{bmatrix} -R^{\frac{1}{2}}K \\ Q^{\frac{1}{2}} \end{bmatrix} X \begin{bmatrix} -K^T R^{\frac{T}{2}} & Q^{\frac{T}{2}} \end{bmatrix}\right) \tag{A.8}$$

as long as $R^{\frac{1}{2}}$ and $Q^{\frac{1}{2}}$ are symmetric,

$$trace(zz^T) = trace(R^{\frac{1}{2}}KXK^T R^{\frac{1}{2}}) + trace(Q^{\frac{1}{2}}XQ^{\frac{1}{2}}) \tag{A.9}$$

which is same as equation (3.3).

Appendix B

Convex Sets and Convex Hulls

Definitions for some of the terms used in the thesis work are given here.

Convex Set: A set S in $n-$dimensional space is called a convex set if the line segment joining any pair of points in S lies entirely in S. For example, a ball of $r > 0$ about a point in space is a convex set, but its boundary (the sphere of radius r about that point) is not convex.

Convex Hull: A convex hull of a set of points S in n-dimensional space is the intersection of all the convex sets containing S. Thus, it is the smallest convex set containing S. For example, the convex hull of three points in space is the planar triangle that has the three points as its vertices. The convex hull of a unit circle in a plane is the unit disk, and the convex hull of a cube in space is the solid cube.

Convex Polytope: A convex hull of a finite number of points is called convex polytope.

Affine Function: Let $f : X -> R^n$, which says that f is a function that maps a set X to n dimensional space. Then f is affine if X is a convex set and

$$f(ax + (1 - a)y) = af(x) + (1 - a)f(y)$$

for all x, y in x and a in $[0, 1]$.

Appendix C

F-16 Geometry and Aerodynamic Coefficients

Table 1: F-16 Geometry

Weight	$W = 20{,}500\ lb$
Moment of Inertia	$I_{yy} = 55{,}814\ slug - ft^2$
Wing Reference Area	$S = 300\ ft^2$
Mean Aerodynamic Chord	$\bar{c} = 11.32\ ft$
Reference longitudinal c.g. position	$\bar{x}_{cg_{ref}} = 0.35$

Table 2: Aerodynamic Coefficients

$$C_Z(\alpha, \beta, \delta_e) = (f_0 + f_1\alpha + f_2\alpha^2 + f_3\alpha^3 + f_4\alpha^4)(1 - \beta^2) + f_5\delta_e \quad \text{(C.1)}$$

$$C_m(\alpha, \delta_e) = m_0 + m_1\alpha + m_2\delta_e + m_3\alpha\delta_e + m_4{\delta_e}^2 + m_5\alpha^2\delta_e + m_6{\delta_e}^3 + m_7\alpha{\delta_e}^2 \quad \text{(C.2)}$$

$$C_{m_q}(\alpha) = n_0 + n_1\alpha + n_2\alpha^2 + n_3\alpha^3 + n_4\alpha^4 + n_5\alpha^5 \quad \text{(C.3)}$$

The Coefficients of the Polynomial

$f0$	= -.1378	$m0$	= -.0203	$n0$	= -5.159
$f1$	= -4.211	$m1$	= .0466	$n1$	= -3.554
$f2$	= 4.775	$m2$	= -.6012	$n2$	= -35.98
$f3$	= -10.26	$m3$	= -.0806	$n3$	= 224.73
$f4$	= 8.399	$m4$	= .0832	$n4$	= -412.09
$f5$	= -.4354	$m5$	= .5018	$n5$	= 241.17
		$m6$	= .6378		
		$m7$	= .4226		

Bibliography

[1] Shamma, J.S., and Athans, M., "Guaranteed Properties of Gain Scheduled Control for Linear Parameter Varying Plants," *Automatica,* 27, 1991, pp. 559-564.

[2] Sun, X.D., and Postlethwaite, I., "Affine LPV modeling and its use in Gain-Scheduled Helicopter Control," *UKACC International Conference on Control*, September 1998, Conference Publication No. 455, pp. 1504-1509.

[3] Apkarian, P., Gahinet, P., and Becker, G., "Self-scheduled H_∞ Control of Linear Parameter Varying Systems: a Design Example," *Automatica,* 31, 1995, No. 9, pp. 1251-1261.

[4] Terui, F., and Tsukamoto, T., "LPV Controller Design for the Lateral-Directional Motion of Re-entry Vehicle," Paper No. 99-4138, *AIAA Guidance, Navigation and Control Conference,* 1999, pp. 988-994.

[5] Wu, Fen, and Packard, Andy, "LQG Control Design for LPV Systems," *American Control Conference,* June 1995, pp. 4440-4444.

[6] Shamma, J.S., and Cloutier, J.R., "Gain-Scheduled Missile Autopilot Design Using Linear Parameter Varying Transformations," *Journal of Guidance, Control, and Dynamics,* Vol. 16, No. 2, 1993, pp. 256-263.

[7] Boyd, S., Feron, E., and Balakrishnan, V., "Numerical Methods for H_2 related Problems," *Automatic Control Conference,* 1992, pp. 2921-2922.

[8] Boyd, S., Balakrishnan, V., Feron, E. and El Ghaoui L., "Control System Analysis and Synthesis via Linear Matrix Inequalities," *American Control Conference*, 1993, pp. 2147-2154.

[9] Morelli, E.A., "Global Nonlinear Parameter Modeling with Application to F-16 Aerodynamics," *American Control Conference,* 1998, June 24-26, pp. 1-5.

[10] Barber, C. B., Dobkin, D.P., and Huhdanpaa, H.T., "The Quickhull Algorithm for Convex Hulls," *ACM Trans. on Mathematical Software,* Vol. 22, No. 4, Dec. 1996, pp. 469-483.

[11] Brogan, W.L., *Modern Control Theory*, Third Edition, Prentice Hall, New Jersey, 1991.

[12] Stengel, R.F., and Ray, L.R., "Stochastic Robustness of Linear Time-Invariant Control Systems," *IEEE Transactions on Automatic Control,* Vol. 36, No. 1, January 1991, pp. 82-87.

[13] Ray, L.R., and Stengel, R.F., "A Monte Carlo Approach to the Analysis of Control System Robustness," *Automatica,* Vol. 29, No. 1, 1993, pp. 229-236.

[14] Ray, L.R., and Stengel, R.F., "Application of Stochastic Robustness to Aircraft Control Systems," *Journal of Guidance, Control, and Dynamics,* Vol. 14, No. 6, Nov.-Dec. 1991, pp. 1251-1259.

Vita

Abhishek Kumar was born on September 28, 1976 in Bihar, India. He did his high school from Mount Assisi School, Bhagalpur and Delhi Public School, Delhi. He received his bachelor degree in Aerospace Engineering in 1998 from Indian Institute of Technology, Kanpur. Then he came to Virginia Polytechnic Institute and State University, Virginia, USA where he received his Masters of Science in Aerospace Engineering in 2000. His area of concentration for masters thesis was Dynamics and Control with emphasis on modeling.