DISSERTATION

SKEW μ WITH A POWER SYSTEMS APPLICATION

Submitted by

Rodney M. Holland P.E.

Department of Electrical and Computer Engineering

In partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2003

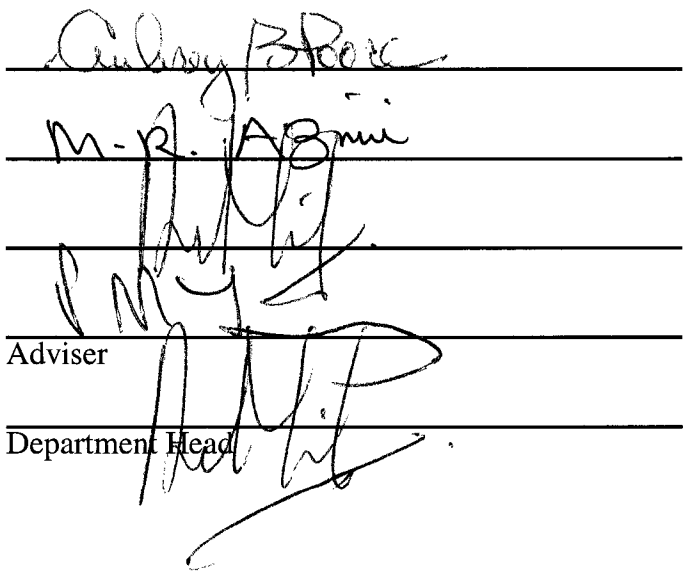
ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

COLORADO STATE UNIVERSITY

December 2, 2002

WE HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER OUR SUPERVISION BY RODNEY M. HOLLAND P.E. ENTITLED **SKEW μ WITH A POWER SYSTEMS APPLICATION** BE ACCEPTED AS FULFILLING IN PART REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY.

Committee on Graduate Work



Adviser

Department Head

ABSTRACT OF DISSERTATION

SKEW μ WITH A POWER SYSTEMS APPLICATION

Exploitation of the NP hard, mixed μ problem structure provides a polynomial time algorithm that approximates μ with usually reasonable answers. When the problem is extended to the skew μ formulation an extension of the existing μ methodology to the skew μ problem is required. The focus of this paper is to extend the μ bounds development to the skew μ problem and show skew μ bounds computation by way of various algorithms.

The motivation for extending the μ problem to skew μ is driven by large scale power systems stability analysis. Power system analysis is computationally expensive and contains narrow hard-to-identify peaks in instability. To reduce computational load and improve accuracy, the power system analysis problem is translated into a skew μ problem. This is done by reformulating the state space power system model as a robust analysis problem with the frequency range of interest augmented to the state space model as an uncertainty. The superiority of the skew μ methods for many μ problems is demonstrated via analysis of 2 power system benchmark problems.

Rodney M. Holland
Electrical and Computer Engineering Department
Colorado State University
Fort Collins, CO 80523
Spring 2003

ACKNOWLEDGMENTS

This project is funded by a grant from NSF/EPRI (NSF Award Number 9810081), and is being performed in cooperation with Iowa State University (ISU).

In addition, this dissertation is indebted to the following professors upon whose framework, this structure is built.

Peter Young-Colorado State University

Mustafa Khammash-Formerly of Iowa State University, Now at USC Santa Barbara

Vijay Vittel-Iowa State University

A special note to some members of my committee who have left me with memorable moments:

Dr Lile, "I control teaching assistantships. If you want one, its yours."

Dr Azimi, "You're going to do your PhD...are you crazy? Hey, you want an assistantship?"

Dr Poore, "I'm sorry, I only sit on committees of students who have taken my classes" (at which point I said "YES!!!" because I had taken his M340 class 17 years ago in 1982...)

DEDICATION

Undying devotion to God, who sent his son Jesus Christ as my savior, and through who's blessings, and sense of humor all of this was made possible.

Thanks to my parents, Robert and Elzora Holland, for instilling in me the values and morals it takes to reach this point.

Gratitude to my wife, Kris, who said "Why don't you go back for your PhD full time", to which I promptly replied "There's no way we can do that...." (obviously there was a way).

Love to my kids, Wilson, Ross and Derek, who don't seem to find it strange that their father is a 40 year old student.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOTATION AND SYMBOLS

A,B,C,D	State space parameters when appropriate
α^*	Complex conjugate of $\alpha \in \mathbb{C}$
$\|\alpha\|$	absolute value of $\alpha \in \mathbb{R}$ or magnitude of $\alpha \in \mathbb{C}$
$\mathbb{C}$	Field of complex numbers
$\mathbb{C}^{n\times m}$	Complex matrix with n rows and m columns
$\mathbb{C}^n$	Complex column vector with n elements
Δ	Uncertainty
Δ_f	Fixed portion of the uncertainty matrix
Δ_v	Varying portion of the uncertainty matrix
$\det(M)$	Determinant of a square matrix M
I_n	Identity matrix of size $n \times n$
$Im(M)$	Imaginary part of a matrix M
inf	Infimum, is the greatest lower bound of a set S, defined as a quantity x such that no member of the set is less than x.
K	Controller, in the appropriate configuration
$\overline{\lambda}(M)$	Largest (real) eigenvalue of a Hermitian matrix M
$\lambda(M)$	Eigenvalue of a matrix M
$\lambda_{min}(M)$	Smallest (real) eigenvalue of a Hermitian matrix M
$\lambda_k(M)$	k^{th} largest (real) eigenvalue of a Hermitian matrix M
M	Plant model
M^T	Transpose of matrix or vector M
M^H	Complex conjugate transpose of matrix or vector M
M^{-1}	Inverse of a square matrix M
m_{ij}	i^{th} row and j^{th} column element of matrix M
$\mu(M)$	Structured singular value of a matrix M
μ_{ub}	μ upper bound
μ_{lb}	μ lower bound
$\mathbb{R}$	Field of real numbers
$\mathbb{R}^{n\times m}$	Real matrix with n rows and m columns
$\mathbb{R}^n$	Real column vector with n elements
$\rho(M)$	Spectral radius of a matrix M (see page 204)
$\rho_R(M)$	Real spectral radius of a matrix M (see page 204)
$Re(M)$	Real part of a matrix M

$\overline{\sigma}(M)$	Largest singular value of a matrix M
$\underline{\sigma}(M)$	Smallest singular value of a matrix M
sup	Supremum, is the least upper bound of a set S, defined as a quantity x such that no member of the set exceeds x.
$Tr(M)$	Trace of a square matrix M
$\in$	element of, or belongs to
$\forall$	for all
$\triangleq$	Defined as

LIST OF ABBREVIATIONS

AVR	Auto Voltage Regulator
B & B	Branch and Bound
ETMSP	Extended Transient Midterm-Stability Program [32]
F_l	F-lower, LFT with interfacing matrix graphically below system matrix
F_u	F-upper, LFT with interfacing matrix graphically above system matrix
GB	Giga-Byte
GEVP	Generalized Eigenvalue Problem
HD	Hard Drive
LFT	Linear Fractional Transform
LHP	Left Half Plane
LHS	Left Hand Side
LMI	Linear Matrix Inequality
MIMO	Multiple Input Multiple Output
PC	Personal Computer
PD	Positive Definite
PSS	Power Systems Stabilization
RAM	Random Access Memory
RHP	Right Half Plane
RHS	Right Hand Side
ROC	Region Of Convergence
SISO	Single Input Single Output
SS	State-Space
SSV	Structured Singular Value
SV	Singular Value
WLOG	Without Loss Of Generality

Chapter 1

Introduction

1.1 Objectives

The objectives of this project are four-fold:

- Provide an enhanced ability to predict power grid instability in comparison to current methods of analysis.
- Investigate the improvement of current power system stabilizer(PSS) design methods using skew μ tools.
- Extend the theoretical boundaries associated with the robust control tool skew μ.
- Provide computational tools for characterizing systems using skew μ.

The characteristics of power grid analysis provide significant motivation for the development of robust analytical tools. First, power systems are MIMO (multi-input multi-output) in nature, which lends itself to characterization and design using robust analysis and robust control techniques. Secondly, power grid systems typically fall into the class of large or enormous in size, necessitating development of analysis and computational tools to improve the ability to predict power grid instability.

In robust analysis, stability characterization is quantified by the property μ; however, the power grid analysis problem lends itself to a particular robust analysis generalization called

skew μ. [1] Small problems can achieve skew μ results through the iterative application of existing μ tools and B & B (Branch and Bound) techniques, however, large problems become prohibitively slow computationally when the same techniques are applied.

While the tools created during the course of this project will immediately benefit the power industry, they also have the potential to impact the controls industry in general. Many problems tackled with μ computational tools can be more favorably cast as skew μ problems. Using skew μ computational tools, more accurate and more useful results can be obtained more rapidly. Clarifications of these broad statements can be found in this dissertation, as well as in [22].

1.1.1 Overture

The disparities between modeling, simulation and the real world are often the butt of technical jokes. Real physical systems can not be exactly described by mathematical models, but mathematical models are the most convenient and elegant method available for allowing us to gain foresight into the real physical world.

Often times these models are developed on the concept of an average or nominal system; however, the actual system may vary significantly from the nominal system. These variations can be the result of environmental changes, manufacturing variations, chemical property variations, and a host of other conceivable alterations to the nominal system. In order to provide control solutions with confidence, these variations in a system must be accounted for. This is one of the major motivations behind robust analysis and robust control theory.

Robust control theory provides for analysis of system variations through the use of perturbations (Δ) to a nominal model. The intent is to provide information or a solution for all expected system combinations, including the worst case scenario of parameter variations.

---

[1] Precise explanations of terms will be given in later sections. This introduction serves as a road map so that the reader does not lose sight of the objectives.

Easily said, finding the worst case situation is not always easy. Worst case scenarios can often occur at combinations that are not the extremum of the parameter variations. An equally vexing problem is the prospect of a finite, but extremely large number of possibilities to check, too many to check on an individual case by case basis.

The second major motivation behind robust analysis is the ability to quantify the stability and performance of a system when it is multi-variable. The stability of a MIMO system is generally described by a single quantity known as μ. While SISO systems can be accurately quantified in terms of the familiar Bode or Nyquist plot, MIMO systems cannot. Loop-at-a-time analysis techniques do not account for the interaction between MIMO system variables, thus these techniques are invalid. The quantity μ provides a stability characterization for the whole system. At any given frequency the distance to instability is known.

While a significant amount of effort has gone into the development of μ analysis and μ computational tools, very little has been done with skew μ. The intent of this project is to provide a significant improvement in the knowledge and computational tools available for skew μ.

1.2 Primary Inspiration-Modeling and Design of Power System Stabilizers

1.2.1 The Conventional Robust Analysis Solution

The conventional robust analysis method for power systems is to model the system mathematically as a problem of finding the μ bounds of the system, combined with a frequency gridding technique to examine the system for areas of instability. Once a frequency range of instability is found, that range is finely gridded to allow for more complete characterization of the instability. Unfortunately, this method does not check all frequencies. There are always missed points, no matter how fine the grid, and thus the potential for significant error. Additionally, this methodology is computationally very expensive.

1.2.2 The Better Solution

An important technique in robust analysis, introduced by Sideris [45] in 1992, is to model the frequency variable as an uncertainty parameter (perturbation) in the robust system analysis. This extremely useful formulation treats frequency as just another variable in the stability search, thus it searches the entire frequency range for the worst case frequency perturbation and eliminates frequency gridding in the process.

Formulation of frequency as an uncertainty parameter is a skew μ problem. The current power system analysis methodology formulates the skew μ problem, but calculates μ over a region of frequency. Thus the resulting answer can only indicate if there is a potential problem in the frequency interval (yes/no) because of the interdependency of frequency in the calculation of μ.

If frequency is skewed (fixed in range) in the problem, the resulting answer is a stability characterization according to the variation of all the other perturbed variables. In short, a stability characterization is obtained for the power system which is guaranteed across the entire range of frequency of operation.

1.2.3 The Big Benefit To The Power System Study

The underlying benefit to power system study will be:

- An improvement in the computational accuracy of a system's stability characteristics.
- A significant reduction in time for analyzing the stability of a power system (both from a system model setup and computational view).

These advantages are accomplished in the following manner:

Using one test over a given frequency range, the stability bounds will be exactly known. There will be no missed search areas in the region, unlike frequency gridding.

Ultimately, a B&B test may be used to improve accuracy. However, the functional search will be different. Search of contiguous frequency areas will be guaranteed in one test. Since

computational tools are designed to return the worst case perturbation of the variables, it is possible that a skew μ computation could return the exact values of the variables at the worst case. This would include the specific frequency location, with only a single test.

1.3 The Deeper Motivation-The Skew μ problem

1.3.1 The Concept of Skew μ

A value of $\mu = 1.1$ in relation to robust stability, means that all the perturbations in the system must be decreased in magnitude by a factor of 1.1 to guarantee stability. But what if the desire is to have the maximum uncertainty range of some blocks fixed, then how large can the other sources of uncertainty be before one encounters instability. This value is defined as skew μ.

For example: One can't control the outside temperature (one level of uncertainty), but one can choose the size of valve (another item with uncertainty) that they install. So, if a person chooses to fix the range of the temperature uncertainty, then what is the maximum flow before the system encounters instability? This is a skew μ problem, and the generalities implied by this simple example continually arise. Skew μ refers to the fact that some perturbations (Δ_{fixed}) are restricted in range while trying to find the smallest values for other perturbations (Δ_i) that cause instability.

Perhaps the most common motivating problem for skew μ is the robust performance problem. The skew μ robust performance problem defines all system parameter uncertainties as the fixed range uncertainties, and the varying range uncertainty as a hypothetical performance uncertainty block from system output to input. This skew μ problem asks the question "For a system with these uncertainties, what level of performance is available from the system?". This problem and its relation to controller design is discussed in chapter 8.

Another significant application for skew μ is analysis of frequency augmented state space models. In this problem, frequency is augmented to the state space structure as a perturbation of a fixed range, while other uncertainties are considered as the varying range

perturbations. This allows worst case robust stability bounds to be established for the state space system over a frequency range of interest . The prevalent state space robust analysis method analyzes the state space system at closely spaced frequency points. Although this provides a reasonable analysis, it does not guarantee that potential instability points are found. The skew μ analysis of the frequency augmented state space system searches over frequency to identify the worst case condition and guarantee a stability bound. The details of this problem are discussed in section 3.4.

1.3.2 Skew μ in Terms of the Power System Stabilization Problem

In the PSS analysis, the skew variable will be frequency. This implies the question "If frequency is fixed with a certain range, then how large can the other system perturbations be before instability is encountered?". This is precisely the result that is desired. In relation to the system uncertainties, quantify the system stability over a specific frequency range and guarantee there are no missed points. Frequency gridding could not guarantee un-missed points. Skew μ performs a worst case search over frequency, guaranteeing the answer.

A second important feature of this formulation is that it is not influenced by the implied feedback loop mentioned at the top of section 1.3.1. The implied feedback loop of μ is the decrease in uncertainty size to guarantee stability. If the uncertainty size of all perturbations is decreased to guaranteed stability, then the frequency range is also decreased. This is exactly the advantage of skew μ in the power problem. Frequency is fixed in range , it is all the other perturbations that are allowed to vary in size to quantify stability.

1.4 Historical Review of Skew μ

The intent of this review is to highlight some of the developments that lead to the topic of this dissertation. It is not intended to be comprehensive in nature.

Very few ideas appear out of nowhere, and robust control theory is no exception. It too, is built on the work of predecessors. However, to pick a reasonable starting point for the purposes of this dissertation, start with the November 1982 issue of the *IEE Proceedings, Part D*. In that proceeding, two papers appeared, one right after the other, dealing with the stability issues of systems with structured uncertainties or perturbations. The first paper was by John Doyle [11], the second was by M.G. Safonov [43].

These papers provided the impetus for extending research in the use of singular values and in particular, the SSV (Structured Singular Value) as a method for determining the stability of multi-variable systems with structured perturbations. The interest in structured perturbations was motivated by their ease of representation for simultaneous variations in distinct system components.

Specifically, Doyle's paper [11] introduced the use of upper and lower bounds for SSV problems as a means for determining the stability of a system.

A host of researchers followed in Doyle's and Safanov's footsteps, extending and improving the the SSV concept of system analysis. A partial list of highlights are:

- Proposed control synthesis method, The *D-K iteration* [13]-1985
- Consideration of both parametric and dynamic uncertainty within the μ analysis framework [13]-1985
- Development of a lower bound for the complex μ problem [41]-1988
- Release of the *μ-Tools Toolbox* for Matlab®[2]-1991
- Introduction of G scales for improving real parametric uncertainty [18]-1991

- Introduction of frequency as an uncertainty variable [45]-1992
- Theoretical and computational result for a mixed μ lower bound [53]-1997

The concept of skew μ has been known since the early days of μ [49], however, investigations into the skew μ problem have been relatively limited in scope. The bulk of the material in circulation has been published by Ferreres ([22],[19],[20] and [21]), with an additional article by [25], in which a lower bound algorithm and and LMI upper bound are proposed for the skew μ problem. While these citations provide insight, the intent of this dissertation is to provide a more detailed analysis with new revelations in the area of skew μ and additionally provide computational tools for skew μ which have not existed in the past.

1.5 Outline for Development of Skew μ

Because skew μ is an extension of μ, it will be developed in terms of the robust stability quantity μ. The following list gives a rough map of the path that will be followed for the remainder of this dissertation.

- Modeling uncertainty in MIMO systems
- Casting a MIMO system as a μ problem
- Calculating bounds on μ
- Formulating the skew μ problem and extending μ bounds to skew μ
- Application of μ and skew μ tools to the power system problem
- Development of a skew μ control design tool for potential use with power systems
- Conclusions

Chapter 2

Establishing the General μ Problem

In the world of control systems analysis, the model is at the core of the situation. Adjacent to the model are always concerns about the accuracy of the model. No model is perfect. There are always approximations. Dealing with model uncertainties requires a methodology for incorporating these uncertainties into the system. In turn, the system model may change from the normal expectation because of incorporation of these uncertainties, and ultimately the manner in which system stability and performance are quantified may also change from the previous norm. This is the case with robust analysis. Some of the techniques will look familiar to the control engineer. Some will not. In order to provide a clear path to the results of this project, a basic overview of robust control theory will be developed in the following sections so that the innovations may be built upon it.
First, methods for modeling uncertainties will be developed.

2.1 Types of Perturbations

Model uncertainties are termed perturbations (Δ). In typical control systems modelling there are two primary reasons for uncertainty of the model:

- Un-modelled dynamics

- Most created models are of lower order than the actual system. Even if highly detailed models are available, a lower order model may be chosen to facilitate analysis. The neglected dynamics are termed *uncertainty*.

- Parametric inaccuracies.

  - Parameters in models are usually approximately known, and are sometimes just plain in error.
  - Parameters may have non-linear characteristics, varying with changes in other parameters. Robust analysis is linear in form, hence non-linearities cause uncertainty in parameter values.
  - Measurement devices are imperfect. Any measured quantity in a control system is subject to error.
  - Parametric uncertainties are structured uncertainties since they model the uncertain parameters in a structured manner.
  - Parametric uncertainties are real (as opposed to complex) in nature. The uncertainty is directly related to a real-valued parameter.

Dynamic and parametric uncertainties can be modeled in various ways to interact with the nominal system. The following sections give a representation of the most common methods of illustrating perturbations and system interaction. In all cases and through out this document, Δ will be used to represent uncertainty.

2.1.1 Multiplicative Uncertainty

Multiplicative uncertainty can be represented as

$$Gp = (1 + \Delta w)G, \tag{2.1}$$

which is graphically represented in figure 2.1. In addition to the uncertainty, there is an arbitrary weighting function w that allows for the modification of the size of the uncertainty.

This weighting function is usually based on the confidence in the range of the uncertainty, and may be frequency biased (i.e. potentially $w(s)$).

Multiplicative uncertainty covers un-modelled dynamics, and is used as a crude approximation to parametric uncertainties. It covers these uncertainties with a stable transfer function (Δ) which can provide the appropriate response characteristics of the grouped uncertainties over a wide frequency range.

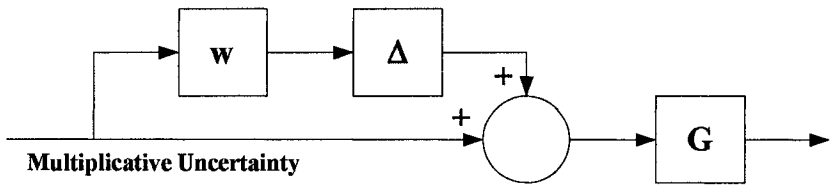


Figure 2.1: Block diagram of multiplicative uncertainty with a weighting function w

2.1.2 Additive Uncertainty

Additive uncertainty can be represented as

$$Gp = G + \Delta w. \tag{2.2}$$

This is graphically represented in figure 2.2.

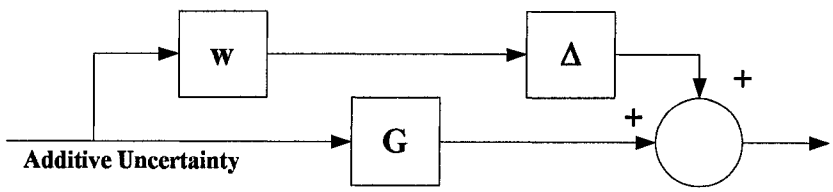


Figure 2.2: Block diagram of additive uncertainty

Additive uncertainty covers the irregular perturbations of a nominal system with a suitably sized disk. All Δ's used in constructing this disk are stable transfer functions.

Figure 2.3 displays a Nyquist plot with the trajectory of a nominal system as frequency increases. On the trajectory is placed a shaded region which represents the potential area of variation due to parametric uncertainty or un-modeled dynamics. The disk represents the ability of additive or multiplicative uncertainty to cover that region. The conservative nature of the disk depends on the shape of the uncertainty region. Compatibility between the disk and the uncertainty region may cause the disk to cover a significantly larger area than the uncertainty region to guarantee full coverage of the uncertainty region. Since robust analysis provides a solution which accounts for the disk, the analysis may be overly conservative. Potentially useful regions that lie inside the disk, but outside the actual uncertainty region are guarded against but pose no problem in the real world application.

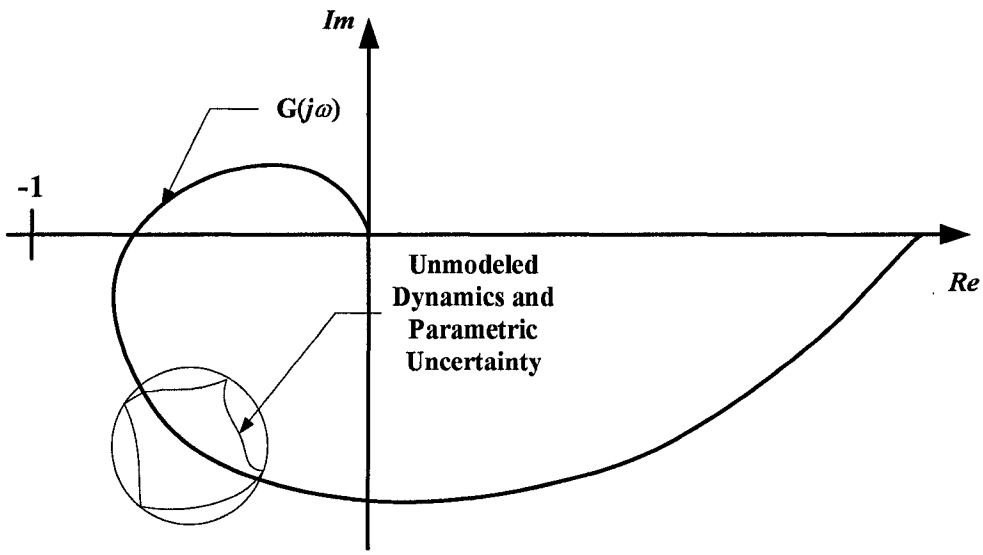


Figure 2.3: Uncertainty covered with disk on a Nyquist plot

2.1.3 Feedback or Inverse Multiplicative Uncertainty

Feedback uncertainty can be represented as

$$\frac{G}{1-\Delta wG}. \tag{2.3}$$

This is graphically represented in figure 2.4.

Feedback uncertainty is better suited for representing pole location uncertainty. Care should be taken in the use of feedback uncertainty, since it is possible to create an unstable loop in the uncertainty through use of an inappropriate model.

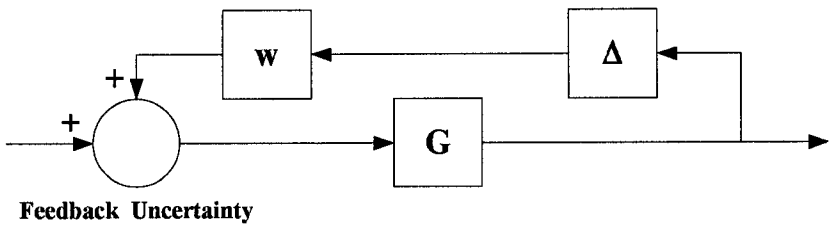


Figure 2.4: Block diagram of feedback uncertainty

These perturbation techniques allow for the modeling of uncertainties in the system. However, these models do no good unless they can be integrated into the system in a reasonable fashion that will allow for analysis of the system and quantification of stability and performance. To meet this requirement a method of system construction called the LFT (Linear Fractional Transformation) is used. The creation and manipulation of the LFT is explained in the following sections.

2.2 Linear Fractional Transformations

The LFT was introduced into robust analysis in 1984 by Doyle [12], and since has become an icon of robust analysis. A simple development of the fundamental forms is given here. Additional information can be found in the appendix of [46].

In figure 2.5 a simple input output relationship is implied, but can be more readily understood in figure 2.6.

In figure 2.6 the input output relationship is easy to read as

$$y = M_{11}u + M_{12}w \tag{2.4}$$

$$z = M_{21}u + M_{22}w \tag{2.5}$$

$$. \tag{2.6}$$

Figure 2.5: Standard LFT form for two inputs

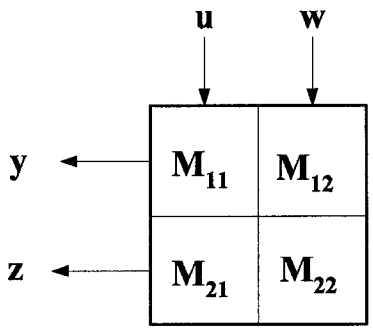


Figure 2.6: LFT form for two inputs with vertical input format

If a loop is closed between u and y as in figure 2.7, then the relationship $u = \Delta y$ can be substituted into equation 2.4 to derive

$$z = F_u(M,\Delta)w, \tag{2.7}$$

where

$$F_u(M,\Delta) \triangleq M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12}. \tag{2.8}$$

The notation here denotes the term F-upper (F_u), which is commonly used to indicate an LFT with a loop closed around the upper portion or the system matrix M.

Likewise, a loop can be closed around the lower portion of the system matrix (i.e., $w = \Delta z$) as in figure 2.8. In this case, the form is termed F-lower (F_l) and has an input output relationship of

$$y = F_l(M,\Delta)u, \tag{2.9}$$

where

$$F_l(M,\Delta) \triangleq M_{11} + M_{12}\Delta(I - M_{22}\Delta)^{-1}M_{21}. \tag{2.10}$$

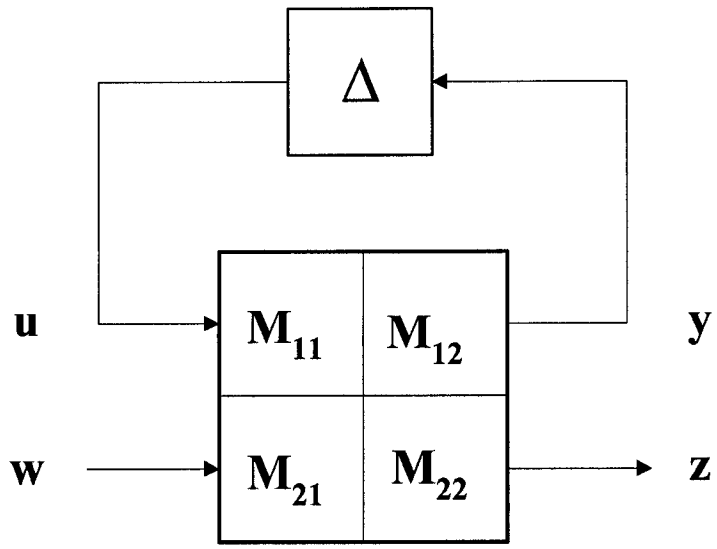


Figure 2.7: Standard F-upper LFT form used in robust analysis

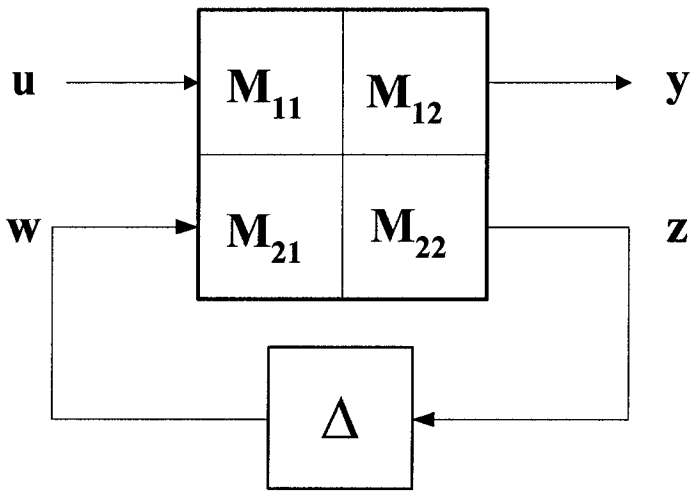


Figure 2.8: Standard F-lower LFT form used in robust analysis

LFT's formed as an F-upper will be predominantly used in this document.

Now, consider an LFT formed by the two operators in figure 2.9.

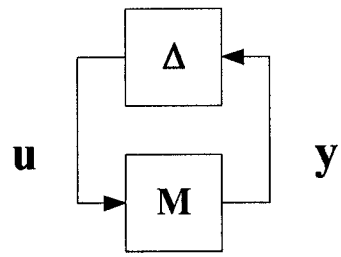


Figure 2.9: Input/output relationship of LFT

The output relationships are

$$y = Mu \tag{2.11}$$

$$u = \Delta y. \tag{2.12}$$

Presuming that $M \in \mathbb{C}^{m \times n}$ and Δ of size $n \times m$ the loop formula can be reduced to $(I_m - M\Delta)y = 0$ which yields a unique solution if and only if $\det(I_m - M\Delta)$ is non-zero. This is one of the main premises behind μ, and this thought will be expanded to define μ in section 2.3.

The LFT structure is the basis on which much μ theory is built, and all perturbations will be pushed into the appropriate structure to facilitate this form. Two major items of robust analysis have been introduced, modeling of uncertainty and modeling systems as LFT's. Now it becomes necessary to integrate the two parts together such that whole systems may be adequately described.

2.2.1 Changing a Model with Uncertainty into LFT Form

How an uncertainty enters a block diagram does not affect the ability to form an LFT from the system, as evidenced in figure 2.10 and figure 2.11. These two figures form an example

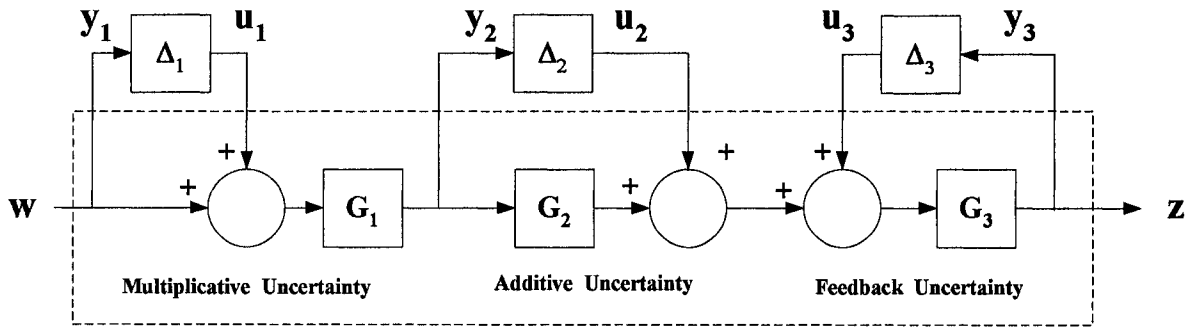


Figure 2.10: System with uncertainty

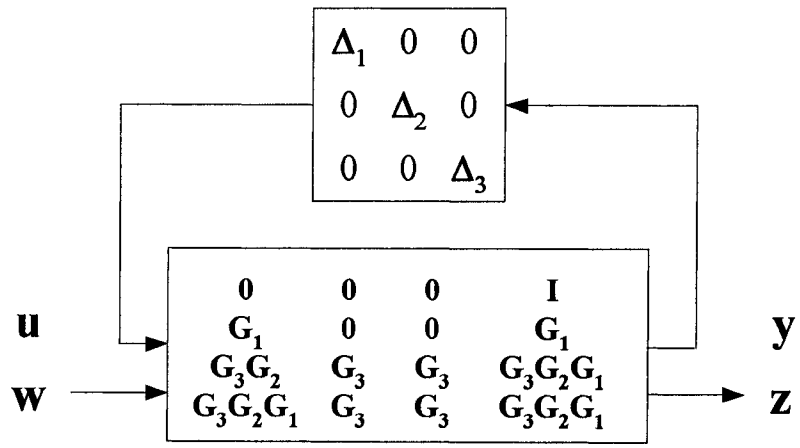


Figure 2.11: Conversion of system with uncertainty to an LFT-Standard representation

of conversion from a generic system to an LFT. Essentially, any block diagram that can be drawn, can be drawn as an LFT.

The procedure for obtaining the LFT from a perturbed model is as follows:

1. Create perturbed system.
2. Identify inputs and outputs of system for conversion to an LFT
3. Identify path from a specific input to a specific output, paying attention to signs along the way. The system entry is the reverse order encountered. For instance, from input A to output B might encounter G_1, G_2, G_3 in that order. The corresponding system factor from A to B is then $G_3G_2G_1$, and is entered in the system matrix in column A, row B. Note that the perturbations are considered external to the system or as an *open circuit* during this procedure.
4. Form the appropriate uncertainty block matrix. This will be the block uncertainties on the diagonal of the matrix if the previous steps have been followed.

As an example, consider the system described by figure 2.10 and figure 2.11. The system matrix can alternately be represented by figure 2.12. Applying the algorithm above in order to obtain the component from u_1 to y_3 go from u_1 to y_3 via G_1,G_2, and G_3. Δ_1,Δ_2, and Δ_3 are considered open in this step and flow must go in the direction of the arrows. The component of the system matrix is then written in reverse order, $G_3G_2G_1$. Consider going from u_2 to y_1. There is no way to get there, therefore the answer is 0. Finally, consider going from u_2 to z. Only the block G_3 is in the path, therefore the entry at the (z, u_2) position of the system matrix is G_3. There are no sign changes in this block diagram, so all entries are positive. Intuitively, this game can be played with any block diagram that can be drawn; therefore any method that can be constructed for describing a perturbation can be used to generate an LFT of the system.

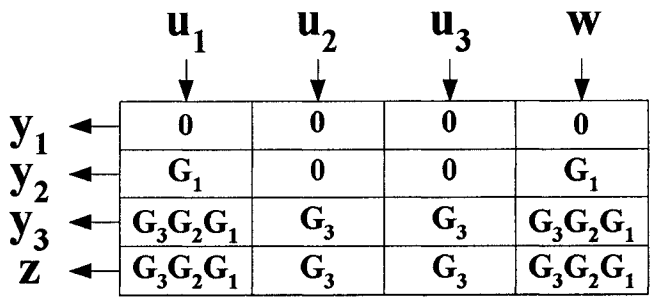


Figure 2.12: Conversion of system with uncertainty to an LFT-Matrix representation

At this point the necessary infrastructure has been created for developing system descriptions which can model all linear systems and fit the LFT structure of figure 2.9. In order for this model and methodology to be of value, a criterion needs to be developed that relates the LFT and perturbation structure to its stability and performance. This is the focus of the next sections.

2.3 μ and Stability Criterion

The MIMO system stability criterion, referred to as μ, can be derived using the LFT depicted in figure 2.9. The input/output relationships are

$$y = Mu \tag{2.13}$$

$$u = \Delta y. \tag{2.14}$$

By substitution,

$$y = M\Delta y, \tag{2.15}$$

which can be regrouped as

$$(I - M\Delta)y = 0. \tag{2.16}$$

This is essentially equivalent to solving the eigenvalue problem

$$(\lambda I - A)x = 0, \tag{2.17}$$

where $\lambda = 1$. The solution to the eigenvalue problem is well known, i.e. λ is an element of the set of eigenvalues of A if and only if $\lambda I - A$ is singular, or $\det(\lambda I - A) = 0$. In this case, presume $\lambda = 1$ and a Δ has been found such that

$$\det(I - M\Delta) = 0. \tag{2.18}$$

Then there exists a $y \neq 0$ which makes

$$(I - M\Delta)y = 0.$$

This implies that

$$Iy = M\Delta y,$$

or

$$\|y\|_2 = \|M\Delta y\|_2 \leq \overline{\sigma}(M\Delta)\|y\|_2 \leq \overline{\sigma}(M)\overline{\sigma}(\Delta)\|y\|_2.$$

Rearranging the first and last components of the previous equation one gets

$$1 \leq \overline{\sigma}(M)\overline{\sigma}(\Delta)$$

or

$$\overline{\sigma}(\Delta) \geq \frac{1}{\overline{\sigma}(M)}. \tag{2.19}$$

Equation 2.19 states that a Δ which causes singularity must have a maximum singular value equal to or larger than $\frac{1}{\overline{\sigma}(M)}$.

Now suppose one has the SVD of M as

$$M = \sum_i \sigma_i u_i v_i^H.$$

Choose

$$\Delta = \frac{1}{\sigma_1} v_1 u_1^H,$$

then one immediately knows that

$$\overline{\sigma}(\Delta) = \frac{1}{\sigma_1} = \frac{1}{\overline{\sigma}(M)}.$$

Combining the chosen Δ with the SVD of M, one gets

$$M\Delta = \sum_i \frac{\sigma_i}{\sigma_1} u_i v_i^H v_1 u_1^H = u_1 u_1^H = I.$$

Noting that $v_i^H v_j = \delta_{ij}$ and choosing $y = u_1$, then one can say

$$(I - M\Delta)u_1 = u_1 u_1^H u_1 - u_1 v_1^H v_1 u_1^H u_1 = u_1 - u_1 = 0,$$

therefore $I - M\Delta$ is singular if $\overline{\sigma}(\Delta) = \frac{1}{\overline{\sigma}(M)}$. Thus the smallest perturbation Δ, that can drive the system to instability (or equivalently, make $\det(I - M\Delta) = 0$) has a maximum singular value of $\overline{\sigma}(\Delta) = \frac{1}{\overline{\sigma}(M)}$.

The previous derivation is under the condition that Δ is a full matrix. However, from previous section 2.1 one sees that for real/physical systems, Δ is not some full matrix. It has structure. Specifically, it is a block diagonal structure, based on the nature of the uncertainties in the system setup.

Naturally then, finding the point of instability for a MIMO system is not going to be so easy as calculating the $\overline{\sigma}(M)$. We are actually looking for the first point of instability caused by a variation in an uncertainty direction that is available. Because of the structure of Δ, not all directions of matrix manipulation are available.

To grasp the stability concept easier, lets provide the following characterization of Δ,

$$\|\Delta\| \leq 1. \tag{2.20}$$

This is a unit ball ($\|\Delta\| \leq 1$ for $\Delta \in \mathbf{B}\Delta$) characterization of Δ in the appropriate space. Depending on M, the unit ball may or may not cause the perturbed system to pass through the origin (think of $\det(I - M\Delta)$ as a multi-variable Nyquist plot) . The desired information is: if the ball grows or shrinks, at what point does $det(I - M\Delta) = 0$. Alternately, by introduction of the appropriate positive real scalar β, it can also be written for $\Delta \in \mathbf{B}\Delta$ as

$$(I - \frac{1}{\beta} M\Delta) = 0. \tag{2.21}$$

If $\beta \gg 1$, it implies it takes a very small perturbation to the nominal system to drive the system into instability. If $\beta \ll 1$, it implies it takes a very large perturbation to the nominal system to drive the system into instability. This value β is the value of μ, the MIMO measurement of stability. In short,

$$\mu(M) = \beta. \tag{2.22}$$

If $\mu > 1$, the system has the *possibility* of being unstable for $\Delta \in \mathbf{B}\Delta$. If $\mu < 1$ the system does not have the possibility of being unstable within the range of expected uncertainties. It should be noted that this does not guarantee stability or instability (presuming the nominal system is stable). It indicates the potential for stability or instability based on the range of uncertainty that the system has been modeled with. If the engineer makes a poor estimate of the range of uncertainty, well, "Bad guess, bad results".

An inherent problem with the previous development is that finding μ has been shown to be an NP hard problem [6]. Background material on NP complete and associated NP hard problems is not presented here, but can be found in [24]. More details on the impact of NP hardness can be obtained from [50] and [24], however, it is *widely believed* that NP hardness implies that μ for a problem cannot be calculated in polynomial time. While the exact computational consequences of being NP hard are not known, the following conjecture is believed to be true.

> Open Conjecture: *Computation time of an NP hard problem-* If the problem contains n variables and has the property of being NP hard, then it is believed the computation time is higher than a^n for some $a > 1$. This will result in a great increase in computation time as n increases. (note: typically it is believed that $a > 2$ for most real world problems)

For example, consider a problem that grows in complexity with polynomial time n^2. If a size 10 (n=10) problem can be solved in .01 seconds (trivial) then a size 50 (n=50) problem

can be solved in .25 seconds (still trivial). If a problem grows in exponential time according to 2^n and a size 10 (n=10) problem can be solved in .01 seconds (trivial) then a size 50 (n=50) can be solved in 348.4 years (non-trivial).

Because of apparent computational problems of being NP hard, upper and lower bound calculations have been established which give the range of μ for a particular problem. The range between these bounds of μ can often be compressed to a point that renders exact computation of μ unnecessary. The previous paragraph gives only the flavor of the computational problems of μ. The more interested readers should see [24] and [6] for more extensive details on the subject.

The relationship of these bounds to μ is extracted in the following sections. Prior to that, a short explanation and clarification on structured perturbation must be done to set the stage for the bounds development.

2.4 Types of Structured Perturbation

As described in section 2.1 there are a few ways that perturbations emerge, and section 2.1 gives their relationship to physical quantities. In this section, the main concern will be the mathematical possibilities for a perturbation.

The types of perturbations are:

Real This is a single real perturbation denoted as δ^r, where $\delta^r \in \mathbb{R}$

Repeated Real This is a repeated instance of a real perturbation denoted as $\delta^r I_n$, where $\delta^r \in \mathbb{R}$. For example, suppose $n = 3$, then

$$\delta^r I_3 = \begin{bmatrix} \delta^r & 0 & 0 \\ 0 & \delta^r & 0 \\ 0 & 0 & \delta^r \end{bmatrix} \tag{2.23}$$

Complex This is a single complex perturbation denoted as δ^c, where $\delta^c \in \mathbb{C}$

Repeated Complex This is a repeated instance of real perturbation denoted as $\delta^c I_n$, where $\delta^c \in \mathbb{C}$. For example, suppose $n = 3$, then

$$\delta^c I_3 = \begin{bmatrix} \delta^c & 0 & 0 \\ 0 & \delta^c & 0 \\ 0 & 0 & \delta^c \end{bmatrix} \tag{2.24}$$

Full Block This is a full matrix perturbation denoted as Δ^C, where $\Delta^C \in \mathbb{C}^{m\times n}$. For example, suppose $m = 2$ and $n = 3$, then

$$\Delta^C = \begin{bmatrix} \Delta^C_{1,1} & \Delta^C_{1,2} & \Delta^C_{1,3} \\ \Delta^C_{2,1} & \Delta^C_{2,2} & \Delta^C_{2,3} \end{bmatrix} \tag{2.25}$$

Any perturbation matrix Δ may conceivably be made of any combination of these types. It is possible to have a perturbation matrix that looks like

$$\Delta = block\ diag(\delta^r_1 I_2, \delta^c, \delta^r_2 I_3, \Delta^C_{2\times 3}) \tag{2.26}$$

or, more specifically

$$\Delta = \begin{bmatrix} \delta^r_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \delta^r_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta^c_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \delta^r_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \delta^r_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta^r_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta^C_{11} & \Delta^C_{12} & \Delta^C_{13} \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta^C_{21} & \Delta^C_{22} & \Delta^C_{23} \end{bmatrix}. \tag{2.27}$$

The purpose in the definitions and examples above is to make the following general structure for perturbations less obscure when it is presented.

First, given a matrix $M \in \mathbb{C}^{m\times n}$ as in figure 2.9 and three non-negative integers m_r, m_c, and m_C with $p \triangleq m_r + m_c + m_C \leq n$, the block structure $\mathcal{K}(m_r, m_c, m_C)$ is an p-tuple of positive integers

$$\mathcal{K} = (k_1, \ldots, k_{m_r}, k_{m_r+1}, \ldots, k_{m_r+m_c}, k_{m_r+m_c+1}, \ldots, k_m) \tag{2.28}$$

where we require $\sum_{i=1}^{p} k_i = n$ in order that these dimensions are compatible with M. This now determines the set of allowable perturbations, namely define

$$X_{\mathcal{K}} = \{\Delta = block\ diag(\delta_1^r I_{k_1}, \ldots, \delta_{m_r}^r I_{k_{m_r}}, \delta_1^c I_{k_{m_r+1}}, \ldots, \delta_{m_c}^c I_{k_{m_r+m_c}}, \Delta_1^C, \ldots, \Delta_{m_C}^C) : \delta_i^r \in \mathbb{R}, \delta_i^c \in \mathbb{C}, \Delta_i^C \in \mathbb{C}^{k_{m_r+m_c+i} \times k_{m_r+m_c+i}}\} \tag{2.29}$$

Note that $X_{\mathcal{K}} \subset \mathbb{C}^{n \times m}$ and that this block structure is sufficiently general to allow for repeated real scalars, repeated complex scalars, and full complex blocks. The purely complex case corresponds to $m_r = 0$, and the purely real case to $m_c = m_C = 0$.

The vector $\mathcal{K}$ is a listing of positive integers that *represent the number of* real perturbations, complex perturbations and full block perturbations, including the dimensions of each specific block. $X_{\mathcal{K}}$ represents the set of *actual allowable Δ's (perturbations)*. The notation may appear complicated, but the concept it represents, the number and arrangement of the perturbation blocks, is relatively straight forward.

In terms of arrangement, note that the real perturbations are listed first, the complex second and the full complex blocks third. This is not a requirement, but it does facilitate notation and development. Any other arrangement of perturbation blocks is only an elementary transformation away. Additionally, equation 2.29 defines all blocks as square. This is for the ease of presentation. The theory developed in this document (as well as the software) handles the general case where full block perturbations are not necessarily square.

2.4.1 Perturbations, Complex μ and Mixed μ

Complex μ is a term typically given to problems involving finding bounds on μ where the perturbations are only complex in nature.

Mixed μ is a term typically given to problems involving finding bounds on μ where the perturbations are both real and complex.

Examples in section 2.5 will be set in the format of upper bounds on complex μ. These bounds are also valid for the mixed μ case, but are typically more crude than the mixed μ bounds.

These sections are provided for informational purposes, and thus are not extended to the mixed μ case. However, all skew μ results obtained in this document are valid for the mixed μ case, as is the software created for this project.

Keeping this in mind, and noting that a perturbation structure has been defined, the bounds can be approached.

2.5 μ Upper Bound

Starting with the basic definitions from section 2.3, an upper bound for μ is developed in the following manner. Suppose that

$$\det(I - M\Delta) = 0 \tag{2.30}$$

for some Δ. Then for some vector $x \neq 0$,

$$(I - M\Delta)x = 0 \tag{2.31}$$

or

$$M\Delta x = x. \tag{2.32}$$

Now, using norms to bound this equation, one gets

$$\|x\|_2 = \|M\Delta x\|_2 \leq \overline{\sigma}(M\Delta)\|x\|_2 \leq \overline{\sigma}(M)\overline{\sigma}(\Delta)\|x\|_2. \tag{2.33}$$

Note that $\|\cdot\|_2 = \overline{\sigma}(\cdot)$.

A little rearranging of equation 2.33 allows the statement

$$\overline{\sigma}(\Delta) \geq \frac{1}{\overline{\sigma}(M)}. \tag{2.34}$$

This lower bound on the perturbation size gives us an upper bound for μ,

$$\mu(M) \leq \frac{1}{\frac{1}{\overline{\sigma}(M)}} = \overline{\sigma}(M). \tag{2.35}$$

This is a nice result. However, it has an intrinsic problem. It does not exploit the structure of Δ, and M is not always well conditioned, thus $\overline{\sigma}(M)$ may be significantly larger than $\mu(M)$. One way to improve this upper bound is through the introduction of a similarity transform to improve the condition of the matrix. The requirements necessary for being able to perform such a transform are discussed in the next section.

2.5.1 Commutation of D-Scales

Scaling of matrices in the robust control framework increases the accuracy of singular value calculations via minimization of the matrix norm through a similarity transform.

Graphically, this useful technique is carried out in the three figures of 2.13.

The ability to re-distribute the order of the D and Δ matrices is provided by using a structure that commutes. The requirement is that D be chosen such that

$$D\Delta = \Delta D. \tag{2.36}$$

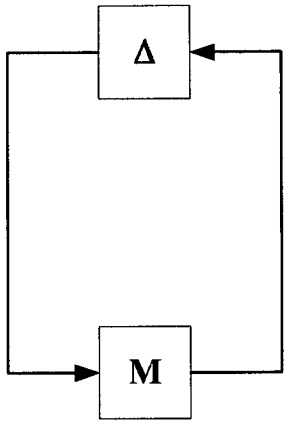
Using the fact that a scalar commutes, $\alpha \cdot A = A \cdot \alpha$ and $A \cdot I = I \cdot A$, it becomes evident that D must have a *complimentary block structure* to commute with Δ. If Δ has the structure

$$\Delta = \begin{bmatrix} \delta_i & 0 & 0 \\ 0 & \delta_j I_p & 0 \\ 0 & 0 & \Delta_{F_{k\times k}} \end{bmatrix}, \tag{2.37}$$

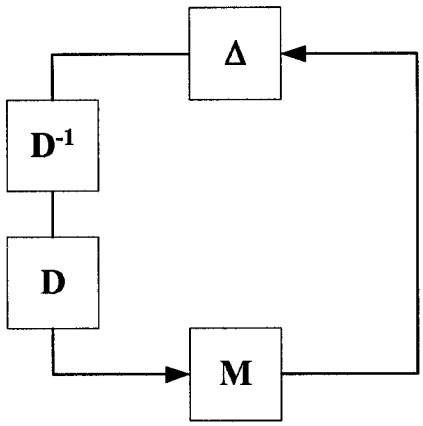
where δ_i and δ_j are complex scalars, I is an $p \times p$ sized identity matrix, and Δ_F is a full complex matrix of size $k \times k$. The commuting D structure can be expressed as

$$D = \begin{bmatrix} d_x & 0 & 0 \\ 0 & D_{G_{p\times p}} & 0 \\ 0 & 0 & d_y I_k \end{bmatrix} \tag{2.38}$$

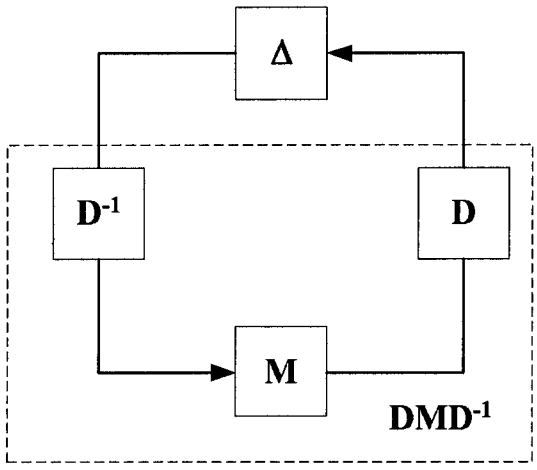
where d_x and d_y are complex scalars, I is an $k \times k$ sized identity matrix, and D_G is a $p \times p$ sized full complex matrix block.



(a) M-Δ



(b) M-D cancelling



(c) DMD^{-1}

Figure 2.13: Explanatory diagram for similarity transformation of M

A quick reference is given in table 2.1. Note that all D-scale entries can be elements of complex numbers, however, given a Δ_F, the corresponding $d_y I$ is chosen with $d_y \in \mathbb{R}$, since the added degree of freedom gains no advantage, and increases the number of calculations and variables.

Note that this example was done for the case where Δ_F is square. Once again, this can be extended to the more general case where Δ_F is non-square.

D	Δ
repeated scalar	full block
full block	repeated scalar
single scalar	single scalar

Table 2.1: Table of Commuting Pairs

This covers the functional considerations for choice of D, however, to complete the concept an explicit definition of D is required.

Consider D, an element of a set of block diagonal matrices $\mathcal{D}_{\mathcal{K}}$, picked to have the complimentary structure to Δ such that it commutes with any $\Delta_k \in \Delta$ in the structured perturbation from equation 2.29: then $\mathcal{D}_{\mathcal{K}}$ is defined as

$$\mathcal{D}_{\mathcal{K}} = \{block\ diag(D_1, \ldots, D_{m_r+m_c}, d_1 I_{k_{m_r+m_c+1}} \ldots, d_{m_C} I_{k_m}) : \det(D_i) \neq 0, D_i \in \mathbb{C}^{k_i \times k_i}, d_i \neq 0, d_i \in \mathbb{C}\}. \quad (2.39)$$

In addition, a second set of D-scales is defined at this point for use later in this document. Because these scaling will be used throughout the remainder of the document it is helpful to have them defined early at a single point the reader can refer to.

$$\hat{\mathcal{D}}_{\mathcal{K}} = \{block\ diag(D_1, \ldots, D_{m_r+m_c}, d_1 I_{k_{m_r+m_c+1}} \ldots, d_{m_C} I_{k_m}) : 0 < D_i = D_i^H \in \mathbb{C}^{k_i \times k_i}, 0 < d_i \in \mathbb{R}\}. \quad (2.40)$$

The second set of D-scales differs from the first set by requiring that the D's be Hermitian. This is not required for general D's, however, this property will emerge as the result of some matrix manipulations later in this document, and will be only for specified cases. With commutation of D-scales defined, an improved upper bound can be specified.

2.5.2 Improved μ Upper Bound

A modified upper bound can be obtained by inspecting figure 2.13. By equivalence,

$$\mu(M) = \mu(DMD^{-1}) \tag{2.41}$$

and, most importantly,

$$\mu(M) = \mu(DMD^{-1}) \leq \overline{\sigma}(DMD^{-1}). \tag{2.42}$$

Finally, since this holds for and $D \in \mathcal{D}_{\mathcal{K}}$,

$$\mu(M) \leq \inf_{D \in \mathcal{D}_{\mathcal{K}}} \overline{\sigma}(DMD^{-1}) \tag{2.43}$$

Pictorially, its easy to understand what's happening in figure 2.13, but the exact impact of equation 2.43 is slightly more obscure. Consider, D and D^{-1} are a similarity transform where the matrix D is chosen to have a particular structure to allow it to commute with Δ. The additional requirement on choice of D is that it minimize $\overline{\sigma}(DMD^{-1})$ as much as possible (hence the inf process).

Clearly, the system properties (eigenvalues) are not being affected because the D's constitute a similarity transform, however, the μ upper bound is being shrunk by the minimization of the 2 norm (Euclidean norm). This is the goal, a tighter bound on the exact value of μ. This constrains $\mu(M)$ from above, now its necessary to constrain μ from below.

2.6 μ Lower Bound

To arrive at a useful lower bound, a special Δ needs to be defined. This Δ will be denoted as Q, where Q is an element of the set $X_{\mathcal{K}}$ (i.e. the set of allowable Δ's), and have the additional requirement that Q is pseudo unitary (i.e. all complex perturbations are unitary). The set of permissible Q's may be explicitly defined as:

$$Q_{\mathcal{K}} = \{\Delta \in X_{\mathcal{K}} : \delta_i^r \in [-1\ 1], \delta_i^{cH}\delta_i^c = 1, \Delta_i^{CH}\Delta_i^C = I_{k_{m_r+m_c+i}}\} \tag{2.44}$$

which satisfies the requirement of a unitary matrix that $QQ^H = I$ for the complex perturbations, but is denoted here on a block by block basis, similar to the perturbation Δ.

To develop a lower bound for $\mu(M)$, suppose that $k_{m_r} = 0$. In general this is not required, however, for the time being, only the complex Δ case will be shown. Additionally, presume $Q \in \Delta$ is unitary, and $\rho(QM) = \gamma$ where $\rho = max|\lambda_i|$. Then

$$QMx = \gamma e^{j\theta}x, \tag{2.45}$$

with $x \neq 0$. Rearrange this to

$$\frac{Q}{\gamma e^{j\theta}}Mx = x. \tag{2.46}$$

If we denote $\frac{Q}{\gamma e^{j\theta}} = \Delta$, then the equation reduces to

$$\Delta Mx = x \tag{2.47}$$

where $\overline{\sigma}(\Delta) = \frac{1}{\gamma}$. Manipulating the previous equation,

$$(I - \Delta M)x = 0 \tag{2.48}$$

or

$$\det(I - \Delta M) = 0, \tag{2.49}$$

which is the definition of $\mu(M)$.

Therefore,

$$\mu(M) \geq \frac{1}{\frac{1}{\gamma}} = \gamma \tag{2.50}$$

and

$$\rho(QM) \leq \mu(M) \tag{2.51}$$

or, since this holds for any $Q \in Q_{\mathcal{K}}$

$$\max_{Q \in Q_{\mathcal{K}}} \rho(QM) \leq \mu(M), \tag{2.52}$$

where Q is chosen to maximize the spectral radius of the product QM.

Now, to complete the proof in the other direction, suppose a worst case Δ has been found. Call this worst case $\hat{\Delta}$, where $\overline{\sigma}(\hat{\Delta}) = \frac{1}{\beta}$. Therefore $\mu(M) = \beta$, and

$$\det(I - \hat{\Delta}M) = 0$$

$$(I - \hat{\Delta}M)y = 0$$

$$\hat{\Delta}My = y$$

$$\frac{1}{\beta}\Delta My = y$$

$$\Delta My = \beta y$$

$$\rho(\Delta M) \geq \beta. \tag{2.53}$$

Thus

$$\max_{\Delta \in \mathcal{B}\Delta} \rho(\Delta M) \geq \mu(M). \tag{2.54}$$

By coupling the forward and backward proofs together, the following is obtained:

$$\max_{Q \in Q_{\mathcal{K}}} \rho(QM) \leq \mu(M) \leq \max_{\Delta \in \mathcal{B}\Delta} \rho(\Delta M). \tag{2.55}$$

Underlying equation 2.55 is a requirement that $\rho(QM) = \mu(M)$ for complex perturbations. This requirement stems from the maximum modulus theorem for rational functions, extended to LFT's (see [39] and [37] for details). The maximum modulus theorem for rational functions states that given a function f analytic in an open connected region of the complex plane, then $|f|$ does not attain its maximum at an interior point of the region. This implies the maximum of $|f|$ is on the boundary. When this is extended to μ theory and LFT's, it

implies for complex perturbations that the maximum perturbation is not less than $\rho(QM)$, and therefore must be equal to $\rho(QM)$.

An underlying problem here is that the lower bound problem is not convex, so finding an optimal solution for the lower bound is not necessarily guaranteed. However, any local solution still gives a valid lower bound. Note that this proof is for the case where Δ is complex only. In the case where real uncertainties are included, the spectral radius must be replaced with the real spectral radius (ρ_R).

2.7 Limits of $\mu(M)$

The examples of the previous two sections have provided a complex μ upper bound and a mixed μ lower bound to confine the value of μ to

$$\max_{Q \in Q_{\mathcal{K}}} \rho_R(QM) \leq \mu(M) \leq \inf_{D \in \mathcal{D}} \overline{\sigma}(DMD^{-1}). \tag{2.56}$$

In the lower bound, the spectral radius has been replaced with the real spectral radius, hence, the lower bound is valid for all possible combinations of uncertainties, including real uncertainties.

Because finding μ is intractable, we settle for finding bounds on μ. These bounds can be improved by manipulations that do not change the value of μ for the system. Ultimately, this reduces our problem down to finding a unitary Q that maximizes the spectral radius of QM, and a structured similarity transform D, that commutes with Δ and minimizes the 2 norm of DMD^{-1}. These are the fundamentals of the μ problem with the exception of extending the μ upper bound to the mixed μ case. In depth coverage of the mixed μ case can be found in [18].

With the general μ problem well defined, it is possible to proceed to the skew μ problem.

Chapter 3

Formulating the General Skew μ Problem

3.1 Why Skew μ?

Suppose we have a gas turbine (jet engine) power generation system with uncertainty in some of the parameters and dynamics of the system. Specifically, let's consider the fuel BTU content, the ambient temperature, the system load, and the air inlet guide vane position. Even with detailed modeling, there are unmodeled dynamics and parameter uncertainty due to differences between theoretical and actual parts, as well as variations associated with the accuracy of sensor readings.

Presume that μ analysis returns a value of $\mu = 1.1$. This implies that reducing the uncertainty ranges by a factor of 1.1 would guarantee stability. Thus the expected range of Δ should be further restricted to $\frac{\Delta}{1.1}$.

Examine the reality of this. The loading of a generator can be restricted (relatively easily), and typically the inlet guide vane position can be measured and controlled tolerably well. This solves half the problem. BTU content is somewhat controllable, depending on the situation. If this were a stand-alone power station on an offshore platform in a gas field, its fuel would be supplied from the gas wells after being scrubbed (cleaned). The BTU content would be approximately known, but not guaranteed, nor absolutely controllable. The final item, ambient temperature plays a critical role in the air/fuel ratio and cannot be ignored.

The vast volumes of air consumed by a gas turbine prevent pre-conditioning the inlet air on a large scale. Since the range of ambient temperature variation cannot be changed, it must be considered fixed, and one must learn to deal with it. Thus, one can work to reduce the amount of uncertainty associated with 3 of the parameters, but ambient temperature range must be considered fixed (skewed).

Now the question. If one can vary the uncertainty size of the other parameters, but leave ambient air temperature fixed in range, what is the associated value of μ in this case? This value is referred to as *skew* μ since one or more of the parameters are fixed in variation range.

These situations arise often, perhaps more often than the standard μ problems (there's always something one doesn't have control over), hence the need for theory and numerical tools surrounding the skew μ problem.

Incidently, the standard method for dealing with ambient temperature variations in gas turbines is to recycle some of the turbine exhaust to warm the inlet air when the ambient temperature is too cold, and to de-rate the engine performance when the weather is too warm. Both cases sacrifice engine performance. If problem points can be more accurately predicted using skew μ then less conservative margins can be set and higher performance achieved from a gas turbine.

The primary point of this illustration is that skew μ is a very real tool, not just a theoretical exercise.

3.2 Skew μ-The Technical Definition

3.2.1 Structure of Perturbations

An LFT such as a standard $M-\Delta$ connection (see figure 2.9) can have the elements of M and Δ re-arranged via elementary row-column transformations with out affecting the intrinsic properties of the system or the interconnection. This was discussed in section 2.4. In other words, the rows and columns in M can be swapped and correspondingly in Δ, as

long as the interconnections are kept the same. This ability is needed for (1) understanding the process and (2) creating a numerical solution to problems. Any $M-\Delta$ connection can be taken and robust analysis will be valid if it is not rearranged; however, rearrangement aids the conceptual ease in deriving algorithms and designing numerical solutions.

Having made that point clear, it's now time to lay out some perturbation block structure that is important, but will look particularly messy on the page. Fortunately, it is similar to the structure discussed in section 2.4, so the extension should be reasonably understandable.

Presume there is a system with a perturbation Δ. Using elementary transformations on M and Δ, the structure of Δ can be rearranged to

$$\Delta = \begin{bmatrix} \Delta_f & 0 \\ 0 & \Delta_v \end{bmatrix}. \tag{3.1}$$

In this case, Δ_f represents perturbations of a maximum fixed range. All searches for the combination of perturbations that set $\det(I-M\Delta)=0$ must search on Δ_f less than or equal to the specified maximum. Δ_v, on the other hand, represents perturbations of a varying nature, which are shrunk or expanded to the size necessary to make $\det(I-M\Delta)=0$.

To be clear on this point, the Δ_f may be expanded as necessary up to a maximum fixed size in setting $\det(I-M\Delta)=0$. Δ_v may be expanded to any size necessary in setting $\det(I-M\Delta)=0$. Of course Δ_f and Δ_v must also satisfy block structure requirements per problem specifications.

As was explained in sections 2.3 and 2.4, we consider three ways to categorize perturbations, real (possibly repeated), complex (possibly repeated), and a full complex matrix block when the perturbation in question has no structure. The mathematical representation used for a real block is δ^r, the representation for a complex block is δ^c, and a full complex block representation is Δ^C. Each one of these types of perturbations can occur as a fixed or varying perturbation. It depends **only** on the physical nature of the problem.

The definition of skew μ is dependent upon the underlying block structure of the uncertainties. This structure is defined in the following paragraphs.

Given a matrix $M \in \mathbb{C}^{n\times n}$ which interacts with perturbations of a fixed and varying nature, then, define three non-negative integers relating to the number of fixed perturbations m_{r_f}, m_{c_f}, and m_{C_f} with $m_f \triangleq m_{r_f} + m_{c_f} + m_{C_f} \leq n_f$, where n_f is the total number of fixed perturbations. The block structure $\mathcal{K}_f(m_{r_f}, m_{c_f}, m_{C_f})$ is an m_f-tuple of positive integers

$$\mathcal{K}_f = (k_1, \ldots, k_{m_{r_f}}, k_{m_{r_f}+1}, \ldots, k_{m_{r_f}+m_{c_f}}, k_{m_{r_f}+m_{c_f}+1}, \ldots, k_{m_{r_f}+m_{c_f}+m_{C_f}}). \tag{3.2}$$

Similarly, define for varying perturbations three non-negative integers relating to the number of varying perturbations m_{r_v}, m_{c_v}, and m_{C_v} with $m_v \triangleq m_{r_v} + m_{c_v} + m_{C_v} \leq n_v$, where n_v is the total number of varying perturbations. The block structure $\mathcal{K}_v(m_{r_v}, m_{c_v}, m_{C_v})$ is an m_v-tuple of positive integers

$$\mathcal{K}_v = (k_1, \ldots, k_{m_{r_v}}, k_{m_{r_v}+1}, \ldots, k_{m_{r_v}+m_{c_v}}, k_{m_{r_v}+m_{c_v}+1}, \ldots, k_{m_{r_v}+m_{c_v}+m_{C_f}}). \tag{3.3}$$

It is also required that $\sum_{i=1}^{m_f} k_{f_i} + \sum_{i=1}^{m_v} k_{v_i} = n$ in order that these dimensions are compatible with M.

This determines the set of allowable perturbations, namely define for the fixed variations

$$\begin{aligned} Z_{\mathcal{K}_f} = \{\Delta_f = block\,diag(&\delta_1^r I_{k_1}, \ldots, \delta_{m_{r_f}}^r I_{k_{m_{r_f}}}, \delta_1^c I_{k_{m_{r_f}+1}}, \\ &\ldots, \delta_{m_{c_f}}^c I_{k_{m_{r_f}+m_{c_f}}}, \Delta_1^C, \ldots, \Delta_{m_{C_f}}^C) : \delta_i^r \in \mathbb{R}, \delta_i^c \in \mathbb{C}, \\ &\Delta_i^C \in \mathbb{C}^{k_{m_{r_f}+m_{c_f}+i} \times k_{m_{r_f}+m_{c_f}+i}}\}. \end{aligned} \tag{3.4}$$

The varying perturbations requires a similar definition (with subscript v dropped)

$$\begin{aligned} Y_{\mathcal{K}_v} = \{\Delta_v = block\,diag(&\delta_1^r I_{k_1}, \ldots, \delta_{m_{r_v}}^r I_{k_{m_{r_v}}}, \delta_1^c I_{k_{m_{r_v}+1}}, \\ &\ldots, \delta_{m_{c_v}}^c I_{k_{m_{r_v}+m_{c_v}}}, \Delta_1^C, \ldots, \Delta_{m_{C_v}}^C) : \delta_i^r \in \mathbb{R}, \delta_i^c \in \mathbb{C}, \\ &\Delta_i^C \in \mathbb{C}^{k_{m_{r_v}+m_{c_v}+i} \times k_{m_{r_v}+m_{c_v}+i}}\}. \end{aligned} \tag{3.5}$$

Note that $Y_{\mathcal{K}_v} \subset \mathbb{C}^{n_v \times n_v}$ and $Z_{\mathcal{K}_f} \subset \mathbb{C}^{n_f \times n_f}$. This block structure is sufficiently general to allow for repeated real scalars, repeated complex scalars, and full complex blocks in both fixed and varying cases.

Note that all the results which follow are easily generalized to the case where the full complex blocks need not be square, and the blocks may come in any order. We make these restrictions in equations 3.4 and 3.5 purely for notational convenience.

To clarify this explanation, its best to give an example. Consider a set

$$X_{\mathcal{K}_f} = \{\Delta_f = block\ diag(\delta_1^r I_1, \delta_1^c I_2, \Delta^C_{1_{2\times 2}}, \delta_2^r I_2, \delta_2^c I_1)\}. \tag{3.6}$$

As a matrix, this looks like

$$\Delta_f = \begin{bmatrix} \delta_1^r & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \delta_1^c & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta_1^c & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta^F_{1_{11}} & \Delta^F_{1_{12}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \Delta^F_{1_{21}} & \Delta^F_{1_{22}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta_2^r & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \delta_2^r & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \delta_2^c \end{bmatrix}. \tag{3.7}$$

There are two real perturbations, one with a multiplicity of 1, and a second with a multiplicity of 2. There are two complex perturbations, the first with a multiplicity of 2, and a second with a multiplicity of 1. There is one full complex block perturbation of size 2×2. The perturbation blocks of equation 3.6 can be rearranged to group the reals first, the repeated complex second, and the full block complex third,

$$\Delta_f = \begin{bmatrix} \delta_1^r & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \delta_2^r & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \delta_2^r & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \delta_1^c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \delta_1^c & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \delta_2^c & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta^C_{1_{11}} & \Delta^C_{1_{12}} \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta^C_{1_{21}} & \Delta^C_{1_{22}} \end{bmatrix}. \tag{3.8}$$

Likewise, the system matrix M must be rearranged to reflect the rearrangement in the perturbations. Note that this is an example for the the fixed perturbations. An identical procedure must be done for the varying perturbations as well. Ultimately, the fixed and varying perturbations are grouped in a diagonal matrix as in equation 3.1.

Δ^C in this example is square, but it is not necessarily required to be square, thus Δ and M are also not required to be square. Most results in this document will be demonstrated via examples using the square case. However, the theory and software tools handle the general case.

Now that the structure for the perturbations has been specified, skew μ can be properly defined.

3.2.2 Definition of Skew μ

The definition for skew μ is as follows:

Definition 1 *The skewed structured singular value $\mu_s(M)$ of a matrix $M \in \mathbb{C}^{m\times n}$ with respect to a block structure $\mathcal{K}_f(m_{r_f}, m_{c_f}, m_{C_f})$ and $\mathcal{K}_v(m_{r_v}, m_{c_v}, m_{C_v})$ is defined as:*

$$\mu_s(M) = \frac{1}{\min\limits_{\Delta \in W_{\mathcal{K}_f,\mathcal{K}_v}} \{\overline{\sigma}(\Delta_v) |\ det(I - \Delta M) = 0 for\ structured\ \Delta\}}$$

with $\mu_s(M) = 0$ if no $\Delta \in W_{\mathcal{K}_f,\mathcal{K}_v}$ solves $det(I - \Delta M) = 0$, and $\mu_s(M) = \infty$ if $\exists$ a $\begin{bmatrix} \Delta_f & 0 \\ 0 & 0 \end{bmatrix} \in W_{\mathcal{K}_f,\mathcal{K}_v}$ solving $\det(I - \Delta M) = 0$.

For a system M, with a set of perturbations, the definition of skew μ is the smallest SSV of a subset of perturbations that destabilizes the system M with the remainder of the perturbations being within a fixed size. Formally stating the restriction on perturbations of a fixed size,

$$\mathbf{B}Z_{\mathcal{K}_f} = \{\Delta_f \in Z_{\mathcal{K}} : \overline{\sigma}(\Delta_f) \leq 1\} \tag{3.10}$$

with the composite Δ perturbations as

$$W_{\mathcal{K}_f,\mathcal{K}_v} = \{\Delta = block\ diag(\Delta_f, \Delta_v)\}. \tag{3.11}$$

Splitting of perturbations into two sets with certain characteristics may be more easily understood by studying figure 3.1.

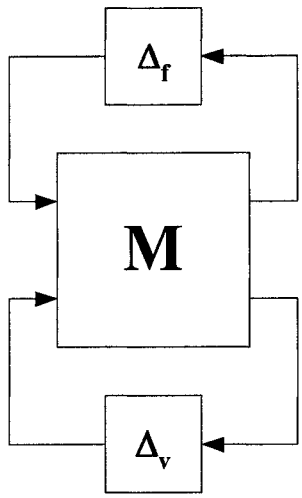
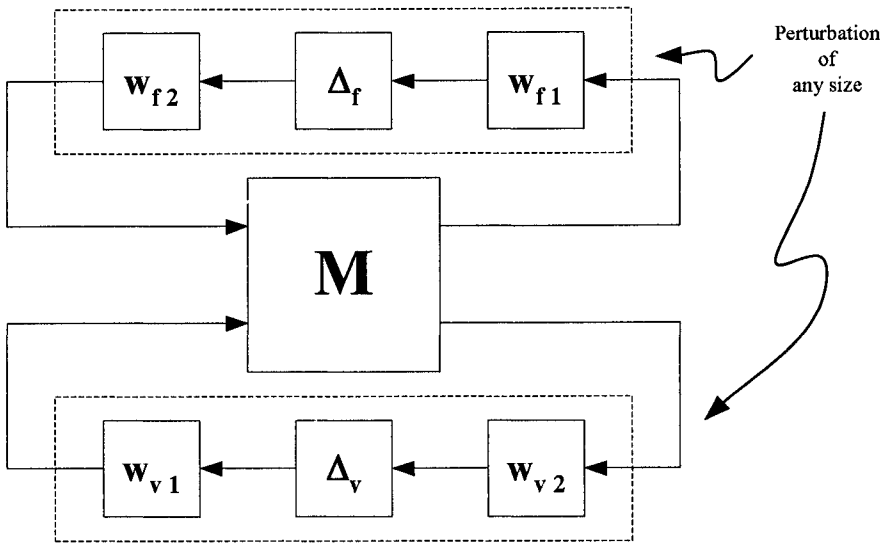


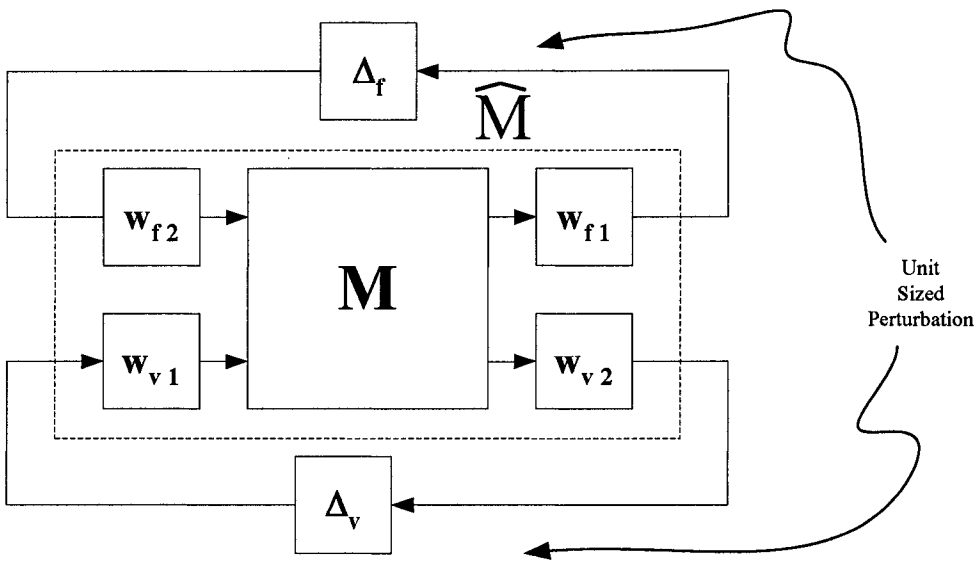
Figure 3.1: Spitting of perturbations into fixed range (Δ_f) and varying range (Δ_v) perturbations

Here **B** represents then n_f dimensional ball of Δ_f, which is restricts one to perturbations of size 1. The unit ball is indicated as a formality. In practice, the fixed perturbation has some fixed range, -1 to 10, 0 to 100, -50 to 50, etc., to which it is restricted. These ranges are normalized to unity and the associated scalings absorbed into M as demonstrated in figures 3.2.2 (a) and 3.2.2 (b). In these two figures, w represents scaling of the perturbation. This scaling is absorbed into M to form a new matrix $\hat{M}$, and the remaining perturbations, Δ_f and Δ_v, are of unit size.

The two extremes of μ_s are $\mu_s = 0$ and $\mu_s = \infty$. The case of $\mu_s = 0$ indicates there is no perturbation Δ in the perturbation set that has the potential to destabilize the system. As the destabilizing perturbations grow larger and larger, the value of skew μ_s grows smaller and smaller. Intuitively, as the destabilizing perturbation gets infinitely large,μ_s approaches 0. The other extreme case, $\mu_s = \infty$ indicates that as the destabilizing perturbation becomes tiny,μ_s grows extremely large. The implication of this is that there is no varying range perturbation (Δ_v) that can be found that does not potentially destabilize the system, hence $\mu_s = \infty$. A secondary implication of this condition is that the potential for destabilizing the



(a) System with perturbations



(b) System M with perturbations normalized and weights pulled into new system matrix $\hat{M}$

Figure 3.2: Normalization of perturbations and inclusion of weights in M

system can be accomplished by a fixed range perturbation (Δ_f) alone, irrespective of the varying range perturbation (Δ_v).

At this point, one more concept will be introduced. The concept is straight forward, but much of the remainder of this document relies on and uses this concept in deriving its results. Figure 3.3 displays the case where the perturbations have been split into fixed range and varying range components as previously explained. Additionally, in figure 3.3, it has been presumed that a value of skew $\mu = \nu$ has been found (some how) as well as a perturbations Δ_f and Δ_v that set the $\det(I - \Delta M) = 0$. This value ν has been included such that it can be wrapped either into the destabilizing varying perturbation or into the section of M that interacts with the varying perturbations. Given this setup, the following theorem can be formed.

Theorem 1 *Given a system M, $\mu_s(M) = \nu$ if and only if a matrix S can be constructed such that*

$$S = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \nu I_v \end{array}\right], \tag{3.12}$$

where I_f is of the dimension of the fixed range perturbations and I_v is of the dimension of the varying range perturbations. Additionally, the matrix M_s can be constructed where

$$M_s = S^{-1}M \tag{3.13}$$

and

$$\mu(M_s) = 1. \tag{3.14}$$

Proof: Apply definitions of skew μ and μ to figure 3.3.

Note that if $\mu_s(M) = \nu$, then wrapping $\frac{1}{\nu}$ into the section of M that interacts with the varying perturbation gives $\mu(M_s) = 1$, by the definitions of skew μ and μ. □

Remarks: Two other things should be noted between this theorem and figure 3.3. First, if a varying range perturbation, Δ_v has been found that could potentially destabilize the system, then constructing the perturbation

$$\hat{\Delta}_v = \frac{1}{\nu}\Delta_v$$

normalizes Δ_v such that

$$\|\hat{\Delta}_v\|_2 = 1.$$

Secondly, the product $S^{-1}M$ could just as well have been constructed as MS^{-1}, or $\sqrt{S^{-1}}M\sqrt{S^{-1}}$. Typically this document and its associated software tools uses the form $\sqrt{S^{-1}}M\sqrt{S^{-1}}$ for reasons explained in section 5.3.1.

The concept of forming $M_s = S^{-1}M$ and $\mu(M_s) = 1$ will be used throughout the remainder of this document to facilitate the construction of explanations, proofs and algorithms.

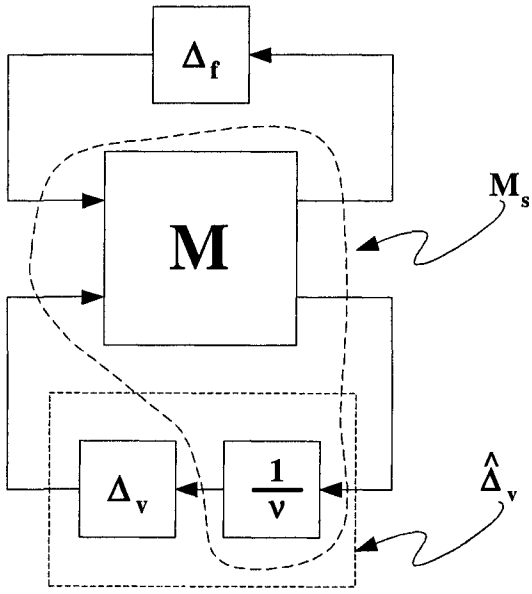


Figure 3.3: Graphical interpretation of skew μ

3.3 What's Gained with Skew μ ?

Clearly, the advantage of skew μ is an increased ability to accurately model and assess the stability of a multi-variable control system. As was previously pointed out, parameter

variations and modeling inaccuracies are not necessarily an option. They're a fact of life. Succinctly, skew μ allows for the quantification of multi-variable system stability where the variation of some parameters are of a known fixed range, and others are of an undetermined magnitude of variation. This is a more general case than μ, and has many applications. Like μ, it is NP hard. Consider a system M such that

$$M = \begin{bmatrix} 0 & 0 \\ 0 & M_{22} \end{bmatrix}. \tag{3.15}$$

If M were subject to a perturbation $\Delta = diag(\Delta_f, \Delta_v)$, then one could also say that

$$\mu_s(M) = \mu(M_{22}). \tag{3.16}$$

Thus finding skew μ (μ_s) is at least as hard as finding μ.

This implies that bounds need to be found in the skew μ case, just as bounds were found in the μ case. However, before going on to the development of skew μ bounds, a special application of skew μ will be developed. This particular problem is integral to the power system problem, as well as being an important case in general. As was alluded to earlier, this is the case where frequency is the skew variable.

3.4 Frequency as a Skew Variable

3.4.1 Sideris' Proposal

A continuous time SS system has the system transfer function ([46] among others)

$$G(s) = D + C(sI - A)^{-1}B. \tag{3.17}$$

To analyze this standard system format, μ analysis is performed in the frequency domain by analyzing systems at closely spaced individual frequencies of interest (*gridding*) in computing $\mu(G(j\omega))$. Using this method, there's a risk that the worst case value of $\mu(G(j\omega))$ is missed because a particular frequency point was not searched. To formally prove all relevant frequencies are examined requires verifying an infinite number of frequency points in a finite number of tests. A method for checking the robust stability of a linear system over a frequency range in a finite number of tests is given in [45]. This method treats the frequency as an uncertainty, which is augmented to the original system. In this manner, a worst case search is done over frequency as well as the other perturbations, insuring that no important frequency point is missed.

Using μ (not skew μ) frequency as a perturbation variable is subject to resizing to make $\det(I - M\Delta) = 0$. It's not desirable to specify a range of frequency as a perturbation only to have the range changed. Once a frequency range is chosen, one would like to have it stay fixed in the problem. Note that this is synonymous with the definition of a skew perturbation. Thus inclusion of frequency in a robust analysis is an ideal skew μ problem.

Before moving on to the development of skew μ bounds, the development of frequency as a skew variable needs to be formalized, and the equivalency between $\mu(G(j\omega)$ and $\mu(G(s))$ needs to be shown.

3.4.2 Equivalency of μ for Standard Systems and Frequency Augmented Systems

Inclusion of frequency as a perturbation variable is done via the Laplace variable s. The value of Sideris' proposal is directly coupled to the ability to show that $\mu(G(j\omega))$ (where ω indicates the system is examined at a specific frequencie) is equivalent to $\mu(G(s))$ where s varies over the right half of the Laplace plane (RHP). Stating this mathematically, it is necessary to show that

$$\sup_{\omega} \mu(G(j\omega)) = \sup_{Re(s)\geq 0} \mu(G(s)). \tag{3.18}$$

Equation 3.18 indicates that the bound on the largest value of μ found by examining $G(j\omega)$ over all frequencies is equivalent to the bound on the value of μ obtained by examining $G(s)$ over all $\{s : Re(s) \geq 0\}$. Formulating this appropriately:

Theorem 2 *Given a system $G(s)$, where $\mu(G(j\omega))$ indicates the value of $\mu(G(s))$ evaluated at a specific frequency ω, and $\mu(G(s))$ indicates the value of μ for the system evaluated over $\{s : Re(s) \geq 0\}$, then*

$$\sup_{\omega} \mu(G(j\omega)) = \sup_{Re(s)\geq 0} \mu(G(s)). \tag{3.19}$$

Proof: By the definition of ω, the LHS of equation 3.19 covers the imaginary axis through the range of the variable ω. Secondly, from complex variable analysis, the maximum modulus theorem states that: Given a non-empty, open connected region Ω, in the complex plane, and an analytic function F, then if $F \neq c$, a constant, then $|F|$ does not attain its maximum at an interior point of Ω.

Via the maximum modulus of the function $F(s)$, it can then be stated

$$\|F(s)\|_\infty = \sup_{Re(s)\geq 0} |F(s)| = \sup_{Re(s)=0} |F(s)|. \tag{3.20}$$

But

$$\sup_{Re(s)=0} |F(s)| = \sup_{\omega} \mu(F(j\omega)), \tag{3.21}$$

thus

$$\|F(s)\|_\infty = \sup_{Re(s)\geq 0} |F(s)| = \sup_{\omega} \mu(F(j\omega)). \tag{3.22}$$

This shows the equivalency between the maximum of $F(j\omega)$ and the maximum of $F(s)$. Next it is necessary to show that the maximum modulus property holds for μ. However, this has been previously done in theorem 8.6 of [37], thus $\sup_\omega \mu(G(j\omega)) = \sup_{Re(s)\geq 0}\mu(G(s))$.

□

Now that equivalency between $\mu(G(j\omega)$ and $\mu(G(s))$ has been shown, the framework surrounding incorporation of frequency as perturbation variable can be developed.

3.4.3 Incorporation of Frequency as an Uncertainty

Consider a SS system where all uncertainty variables are based on frequency. Pictorially the system would be represented as in figure 3.4, and is described by the equations

$$z = Cu + Dw \tag{3.23}$$

$$y = Au + Bw. \tag{3.24}$$

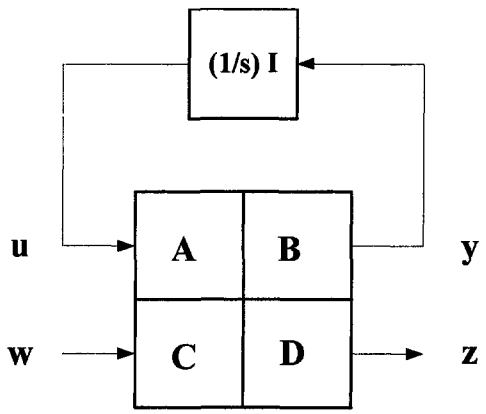


Figure 3.4: State space system LFT block diagram with frequency as the perturbation

The general equation for an upper (F_u) arrangement like 3.4 was given in equation 2.8, and is repeated here,

$$F_u(P,\Delta) \triangleq P_{22} + P_{21}\Delta(I - P_{11}\Delta)^{-1}P_{12}. \tag{3.25}$$

A little careful inspection reveals that equation 2.8 is equivalent to equation 3.17 when figure 3.4 is considered. This can be demonstrated as follows:

$$u = \frac{1}{s} Iy$$

$$y = A\frac{1}{s} Iy + Bw$$

$$(I - A\frac{1}{s})y = Bw$$

$$y = (I - A\frac{1}{s})^{-1}$$

$$u = \frac{1}{s} I(I - A\frac{1}{s})^{-1} Bw$$

$$z = C\frac{1}{s} I(I - A\frac{1}{s})^{-1} Bw + Dw$$

$$z = C(sI - A)^{-1} Bw + Dw$$

$$z = G(s)w$$

By noting that $\frac{1}{s} = \frac{1}{j\omega} = -j\frac{1}{\omega}$ and absorbing the $-j$ into the matrix M, figure 3.4 can be redrawn as figure 3.5.

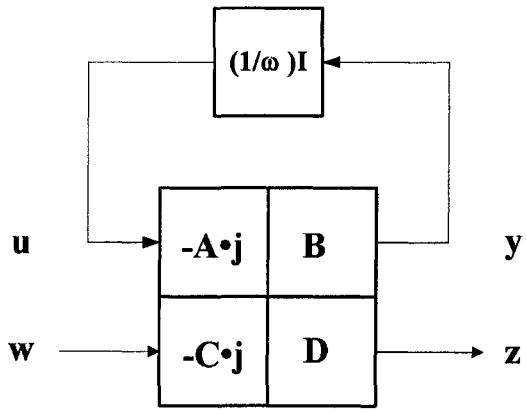


Figure 3.5: SS LFT block diagram with frequency perturbation. Here j is wrapped into system matrix.

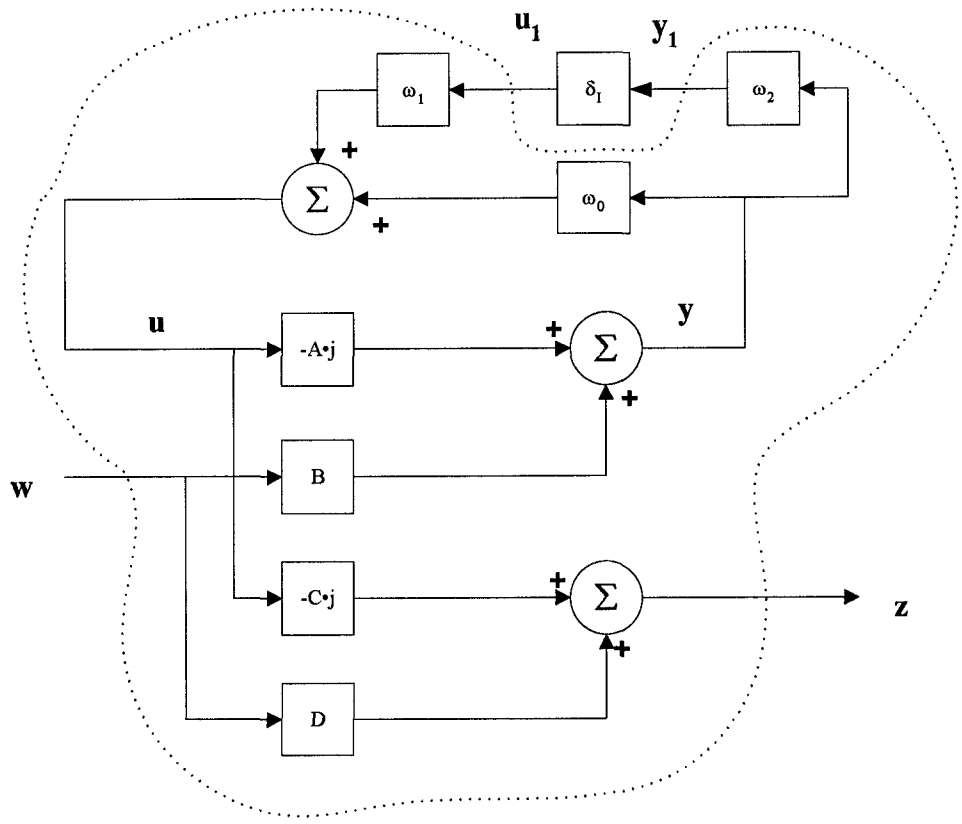


Figure 3.6: Detailed block diagram from which equations were derived

Defining the system output in terms of figure 3.5 gives a slightly modified version of equations 3.23 and 3.24,

$$z = -Cju + Dw \tag{3.26}$$

$$y = -Aju + Bw. \tag{3.27}$$

Additional figure 3.6 is given for clarity. Figure 3.6 includes ω_2, ω_1 and ω_0 for scaling the perturbations (δ_I), as well as the individual matrices broken out as single blocks. The details of δ_I,ω_2, ω_1 and ω_0 are explained later in this section.

Defining new relationships based on manipulating figures 3.4, 3.5 and 3.6

$$y_1 = \omega_2 y \tag{3.28}$$

$$u = \omega_1 u_1 + \omega_0 y. \tag{3.29}$$

Via substitution the following can be achieved,

$$y_1 = -Aj\omega_2(\omega_1 u_1 + \omega_0 y) + \omega_2 Bw = -Aj\omega_2\omega_1 u_1 - Aj\omega_2\omega_0 y_1 + \omega_2 Bw. \tag{3.30}$$

Reducing equation 3.30 gives

$$(I + Aj\omega_0)y_1 = -Aj\omega_2\omega_1 u_1 + \omega_2 Bw, \tag{3.31}$$

or

$$y_1 = [-j\omega_2(I + Aj\omega_0)^{-1}A\omega_1]u_1 + [\omega_2(I + Aj\omega_0)^{-1}B]w. \tag{3.32}$$

This is the first relationship needed. Now, to work on the relationship for z. First, a closed form solution for y is needed. Achieving this form via substitution,

$$y = -Aj\omega_0 y - Aj\omega_1 u_1 + Bw, \tag{3.33}$$

and re-arrangement,

$$y = (I + j\omega_0 A)^{-1}(-Aj\omega_1 u_1 + Bw). \tag{3.34}$$

Substituting this into z (via equations 3.26 and 3.29),

$$z = -Cj\omega_1 u_1 + Cj\omega_0(I + Aj\omega_0)^{-1}Aj\omega_1 u_1 + Dw - Cj\omega_0(I + Aj\omega_0)^{-1}Bw. \tag{3.35}$$

Rearranging equation 3.35 gives

$$z = [-Cj\omega_1 + Cj\omega_0(I + Aj\omega_0)^{-1}Aj\omega_1]u_1 - [Cj\omega_0(I + Aj\omega_0)^{-1}B - D]w. \tag{3.36}$$

Since $(I + A)^{-1}A - I = -(I + A)^{-1}$, we can rewrite equation 3.36 as

$$z = [-Cj\omega_1(I + Aj\omega_0)^{-1}]u_1 - [Cj\omega_0(I + Aj\omega_0)^{-1}B - D]w. \tag{3.37}$$

Equations 3.32 and 3.37 translate into equation 3.38,

$$\begin{aligned} G_{11} &= -j\omega_2(I + Aj\omega_0)^{-1}A\omega_1 \\ G_{12} &= \omega_2(I + Aj\omega_0)^{-1}B \\ G_{21} &= -Cj\omega_1(I + Aj\omega_0)^{-1} \\ G_{22} &= -Cj\omega_0(I + Aj\omega_0)^{-1}B + D. \end{aligned} \tag{3.38}$$

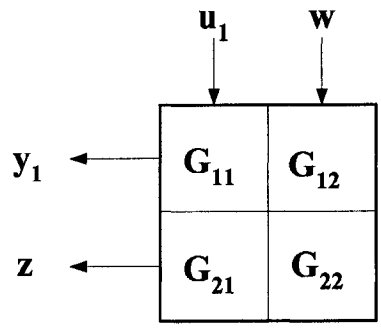


Figure 3.7: Transformed system matrix with system inputs

which can be drawn as a new system diagram, and is depicted in figure 3.7.

Typically, it is desired to have δ_I to range from 1 to -1. Figure 3.6 shows the relationship between δ_I and $\frac{1}{\omega}$ to be

$$\frac{1}{\omega} = \omega_0 + \omega_1 \delta_I \omega_2. \tag{3.39}$$

This implies using the variable ω_0 to establish the frequency range midpoint, and the variables ω_1,ω_2 to span from the midpoint to the minimum and maximum variations. Thus the midpoint can be defined as

$$\omega_0 = \frac{\frac{1}{\omega_{min}} + \frac{1}{\omega_{max}}}{2}, \tag{3.40}$$

and the spanning parameter as

$$\omega_1 = \omega_2 = \sqrt{\frac{\frac{1}{\omega_{min}} - \frac{1}{\omega_{max}}}{2}}. \tag{3.41}$$

The spanning parameter has been split between the two variables ω_1 and ω_2 to improve the numerical properties of matrix computations associated with the equations in 3.38, used later in this work in section 6.5.

To verify that formulating the frequency component as a perturbation finds the value of μ over the frequency range, consider the following proof.

Let M be the original system, $\Delta = diag\{\Delta_1, \ldots, \Delta_n\}$ be the original system perturbations, and $j = \sqrt{-1}$. Starting from the non-normalized formulation in equations 3.26 and 3.27

form the matrix

$$G = \begin{bmatrix} -jA & B \\ -jC & D \end{bmatrix}. \tag{3.42}$$

Next, exploit the fact that ω is real, and form the augmented perturbation matrix

$$\tilde{\Delta} = \begin{bmatrix} \frac{1}{\omega} I_m & 0 \\ 0 & \Delta \end{bmatrix}. \tag{3.43}$$

In this matrix, it has been assumed WLOG the frequency components occupy the upper left quadrant of $\tilde{\Delta}$.

Then $\det[I - M(j\omega)\Delta] = 0$ if and only if $\det[I - G\tilde{\Delta}] = 0$.

To show this, construct

$$\begin{aligned} \det[I - G\tilde{\Delta}] &= \det\left\{ \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} -jA & B \\ -jC & D \end{bmatrix} \cdot \begin{bmatrix} \frac{1}{\omega} I_m & 0 \\ 0 & \Delta \end{bmatrix} \right\} \\ &= \det \begin{bmatrix} I + \frac{j}{\omega} A & -B\Delta \\ \frac{j}{\omega} C & I - D\Delta \end{bmatrix} \\ &= \det[I - \tfrac{1}{j\omega} A] \det[I - D\Delta + \tfrac{j}{\omega} C(I + \tfrac{j}{\omega} A)^{-1} B\Delta] \\ &= \det[I - \tfrac{1}{j\omega} A] \det[I - (C(j\omega I - A)^{-1} B + D)\Delta] \\ &= \det[I - \tfrac{1}{j\omega} A] \det[I - M(j\omega)\Delta] \end{aligned} \tag{3.44}$$

This is achieved by applications of Schur's determinant formula [46], and some manipulation.

Since the determinant of $[I - \frac{1}{j\omega} A]$ must be

$$det[I - \frac{1}{j\omega} A] \neq 0 \tag{3.45}$$

to satisfy nominal stability, then the determinant can only be made 0 if

$$\det[I - M(j\omega)\Delta] = 0. \tag{3.46}$$

The concept of skew μ applied to this case implies the uncertainty range of ω is just sufficient to set the determinant of the system to 0. However, frequency is not the main concern. The item of interest is the value of the remaining perturbations (Δ) that sets the

determinant to 0. Clearly this is the value of Δ that makes $\det[I - M(j\omega)\Delta] = 0$. Note that $\det[I - M(j\omega)\Delta] = 0$ is equivalent to the requirement in the definition of $\mu(M)$ with a perturbation set Δ. The frequency augmented skew μ problem equates to the $\mu(M)$ problem with the added stipulation of being computed over a frequency interval rather than at specific frequency points.

This formulates the skewed frequency robust analysis problem. An alternate approach to this formulation can be made by performing a bilinear transformation of the Laplace variable s to the discrete variable z and then covering with a complex (versus real) parameter. This alternative is shown in the following section.

3.4.4 Alternate Single Test Method from the z -Plane

3.4.4.1 Motivation for Alternate Test Method

To provide for a complete view of augmenting SS systems with frequency, an alternate method is described in the following section. This method augments frequency to the SS model via the use of z-plane methods rather than s-plane methods as done previously in this chapter. This yields some advantages, but also has some disadvantages. However, a review of this method adds to the overall knowledge of the reader and thus it is elucidated here.

The computational aspects of computing bounds on μ are more troublesome for real variables than complex. The conservative nature of real perturbation bounds can be improved by the application of additional techniques (see section 5.6.3 for more explanation). However, in the case of augmenting the frequency variable as a perturbation, an alternate approach is afforded by applying a secondary transformation, the bilinear z transform, to the Laplace variable.

There are several motivating factors for this. The first is that the z-plane rendition augments frequency as a complex perturbation rather than a real perturbation, which typically yields a tighter bound than real perturbations. The second is that the frequency range $\omega = 0$ to

∞ in the Laplace plane is mapped onto the circumference of the semi-circle $z = 1$ in the z-plane. In short, one test over the parameter range $z \leq 1$ tests the whole frequency spectrum. A more detailed discussion of these factors follows.

3.4.4.2 Discussion of Differences Between the Two Test Methods

Sideris' technique could be represented as frequency augmentation where frequency was a complex perturbation. The drawback with this approach is that the $Re(s) = 0$, thus the perturbation was along the imaginary axis. Complex perturbations are modeled as a disc in the complex plane, hence a bound provided using frequency as a complex perturbation might be overly conservative because it does not make use of the extra phase information (i.e. $Re(s) = 0$). This is improved somewhat by including the frequency perturbation as a real variable (ω), wrapping the complex parameter j into the matrix M, and then applying additional techniques of section 5.6.3 to improve bounds computation. However, real perturbations can yield bounds that may be significantly far away from the actual value of μ or skew μ. Thus, this is the first method of handling frequency augmentation: Change the perturbation to a real perturbation problem where techniques are known for handling real perturbations.

The second method of handling frequency augmentation is to change the perturbation from a complex perturbation where $Re(s) = 0$ to a complex perturbation where the frequency varies on a disc. This option is provided by the z-transform since $\omega = 0$ to ∞ along the imaginary axis in the Laplace variable maps to a disc with a radius of $z = 1$ in the z-plane.

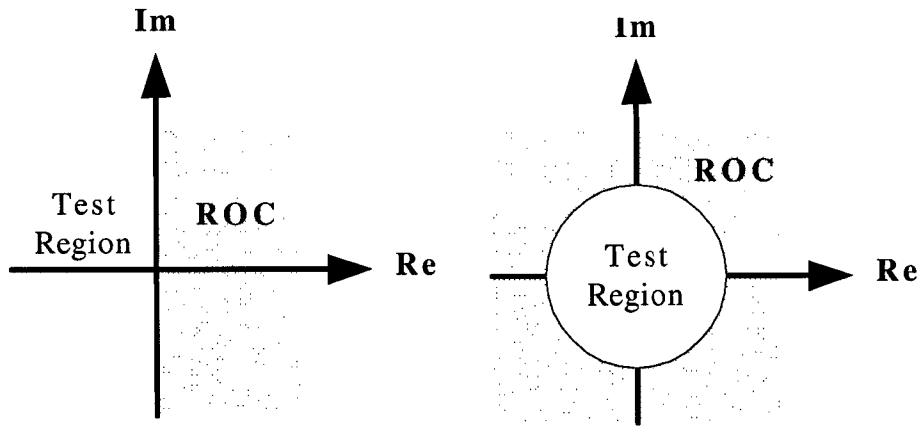
3.4.4.3 Mapping Laplace Plane to the Z Plane

SS systems generate the well known transfer function

$$M(s) = C(sI - A)^{-1}B + D.$$

For practical problems, one expects the system to be nominally stable, or equivalently, all its poles in the LHP. In terms of one sided Laplace transformation theory this implies the

transfer function has a region of convergence (ROC) in the RHP. The implication of the ROC in terms of using s as a perturbation variable means that the perturbation exists in the LHP. Consequently, when the Laplace plane is mapped to the z plane, then the ROC for the z transform is external to the unit disk, and the search region for the variable z is inside the unit disk. This correspondence is depicted in figure 3.8.



(a) Laplace plane for half plane perturbation

(b) z or δ plane for disk perturbation

Figure 3.8: Mapping ROC and search region from s-plane to z-plane for complex perturbations

Since the one sided Laplace transform exists on the half-plane, and the z transform for causal systems exists on a disk, the perturbation boundary described by the Laplace transform would exist along a line in the half plane, where as the perturbation boundary described by the z-transform would exist on the circumference of a disc. Bounding of μ values attempts to cover uncertainties with disks of various radii. If the uncertainty boundary exists along a line in a half plane then the bounding disk is far more conservative than necessary. Figure 3.9 gives a graphical representation of this idea.

3.4.4.4 Developing a One Shot Test

A finite frequency interval is easy to cover using Sideris' frequency augmentation technique, but an infinite frequency range such as zero to infinity is hard to do. When the problem is couched in the z-plane format, the frequency range from zero to infinity may be

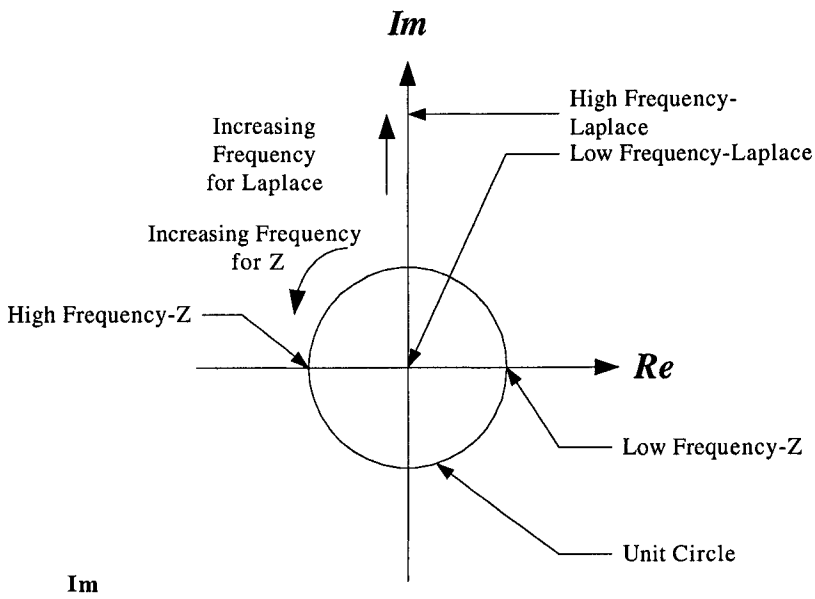


Figure 3.9: Laplace and Z transform variable mapping

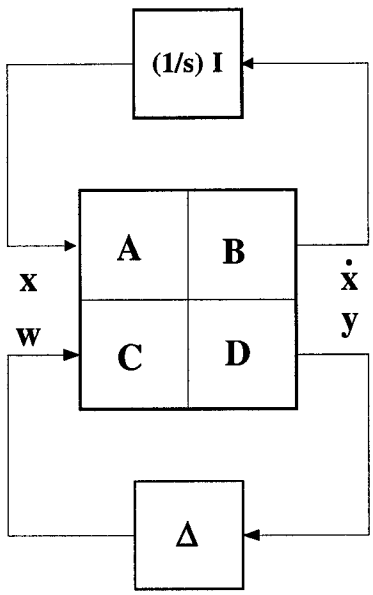
done as a one shot test from $z = 1$ to $z = -1$. If the one shot test yields an answer suitable to the purpose, then no more investigation is required. However, if the one shot test does not yield a definitive answer, then it may be necessary to subdivide the frequency range. Subdividing the frequency interval is hard to do when the problem is placed in the z-plane format since z is a complex variable. This is the primary drawback to using the z-plane formulation of the perturbation.

The mechanics of conversion from a real parameter variation to a complex parameter variation is performed by replacing the Laplace variable with the bilinear z transform,

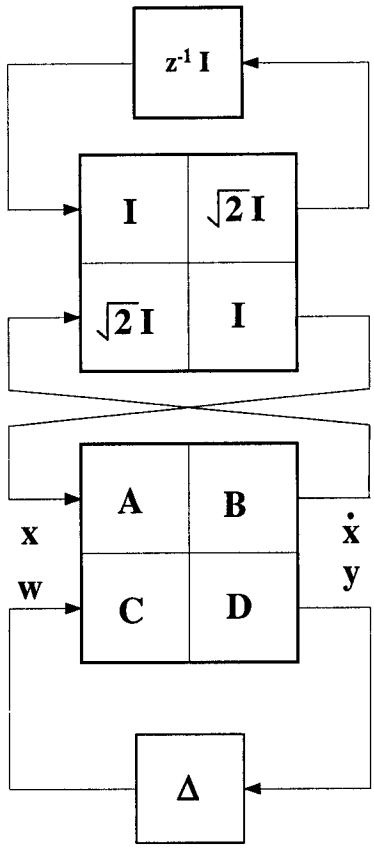
$$\frac{1}{s} = \frac{z+1}{z-1} = \frac{1+\delta}{1-\delta}, \tag{3.47}$$

where $\delta = z^{-1}$. Using a bilinear z transform of the Laplace variable, a second LFT can be formed from equation 3.47 and substituted in to replace the Laplace variable as shown in figure 3.10 (a)-(c).

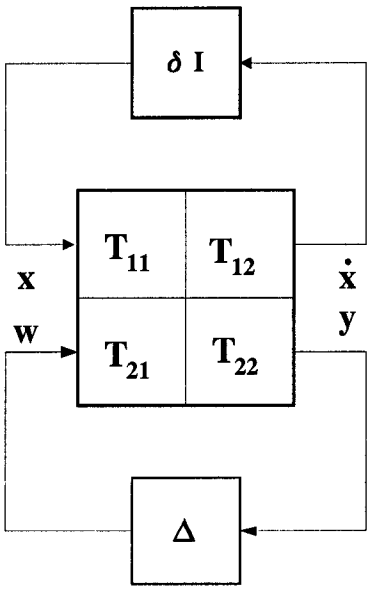
An interconnection of LFT's is itself, an LFT [46]. Thus, reduction to figure 3.10 (c) is accomplished through successive applications of the F_u equation for reducing LFT's. The



(a) Original frequency augmentation



(b) Bilinear Z replaces Laplace variable



(c) Simplification of the interconnected LFT's, with δ replacing z^{-1}

Figure 3.10: Mapping frequency from an *s*-plane to a *z*-plane variable via LFT's

system matrix for the reduced figure 3.10 (c) is

$$T = \begin{bmatrix} (I+A)(I-A)^{-1} & \sqrt{2}(I-A)^{-1}B \\ \sqrt{2}C(I-A)^{-1} & C(I-A)^{-1}B+D \end{bmatrix}. \tag{3.48}$$

Inclusion of frequency as a perturbation has come down to the complex parameter, $z \in \mathbb{C}$. Generally problems with complex uncertainties provide computationally less conservative answers than the real parameter ω. However, complex parameters provide difficulty when one wants to divide up the perturbations in sectors, hence the methods of chapter 7 must be applied. The pros and cons of these two approaches are summarized in table 3.1.

Approach	Sideris	Bilinear
Pros	Easy for intervals and B&B	Handled with complex Δ's
Cons	Real Δ's	Not easy for intervals, B&B

Table 3.1: Table of frequency augmentation approaches, Pros and Cons

During analysis of the power system problem, it is the real parameter ω that will be used. This decision is primarily based on the need to cut up the frequency interval into smaller sections for more accurate results. Before the power system problem can be analyzed,however, skew μ bounds need to be developed so that computational methods can be created for bounding skew μ in real world problems.

Chapter 4

Developing a Skew μ Lower Bound

Following the same pattern as previous sections, a lower bound for skew μ will be developed out of existing proofs for μ. The rationale for this is that the people who are sincerely interested in the derivations and proofs contained here will most likely be familiar with the existing proofs for the standard μ lower bound. The evolution of the lower bound for μ can be traced over a ten year period through a number of papers. The first impressions and commentaries can be found in [11] and [16]. These concepts are developed through a number of other works, notably [37] and [53]. The philosophy behind the lower bound is that (complex) μ is invariant to unitary transformations, Q, but the spectral radius of a matrix, $\rho(M)$, is not. If the *right* unitary transformation is chosen, then the spectral radius of the product, $\rho(QM)$ is maximized. This spectral radius is a lower bound on μ, as was shown for the complex case in section 2.6. To extend this to skew μ requires first extending the μ lower bound to real and full block perturbations, and then extending that situation to the problem where some of the perturbations are fixed in range. Keeping this in mind, the next topic is to review the results for the lower bound for the μ problem which includes real uncertainties as well as complex.

4.1 Notation for the Lower Bound

In order to refine the lower bound, some sets need to be defined which will be of use later. As a reminder, the notation for the set of structured perturbations previously defined in

section 2.4is repeated here.

$$X_{\mathcal{K}} = \{\Delta = block\ diag(\delta_1^r I_{k_1}, \ldots, \delta_{m_r}^r I_{k_{m_r}}, \delta_1^c I_{k_{m_r+1}}, \ldots, \delta_{m_c}^c I_{k_{m_r+m_c}}, \Delta_1^C, \ldots, \Delta_{m_C}^C) : \\ \delta_i^r \in \mathbb{R}, \delta_i^c \in \mathbb{C}, \Delta_i^C \in \mathbb{C}^{k_{m_r+m_c+i} \times k_{m_r+m_c+i}}\} \quad (4.1)$$

The following sets of block diagonal matrices are dependent on the underlying block structure and are noted as

$$Q_{\mathcal{K}} = \{\Delta \in X_{\mathcal{K}} : \delta_i^r \in [-1\ 1], \delta_i^{c*}\delta_i^c = 1, \Delta_i^{C*}\Delta_i^C = I_{k_{m_r+m_c+i}}\} \quad (4.2)$$

and

$$\breve{\mathcal{D}}_{\mathcal{K}} = \{block\ diag(e^{j\theta_1}D_1, \ldots, e^{j\theta_{m_r}}D_{m_r}, D_{m_r+1}, \ldots, D_{m_r+m_c}, d_1 I_{k_{m_r+m_c+1}}, \ldots, \\ d_{m_C} I_{k_m}) : \theta_i \in [-\frac{\pi}{2}\ \frac{\pi}{2}], 0 < D_i = D_i^* \in \mathbb{C}^{k_i \times k_i}, 0 < d_i \in \mathbb{R}\}. \quad (4.3)$$

The set $\breve{\mathcal{D}}_{\mathcal{K}}$ is defined differently from $\mathcal{D}_{\mathcal{K}}$ in chapter 2 to specifically address the issue of real perturbations. Note that these are not quite the "usual" scaling sets associated with the complex μ problems as explained in chapter 2 (or see [38]), since for $m_r \neq 0$ matrices in $Q_{\mathcal{K}}$ are not necessarily unitary, and matrices in $\breve{\mathcal{D}}_{\mathcal{K}}$ are not necessarily Hermitian. These sets are chosen for reasons which will become clear later. Note that for any $\Delta \in X_{\mathcal{K}}$ and any $D \in \breve{\mathcal{D}}_{\mathcal{K}}$, $D\Delta = \Delta D$.

Further, in order to refine the lower bound, define the unit ball in the perturbation set as

$$\mathbf{B}X_{\mathcal{K}} = \{\Delta \in X_{\mathcal{K}} : \overline{\sigma}(\Delta) \leq 1\}. \quad (4.4)$$

4.1.1 Notational and Conceptual Problems Caused by Real Perturbations

The μ lower bound for the purely complex case ($m_r = 0$) is an eigenvalue maximization problem, whereas in the mixed case it is a *real* eigenvalue maximization (i.e. ρ_R) problem. While $\max_{Q \in Q_{\mathcal{K}}} \rho(QM) = \mu_{\mathcal{K}}(M)$ for the purely complex case, as was shown in section 2.6, it also holds with equality for the mixed case as shown in the following theorem from [53].

Theorem 3 *[53] For any matrix $M \in \mathbb{C}^{n\times n}$, and any compatible block structure $\mathcal{K}$*

$$\max_{Q\in Q_{\mathcal{K}}} \rho_R(QM) = \mu_{\mathcal{K}}(M) \tag{4.5}$$

However, the definition of $Q_{\mathcal{K}}$ requires the search over the full range of the real perturbations. This causes a few notational problems and difficulties which are explained in the following paragraphs.

Note that for any $Q \in Q_{\mathcal{K}}$ and any $\Delta \in \mathbf{B}X_{\mathcal{K}}$, then $Q\Delta \in \mathbf{B}X_{\mathcal{K}}$ and $\Delta Q \in \mathbf{B}X_{\mathcal{K}}$, since a unitary rotation leaves the 2-norm of a matrix unchanged. Suppose some matrix $Q \in Q_{\mathcal{K}}$ achieves a local maximum of $\rho_R(QM)$ over $Q \in \mathbf{B}X_{\mathcal{K}}$, then it is easy to show that the matrix $\hat{M} \triangleq QM$ has a local maximum of $\rho_R(\hat{Q}\hat{M})$ over $\hat{Q} \in \mathbf{B}X_{\mathcal{K}}$ at $\hat{Q} = I$. This point may seem rather obvious, but its necessary for the next stage of this explanation.

Since the *real* elements of Q are not restricted to be on their boundary one can say more than this. For any matrix $Q \in Q_{\mathcal{K}}$ (see equation (4.2)) define the index sets

$$\mathcal{J}(Q) = \{i \le m_r : |\delta_i^r| = 1\} \tag{4.6}$$

$$\hat{\mathcal{J}}(Q) = \{i \le m_r : |\delta_i^r| < 1\} \tag{4.7}$$

and define the allowable perturbation set as before, except with the real perturbations noted in terms of these index sets, i.e.

$$\hat{\mathbf{B}}\Delta_\varepsilon(\mathcal{J},\hat{\mathcal{J}}) = \{\Delta \in X_{\mathcal{K}} : |\delta_i^r| \le 1, i \in \mathcal{J}, |\delta_i^r| < 1+\varepsilon, i \in \hat{\mathcal{J}}, |\delta_i^c| \le 1, i = 1,\ldots,m_c, \\ \overline{\sigma}(\Delta_i^C) \le 1, i = 1,\ldots,m_C\} \tag{4.8}$$

One sees that for sufficiently small $\varepsilon > 0$ for any $Q \in Q_{\mathcal{K}}$ and any $\Delta \in \hat{\mathbf{B}}\Delta_\varepsilon(\mathcal{J}(Q),\hat{\mathcal{J}}(Q))$, then $Q\Delta \in \mathbf{B}X_{\mathcal{K}}$ and $\Delta Q \in \mathbf{B}X_{\mathcal{K}}$. The point of all this is that if some matrix $Q \in Q_{\mathcal{K}}$ achieves a local maximum of $\rho_R(QM)$ over $Q \in \mathbf{B}X_{\mathcal{K}}$ then the matrix $\hat{M} \triangleq QM$ has a local maximum of $\rho_R(\hat{Q}\hat{M})$ *over* $\hat{Q} \in \hat{\mathbf{B}}\Delta_\varepsilon(\mathcal{J}(Q),\hat{\mathcal{J}}(Q))$ (for some $\varepsilon > 0$) at $\hat{Q} = I$ (and in fact the converse is true provided the assumption that for every i, $\delta_i^r \neq 0$).

The notation here is rather cumbersome and tends to obscure a rather simple concept. All the above says is that if $\rho_R(QM)$ is at a maximum point with some of the real perturbations at interior points of the ball (this is not a possibility for the complex perturbations) then one stays inside the allowable set, and cannot increase $\rho_R(QM)$ if the real perturbations are moved *up* or down in magnitude. Hence the requirement of $\hat{Q} = I$. If this were not so, then there would be a $\hat{Q}$ that could rotate the magnitude of the real perturbations (up or down) to increase $\rho_R(QM)$. Complex perturbations achieve their maximum on the boundary. However, this is not always true for real perturbations. This proceeding explanation clarifies that if a maximum is achieved with the real perturbation at an interior point of the ball, any change in the real perturbation will decrease $\rho_R(QM)$, regardless if the change moves the real perturbation closer to the boundary of the ball or not.

One further piece of notation is introduced here. Suppose $M \in \mathbb{C}^{n\times n}$ has an eigenvalue λ with right and left eigenvectors x and y respectively. Then partition x and y compatibly with the block structure as

$$x = \begin{bmatrix} x_{r_1} \\ \vdots \\ x_{r_{m_r}} \\ x_{c_1} \\ \vdots \\ x_{c_{m_c}} \\ x_{C_1} \\ \vdots \\ x_{C_{m_C}} \end{bmatrix}, \qquad y = \begin{bmatrix} y_{r_1} \\ \vdots \\ y_{r_{m_r}} \\ y_{c_1} \\ \vdots \\ y_{c_{m_c}} \\ y_{C_1} \\ \vdots \\ y_{C_{m_C}} \end{bmatrix} \tag{4.9}$$

where $x_{r_i}, y_{r_i} \in \mathbb{C}^{k_i}$, $x_{c_i}, y_{c_i} \in \mathbb{C}^{k_{m_r+i}}$, $x_{C_i}, y_{C_i} \in \mathbb{C}^{k_{m_r+m_c+i}}$. These will be referred to as the "block components" of x and y, and the "non-degeneracy" assumption is defined to be that for every i (in the appropriate set), $|y_{r_i}^* x_{r_i}| \neq 0, |y_{c_i}^* x_{c_i}| \neq 0, |y_{C_i}||x_{C_i}| \neq 0$.

The creation of notational tools has been lengthy, but using these notations, development for the standard μ problem can now proceed.

4.2 Results for the Standard μ Problem

Theorem 2 from [53] gives a characterization of a maximum point of $\rho_R(QM)$ in terms of an alignment of the right and left eigenvectors of QM. This leads directly to the following decomposition from [53].

Theorem 4 *[53] Suppose $Q \in Q_{\mathcal{K}}$ achieves a local maximum of $\rho_R(QM)$ over $Q \in \mathbf{B}X_{\mathcal{K}}$, and that the eigenvalue achieving $\rho_R(QM)$, denoted β, is distinct and positive. Then if the right and left eigenvectors of QM, denoted x and y respectively, satisfy the non-degeneracy assumption, there exists a matrix $D \in \breve{\mathcal{D}}_{\mathcal{K}}$ with $D^2 \in \breve{\mathcal{D}}_{\mathcal{K}}$ and $\theta_i = \pm\frac{\pi}{4}$ for $i \in \hat{\jmath}(Q)$ such that*

$$\begin{aligned} QDMD^{-1}(Dx) &= \beta Dx \\ (x^*D^*)QD^*M(D^*)^{-1} &= \beta x^*D^* \end{aligned} \tag{4.10}$$

with $\beta \leq \mu_{\mathcal{K}}(M)$. Furthermore if the above maximum is global then $\beta = \mu_{\mathcal{K}}(M)$.

It is well known that for the purely complex case there is a decomposition at μ (see Packard et. al. [41] and related work by Daniel et. al. [7]) and equation (4.10) extends this result to the mixed case ($m_r \neq 0$).

Thus one (almost) always has a decomposition at μ of the form of equation (4.10), and any such decomposition gives a lower bound for μ. From [53], this condition is reformulated into a set of vector equations as follows.

Lemma 1 *[53] Suppose there exists matrices $Q \in Q_{\mathcal{K}}$ with $\delta_i^r \neq 0$ for $i = 1, \ldots, m_r$ and $\hat{D} \in \breve{\mathcal{D}}_{\mathcal{K}}$ with $\hat{D}^2 \in \mathcal{D}_{\mathcal{K}}$ and $\hat{\theta}_i = \pm\frac{\pi}{4}$ for $i \in \hat{\jmath}(Q)$. Then there exists a non-zero vector $\hat{x}$, and a real positive scalar β such that*

$$\begin{aligned} Q\hat{D}M\hat{D}^{-1}(\hat{D}\hat{x}) &= \beta\hat{D}\hat{x} \\ (\hat{x}^*\hat{D}^*)Q\hat{D}^*M(\hat{D}^*)^{-1} &= \beta\hat{x}^*\hat{D}^* \end{aligned} \tag{4.11}$$

iff there exists a matrix $D \in \breve{\mathcal{D}}_{\mathcal{K}}$ *with* $\theta_i = \pm\frac{\pi}{2}$ *for* $i \in \hat{\jmath}(Q)$ *and non-zero vectors* b,a,z,w *such that*

$$\begin{aligned} Mb &= \beta a & M^*z &= \beta w \\ b &= Qa & b &= D^{-1}w \\ z &= Q^*QDa & z &= Q^*w \end{aligned} \tag{4.12}$$

Proof: ($\Rightarrow$) Define $x = \hat{D}\hat{x}$ and b,a,z,w as

$$\begin{aligned} b &= \hat{D}^{-1}x & a &= \hat{D}^{-1}Q^{-1}x \\ z &= \hat{D}Q^*x & w &= \hat{D}x \end{aligned}$$

Finally define $D = \hat{D}^2$ and the result follows.

($\Leftarrow$) Defining $\hat{D}$ as the unique matrix $\hat{D} \in \breve{\mathcal{D}}_{\mathcal{K}}$ such that $\hat{D}^2 = D$, and $\hat{x} = b$ the result follows directly. □

4.3 A Lower Bound Power Algorithm for the Standard μ Problem

In light of lemma 1, the problem of computing a lower bound for $\mu_{\mathcal{K}}(M)$ is reduced to one of finding a solution to the set of equations in 4.12 which gives a decomposition as in equation 4.10. The desire is to develop an algorithm for computing such a solution. In order to do this one first notes that if b,a,z,w are partitioned compatibly with the block structure as in 2.29 then the set of constraint equations

$$\begin{aligned} b &= Qa & b &= D^{-1}w \\ z &= Q^*QDa & z &= Q^*w \end{aligned}$$

can be broken down into a series of m similar independent constraint equations on the block components (since Q and D are block diagonal). These equations are of three types corresponding to a repeated real scalar block, a repeated complex scalar block, or a full complex block. Algorithm development is aided by the following lemmas, which address a generic constraint of each type. Three of the following lemmas are due to Packard [41],

and reproduced here with an additional lemma, all which can be found in [53]. Lemma 2 is necessary in the proof of lemma 5, thus the reason for its inclusion here.

Lemma 2 ([41]) *Let $y \in \mathbb{C}^n$ and $x \in \mathbb{C}^n$ be non-zero vectors. Then there exists a Hermitian, Positive Definite $D \in \mathbb{C}^{n\times n}$ such that $y = Dx$ iff $y^*x \in \mathbb{R}$ and $y^*x > 0$.*

Lemma 3 (Repeated Complex Scalar Block [41]) *Let $b, a, z, w \in \mathbb{C}^k$ be non-zero vectors with $a^*w \neq 0$. Then there exists a complex scalar q with $|q| = 1$, and a complex matrix $D \in \mathbb{C}^{k\times k}$ with $0 < D = D^*$ such that*

$$\begin{aligned} b &= qa & b &= D^{-1}w \\ z &= q^*qDa & z &= q^*w \end{aligned}$$

if and only if

$$z = \frac{w^*a}{|w^*a|}w \qquad b = \frac{a^*w}{|a^*w|}a \tag{4.13}$$

Lemma 4 (Full Complex Block [41]) *Let $b, a, z, w \in \mathbb{C}^k$ be non-zero vectors. Then there exists a complex matrix $Q \in \mathbb{C}^{k\times k}$ with $Q^*Q = I_k$, and a real positive scalar d such that*

$$\begin{aligned} b &= Qa & b &= d^{-1}w \\ z &= Q^*Qda & z &= Q^*w \end{aligned}$$

if and only if

$$z = \frac{|w|}{|a|}a \qquad b = \frac{|a|}{|w|}w \tag{4.14}$$

Lemma 5 (Repeated Real Scalar Block [53]) *Let $b, a, z, w \in \mathbb{C}^k$ be non-zero vectors with $a^*w \neq 0$. Then there is a real scalar q with $|q| \leq 1$, a real scalar $\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}]$, and a complex matrix $D \in \mathbb{C}^{k\times k}$ with $0 < D = D^*$ such that*

$$\begin{aligned} b &= qa & b &= e^{-j\theta}D^{-1}w \\ z &= q^*qe^{j\theta}Da & z &= q^*w \end{aligned}$$

with $\theta = \pm\frac{\pi}{2}$ for $|q| < 1$ iff

$$z = qw \qquad b = qa \tag{4.15}$$

with

$$\begin{aligned} Re(a^*w) \geq 0 \quad & for \quad q = 1 \\ Re(a^*w) \leq 0 \quad & for \quad q = -1 \\ Re(a^*w) = 0 \quad & for \quad |q| < 1 \end{aligned} \tag{4.16}$$

Proof: ($\Rightarrow$) Immediately it follows that $z = qw$ and $b = qa$. Thus $a^*w = \frac{1}{q}b^*w = \frac{1}{q}e^{j\theta}w^*(D^*)^{-1}w$. Now $q = 1$ implies $arg(a^*w) = \theta$ and hence $Re(a^*w) \geq 0$. Similarly $q = -1$ implies $arg(a^*w) = \theta + \pi$ and hence $Re(a^*w) \leq 0$. Finally $|q| < 1$ implies $arg(a^*w) = \theta$ or $\theta + \pi$ with $\theta = \pm\frac{\pi}{2}$. Thus $arg(a^*w) = \pm\frac{\pi}{2}$ and so $Re(a^*w) = 0$.

($\Leftarrow$) Immediately then $b = qa$ and $z = q^*w$, and so $b^*w = qa^*w$. Denoting $\theta = arg(b^*w)$ it is apparent that for $q = 1$ $Re(a^*w) \geq 0$ which implies $Re(b^*w) \geq 0$ and so $\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}]$. Similarly for $q = -1$ $Re(a^*w) \leq 0$ which implies $Re(b^*w) \geq 0$ and so $\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}]$. Finally for $|q| < 1$ $Re(a^*w) = 0$ which implies $Re(b^*w) = 0$ and so $\theta = \pm\frac{\pi}{2}$. Now $b^*(e^{-j\theta}w)$ is real and positive and so applying lemma 2 gives a matrix $\hat{D}$ with $0 < \hat{D} = \hat{D}^*$ such that $b = e^{-j\theta}\hat{D}w$. Define $D = \hat{D}^{-1}$ and which gives $b = e^{-j\theta}D^{-1}w$ and $z = q^*w = q^*e^{j\theta}Db = q^*qe^{j\theta}Da$. □

These lemmas allow one (with a few technical assumptions) to eliminate the matrices Q and D from (4.12). In order to avoid the notation becoming excessive, consider a simple block structure with $m_r = m_c = m_C = 1$ for the remainder of this section. *It must be stressed that this is purely for notational convenience, and that the general formulae for an arbitrary block structure, as defined in Section 2.4, are simply obtained by duplicating the appropriate formulae for each block.* So given $\mathcal{K} = (k_1, k_2, k_3)$the appropriate scaling sets become

$$\begin{aligned} Q_{sub} = \{&block\ diag(q^r I_{k_1}, q^c I_{k_2}, Q^C) : \\ &q^r \in [-1\ 1], q^{c*}q^c = 1, Q^{C*}Q^C = I_{k_3}\} \end{aligned} \tag{4.17}$$

$$\mathcal{D}_{sub} = \{block\ diag(e^{j\theta}D_1, D_2, dI_{k_3}) :$$
$$\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}], 0 < D_i = D_i^* \in \mathbb{C}^{k_i \times k_i}, 0 < d \in \mathbb{R}\} \tag{4.18}$$

and one partitions b, a, z, w compatibly with this block structure as

$$b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}, \quad a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix}, \quad z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}, \quad w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{4.19}$$

where $b_i, a_i, z_i, w_i \in \mathbb{C}^{k_i}$. Then the final form of (4.12) can be obtained as in the following theorem from [53] (which will form the basis of a power iteration [26] to compute a lower bound for $\mu_{\mathcal{K}}(M)$).

Theorem 5 ([53]) *Suppose there are vectors $b, a, z, w \in \mathbb{C}^n$ partitioned as in (4.19) with $b_i, a_i, z_i, w_i \neq 0$ and $a_1^* w_1, a_2^* w_2 \neq 0$. Then there exist matrices $Q \in \mathcal{Q}_{sub}$ and $D \in \mathcal{D}_{sub}$, and a positive real scalar β such that*

$$\begin{array}{rr} Mb = \beta a & M^* z = \beta w \\ b = Qa & b = D^{-1} w \\ z = Q^* QDa & z = Q^* w \end{array}$$

with $\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}]$ and $\theta = \pm\frac{\pi}{2}$ for $|q^r| < 1$ iff

$$\begin{array}{lll} Mb = \beta a & & \\ z_1 = q w_1 & z_2 = \dfrac{w_2^* a_2}{|w_2^* a_2|} w_2 & z_3 = \dfrac{|w_3|}{|a_3|} a_3 \\ M^* z = \beta w & & \\ b_1 = q a_1 & b_2 = \dfrac{a_2^* w_2}{|a_2^* w_2|} a_2 & b_3 = \dfrac{|a_3|}{|w_3|} w_3 \end{array} \tag{4.20}$$

for some real scalar $q \in [-1\ 1]$ with

$$\begin{array}{l} Re(a_1^* w_1) \geq 0 \quad for \quad q = 1 \\ Re(a_1^* w_1) \leq 0 \quad for \quad q = -1 \\ Re(a_1^* w_1) = 0 \quad for \quad |q| < 1 \end{array} \tag{4.21}$$

Proof: Apply lemmas 3, 4 and 5 to the appropriate block components. □

Any solution to (4.20) and (4.21) immediately gives a decomposition as in (4.10) and thus β is a lower bound for $\mu_{\mathcal{K}}(M)$. Also note that, under certain technical assumptions (as given), there always exists a solution to these equations with $\beta = \mu_{\mathcal{K}}(M)$. Since one would like to find the largest β one can that solves (4.20) and (4.21), a method for finding a solution to this system of equations via the following power iteration is proposed:

$$\begin{aligned}
\tilde{\beta}_{k+1} a_{k+1} &= M b_k \\
z_{1_{k+1}} &= \tilde{q}_{k+1} w_{1_k} \qquad z_{2_{k+1}} = \frac{w_{2_k}^* a_{2_{k+1}}}{|w_{2_k}^* a_{2_{k+1}}|} w_{2_k} \qquad z_{3_{k+1}} = \frac{|w_{3_k}|}{|a_{3_{k+1}}|} a_{3_{k+1}} \\
\hat{\beta}_{k+1} w_{k+1} &= M^* z_{k+1} \\
b_{1_{k+1}} &= \hat{q}_{k+1} a_{1_{k+1}} \qquad b_{2_{k+1}} = \frac{a_{2_{k+1}}^* w_{2_{k+1}}}{|a_{2_{k+1}}^* w_{2_{k+1}}|} a_{2_{k+1}} \qquad b_{3_{k+1}} = \frac{|a_{3_{k+1}}|}{|w_{3_{k+1}}|} w_{3_{k+1}}
\end{aligned} \tag{4.22}$$

where $\tilde{q}_{k+1}$ and $\hat{q}_{k+1}$ evolve as

$$\begin{aligned}
\tilde{\alpha}_{k+1} &= sgn(\hat{q}_k) \frac{|b_{1_k}|}{|a_{1_{k+1}}|} + Re(a_{1_{k+1}}^* w_{1_k}) \\
\text{If } |\tilde{\alpha}_{k+1}| \geq 1 \quad \text{Then } \tilde{q}_{k+1} &= \frac{\tilde{\alpha}_{k+1}}{|\tilde{\alpha}_{k+1}|} \quad \text{Else } \tilde{q}_{k+1} = \tilde{\alpha}_{k+1} \\
\hat{\alpha}_{k+1} &= sgn(\tilde{q}_{k+1}) \frac{|b_{1_k}|}{|a_{1_{k+1}}|} + Re(a_{1_{k+1}}^* w_{1_{k+1}}) \\
\text{If } |\hat{\alpha}_{k+1}| \geq 1 \quad \text{Then } \hat{q}_{k+1} &= \frac{\hat{\alpha}_{k+1}}{|\hat{\alpha}_{k+1}|} \quad \text{Else } \hat{q}_{k+1} = \hat{\alpha}_{k+1}
\end{aligned} \tag{4.23}$$

and $\tilde{\beta}_{k+1}, \hat{\beta}_{k+1}$ are chosen positive real so that $|a_{k+1}| = |w_{k+1}| = 1$.

It is now straightforward to verify that if the algorithm converges to some equilibrium point then we satisfy the appropriate constraints on each block component and hence by lemmas 3, 4, and 5 we have have non-zero vectors $b, a, z, w \in \mathbb{C}^n$, matrices $Q \in \mathcal{Q}_{sub}, D \in \mathcal{D}_{sub}$, and positive real scalars $\tilde{\beta}, \hat{\beta}$ such that

$$\begin{aligned}
Mb = \tilde{\beta} a \qquad & M^* z = \hat{\beta} w \\
b = Qa \qquad & b = D^{-1} w \\
z = Q^* Q D a \qquad & z = Q^* w
\end{aligned} \tag{4.24}$$

Thus if $\tilde{\beta} = \hat{\beta}$ then (4.12) is satisfied and there is a decomposition as in (4.10). Hence $\tilde{\beta}$ is a lower bound for $\mu_{\mathcal{K}}(M)$ (associated with a local maximum of $\rho_R(QM)$).

One notes that if $\tilde{\beta} \neq \hat{\beta}$ then a decomposition as in (4.10) has not been found. However from (4.24) one finds that $QMb = \tilde{\beta}b$ and $w^*QM = \hat{\beta}w^*$. Thus both $\tilde{\beta}$ and $\hat{\beta}$ are real positive eigenvalues of QM, and by lemma 3 of [53], $\max(\tilde{\beta}, \hat{\beta})$ still gives a lower bound for $\mu_{\mathcal{K}}(M)$.

Note that the equilibrium points of the algorithm are unaffected if one multiplies the terms $Re(a^*_{1_{k+1}} w_{1_k}), Re(a^*_{1_{k+1}} w_{1_{k+1}})$ by arbitrary real positive scalars. Thus one may employ this degree of freedom to select scaling parameters to aid convergence.

4.4 Extension to a Power Algorithm for the Skew μ Lower Bound

Section 4.3's power iteration algorithm can be extended to the skew μ case. The first requirement is to show that the power iteration works if the system is skewed via the inclusion of a skewing matrix. However, before that is done, the framework built up in section 4.3 needs to be increased to cover the situation where some perturbations are declared of a fixed range, and some perturbations are declared of a varying range.

Like the previous section, in order to avoid the notation becoming excessive, a simple block structure will be considered. In this case it will be presumed that $m_{r_f} = m_{c_f} = m_{C_f} = 1$ for fixed range perturbations, and that $m_{r_v} = m_{c_v} = m_{C_v} = 1$ for the varying range perturbations. *Once again, it must be stressed that this is purely for notational convenience, and that the general formulae for an arbitrary block structure are simply obtained by duplicating the appropriate formulae for each block.* So given $\mathcal{K}_f = (k_{1_f}, k_{2_f}, k_{3_f})$ and $\mathcal{K}_v = (k_{1_v}, k_{2_v}, k_{3_v})$ the appropriate scaling sets become (where $_s$ indicates the skew μ case)

$$\begin{aligned} Q_{sub_s} = \{ & block\ diag(q^r I_{k_{1_f}}, q^c I_{k_{2_f}}, Q_f^C, q^r I_{k_{1_v}}, q^c I_{k_{2_v}}, Q_v^C) : \\ & q^r \in [-1\ 1], q^{c*} q^c = 1, Q_f^{C*} Q_f^C = I_{k_{3_f}}, Q_v^{C*} Q_v^C = I_{k_{3_v}} \} \end{aligned} \tag{4.25}$$

$$\mathcal{D}_{sub_s} = \{block\,diag(e^{j\theta_f}D_{1_f}, D_{2_f}, d_f I_{k_{3_f}}, e^{j\theta_v}D_{1_v}, D_{2_v}, d_v I_{k_{3_v}}) : \\ \theta_i \in [-\frac{\pi}{2}\,\frac{\pi}{2}], 0 < D_i = D_i^* \in \mathbb{C}^{k_i \times k_i}, 0 < d_i \in \mathbb{R}\} \tag{4.26}$$

and one partitions b, a, z, w compatibly with this block structure as

$$b = \begin{bmatrix} b_{1_f} \\ b_{2_f} \\ b_{3_f} \\ b_{1_v} \\ b_{2_v} \\ b_{3_v} \end{bmatrix}, \quad a = \begin{bmatrix} a_{1_f} \\ a_{2_f} \\ a_{3_f} \\ a_{1_v} \\ a_{2_v} \\ a_{3_v} \end{bmatrix}, \quad z = \begin{bmatrix} z_{1_f} \\ z_{2_f} \\ z_{3_f} \\ z_{1_v} \\ z_{2_v} \\ z_{3_v} \end{bmatrix}, \quad w = \begin{bmatrix} w_{1_f} \\ w_{2_f} \\ w_{3_f} \\ w_{1_v} \\ w_{2_v} \\ w_{3_v} \end{bmatrix} \tag{4.27}$$

where $b_i, a_i, z_i, w_i \in \mathbb{C}^{k_i}$. Using this extended structure one can now proceed to derive machinery for obtaining a skew μ lower bound.

Theorem 6 *Suppose we have vectors $b, a, z, w \in \mathbb{C}^n$ partitioned as in (4.19), but extended to account for the varying and fixed cases $(_v,_f)$ with $b_{i_v}, a_{i_v}, z_{i_v}, w_{i_v} \neq 0, b_{i_f}, a_{i_f}, z_{i_f}, w_{i_f} \neq 0$ and $a_{1_v}^* w_{1v}, a_{2v}^* w_{2v}, a_{1f}^* w_{1f}, a_{2f}^* w_{2f} \neq 0$. Then there exist matrices $Q \in \mathcal{Q}_{sub_s}$ and $D \in \mathcal{D}_{sub_s}$, and a matrix of the block form:*

$$S = \left[\begin{array}{c|c} I & 0 \\ \hline 0 & \nu I \end{array}\right] \tag{4.28}$$

Where the S matrix is partitioned such that the blocks I and νI are sized to correspond to the fixed (skew) and varying uncertainties, Δ_f and Δ_v respectively, as defined in (3.4),(3.5), and (3.11). Additionally, ν is a positive real scalar ($\nu \in \mathbb{R} : \nu > 0$) such that the following holds

$$\begin{array}{rl} Mb = S\beta a & M^* z = S\beta w \\ b = Qa & b = D^{-1} w \\ z = Q^* QDa & z = Q^* w \end{array} \tag{4.29}$$

with $\theta \in [-\frac{\pi}{2}\ \frac{\pi}{2}]$ *and* $\theta = \pm\frac{\pi}{2}$ *for* $|q^r| < 1$ *iff*

$$
\begin{aligned}
& Mb = S\beta a \qquad (4.30)\\
& z_{1v} = qw_{1v} \qquad z_{2v} = \frac{w_{2v}^* a_{2v}}{|w_{2v}^* a_{2v}|} w_{2v} \qquad z_3 = \frac{|w_{3v}|}{|a_{3v}|} a_{3v}\\
& z_{1f} = qw_{1f} \qquad z_{2f} = \frac{w_{2f}^* a_{2f}}{|w_{2f}^* a_{2f}|} w_{2f} \qquad z_{3f} = \frac{|w_{3f}|}{|a_{3f}|} a_{3f}\\
& M^* z = S\beta w\\
& b_{1v} = qa_{1v} \qquad b_{2v} = \frac{a_{2v}^* w_{2v}}{|a_{2v}^* w_{2v}|} a_{2v} \qquad b_{3v} = \frac{|a_{3v}|}{|w_{3v}|} w_{3v}\\
& b_{1f} = qa_{1f} \qquad b_{2f} = \frac{a_{2f}^* w_{2f}}{|a_{2f}^* w_{2f}|} a_{2f} \qquad b_{3f} = \frac{|a_{3f}|}{|w_{3f}|} w_{3f}
\end{aligned}
$$

for some real scalars $q_v \in [-1\ 1]$ *and* $q_f \in [-1\ 1]$ *with*

$$
\begin{aligned}
& Re(a_{1v}^* w_{1v}) \geq 0 \quad for \quad q_v = 1\\
& Re(a_{1v}^* w_{1v}) \leq 0 \quad for \quad q_v = -1 \qquad (4.31)\\
& Re(a_{1v}^* w_{1v}) = 0 \quad for \quad |q_v| < 1\\
& Re(a_{1f}^* w_{1f}) \geq 0 \quad for \quad q_f = 1\\
& Re(a_{1f}^* w_{1f}) \leq 0 \quad for \quad q_f = -1\\
& Re(a_{1f}^* w_{1f}) = 0 \quad for \quad |q_f| < 1
\end{aligned}
$$

(4.32)

Proof: Note that the first line of 4.29 can be rewritten as:

$$\beta a = S^{-1} Mb \qquad (4.33)$$

and the S^{-1} matrix wrapped into M to form:

$$\beta a = M_s b, \qquad (4.34)$$

with $M_s = S^{-1}M$. This transforms the problem into a standard μ problem on M_s. The remainder of the proof is via theorem 5 and application of lemmas 3, 4 and 5 to the appropriate block components. □

The next theorem uses normalization of a power iteration parameter to allow computation of the skew μ lower bound as a step that can be included in the power iteration.

Theorem 7 *If* β *of theorem (6) is set to 1, the value of* μ *for the fixed section, then the value of* ν *is a lower bound on* $\mu_s(M)$ *that can be calculated from*

$$\nu = \frac{\|\gamma_2\|_2^2}{\sqrt{1 - \|\gamma_1\|_2^2}} \tag{4.35}$$

or

$$\nu = \frac{\|\psi_2\|_2^2}{\sqrt{1 - \|\psi_1\|_2^2}}, \tag{4.36}$$

where $\begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = Mb$ *and* γ_1 *has the number of elements equal to the length of the fixed columns of S and* γ_2 *has the number of elements equal to the length of the varying columns of S. Likewise,* $\begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix} = M^* z$ *and* ψ_1 *has the number of elements equal to the length of the fixed rows of S and* ψ_2 *has the number of elements equal to the length of the varying rows of S.*

When $\|\gamma_1\|_2^2 \geq 1$ *or* $\|\psi_1\|_2^2 \geq 1$ *then the system can be destabilized by a fixed perturbation only. In this case* $\mu_s(M) = \infty$. *If* $\|\gamma_2\|_2^2 = 0$ *or* $\|\psi_2\|_2^2 = 0$, *then no varying perturbation of any size can destabilize the system and* $\mu_s(M) = 0$.

Proof: Define S_L, similar to equation 4.28, as a suitably sized skewing matrix for left sided multiplication such that:

$$M_s = S_L^{-1} M. \tag{4.37}$$

Although equation 4.38 is implied by earlier developments in this chapter, the following derivation is provided for those who desire more detail. This derivation follows along the lines of the proof of Theorem 4, which can be found in [53]:

$$QM_s\hat{x} = \beta\hat{x}$$

$$Q\hat{D}M_s\hat{D}^{-1}(\hat{D}\hat{x}) = \beta\hat{D}\hat{x}$$

$$x = \hat{D}\hat{x}$$

$$Q\hat{D}M_s(\hat{D}^{-1}x) = \beta x$$

$$Q\hat{D}M_s b = \beta x$$

$$M_s b = \hat{D}^{-1}\hat{Q}^{-1}\beta x$$

$$M_s b = \beta a$$

$$S_L^{-1} M b = \beta a$$

$$Mb = \beta S_L a. \tag{4.38}$$

Then consider for the left side of equation 4.38:

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \end{bmatrix} = \beta S_L a, \tag{4.39}$$

where γ_1 is of the length of the fixed columns of S_L and γ_2 is of the length of the varying columns of S_L.

For skew systems, $\beta = 1$, and it can be assumed $|a| = 1$ WLOG (see remark below) , then:

$$S_L^{-1}\gamma = 1 \tag{4.40}$$

where

$$S_L^{-1} = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\nu} I_v \end{array}\right]. \tag{4.41}$$

Taking the L_2 norm of both sides of equation 4.39 and manipulating gives

$$1 = \|\gamma_1\|_2^2 + \frac{1}{\nu^2}\|\gamma_2\|_2^2, \tag{4.42}$$

which implies

$$\nu = \frac{\|\gamma_2\|_2^2}{\sqrt{1 - \|\gamma_1\|_2^2}}. \tag{4.43}$$

Because left and right sided skew matrices will be used in the computational aspects, a similar development is demonstrated for the RHS multiplication. Although equation 4.45 is implied by earlier developments in this chapter, the following derivation is provided

for those who desire more detail. Define S_R, similar to equation 4.28, as a suitably sized skewing matrix for left sided multiplication such that:

$$M_s = MS_R^{-1}. \tag{4.44}$$

Following along the lines of the proof of Theorem 4, which can be found in [53]:

$$\hat{x}^H Q M_s = \beta \hat{x}^H$$

$$(\hat{x}^H \hat{D}^H) Q \hat{D}^H M_s (\hat{D}^H)^{-1} = \beta \hat{x}^H \hat{D}^H$$

$$x = \hat{D}\hat{x}$$

$$x^H Q \hat{D}^H M_s = \beta \hat{x}^H \hat{D}^H \hat{D}^H$$

$$x^H Q \hat{D}^H M_s = \beta x^H \hat{D}^H$$

$$x^H Q \hat{D}^H M_s = \beta w^H$$

Taking the complex conjugate transpose of both sides (^{H})

$$M_s^H \hat{D} Q^H x = \beta w$$

$$z = \hat{D} Q^H x$$

$$M_s^H z = \beta w$$

$$(MS_R^{-1})^H z = \beta w$$

$$S_R^{-H} M^H z = \beta w$$

$$M^H z = \beta S_R^H w. \tag{4.45}$$

Then consider for the left side of the equation 4.45:

$$M^H z = \beta S_R w \tag{4.46}$$

$$\begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix} = \beta S_R^H w \tag{4.47}$$

For fixed systems, $\beta = 1$, and it can be assumed $|w| = 1$ WLOG (see remark below), then:

$$S_R^{-H}\psi = 1 \tag{4.48}$$

where

$$S_R^{-1} = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\nu} I_\nu \end{array}\right]. \tag{4.49}$$

Taking the 2 norm of both sides of equation 4.47 and manipulating gives

$$1 = \|\psi_1\|_2^2 + \frac{1}{\nu^2}\|\psi_2\|_2^2, \tag{4.50}$$

which implies

$$\nu = \frac{\|\psi_2\|_2^2}{\sqrt{1 - \|\psi_1\|_2^2}}. \tag{4.51}$$

Note that for equations 4.35 and 4.36 that as $\|\gamma_1\|_2^2$ or $\|\psi_1\|_2^2$ approaches 1, ν approaches ∞. Additionally, note that as $\|\gamma_2\|_2^2$ or $\|\psi_2\|_2^2$ approaches 0, ν approaches 0. □

Remarks: Since the relationships (4.30) and (4.31) are unaffected if b and a are multiplied by an arbitrary positive real scalar η, and z and w by an arbitrary positive real scalar θ, then in searching for solutions to these equations one may impose the additional restriction $|a| = |w| = 1$, which essentially sets up the methodology for calculating a lower bound on ν.

It should also be noted that per theorem 7, there always exists a solution to these equations with $\frac{1}{\nu} = \mu_s(M)$. Since one would like to find the largest ν that solves (4.30) and (4.31), the following power iteration is proposed for finding a solution to this system of equations.

This is similar to [53] with an extension to cover the varying and fixed (v,f) case:

$$\tilde{S}_{k+1}a_{k+1} = Mb_k$$

$$z1_{v_{k+1}} = \tilde{q}_{v_{k+1}} w1_{v_k}$$

$$z2_{v_{k+1}} = \frac{w^*_{2v_k} a2_{v_{k+1}}}{|w^*_{2v_k} a2_{v_{k+1}}|} w2_{v_k} \qquad z3_{v_{k+1}} = \frac{|w3_{v_k}|}{|a3_{v_{k+1}}|} a3_{v_{k+1}}$$

$$z1_{f_{k+1}} = \tilde{q}_{f_{k+1}} w1_{f_k}$$

$$z2_{f_{k+1}} = \frac{w^*_{2f_k} a2_{f_{k+1}}}{|w^*_{2f_k} a2_{f_{k+1}}|} w2_{f_k} \qquad z3_{f_{k+1}} = \frac{|w3_{f_k}|}{|a3_{f_{k+1}}|} a3_{f_{k+1}}$$

$$\hat{S}_{k+1}w_{k+1} = M^* z_{k+1}$$

$$b1_{v_{k+1}} = \hat{q}_{v_{k+1}} a1_{v_{k+1}}$$

$$b2_{v_{k+1}} = \frac{a^*_{2v_{k+1}} w2_{v_{k+1}}}{|a^*_{2v_{k+1}} w2_{v_{k+1}}|} a2_{v_{k+1}} \qquad b3_{v_{k+1}} = \frac{|a3_{v_{k+1}}|}{|w3_{v_{k+1}}|} w3_{v_{k+1}}$$

$$b1_{f_{k+1}} = \hat{q}_{f_{k+1}} a1_{f_{k+1}}$$

$$b2_{f_{k+1}} = \frac{a^*_{2f_{k+1}} w2_{f_{k+1}}}{|a^*_{2f_{k+1}} w2_{f_{k+1}}|} a2_{f_{k+1}} \qquad b3_{f_{k+1}} = \frac{|a3_{k+1}|}{|w3_{k+1}|} w3_{k+1}$$

where $\tilde{q}_{f_{k+1}}$, and $\hat{q}_{f_{k+1}}$ evolve as

$$\tilde{\alpha}_{f_{k+1}} = sgn(\hat{q}_{f_k}) \frac{|b1_{f_k}|}{|a1_{f_{k+1}}|} + Re(a^*_{1_{f_{k+1}}} w1_{f_k})$$

If $|\tilde{\alpha}_{f_{k+1}}| \geq 1$ Then $\tilde{q}_{f_{k+1}} = \frac{\tilde{\alpha}_{f_{k+1}}}{|\tilde{\alpha}_{f_{k+1}}|}$

Else $\tilde{q}_{f_{k+1}} = \tilde{\alpha}_{f_{k+1}}$

$$\hat{\alpha}_{f_{k+1}} = sgn(\tilde{q}_{f_{k+1}}) \frac{|b1_{f_k}|}{|a1_{f_{k+1}}|} + Re(a^*_{1_{f_{k+1}}} w1_{f_{k+1}})$$

If $|\hat{\alpha}_{f_{k+1}}| \geq 1$ Then $\hat{q}_{f_{k+1}} = \frac{\hat{\alpha}_{f_{k+1}}}{|\hat{\alpha}_{f_{k+1}}|}$

Else $\hat{q}_{f_{k+1}} = \hat{\alpha}_{f_{k+1}}$

for the section of the system involved with the fixed range perturbations. For the section of the system involved with the varying range perturbations, $\tilde{q}_{v_{k+1}}$, and $\hat{q}_{v_{k+1}}$ evolve as

$$\tilde{\alpha}_{v_{k+1}} = sgn(\hat{q}_{v_k})\frac{|b_{1_{v_k}}|}{|a_{1_{v_{k+1}}}|} + Re(a^*_{1_{v_{k+1}}} w_{1_{v_k}})$$

If $|\tilde{\alpha}_{v_{k+1}}| \geq 1$ Then $\tilde{q}_{v_{k+1}} = \frac{\tilde{\alpha}_{v_{k+1}}}{|\tilde{\alpha}_{v_{k+1}}|}$

Else $\tilde{q}_{v_{k+1}} = \tilde{\alpha}_{v_{k+1}}$

$$\hat{\alpha}_{v_{k+1}} = sgn(\tilde{q}_{v_{k+1}})\frac{|b_{1_{v_k}}|}{|a_{1_{v_{k+1}}}|} + Re(a^*_{1_{v_{k+1}}} w_{1_{v_{k+1}}})$$

If $|\hat{\alpha}_{v_{k+1}}| \geq 1$ Then $\hat{q}_{v_{k+1}} = \frac{\hat{\alpha}_{v_{k+1}}}{|\hat{\alpha}_{v_{k+1}}|}$

Else $\hat{q}_{v_{k+1}} = \hat{\alpha}_{v_{k+1}}$

and $\tilde{\nu}_{k+1}$ (of $\tilde{S}_{k+1}$)and $\hat{\nu}_{k+1}$ (of $\hat{S}_{k+1}$) are chosen positive real so that $|a_{k+1}| = |w_{k+1}| = 1$ according to:

$$\tilde{\gamma} = Mb_k \tag{4.52}$$

$$\hat{\psi} = M^* z_{k+1}$$

where

$$\tilde{\gamma} = \begin{bmatrix} \tilde{\gamma}_1 \\ \hline \tilde{\gamma}_2 \end{bmatrix} \tag{4.53}$$

with $\tilde{\gamma}_1$ of *dimension* $(\Delta_f) = n_f$ and $\tilde{\gamma}_2$ of *dimension* $(\Delta_v) = n_v$. Then solve

$$\tilde{\nu} = \frac{\|\tilde{\gamma}_2\|_2}{\sqrt{1 - \|\tilde{\gamma}_1\|_2^2}} \tag{4.54}$$

and update:

$$\tilde{S}_{k+1} = \begin{bmatrix} I & 0 \\ \hline 0 & \tilde{\nu}I \end{bmatrix} \tag{4.55}$$

Similarly, this is done for $\hat{S}_{k+1}$.

where

$$\hat{\psi} = \left[\frac{\hat{\psi}_1}{\hat{\psi}_2}\right] \tag{4.56}$$

with $\hat{\psi}_1$ of *dimension* $(\Delta_f) = n_f$ and $\hat{\psi}_2$ of *dimension* $(\Delta_v) = n_v$. Then solve

$$\hat{\nu} = \frac{\|\hat{\psi}_2\|_2^2}{\sqrt{1 - \|\hat{\psi}_1\|_2^2}}. \tag{4.57}$$

and update

$$\hat{S}_{k+1} = \left[\begin{array}{c|c} I & 0 \\ \hline 0 & \hat{\nu} I \end{array}\right]. \tag{4.58}$$

This precludes the case of $\|\tilde{\gamma}_1\|_2^2 \geq 1$ or $\|\hat{\psi}_1\|_2^2 \geq 1$. This particular case can cause problems when it occurs as a momentary step in the iteration, thus special handling is provided, and described in section 4.5.

Remarks: This is essentially a variation of the original work of [53], modified to find a lower bound for skew μ, rather than μ. The original method applies here, but the perturbations must be pre-sorted into fixed and varying sections to allow for the convenient segregation as in (4.53). This segregation simplifies the math as well as simplifying the implemented code.

It is now straightforward to verify that if the algorithm converges to some equilibrium point then it satisfies the appropriate constraints on each block component and hence by lemmas 3, 4, and 5 has non-zero vectors $b, a, z, w \in \mathbb{C}^n$, matrices $Q \in \mathcal{Q}_{sub_s}, D \in \mathcal{D}_{sub_s}$, and positive real scalars $\tilde{\nu}, \hat{\nu}$ such that

$$\begin{array}{rr} Mb = \tilde{S}a & M^*z = \hat{S}w \\ b = Qa & b = D^{-1}w \\ z = Q^*QDa & z = Q^*w \end{array} \tag{4.59}$$

Thus if $\tilde{\nu} = \hat{\nu}$ then (4.12) is satisfied and there is a decomposition as in (4.10), and hence $\tilde{\nu}$ is a lower bound for $\mu_s(M)$ (associated with a local maximum of $\rho_R(QM_s)$).

Also note that if $\tilde{\nu} \neq \hat{\nu}$ then a decomposition as in (4.10) has not been found. However from (4.59) it is found that $QMb = \tilde{S}b$ and $w^*QM = \hat{S}w^*$ where both $\tilde{\nu}$ and $\hat{\nu}$ are real. Then by lemma 3 of [53], the $\max(\tilde{\nu}, \hat{\nu})$ still gives a lower bound for $\mu_s(M)$.

Note that the equilibrium points of the algorithm are unaffected if the terms $Re(a^*_{1_{f_{k+1}}} w_{1_{f_k}}), Re(a^*_{1_{f_{k+1}}} w_{1_{f_{k+1}}}), Re(a^*_{1_{v_{k+1}}} w_{1_{f_k}}), Re(a^*_{1_{v_{k+1}}} w_{1_{v_{k+1}}})$ are multiplied by arbitrary real positive scalars, and hence one may employ this degree of freedom to select scaling parameters so as to aid convergence.

4.5 Avoiding Computation Problems in the Lower Bound

If γ_1 is too large then a complex value for ν will result. This is an intuitive result for equation 4.60.

$$\nu = \frac{\|\gamma_2\|_2}{\sqrt{1 - \|\gamma_1\|_2^2}} \tag{4.60}$$

This is a primary indicator that the problem can be destabilized by a perturbation among the fixed perturbations (Δ_f)without consideration of the varying perturbations (Δ_v). If the problem can be destabilized by one of the fixed perturbations, then one may want to reconsider the problem and/or the details surrounding it. However, since the lower bound is found via an iterative process, the condition of $\nu \in \mathbb{C}$ can result during one of the iterations as a momentary abberation. To avoid computational problems with this, the following algorithm was implemented. The following is for γ_1. However, the algorithm is also implemented in relation to ψ_1 as well in software.

For a system M:

1. Set $\hat{M} = M$
2. Set $\nu_{total} = 1$
3. Find γ_1
4. If $\gamma_1 > 1$

- $\nu_{total} = 2 \cdot \nu_{total}$
- Form

$$\hat{M} = \begin{bmatrix} I_f & 0 \\ 0 & .5 \cdot I_v \end{bmatrix} \hat{M} \tag{4.61}$$

- Go to 6

5. If $\gamma_1 < 1$
   - Find ν.
   - if $\nu \approx 1$ then return ν_{total}, END.
   - if $\nu \neq 1$ then:
     - $\nu_{total} = \nu \cdot \nu_{total}$
     - Form

$$\hat{M} = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\nu} \cdot I_v \end{bmatrix} \hat{M} \tag{4.62}$$

6. If $\nu_{total} \approx \infty$, and vectors b, a, z and w have converged, then $\mu_s(M) \approx \infty$ and the system can be destabilized by a perturbation among the fixed perturbations (Δ_f) without consideration of the varying perturbations (Δ_v) so END, else go to 3.

The simplicity of this algorithm is that systems which can be destabilized by a fixed perturbation continuously wrap 2 into ν_{total}, causing the value of ν_{total} to grow towards infinity. Hence, the skew μ lower bound is $\nu_{total} \approx \infty$, as expected with systems that can be destabilized without considering the varying perturbation. As $\mu_s \approx \infty$ and the eigenvectors for the algorithm converge, the value of ν for a given iteration provides a lower bound on $\mu(M_{11})$. This is a valuable piece of information to obtain when trying to interpret the results of a skew μ robust control problem that did not provide the results expected.

From a practical standpoint, this algorithm is easier to implement than a procedure that must error trap complex values of γ_1. For the algorithm above there are only three possible outcomes, regardless of eigenvector alignment conditions in the power algorithm: (1) ν for

an iteration converges to 1, and a value of ν_{total} is found, (2) $\gamma_1 > 1$ consistently and ν_{total} is *error trapped* as it approaches infinity, or (3) ν for each iteration remains continuously below 1, and ν_{total} is *error trapped* as it approaches 0.

Chapter 5

Developing a Skew μ Upper Bound

5.1 D-Scales

As previously discussed in section 2.5.1, scaling of matrices in the robust control framework (D-scales) decreases the conservatism of the μ bounds via minimization of the matrix norm through a similarity transform. This techniques will be one of the primary methods used in developing an upper bound for skew μ. One of these techniques will use a standard Osborne balancing [36] method to develop an algorithm for finding the skew μ upper bound. The second method will include D-scales, but will be formulated in terms of an LMI. Before developing these bounds, a few preliminary notations need to be pointed out about scaling sets and the left and right scaling matrices.

5.1.1 Scaling Sets

Similar to the requirements of the lower bound, it is necessary to define the $\mathcal{D}$ scaling sets in terms of the fixed and varying perturbations. The notation may appear obvious, however the definitions should be made to eliminate any possible confusion.

For the fixed scaling sets, the D-scales will have a form similar to that of equation equation 2.39, with the notation $_f$ indicating these D-scales interact and commute with the fixed range perturbations, Δ_f. This set is specifically defined as

$$\mathcal{D}_{\mathcal{K}_f} = \Big\{ block\ diag(D_1, \ldots, D_{m_{r_f}+m_{c_f}}, d_1 I_{k_{m_{r_f}+m_{c_f}+1}} \ldots, d_{m_{C_f}} I_{k_{m_f}})$$

$$: \det(D_i) \neq 0, D_i \in \mathbb{C}^{k_i \times k_i}, d_i \neq 0, d_i \in \mathbb{C} \Big\}. \tag{5.1}$$

For the fixed scaling sets, the D-scales will have a form similar to that of equation equation 2.39, with the notation $_v$ indicating these D-scales interact and commute with the fixed range perturbations, Δ_v. This set is specifically defined as

$$\mathcal{D}_{\mathcal{K}_v} = \Big\{ block\ diag(D_1, \ldots, D_{m_{r_v}+m_{c_v}}, d_1 I_{k_{m_{r_v}+m_{c_v}+1}} \ldots, d_{m_{C_v}} I_{k_{m_v}})$$

$$: \det(D_i) \neq 0, D_i \in \mathbb{C}^{k_i \times k_i}, d_i \neq 0, d_i \in \mathbb{C} \Big\}. \tag{5.2}$$

The set of D-scales that interacts and commutes with the system perturbation Δ can then be defined as

$$\mathcal{D}_{\mathcal{K}_f, \mathcal{K}_v} = \{ D = block\ diag(D_f, D_v) \} \tag{5.3}$$

in the same way the fixed range/varying range perturbation was defined in equation 3.11. This explanation provides the necessary detail on scaling sets, the next section provides additional information about left and right scaling matrices.

5.1.2 Non-Square Matrices

Consider a Δ with the structure

$$\Delta = \begin{bmatrix} \delta_i & 0 & 0 \\ 0 & \delta_j I & 0 \\ 0 & 0 & \Delta_F \end{bmatrix}. \tag{5.4}$$

The δ_i and $\delta_j I$ components of 5.4 are square by definition, however, the Δ_F component is not necessarily square. A block similarity transform of a non-square Δ does not have identically sized D and D^{-1}. If Δ is of size $m \times n$ due to a non-square Δ_F, then a block similarity transformation for Δ is

$$D_L \Delta D_R^{-1} \tag{5.5}$$

where D_L is of size $m \times m$ and D_R is of size $n \times n$.

This does not cause significant theoretical complications; however, it does significantly complicate the development of software used to compute μ and skew μ. The change occurs

in the size of the block that commutes with Δ_F in Δ. If Δ_F is of size $k \times l$, with a corresponding commuting matrix in D of $d_y I$, then the identity matrix will be the of size $k \times k$ in D_L and $l \times l$ in D_R, but the value of d_y will be the same in both.

In the following sections, the terms D_L and D_R will be used to indicate appropriately sized left and right scaling matrices.

5.2 Defining the Skew μ Upper Bound

In figure 5.1, the fixed perturbations (Δ_f) are confined to the unit ball, $\mathbf{B}\Delta$. This is not specifically a ball of radius 1, but rather a ball of a given radius, which the perturbations are guaranteed not to exceed. The set of perturbations $\hat{\Delta}_v$ are also confined to their unit ball, however, the positive real scalar α is allowed to vary as necessary to set the $\det(I - \Delta M) = 0$. Obviously the choice of α is dependent on the choice of $\hat{\Delta}_v$. The desired choice of $\hat{\Delta}_v$ is the one that yields the smallest α necessary to set $\det(I - \Delta M) = 0$. In standard μ theory, this measurement is

$$\mu(M) = \frac{1}{\|\Delta\|}. \tag{5.6}$$

In skew μ theory, this value is

$$\mu_s(M) = \frac{1}{\|\Delta_v\|}, \tag{5.7}$$

since only some of the perturbations are varied as necessary to find the point of instability. In this chapter, the upper bound on skew μ will be referred to as ν, however, first it must be shown that ν is an upper bound on skew μ.

The relationship demonstrated in figure 5.1 can be extended to equations by first defining a matrix of the block form

$$S = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \nu I_v \end{array}\right], \tag{5.8}$$

where the S matrix is partitioned such that the blocks I_f and νI_v are sized to correspond to the fixed range and varying uncertainties Δ_f and Δ_v respectively. In this case, $\frac{1}{\nu}$ is taking the place of α from figure 5.1. Then let the matrix M_s be defined as

$$M_s = S^{-1}M. \tag{5.9}$$

From the definition of skew μ, there must exist an M_s for some choice of ν such that $\mu(M_s) = 1$, if not, then $\mu_s(M) = 0$. From this, the following theorem can be derived.

Theorem 8 *The value of* ν *is an upper bound on* $\mu_s(M)$.

Proof:

The desire is to have a ν that sets the $\det(I - M_s\Delta) = 0$, thus it can be stated that

$$(I - M_s\Delta)x = 0$$

$$x = M_s\Delta x$$

$$\|x\| \leq \|M_s\Delta\|\|x\|$$

$$1 \leq \|M_s\Delta\| \leq \|M_s\|\|\Delta\|$$

therefore

$$\frac{1}{\overline{\sigma}(\Delta)} \leq \overline{\sigma}(M_s) = \overline{\sigma}(S^{-1}M). \tag{5.10}$$

Since $\mu(M_s) = 1$ for skew μ, then ν is chosen such that $\overline{\sigma}(S^{-1}M) = 1$, where $\overline{\sigma}(S^{-1}M)$ is an upper bound on μ and thus ν is an upper bound on $\mu_s(M)$. □

Remarks: This is predicated on the fact that $\mu(S^{-1}M) = 1$, which follows from the definition of skew μ, i.e. if some perturbations are fixed then how large can the other perturbations become before the system encounters potential instability.

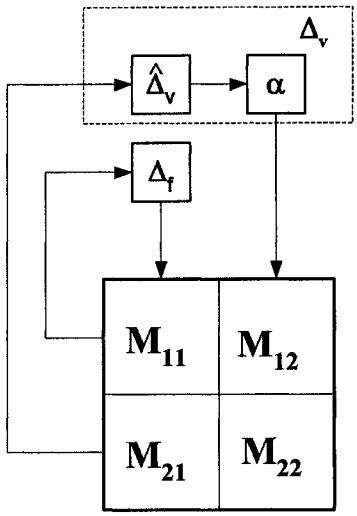


Figure 5.1: A diagram of the skew μ definition

5.3 Obtaining a Skew μ Upper Bound Via Iteration

The first method proposed for finding an upper bound on skew μ is an iterative method. To facilitate this method requires some re-arrangement of M and Δ. WLOG, elementary transformations can be used to arrange M and Δ such that the interacting components of Δ_f and M are arranged to allow Δ_f to occupy the upper left quadrant of Δ, and Δ_v to occupy the lower right quadrant of Δ.

The skew μ upper bound is specified as follows: first, define $\hat{M}$,

$$M_s = S^{-1}M = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\nu}I_v \end{array}\right] \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right]. \tag{5.11}$$

Then, the goal is to find a ν such that $\mu(M_s) = 1$, thus, $\mu_s(M) = \nu$. As previously mentioned, finding μ is difficult, hence a bound on μ is found, i.e.

$$\mu_{ub}(M_s) = \overline{\sigma}(M_s) = \sqrt{\bar{\lambda}(M_s^H M_s)} \geq 1, \tag{5.12}$$

where μ_{ub} indicates the upper bound on μ. This suggests an iterative procedure for finding a rudimentary estimate of $\mu_s(M)$.

1. Estimate ν

2. Form M_s

3. Calculate $\overline{\sigma}(M_s)$

4. If $\overline{\sigma}(M_s) \geq 1$ increase estimate of ν, goto 2

5. If $\overline{\sigma}(M_s) \leq 1$ reduce estimate of ν, goto 2

6. Stop when $\overline{\sigma}(M_s)$ is suitably close to 1, $\mu_{s_{ub}}(M) = \nu$

In this algorithm $\mu_{s_{ub}}$ indicates the skew μ upper bound. However, from previous discussion it is apparent that $\overline{\sigma}(M_s)$ may not bring the estimate of the upper bound as close to the true value of $\mu_s(M)$ as desired. To this end, the following improvements are added to the algorithm.

5.3.1 Implementation Using $S^{-1}M$ Versus $\sqrt{S^{-1}}M\sqrt{S^{-1}}$

There are many ways to form the product of the skewing matrix S^{-1} and the system matrix M. One is to form the product $S^{-1}M$. A second is to form the product $\sqrt{S^{-1}}M\sqrt{S^{-1}}$. These options for forming the product of S^{-1} and M are displayed in figures 5.2, 5.3 and 5.4. The advantage of $\sqrt{S^{-1}}M\sqrt{S^{-1}}$ can be seen by dissecting the products in block form.

Let A be

$$A = S^{-1}M = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\nu}I_\nu \end{array}\right] \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] = \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline \frac{1}{\nu}M_{21} & \frac{1}{\nu}M_{22} \end{array}\right] \tag{5.13}$$

Let B be

$$B = \sqrt{S^{-1}}M\sqrt{S^{-1}} = \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\sqrt{\nu}}I_\nu \end{array}\right] \left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right] \left[\begin{array}{c|c} I_f & 0 \\ \hline 0 & \frac{1}{\sqrt{\nu}}I_\nu \end{array}\right] = \left[\begin{array}{c|c} M_{11} & \frac{1}{\sqrt{\nu}}M_{12} \\ \hline \frac{1}{\sqrt{\nu}}M_{21} & \frac{1}{\nu}M_{22} \end{array}\right]. \tag{5.14}$$

Two things are apparent: first, both systems have the same eigenvalues, and second, if M is norm-balanced (via Osborne balancing [36]), then method B causes less upset on the matrix balance than method A. Hence, method B is the recommended implementation in software.

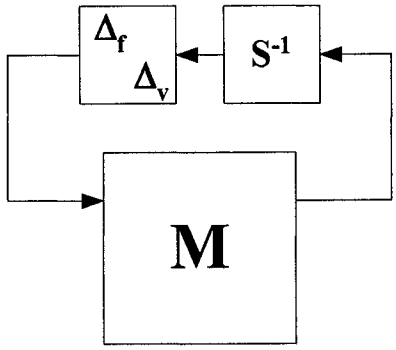


Figure 5.2: Forming the product $S^{-1}M$

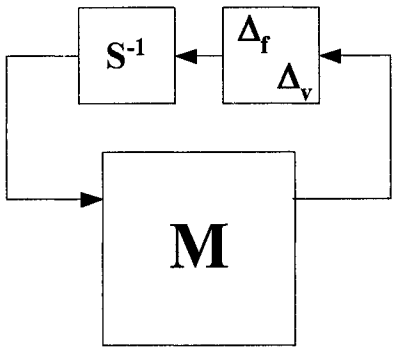


Figure 5.3: Forming the product MS^{-1}

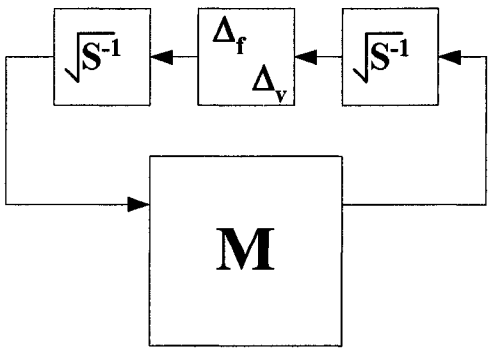


Figure 5.4: Forming the product $\sqrt{S^{-1}}M\sqrt{S^{-1}}$

5.4 The Osborne Skew μ Upper Bound

An upgrade to the method discussed in 5.3 is the application of Osborne balancing [36], where M is norm minimized using DMD^{-1}, and the calculation $\overline{\sigma}(\hat{M})$ is replaced with $\overline{\sigma}(\sqrt{S^{-1}}DMD^{-1}\sqrt{S^{-1}})$. The following sections discuss the background and application of Osborne balancing. In particular, the parameter names have been kept consistent with Osborne's original work to make cross referencing the original paper easier for the reader. Hence, some variable names used are identical with variable names used for different quantities elsewhere in this document. The reader should be aware of this.

5.4.1 Introduction to Osborne Balancing

Some of the difficulties encountered in the problem of obtaining the eigenvalues and eigenvectors of a matrix are due to the small magnitude of its eigenvalues compared to its matrix norm.

Osborne balancing is the method of applying norm-reducing similarity transformations to a matrix in order to facilitate the calculation of its eigenvalues.

Balancing a matrix A consists of finding a diagonal similarity transform DAD^{-1} such that for any i, row i and column i have nearly the same norm. Balancing can improve the speed and accuracy with which the eigenvalues are computed.

5.4.1.1 A Restriction for Osborne Balancing

Assume that the matrix A to be conditioned using Osborne balancing is irreducible by permutation matrices. In other words, there exists no permutation matrix P, obtained by permuting the rows or columns of an identity matrix, such that

$$P^{-1}AP = \begin{bmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{bmatrix} \tag{5.15}$$

This restriction is necessary to bound the elements of the resulting sequence of diagonal matrices, as can be seen later in equation 5.24. This is not a severe restriction, and the

process to obtain the irreducible components of A by permutations is straight forward. Once these components are obtained, they may be conditioned separately.

5.4.1.2 The Osborne Balancing Algorithm

The rows and corresponding columns of $A_{n\times n}$ are operated upon in order WLOG. So, at the k^{th} step of the process the i^{th} row and i^{th} column are being altered, where $i-1$ is the least positive residue of $k-1$ modulo n. Thus, the steps below are followed in order to generate the elements of the norm minimizing similarity transform matrix D.

1. Form

$$D_{k+l} = \bar{D}_k D_k \tag{5.16}$$

where

$$D_1 = I, \tag{5.17}$$

with

$$\bar{D}_k = \left\{ \begin{array}{ll} diag(1,1,\ldots 1,\bar{d}_i,l\ldots,1), & i<n \\ diag(\bar{d}_i^{-1},\bar{d}_i^{-1},\ldots,\bar{d}_i^{-1},1), & i=n \end{array} \right\} \tag{5.18}$$

where d_i is to be determined.

2. Then,

$$A_{k+1} = D_{k+1}A\bar{D}_{k+1}^{-1} = D_k A_k D_k^{-1} \tag{5.19}$$

where

$$A_1 = A. \tag{5.20}$$

A_k is defined as a function of k having elements a_{rs}, or $A(k) = (a_{rs}(k))$.

In general for a matrix A, define the following values,

$$R_j = (\sum_{p=1}^{n} {}' |a_{jp}|^2)^{\frac{1}{2}} \tag{5.21}$$

and

$$S_j = (\sum_{p=1}^{n} {}' |a_{pj}|^2)^{\frac{1}{2}}, \tag{5.22}$$

where the prime ($'$) indicates the omission of the element a_{jj} (the diagonal element).

3. The positive real number $\bar{d}_i$, is chosen so as to minimize the value of

$$\bar{d}_i^2 R_i^2 + \frac{1}{\bar{d}_i^2} S_i^2. \tag{5.23}$$

Setting the partial with respect to d (i.e. $\frac{\partial}{\partial d}$) of this equation equal to 0 implies that

$$\bar{d}_i = \left(\frac{S_i}{R_i}\right)^{\frac{1}{2}} \tag{5.24}$$

4. If $\bar{d}_i$ has been sufficiently close to 1 for m steps, then END, else increment to the next row and column (or row 1, column 1 if at the end) and go to step 1.

The restriction placed earlier on A prevents R_i from equalling 0, which bounds $\bar{d}_i$. As stated in the algorithm, iterations are continued until k is large enough to have $\bar{d}_i$ sufficiently close to 1 for m consecutive steps, indicating that further balancing steps will be of little advantage.

5.4.1.3 Extension of Osborne Method to the Batch Processing Case

Performing an element by element calculation to create D could become prohibitively slow with a large matrix. To efficiently form a D scale, the following equations are used to create vectors of S and R components. These vectors will be used to find the elements of D by performing the algorithm calculations for $\bar{d}_i$ in vector format. Because these summations are intrinsic functions in Matlab®, the calculations are performed significantly faster. As before, the prime notation indicates the element $M_{i,j}$ where $i = j$ has been removed.

$$\underline{S} = \left[\sum_{i=1}^{n} |M_{i,1} \sum_{i=1}^{n} |M_{i,2} \sum_{i=1}^{n} |M_{i,3} \sum_{i=1}^{n} |M_{i,4} \ldots \sum_{i=1}^{n} |M_{i,n}\right]^{\frac{1}{2}} \tag{5.25}$$

$$\underline{R} = \left[\sum_{i=1}^{n} |M_{1,i} \sum_{i=1}^{n} |M_{2,i} \sum_{i=1}^{n} |M_{3,i} \sum_{i=1}^{n} |M_{4,i} \ldots \sum_{i=1}^{n} |M_{n,i}\right]^{\frac{1}{2}} \tag{5.26}$$

Then the $\bar{d}_i$ components are formed by

$$\bar{d} = (\underline{S}./\underline{R})^{\frac{1}{2}}, \tag{5.27}$$

where ./ indicates point by point division of the two vectors (standard Matlab® notation). This method is not equivalent to the original Osborne method, but has been empirically found to have convergence to the required D-scales in practice. Current μ tools code uses this methodology in creating its D-scales, rather than the exact derivation in Osborne's paper.

5.4.2 Implications When Extending Osborne Balancing to Skew μ

Consider the case where M has been previously balanced, then M_s corresponds in the general case to

$$M_s = \sqrt{S^{-1}} D M D^{-1} \sqrt{S^{-1}}, \tag{5.28}$$

or in expanded form

$$M_s = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu}} \cdot I_\nu \end{bmatrix} \begin{bmatrix} d_1 & & \\ & \ddots & \\ & & d_n \end{bmatrix} \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} d_1^{-1} & & \\ & \ddots & \\ & & d_n^{-1} \end{bmatrix} \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu}} \cdot I_\nu \end{bmatrix}. \tag{5.29}$$

In this case, the skewing matrix S has been administered as $\sqrt{S}$ from the left and right. The matrices $\sqrt{S}$ and D commute, so that the matrix M_s is formed,

$$M_s = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu}} \cdot I_\nu \end{bmatrix} \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu}} \cdot I_\nu \end{bmatrix} = \begin{bmatrix} M_{11} & \frac{1}{\sqrt{\nu}} \cdot M_{12} \\ \frac{1}{\sqrt{\nu}} \cdot M_{21} & \frac{1}{\nu} \cdot M_{22} \end{bmatrix}. \tag{5.30}$$

Pictorially, the matrix $\hat{M}$ is displayed in figure 5.5. Since the skewing matrix is block partitioned along lines of the fixed and varying sections, then $\hat{M}$ is also block partitioned along these lines.

Using the Osborne balancing method as guide, the D-scaling matrix can be restated in relation to $\hat{M}$ as follows:

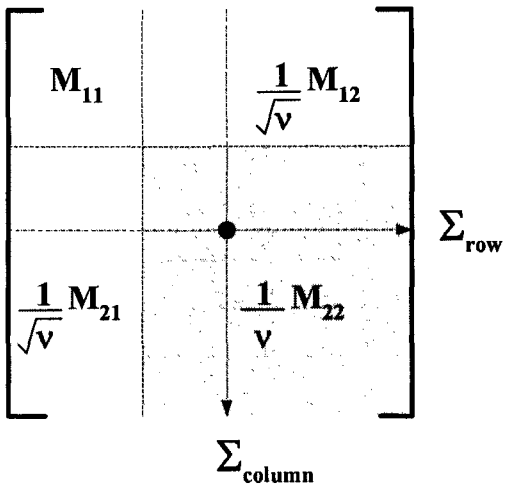


Figure 5.5: Block partitioning of a skewed Osborne matrix

Quadrant 1:

$$\bar{d}_i = \left(\frac{\sum_{p=1}^{f} \prime|a_{pi}|^2 + \frac{1}{\sqrt{\nu}} \sum_{p=f+1}^{n} \prime|a_{pi}|^2}{\sum_{p=1}^{f} \prime|a_{ip}|^2 + \frac{1}{\sqrt{\nu}} \sum_{p=f+1}^{n} \prime|a_{ip}|^2} \right)^{\frac{1}{4}} \tag{5.31}$$

Quadrant 4:

$$\bar{d}_i = \left(\frac{\frac{1}{\sqrt{\nu}} \sum_{p=1}^{f} \prime|a_{pi}|^2 + \frac{1}{\nu} \sum_{p=f+1}^{n} \prime|a_{pi}|^2}{\frac{1}{\sqrt{\nu}} \sum_{p=1}^{f} \prime|a_{ip}|^2 + \frac{1}{\nu} \sum_{p=f+1}^{n} \prime|a_{ip}|^2} \right)^{\frac{1}{4}} \tag{5.32}$$

In this case quadrant 1 is the upper left quadrant, and quadrant 4 is the lower right quadrant, where the point of quadrant separation depends on the number of fixed (f) variables. The factor $\bar{d}_i$ is calculated along the diagonal (index ii) and thus only exists in quadrants 1 and 4. Also note that the summation components of equations 5.31 and 5.32 can be calculated and stored for reuse .

The cyclic problem here is ν affects the value of the D-scales, but the D-scales are being found in order to calculate ν. This implies an iterative approach in finding the skew μ upper bound with Osborne balancing. The iterative procedure can be sped up by using the stored

factors from equations 5.31 and 5.32, thus only requiring a change in choice of ν value and a simple division in recalculating D-scales.

5.4.3 The Osborne Skew μ Upper Bound Algorithm

Combining the information of previous section 5.4.2 with the algorithm introduced in section 5.3, a second, more accurate algorithm can be created for finding the skew μ upper bound. Initialization of the skewing matrix should be $S = I$. S and ν are as indicated in previous section 5.4.2.

1. Identify the system matrix M and rearrange it so the perturbations requiring fixed size can be grouped to interact in a single quadrant.
2. Create an appropriate sized skewing matrix S
3. Form $\sqrt{S^{-1}}M\sqrt{S^{-1}}$
4. Perform Osborne balancing to find D-scales
5. Create $\hat{M} = D\sqrt{S^{-1}}M\sqrt{S^{-1}}D^{-1}$
6. Calculate $\nu = \mu_{ub}$ (upper bound) using formula of section 5.5.
7. Evaluate μ_{ub} as follows:
   - If $1 - \mu_{ub} < tolerance$, stop
   - If $1 - \mu_{ub} > tolerance$, include new value of ν in S , Goto 3

Number 6 above references a method for directly calculating the skew μ upper bound problem. This method is contained in section 5.5, and is used in association with the above algorithm.

This algorithm was the first of the two methods used to compute a skew μ upper bound in the software tools. The second method is the skew μ LMI upper bound, which follows in section 5.6.

5.5 Direct Computation for a Skew μ Upper Bound

Direct computation for the skew μ upper bound is based upon forming the skew μ upper bound problem as a generalized eigenvalue problem. The method for doing this proceeds in the following paragraphs.

From equations 5.12 and 5.14, it is apparent that the upper bound on $\mu_s(M)$ can be translated to

$$\overline{\sigma}(\sqrt{S^{-1}}M\sqrt{S^{-1}}) \leq 1. \tag{5.33}$$

It is useful to note that S could be of two different sizes, and denoted as S_L and S_R, for the left and right sided S. The derivation here describes the square case. However, the principals can be extended to cover the non-square case, and skew μ software tools have the non-square case built into their algorithms.

Equation 5.33 can be expanded to

$$\overline{\sigma}(\sqrt{S^{-1}}M\sqrt{S^{-1}}) = \bar{\lambda}([\sqrt{S^{-1}}M\sqrt{S^{-1}}][\sqrt{S^{-1}}M\sqrt{S^{-1}}]^H) = \bar{\lambda}([\sqrt{S^{-1}}M\sqrt{S^{-1}}\sqrt{S^{-1}}^H M\sqrt{S^{-1}}^H). \tag{5.34}$$

Note that $S^H = S$ and $\sqrt{S^{-1}}^H = \sqrt{S^{-1}}$, which implies that

$$\sqrt{S^{-1}}MS^{-1}M^H\sqrt{S^{-1}} \leq I, \tag{5.35}$$

or

$$MS^{-1}M^H \leq S. \tag{5.36}$$

From previous definitions,

$$S = \begin{bmatrix} I_f & 0 \\ 0 & \nu \cdot I_v \end{bmatrix} \tag{5.37}$$

and

$$S^{-1} = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\nu} \cdot I_v \end{bmatrix}. \tag{5.38}$$

Partition S, such that

$$S = X + \nu Y = \begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix} + \nu \begin{bmatrix} 0 & 0 \\ 0 & I_v \end{bmatrix} \tag{5.39}$$

and

$$S^{-1} = X + \frac{1}{\nu}Y = \begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\nu}\begin{bmatrix} 0 & 0 \\ 0 & I_v \end{bmatrix}. \tag{5.40}$$

Substituting these definitions into equation 5.36 gives

$$M(X + \frac{1}{\nu}Y)M^H \leq X + \nu Y \tag{5.41}$$

or

$$MXM^H + \frac{1}{\nu}MYM^H - X - \nu Y \leq 0. \tag{5.42}$$

Examine the terms of equation 5.42 to get

$$MXM^H - X = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}\begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} M_{11}^H & M_{21}^H \\ M_{12}^H & M_{22}^H \end{bmatrix} - \begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix}$$

$$= \begin{bmatrix} M_{11}M_{11}^H - I_f & M_{11}M_{21}^H \\ M_{21}M_{11}^H & M_{21}M_{21}^H \end{bmatrix} = \begin{bmatrix} U & W \\ W^H & V \end{bmatrix} \tag{5.44}$$

and

$$MYM^H = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}\begin{bmatrix} 0 & 0 \\ 0 & I_v \end{bmatrix}\begin{bmatrix} M_{11}^H & M_{21}^H \\ M_{12}^H & M_{22}^H \end{bmatrix}$$

$$= \begin{bmatrix} M_{12}M_{12}^H & M_{12}M_{22}^H \\ M_{22}M_{12}^H & M_{22}M_{22}^H \end{bmatrix} = \begin{bmatrix} E & G \\ G^H & F \end{bmatrix}. \tag{5.46}$$

At this point, one notes that it is possible to construct a general, augmented problem,

$$\left(\left[\begin{array}{cc|c} U & W & 0 \\ W^H & V & I \\ \hline 0 & I & 0 \end{array}\right] + \frac{1}{\nu}\left[\begin{array}{cc|c} E & G & 0 \\ G^H & F & 0 \\ \hline 0 & 0 & I \end{array}\right]\right)\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0. \tag{5.47}$$

One also notes that this problem is of the form $Ax = \lambda Bx$, a generalized eigenvalue problem, which can be readily solved using conventional software tools.

Now show that this general case applies to the specific problem 5.42, by obtaining and reducing equation set 5.48 from equation 5.47

$$\begin{aligned} Ux_1 + Wx_2 + \frac{1}{\nu}(Ex_1 + Gx_2) &= 0 \\ W^H x_1 + Vx_2 + x_3 I + \frac{1}{\nu}(G^H x_1 + Fx_2) &= 0 \\ x_2 I + \frac{1}{\nu}x_3 I &= 0. \end{aligned} \tag{5.48}$$

Note that

$$x_3 I = -\nu x_2 I. \tag{5.49}$$

Thus the reduced equations are

$$\begin{aligned}(U + \frac{1}{\nu}E)x_1 + (W + \frac{1}{\nu}G)x_2 = 0 \\ (W^H + \frac{1}{\nu}G^H)x_1 + (V - \nu I + \frac{1}{\nu}F)x_2 = 0.\end{aligned} \tag{5.50}$$

Reforming equation 5.50 into a matrix form,

$$\begin{bmatrix} U + \frac{1}{\nu}E & W + \frac{1}{\nu}G \\ W^H + \frac{1}{\nu}G^H & V - \nu I + \frac{1}{\nu}F \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0. \tag{5.51}$$

Now, substituting in the individual components one gets

$$\begin{bmatrix} M_{11}M_{11}^H - I_f + \frac{1}{\nu}M_{12}M_{12}^H & M_{11}M_{21}^H + \frac{1}{\nu}M_{12}M_{22}^H \\ M_{21}M_{11}^H + \frac{1}{\nu}M_{22}M_{12}^H & M_{21}M_{21}^H - \nu I_v + \frac{1}{\nu}M_{22}M_{22}^H \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = 0. \tag{5.52}$$

At this point, one notes that this is easily reduced to the equation of interest, (5.42). Thus equation (5.47) can be readily solved for the desired value ν. This value is found at the equality. The inequality was originally constructed because the desire was to find an upper bound on the value of skew μ.

The problem has been solved for a ν_i that sets $\sigma_i(SM) = 1$. This does not guarantee that $\sigma_i(SM) = \overline{\sigma}(SM)$. Now, suppose that another ν_b is found such that solves equation (5.47), such that $\nu_b < \nu_i$. However, if $\nu_b < \nu_i$, then $\sigma_b(SM) < \sigma_i(SM)$, which means that $\sigma_b(SM) < 1$, a contradiction on the requirements of the solution. Thus if ν_i solves the equations, then $\sigma_i(SM) = \overline{\sigma}(SM)$. □

5.6 Skew μ LMI Upper Bound

Over the past 15 years Linear Matrix Inequalities (LMI)'s have emerged as a powerful tool in solving control problems. The history of LMI's starts in the 1940's, and came into vogue with the focus on optimal control in the 1960's. However, only systems with explicit analytical solutions were solvable. Since that time, powerful numerical interior point techniques [34] have been developed which find a numerical solution to LMIs in a practical, polynomial time manner.

Because of the availability of fast LMI solvers, robust control research has experienced a shift in focus. Rather than arriving at an analytical solution, the goal is to reformulate a given problem to an LMI where it can be verified as solvable, unsolvable, or optimized over constraints. Because of the power of these solvers and the ability to cast the skew μ problem as an LMI, this technique was chosen as a second method for computing an upper bound for skew μ. The material covered in this section is not designed to be an in depth coverage on LMI's. Background material on LMI's for control theory may be found in the books by [44] and [5].

The theory created in this section was implemented using Matlab®. Several software packages are available for Matlab® that allow coding of general LMI problems and provide efficient tools to solve typical control problems. Among these is the LMI Control Toolbox, which was used for implementation of the theory in this section. Other tools may be readily found by searching the internet.

5.6.1 Computational Aspects for Polynomial Time LMI Solvers

The LMI Control Toolbox gives number of flops, $N(\varepsilon)$, needed to compute an ε accurate solution using one of its solvers as bounded by

$$N(\varepsilon) \leq MN^3 log(\frac{V}{\varepsilon}), \tag{5.53}$$

where M is the total row size of the LMI system, N is the total number of scalar decision variables, and V is a data dependent scaling factor. The Matlab® LMI Control Toolbox implements the projective algorithm of Nesterov and Nemirovski [34],[33]. The data dependent scaling factor, V, is believed to be near 5 for many practical problems, but hard data on complexity of practical problems is not readily available. The most significant factor in the formulation is N^3. The number of flops increases on a cubic rate with the number of decision variables. For small to medium sized problems, this is generally not overly detrimental to computation. In the case of large scale problems such as the power systems problem, the number of decision variables can easily exceed 5000. This becomes computationally expensive, at best, and prohibitive if the computer cannot adequately deal with the storage requirements for problems of this size. Specifically, the 4 machine power systems problem generates 5440 different variables and has a tendency to exceed Matlab® memory allocation causing the software to fail. This problem occurred on a current high end PC with a 1.7 Ghz Pentium processor with 512MB of RAM and a 40 GB hard drive. Obviously the number of decision variables generated by the 50 machine power system would be enormous, and software failure almost guaranteed. These issues concerning power systems will be addressed later in chapter 6.

5.6.2 Skew μ LMI Upper Bound Derivation

It is possible to derive an expression that provides for acquiring the D-scales as well as skew μ in terms of an LMI. There are several sources where a basic form for this LMI can be found, in particular [5], [23], and [25]. In [25] the actual skew μ upper bound is given without derivation. Within LMI terminology, this problem is commonly referred to as the generalized eigenvalue problem (GEVP).

To obtain an understanding of the GEVP, start from theorem 1, where it was shown that

$$\mu(M_s) \leq 1.$$

Consequently, from μ theory it can be stated that

$$\overline{\sigma}(D_L M_s D_R^{-1}) \leq 1. \tag{5.54}$$

Equation 5.54 can be expressed as

$$(D_L M_s D_R^{-1})^H (D_L M_s D_R^{-1}) \leq I. \tag{5.55}$$

Noting that the matrix S^{-1} commutes with D, the previous equation can be written as

$$(D_L M D_R^{-1} S^{-1})^H (D_L M D_R^{-1} S^{-1}) \leq I. \tag{5.56}$$

Expanding this equation gives

$$S^{-1}(D_R^{-1})^H M^H D_L^H D_L M D_R^{-1} S^{-1} \leq I. \tag{5.57}$$

Manipulating the previous result provides

$$(D_R^{-1})^H M^H D_L^H D_L M D_R^{-1} \leq S^2, \tag{5.58}$$

which can be further re-arranged to

$$M^H D_L^H D_L M \leq D_R^H D_R S^2. \tag{5.59}$$

Let $P_L \triangleq D_L^H D_L$ and $P_R \triangleq D_R^H D_R$, which have a Hermitian matrix structure [28], then

$$M^H P_L M - \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 I_\nu \end{bmatrix} P_R \leq 0. \tag{5.60}$$

Partitioning along the lines of the fixed and varying components, letting $\lambda_\nu = \nu^2$, and manipulating gives,

$$M^H P_L M - \begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix} P_R \leq \lambda_\nu \begin{bmatrix} 0 & 0 \\ 0 & I_\nu \end{bmatrix} P_R. \tag{5.61}$$

Note that equation 5.61 is approximately in the form of an eigenvalue problem wher λ_ν is the square of the value sought. The desired value to be found is $\sqrt{\lambda_\nu}$.

By partitioning all matrices along the size of the fixed and varying perturbations and re-arranging, the following can be arrived at (notation change $P^L \triangleq P_L$ and $P^R \triangleq P_R$ to help reduce equation clutter),

$$\begin{bmatrix} M_{11}^H & M_{21}^H \\ M_{12}^H & M_{22}^H \end{bmatrix} \begin{bmatrix} P_{11}^L & 0 \\ 0 & P_{22}^L \end{bmatrix} \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} - \begin{bmatrix} P_{11}^R & 0 \\ 0 & 0 \end{bmatrix} \leq \begin{bmatrix} 0 & 0 \\ 0 & \lambda_v \cdot P_{22}^R \end{bmatrix} \tag{5.62}$$

To formulate this as an LMI compatible with the Matlab ® LMI Controls Toolbox [23], equation 5.62 is broken down in the following manner,

$$\begin{bmatrix} M_{11}^H P_{11}^L M_{11} + M_{21}^H P_{22}^L M_{21} - P_{11}^R & M_{11}^H P_{11}^L M_{12} + M_{21}^H P_{22}^L M_{22} \\ M_{12}^H P_{11}^L M_{11} + M_{22}^H P_{22}^L M_{21} & M_{12}^H P_{11}^L M_{12} + M_{22}^H P_{22}^L M_{22} - \lambda_v \cdot P_{22}^R \end{bmatrix} \leq 0. \tag{5.63}$$

Here, 0 is an appropriately size $n \times n$ matrix. The GEVP solver from Matlab ® requires this equation be broken up into several constraints to avoid computational problems with (internal) semi-definite conditions, hence the following form is necessary,

$$\begin{bmatrix} M_{11}^H P_{11}^L M_{11} + M_{21}^H P_{22}^L M_{21} - P_{11}^R & M_{11}^H P_{11}^L M_{12} + M_{21}^H P_{22}^L M_{22} \\ M_{12}^H P_{11}^L M_{11} + M_{22}^H P_{22}^L M_{21} & M_{12}^H P_{11}^L M_{12} + M_{22}^H P_{22}^L M_{22} - Y \end{bmatrix} \leq 0$$

$$\begin{aligned} Y &< \lambda_v \cdot P_{22}^R \\ 0 &< P_{22}^L \\ 0 &< P_{11}^L \\ 0 &< Y. \end{aligned} \tag{5.64}$$

The optimization variables of the LMI in equation 5.64 are the elements of matrices $P_{11}^L, P_{22}^L, P_{11}^R, P_{22}^R$ and Y. These matrices have a Hermitian structure as explained previously in this section. The matrix Y has a structure equivalent to that of P_{22}^R, and acts like a slack variable to allow the GEVP solver to avoid internal computation problems. More details on the requirements of LMI's with similar, semi-definite structure are available in the Matlab® LMI Controls Toolbox manual [23].

This formulation is created for complex skew μ, and therefore, only a crude bound for the mixed skew μ problem. Real perturbations may be included; however, depending on the nature of the problem, the bound may be overly conservative. An extension to this method which improves the bound for real perturbations is found in section 5.6.3.

From this derivation, the following can be said about skew μ and the matrix M:

$$\mu_s(M) \leq \nu, \tag{5.65}$$

and

$$\nu = \inf_{D \in \mathcal{D}_{\mathcal{K}_f, \mathcal{K}_v}} \min_{0 \leq \nu \in \mathbb{R}} \left\{ \nu : (DMD^{-1})^H (DMD^{-1}) \leq \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 \cdot I_v \end{bmatrix} \right\} \tag{5.66}$$

or

$$\mu_s(M) \leq \inf_{D \in \mathcal{D}_{\mathcal{K}_f, \mathcal{K}_v}} \min_{0 \leq \nu \in \mathbb{R}} \left\{ \nu : M^H D_L^H D_L M - \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 \cdot I_v \end{bmatrix} D_R^H D_R \leq 0 \right\}. \tag{5.67}$$

Where λ_v has been replaced by ν^2 to be consistent with other works in this area. A brief reference to this can be found in [25].

Although equation 5.64 is almost directly programmable into Matlab®, there are a few cautions to make. Note that P_R and P_L are Hermitian. Care must be given to setting up the variable structure of these matrices in the LMI toolbox code, especially for the complex part of the variables, which will have no decision variables on the diagonal, since the diagonal of a Hermitian matrix is real. Another note of importance is that the number of variables can grow tremendously as the size of the problem increases. What may appear to be a reasonable size problem can easily generate over 5000 variables, enough to overburden the computational horsepower of many PC's.

Having said this, the skew μ LMI method provides answers that are generally more accurate, and never worse than the answers from the skew μ Osborne method. An additional set of parameters added to the preceding technique can improve the answers of the skew μ LMI bound in the presence of real perturbations. This particular methodology is known as G-scales. It exists in standard μ analysis, and may be extended to skew μ analysis as well.

5.6.3 Inclusion of G-Scales

Real parameter variations take place along the real axis (oddly enough), whereas complex parameter variations are assumed to vary on a disk in the complex plane with a radius of the specified Δ. This extra piece of *phase* information about real perturbations allows for the construction of disks which cover the uncertainties, but which are not necessarily centered at the operation point in the complex plane. Ultimately this yields a less conservative upper bound on μ.

To obtain this less conservative bound in the case of skew μ, first consider a set of scaling matrices constructed as

$$\mathcal{G}_{\mathcal{K}} = \left\{ block\ diag(G_1, \ldots, G_{m_r}, 0_{k_{m_{r+1}}}, \ldots, 0_{k_m}) : G_i = G_i^H \in \mathbb{C}^{k_i \times k_i} \right\} \tag{5.68}$$

for the standard μ case. Also note that a set of scaling will be required for the skew μ case, which will be formed in a similar fashion to D-scales for skew μ. The G-scales associated with the fixed range perturbations are defined as

$$\mathcal{G}_{\mathcal{K}_f} = \left\{ block\ diag(G_1, \ldots, G_{m_{r_f}}, 0_{k_{m_{r_f+1}}}, \ldots, 0_{k_m}) : G_i = G_i^H \in \mathbb{C}^{k_i \times k_i} \right\} \tag{5.69}$$

and the G-scales associated with the varying range perturbations are defined as

$$\mathcal{G}_{\mathcal{K}_v} = \left\{ block\ diag(G_1, \ldots, G_{m_{r_v}}, 0_{k_{m_{r_v+1}}}, \ldots, 0_{k_m}) : G_i = G_i^H \in \mathbb{C}^{k_i \times k_i} \right\}. \tag{5.70}$$

The combined G-scales for skew μ may then be written as

$$\mathcal{G}_{\mathcal{K}_f, \mathcal{K}_v} = \{ G = block\ diag(G_f, G_v) \}. \tag{5.71}$$

The subscript m_r indicates the number of real perturbation blocks, and the subscript m_{r+1} to k_m indicates the number of complex block and full block perturbations. Note that the matrix G only affects the parts of M subject to real perturbations, not the complex or full block perturbations, and is Hermitian in structure. This structure corresponds to the structure set forth in section 2.4 where the complex perturbation elements have been set to 0.

5.6.3.1 G-Scale Derivation

Starting with the basic eigenvalue form $(I - \frac{1}{\nu}\Delta M)x = 0$, the case for G scales can be derived. For $G \in \mathcal{G}$, the following set of equations holds true, assuming $\nu \neq 0$. If there exists a Δ and x such that

$$\Delta M x = \frac{1}{\nu} x, \tag{5.72}$$

then

$$x = \nu \Delta M x \tag{5.73}$$

$$x^H G M x = \nu (x^H M^H \Delta^H) G M x \tag{5.74}$$

$$x^H G M x = x^H M^H G^H \nu \Delta M x \tag{5.75}$$

$$x^H G M x = x^H M^H G^H x \tag{5.76}$$

$$x^H (GM - M^H G^H) x = 0. \tag{5.77}$$

Using properties of Hermitian matrices, section A.1 or [28], $(GM - M^H G^H)$ is skew-Hermitian and $j(GM - M^H G^H)$ is Hermitian. Equation 5.77 is in the form of an LMI with the exception of the equality sign. Equation 5.77 can be used in addition to the previous upper bound LMI derivation 5.62 to achieve a less conservative upper bound.

5.6.3.2 Upper Bound including G-Scales

Using the derivation of the G-scales from the previous section, the following construction of the upper bound can be created (see [18] , [56] or [54] for more background). This is the same form as equation 5.67 with the exception of the addition of G-scales.

$$\mu_s(M) \leq \inf_{\substack{D \in \mathcal{D}_{\mathcal{K}_f, \mathcal{K}_v} \\ G \in \mathcal{G}_{\mathcal{K}_f, \mathcal{K}_v}}} \min_{0 \leq \nu \in \mathbb{R}} \left\{ \nu : M^H D_L^H D_L M + j(GM - M^H G) - \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 \cdot I_\nu \end{bmatrix} D_R^H D_R \leq 0 \right\} \tag{5.78}$$

Note that if $G = 0$ in equation 5.78, then the standard complex skew μ upper bound is recovered.

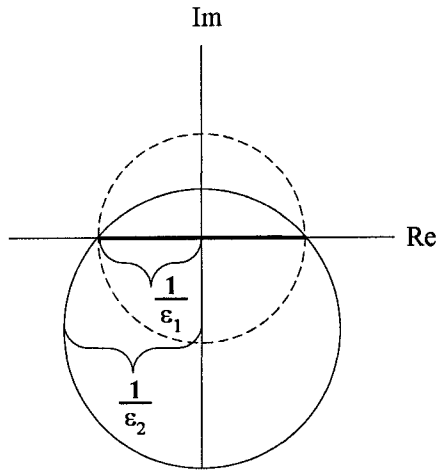


Figure 5.6: Real perturbation and perturbation disks. One disk is centered, and another is adjusted by G-scales

The graphical implication is shown in figure 5.6. The G-scales shift the center of the perturbation disk such that the real perturbations are covered by a less conservative estimate for the upper bound. In figure 5.6, $\mu_s \leq \varepsilon_2 \leq \varepsilon_1$. The use of the phase information of the reals has yielded a less conservative upper bound on μ_s.

The information from this section is applicable to skew-μ as well as μ and is applied to the skew-μ upper bound to arrive at a set of equations in section 5.6.3.3.

5.6.3.3 The Specifics of a Skew μ Upper Bound Including G-Scales

The specifics of the skew μ LMI equations are laid out for clarity. As with the original skew μ LMI care needs to be taken when setting up the variables for solution because of their Hermitian nature.

Referencing equation 5.61 and 5.63, the following LMI's including G-scales are achieved

$$M^H P^L M + j(GM - M^H G^H) - \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 \cdot I_\nu \end{bmatrix} P^R \leq 0. \tag{5.79}$$

Previously, the structure of the G-scales was

$$\begin{bmatrix} G_r & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \leq 0; \tag{5.80}$$

however, when dealing with skew-μ the G-scales must be organized according to fixed and varying reals as well. This makes the structure

$$G = \begin{bmatrix} G_{r_f} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{r_v} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{5.81}$$

required. In these instances, the 0 entries represent the sections pertaining to fixed complex (δ_f^c), fixed full block (Δ_f^C), varying complex (δ_v^c) and varying full block (Δ_v^C) perturbations of the perturbation structure of Δ. The formal definition for $G \in \mathcal{G}_{\mathcal{K}_f,\mathcal{K}_v}$ was given in equation 5.71.

Partitioning the G-scales in a similar fashion to the LMI's of section 5.6, the expression for $j(GM - M^H G^H)$ is achieved,

$$j\left(\begin{bmatrix} G_{11} & 0 \\ 0 & G_{22} \end{bmatrix}\begin{bmatrix} M_{11}^H & M_{21} \\ M_{12} & M_{22} \end{bmatrix} - \begin{bmatrix} M_{11}^H & M_{21}^H \\ M_{12}^H & M_{22}^H \end{bmatrix}\begin{bmatrix} G_{11}^H & 0 \\ 0 & G_{22}^H \end{bmatrix}\right). \tag{5.82}$$

Simplifying this to an expression that can be inserted in the skew μ upper bound LMI,

$$j\left(\begin{bmatrix} G_{11}M_{11} - M_{11}^H G_{11}^H & G_{11}M_{12} - M_{21}^H G_{22}^H \\ \\ G_{22}M_{21} - M_{12}^H G_{11}^H & G_{22}M_{22} - M_{22}^H G_{22}^H \end{bmatrix}\right). \tag{5.83}$$

Finally, inserting the detailed G-scale expression into the LMI upper bound expression, equation 5.63,

$$\begin{bmatrix} M_{11}^H P_{11}^L M_{11} + M_{21}^H P_{22}^L M_{21} + j(G_{11}M_{11} - M_{11}^H G_{11}^H) - P_{11}^R \\ \\ M_{12}^H P_{11}^L M_{11} + M_{22}^H P_{22}^L M_{21} + j(G_{22}M_{21} - M_{12}^H G_{11}^H) \end{bmatrix}$$

$$\begin{bmatrix} M_{11}^H P_{11}^L M_{12} + M_{21}^H P_{22}^L M_{22} + j(G_{11}M_{12} - M_{21}^H G_{22}^H) \\ M_{12}^H P_{11}^L M_{12} + M_{22}^H P_{22}^L M_{22} + j(G_{22}M_{22} - M_{22}^H G_{22}^H) - \lambda_v P_{22}^R \end{bmatrix} \leq 0. \tag{5.85}$$

This expression can be rearranged as in equation set 5.64 to conform to the proper format for implementation via Matlab® LMI Toolbox. One other note must be made on the insertion of G-scales into skew-μ LMI upper bound.

5.6.3.4 A Brief Update to the Requirements of G-Scales

Previous works concerning G-scales in [18] and [54] specifically state that G is Hermitian. This statement needs clarification. In equation 5.81, the requirement is that G_{r_f} and G_{r_v} be Hermitian, however, zeros padded to the matrix G may make G non-Hermitian . For G to be Hermitian requires that G be square, and prevents the use of G-scales with non-square systems. Real perturbations and complex perturbations are square by design, however, full complex block perturbations may be non-square in nature, creating a non-square system, and necessitating a non-square G if G-scales are to be used. Thus, as long as G_{r_f} and G_{r_v} are Hermitian the LMI upper bound that has been developed holds. These comments also extend to the case where G may be of differing sizes for left and right side multiplication. It is possible to have a G_L and G_R of differing dimensions. The previous development of this chapter has assumed that left and right side G have been of the same size.

5.6.4 Complex-Valued LMIs

Typically the matrices encountered in control systems are complex, however, LMI solvers are generally written for real-valued matrices and cannot directly handle LMIs that contain complex valued matrices. However, a complex valued LMI can be turned into a real valued LMI by noting that a complex Hermitian matrix A(x) satisfies

$$A(x) < 0 \iff \begin{bmatrix} Re(A(x)) & Im(A(x)) \\ -Im(A(x)) & Re(A(x)) \end{bmatrix} < 0 \tag{5.86}$$

This suggests the following procedure for turning complex LMIs into real LMIs.

- Decompose the complex matrix variable X as $X_1 + jX_2$, where X_1 and X_2 are real matrices.
- Decompose the complex matrix variable coefficients A as $A_1 + jA_2$, where A_1 and A_2 are real matrices.
- Carry out the complex matrix products to form $A(x)$, and create the matrix as in 5.86.

- If there are no outer factors to the LMI, then matrices such as in 5.86 may be substituted for the original equation matrices. For example:

$$M^H X M < X \tag{5.87}$$

may be replaced with

$$\begin{bmatrix} M_1 & M_2 \\ -M_2 & M_1 \end{bmatrix}^T \begin{bmatrix} X_1 & X_2 \\ -X_2 & X_1 \end{bmatrix} \begin{bmatrix} M_1 & M_2 \\ -M_2 & M_1 \end{bmatrix} < \begin{bmatrix} X_1 & X_2 \\ -X_2 & X_1 \end{bmatrix}, \tag{5.88}$$

where subscripts $_1$ and $_2$ are as defined in the bullets above. If there are outer factors to the LMI, then the outer factors may be combined with the remainder of the equation, however, it complicates the work required to reformulate the system for complex values.

Examples of this may be found in the Matlab® LMI Control Toolbox Reference Manual [23].

5.7 Conversion of D-scales between the Osborne and LMI Upper Bounds

The implementation of the skewing matrix in the Osborne upper bound is through a symmetrical scaling that does not interfere with the norm balance of the system. However, the implementation of the skewing matrix in the LMI form of the upper bound is one sided, thus the D-scales found in the LMI form are not exactly the same as those found for the Osborne upper bound and a method of conversion between the two needs to be found so that the D-scales returned from evaluation are consistent. The following explanation is based on, and similar to derivations found in [18] and [54].

Consider the skew μ form of the LMI including G-scales,

$$M^H D_L^H D_L M + j(GM - M^H G) - \begin{bmatrix} I_f & 0 \\ 0 & \nu^2 \cdot I_v \end{bmatrix} D_R^H D_R \leq 0. \tag{5.89}$$

Construct an S such that

$$S = X + \nu Y = \begin{bmatrix} I_f & 0 \\ 0 & 0 \end{bmatrix} + \nu \begin{bmatrix} 0 & 0 \\ 0 & I_v \end{bmatrix}, \tag{5.90}$$

and make the additional substitution for $D^H D = D_1$ as before, then equation (5.89) becomes

$$M^H D_1^L M + j(G_1 M - M^H G_1) - X D_1^R - \nu^2 Y D_1^R \leq 0. \tag{5.91}$$

The subscript $_1$ is used in this case as an indicator of state 1, the LMI state of the scaling matrices, as they transition to the form of the scaling matrices in the Osborne method. A secondary state can be produced by right multiplying by $D_1^{-\frac{1}{2}}$. At this point the number of subscripts and superscripts becomes excessive, and the ^{L} and ^{R} are dropped. It should be noted that left and right scalings can vary in dimensions due to a non-square M, however the form of the following equations are still valid. The correct size components can be chosen by examining the equations in detail. The resulting equation of the right multiplication is

$$D_1^{-\frac{1}{2}} M^H D_1 M D_1^{-\frac{1}{2}} + j(D_1^{-\frac{1}{2}} G_1 M D_1^{-\frac{1}{2}} - D_1^{-\frac{1}{2}} M^H G_1 D_1^{-\frac{1}{2}}) - D_1^{-\frac{1}{2}} X D_1^{\frac{1}{2}} - \nu^2 D_1^{-\frac{1}{2}} Y D_1^{\frac{1}{2}} \leq 0. \tag{5.92}$$

Let the following substitutions be made, $M_{D_2} = D_1^{\frac{1}{2}} M D_1^{-\frac{1}{2}}$, and $G_2 = D_1^{-\frac{1}{2}} G_1 D_1^{-\frac{1}{2}}$, where $D_2 = D_1^{\frac{1}{2}}$. Note that X and Y remain the same despite the scaling matrices. This resolves equation (5.92) into

$$M_{D_2}^H M_{D_2} + j(G_2 M_{D_2} - M_{D_2}^H G_2) - X \leq \nu^2 Y. \tag{5.93}$$

Equation (5.93) can be manipulated in the following manner to achieve equation (5.96),

$$\frac{1}{\nu^2}(M_{D_2}^H M_{D_2} + j(G_2 M_{D_2} - M_{D_2}^H G_2) - X) \leq Y \tag{5.94}$$

$$\left(\frac{M_{D_2}}{\nu} - \frac{jG_2}{\nu}\right)^H \left(\frac{M_{D_2}}{\nu} - \frac{jG_2}{\nu}\right) - \frac{G_2^2}{\nu^2} - \frac{X^2}{\nu^2} \leq Y \tag{5.95}$$

$$\left(\frac{M_{D_2}}{\nu} - \frac{jG_2}{\nu}\right)^H \left(\frac{M_{D_2}}{\nu} - \frac{jG_2}{\nu}\right) \leq \frac{G_2^2}{\nu^2} + \frac{X}{\nu^2} + Y. \tag{5.96}$$

Equation (5.96), can be arranged to a form convenient for singular values,

$$\left[\left(\frac{M_{D_2}}{\nu}-\frac{jG_2}{\nu}\right)\left(\frac{G_2^2}{\nu^2}+\frac{X}{\nu^2}+Y\right)^{-\frac{1}{2}}\right]^H\left[\left(\frac{M_{D_2}}{\nu}-\frac{jG_2}{\nu}\right)\left(\frac{G_2^2}{\nu^2}+\frac{X}{\nu^2}+Y\right)^{-\frac{1}{2}}\right]\leq I. \tag{5.97}$$

The implication of equation (5.97) in terms of the maximum singular value is

$$\overline{\sigma}\left[\left(\frac{M_{D_2}}{\nu}-\frac{jG_2}{\nu}\right)\left(\frac{G_2^2}{\nu^2}+\frac{X}{\nu^2}+Y\right)^{-\frac{1}{2}}\right]\leq 1. \tag{5.98}$$

Let $D_3=A^{\frac{1}{4}}D_2$ and $G_3=\frac{G_2}{\nu}$, where $A=\left(\frac{G_2^2}{\nu^2}+\frac{X}{\nu^2}+Y\right)=\left(G_3^2+\frac{X}{\nu^2}+Y\right)$. Also note that $A^{\frac{1}{4}}G_3A^{-\frac{1}{4}}=G_3$, then

$$\overline{\sigma}\left[\left(G_3^2+\frac{X}{\nu^2}+Y\right)^{-\frac{1}{4}}\left(\frac{M_{D_3}}{\nu}-jG_3\right)\left(G_3^2+\frac{X}{\nu^2}+Y\right)^{-\frac{1}{4}}\right]\leq 1. \tag{5.99}$$

The strategy at this point changes from pure algebraic manipulation to performing a polar decomposition on D_3. This is done via an SVD on D_3 and regrouping as follows,

$$D_3=U\Sigma V^H=UV^HV\Sigma V^H=[UV^H][V\Sigma V^H]. \tag{5.100}$$

Define a unitary rotation $W=[UV^H]$ and hermitian matrix $P=[V\Sigma V^H]$. Since P is just a unitary rotation of D_3, then it is possible to define $D_4=P$, and $\hat{G}_4=W^HG_3W$ as a unitary rotation of G_3, then equation (5.99) can be written as,

$$\overline{\sigma}\left[\left(\hat{G}_4^2+\frac{X^2}{\nu^2}+Y\right)^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)\left(\hat{G}_4^2+\frac{X^2}{\nu^2}+Y\right)^{-\frac{1}{4}}\right]\leq 1. \tag{5.101}$$

This is close to, but not quite the standard form. To achieve the standard form insert the factor $(\nu^2X+Y)^{-\frac{1}{4}}(\nu^2X+Y)^{\frac{1}{4}}$ on either side of $\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)$. These factors can be combined with the remainder of the equation to achieve

$$\overline{\sigma}\left[\left(\begin{bmatrix}\nu^2\hat{G}_{f4}^2 & 0\\ 0 & \hat{G}_{v4}^2\end{bmatrix}+I\right)^{-\frac{1}{4}}\left(D_4\begin{bmatrix}I_f & 0\\ 0 & \frac{1}{\sqrt{\nu}}I_v\end{bmatrix}M\begin{bmatrix}I_f & 0\\ 0 & \frac{1}{\sqrt{\nu}}I_v\end{bmatrix}D_4^{-1}-\begin{bmatrix}\nu\hat{G}_{f4} & 0\\ 0 & \hat{G}_{v4}\end{bmatrix}\right)\left(\begin{bmatrix}\nu^2\hat{G}_{f4}^2 & 0\\ 0 & \hat{G}_{v4}^2\end{bmatrix}+I\right)^{-\frac{1}{4}}\right]\leq 1. \tag{5.102}$$

If G_4 is defined as

$$G_4 = \begin{bmatrix} \nu I_f & 0 \\ 0 & I_\nu \end{bmatrix} \hat{G}_4, \tag{5.103}$$

then equation (5.102) can be written as

$$\overline{\sigma}\left(\left(I+G_4^2\right)^{-\frac{1}{4}} \left(D_4 S^{-\frac{1}{2}} M S^{-\frac{1}{2}} D_4^{-1} - jG_4\right) \left(I+G_4^2\right)^{-\frac{1}{4}} \right) \leq 1. \tag{5.104}$$

Equation (5.104) is of the same form as the standard μ upper bound,

$$\overline{\sigma}\left(\left(I+G^2\right)^{-\frac{1}{4}} \left(\frac{M}{\beta} - jG\right) \left(I+G^2\right)^{-\frac{1}{4}} \right) \leq 1, \tag{5.105}$$

which can be found in [54].

Note that while $D_3 \in \mathcal{D}_\mathcal{K}$, but due to the polar decomposition, D_4 is in a subset of $\hat{\mathcal{D}}_\mathcal{K}$, i.e. $D_4 \in \hat{\mathcal{D}}_\mathcal{K}$. Likewise, $G_3 \in \mathcal{G}_\mathcal{K}$, but $G_4 \in \hat{\mathcal{G}}_\mathcal{K}$.

Equation (5.104) gives the skew μ upper bound in terms of the maximum singular value while using components from the LMI equations, however, an equally useful item is the conversion of the components themselves. Using the transformations noted in the previous derivation it is possible to show that

$$D_4 = W^H \left(\frac{D_1^{-\frac{1}{2}} G_1^H D_1^{-1} G_1 D_1^{-\frac{1}{2}}}{\nu^2} + \frac{X}{\nu^2} + Y \right) D_1^{\frac{1}{2}} \tag{5.106}$$

and

$$G_4 = \begin{bmatrix} \nu I_f & 0 \\ 0 & I_\nu \end{bmatrix} W^H \frac{D_1^{-\frac{1}{2}} G_1 D_1^{-\frac{1}{2}}}{\nu} W = S W^H D_1^{-\frac{1}{2}} G_1 D_1^{-\frac{1}{2}} W. \tag{5.107}$$

Likewise, a derivation can be performed for conversion from the maximal singular value form to the LMI form, which yields the following conversion factors

$$D_1 = D_4^H \left(G_4^H G_4 + \frac{X}{\nu^2} + Y \right)^{-\frac{1}{2}} D_4 \tag{5.108}$$

and

$$G_1 = \nu D_4^H \left(G_4^H G_4 + \frac{X}{\nu^2} + Y \right)^{-\frac{1}{2}} G_4 D_4. \tag{5.109}$$

In this case, variables subscripted $_1$ are factors associated with the LMI equation form of the skew μ upper bound, and variables subscripted $_4$ are associated with the maximum

singular value form of the upper bound equation, sometimes referred to as the Osborne upper bound.

The underlying result here is a conversion between D and G scales. Since the Osborne upper bound is easier to achieve, its results may be used to create an initial estimation of D's and G's for the LMI upper bound problem in order to increase the speed of calculation and the accuracy. Likewise, the D and G scales from the LMI upper bound may be converted to the more widely used Osborne form when used in supplemental calculations.

For the sake of clarity, going the other direction, from Osborne scalings to LMI scaling is provided in the following paragraphs. Starting with

$$\overline{\sigma}\left(\left(I+G_4^2\right)^{-\frac{1}{4}}\left(D_4S^{-\frac{1}{2}}MS^{-\frac{1}{2}}D_4^{-1}-jG_4\right)\left(I+G_4^2\right)^{-\frac{1}{4}}\right)\leq 1, \tag{5.110}$$

one can let

$$\hat{G}_4=\begin{bmatrix}\frac{1}{\nu}I_f & 0\\ 0 & I_v\end{bmatrix}G_4. \tag{5.111}$$

Then equation (5.110) can be written as

$$\overline{\sigma}\left[\left(\begin{bmatrix}\nu^2\hat{G}_{f4}^2 & 0\\ 0 & \hat{G}_{v4}^2\end{bmatrix}+I\right)^{-\frac{1}{4}}\left(D_4S^{-\frac{1}{2}}M^{-\frac{1}{2}}D_4^{-1}-\begin{bmatrix}\nu\hat{G}_{f4} & 0\\ 0 & \hat{G}_{v4}\end{bmatrix}\right)\left(\begin{bmatrix}\nu^2\hat{G}_{f4}^2 & 0\\ 0 & \hat{G}_{v4}^2\end{bmatrix}+I\right)^{-\frac{1}{4}}\right]\leq 1. \tag{5.112}$$

Which can be re-written as

$$\overline{\sigma}\left[\left(\hat{G}_4^2+\frac{X^2}{\nu^2}+Y\right)^{-\frac{1}{4}}(\nu^2S^{-2})^{-\frac{1}{4}}(\nu^2S^{-2})^{\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)(\nu^2S^{-2})^{\frac{1}{4}}(\nu^2S^{-2})^{-\frac{1}{4}}\left(\hat{G}_4^2+\frac{X^2}{\nu^2}+Y\right)^{-\frac{1}{4}}\right]\leq 1. \tag{5.113}$$

Let $A=\left(\hat{G}_4^2+\frac{X}{\nu^2}+Y\right)$, then write equation (5.113) as

$$\overline{\sigma}\left[A^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)A^{-\frac{1}{4}}\right]\leq 1. \tag{5.114}$$

Changing equation (5.114) to LMI form,

$$\left[A^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)A^{-\frac{1}{4}}\right]^H\left[A^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)A^{-\frac{1}{4}}\right]\leq I. \tag{5.115}$$

Multiplying by $A^{\frac{1}{2}}$ from both sides and employing the ^{H} gives

$$\left[A^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)A^{\frac{1}{4}}\right]^H\left[A^{-\frac{1}{4}}\left(\frac{M_{D_4}}{\nu}-j\hat{G}_4\right)A^{\frac{1}{4}}\right]\leq A. \tag{5.116}$$

Define $\hat{D}=A^{-\frac{1}{4}}D_4$ and $\hat{G}=A^{-\frac{1}{4}}\hat{G}_4A^{\frac{1}{4}}$, and substitute into equation (5.116) to get

$$\left[\left(\frac{M_{\hat{D}}}{\nu}-j\hat{G}\right)\right]^H\left[\left(\frac{M_{\hat{D}}}{\nu}-j\hat{G}\right)\right]\leq A. \tag{5.117}$$

Multiplying out equation (5.117) gives

$$\left(\frac{M_{\hat{D}}}{\nu}\right)^H\frac{M_{\hat{D}}}{\nu}+j\left(\hat{G}^H\frac{M_{\hat{D}}}{\nu}-\left(\frac{M_{\hat{D}}}{\nu}\right)^HG^H\right)+\hat{G}^H\hat{G}\leq A. \tag{5.118}$$

Multiplying the from the left and right by $\hat{D}^H$ and $\hat{D}$ respectively, yields

$$\frac{M^H\hat{D}^H\hat{D}M}{\nu^2}+j\left(\frac{\hat{D}^H\hat{G}^H\hat{D}M}{\nu}-\frac{M^H\hat{D}^H\hat{G}\hat{D}}{\nu}\right)+\hat{D}^H\hat{G}^H\hat{G}\hat{D}\leq\hat{D}^HA\hat{D}. \tag{5.119}$$

Let $\hat{D}^H\hat{D}=\tilde{D}$ and $\nu\hat{D}^H\hat{G}^H\hat{D}=\tilde{G}$, then equation (5.119) reduces to

$$M^H\tilde{D}M+j(\tilde{G}M-M^H\tilde{G})\leq\nu^2\hat{D}^H(A-\hat{G}^H\hat{G})\hat{D}. \tag{5.120}$$

Noting that $\hat{G}=A^{-\frac{1}{4}}G_4A^{\frac{1}{4}}=G_4$, then equation (5.119) can be further reduced to

$$M^H\tilde{D}M+j(\tilde{G}M-M^H\tilde{G})\leq\nu^2\hat{D}^H\left(\frac{X}{\nu^2}+Y\right)\hat{D}=X\tilde{D}+\nu^2Y\tilde{D}. \tag{5.121}$$

Re-writing equation (5.121) into proper form for clean up gives

$$M^H\tilde{D}M+j(\tilde{G}M-M^H\tilde{G})-X\tilde{D}+\nu^2Y\tilde{D}\leq 0, \tag{5.122}$$

which is the original form of the LMI in equation (5.91) where $\tilde{D}=D_1$ and $\tilde{G}=G_1$. Thus the derivation from LMI to Osborne scaling matrices and vice versa has been provided.

Chapter 6

Power Systems

The majority of this chapter is to provide details on the construction of the model for power system analysis. Much of the developmental material contained in this chapter can be found in articles [9], [10] and [8], with additional information from [57], [30] and [48]. It provides clarity for the motivation of the skew μ problem, as well as being a problem of significant importance in its own right.

The material here is not meant to be encompassing, thus for the reader unfamiliar with power systems, a significant amount of basic background material may be required to fully understand this section. This background material may be found in [31], [1] and [29], among others.

The procedure for analyzing a power system is itemized as

1. Identify non-linear models used in modeling an individual generation system.
2. Represent network using steady-state parameters with a constant impedance load model, and assume the generator internal reactance to be constant, which reduces the size of the network model.
3. Couple the reduced network equations to the various generating system components, in a coupled state equation and algebraic equation.
4. Obtain a linearized model of the coupled equation set.

- Obtain a steady-state solution for the equations at several operating points.
- Calculate initial conditions for the state variables using the steady state solutions and power produced at each generator.
- Linearize the state and network equations to achieve a set of SS equations representing the total power system model.

The goal in obtaining a model of a power system is two-fold. First, a model for a power system allows for analysis of the system to determine its relative stability or instability. Secondly, the power system model allows for the design and testing of a controller to improve stability and performance properties of the overall power system. Control of power systems are typically done via Power System Stabilizers (PSS). The PSS is designed to bias the generator excitation system in order to suppress or dampen voltage swings on the transmission line, as shown in figure 6.1. One desirable objective is to apply control design methods developed in this project to power system models with in an attempt to provide a PSS with superior performance to those currently in existence. The techniques that can be applied toward this goal are discussed in chapter 8. Before addressing control design for power systems, the characteristics and dynamics of the generation system must be developed.

Prior to discussing the generation system, the method used for linearizing variables will be given. This is a general method that can be used on any number of problems, and merits a section of its own.

6.1 Linearizing Variables and Forming an LFT

The equations surrounding generation and transmission of electrical current are non-linear. The methods of robust control analysis presented in this document are linear. Thus to analyze the system with these robust analysis techniques generally implies that a method of linearizing the system must be found.

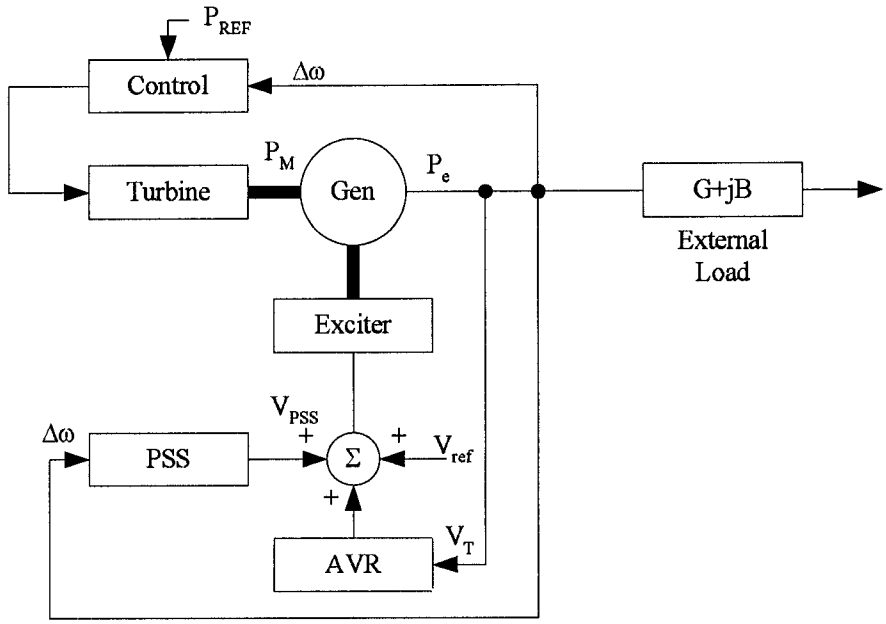


Figure 6.1: Basic generation system diagram showing action of PSS

Consider the A matrix of a SS system. Presume an operating parameter p has values on an interval $[p_{min}, p_{max}]$, then each coefficient of the A matrix depending on p can be approximated by a polynomial expression

$$a_{ij} = a'_{ij0} + a'_{ij1}p + a'_{ij2}p^2. \tag{6.1}$$

It is an advantage to normalize the range of variation of the parameter p to $[-1, 1]$. Thus, the previous equation can be re-written as

$$a_{ij} = a_{ij0} + a_{ij1}\delta + a_{ij2}\delta^2, \tag{6.2}$$

where δ ranges in the interval $[-1, 1]$, and a_{ijk} is dependent on a'_{ijk}, p_{min}, and p_{max}. The dependency of a_{ijk} is as follows: suppose a value of p known as p_k exists on the interval $[p_{min}, p_{max}]$. In terms of δ this can be written as

$$p_k = \frac{p_{max} + p_{min}}{2} + \frac{p_{max} - p_{min}}{2}\delta_k, \tag{6.3}$$

where δ_k is in the range $-1 \leq \delta_k \leq 1$ such that p_k ranges from p_{min} to p_{max} as δ_k ranges from -1 to 1.

Also note equation 6.3 can be substituted in to equation 6.1. One can easily see the product of a'_{ijx} and equation 6.3 will generate extra terms which can be collected in the variables a_{ij0} and a_{ij1} of equation 6.2 to form the new a_{ij} coefficients.

This representation of a_{ij} can be constructed as an LFT of a known matrix with the uncertainty $\delta I^{2\times 2}$. This is graphically represented in figure 6.2.

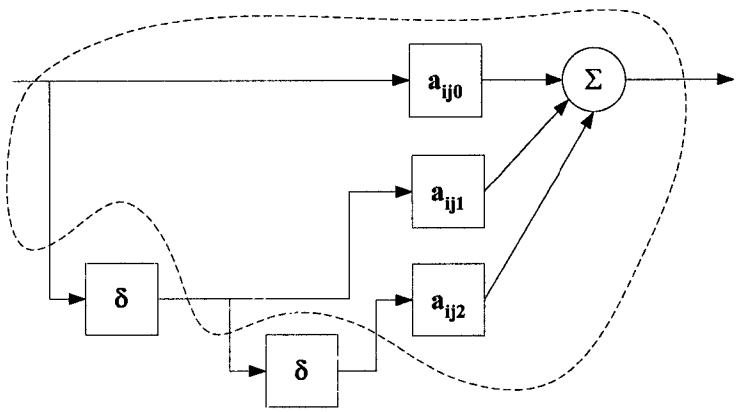


Figure 6.2: LFT diagram of a linearized polynomial

This process of isolating the uncertainties and forming an LFT can be repeated where ever varying parameters appear, such as in the location of the governor system poles or the excitation system poles. Ultimately, the system can be regrouped to form a proper $M-\Delta$ structure using the methods of section 2.2.1, which can then be analyzed using robust analysis techniques.

6.2 Generation System

Generation systems and generators have been researched in theory and practice for over 100 years, and the relationships needed for modeling power system stability have been understood for decades. The models used in this research have been well established, and are available in [31] and [1]. These models are considered standardized, and are widely used to provide for comparison and repeatability of research.

Specifically, the models are the classical generator model, the "two-axis" synchronous machine (SM) with ETMSP Type-30 Excitation model (ES), and the ETMSP Type-1 Power System Stabilizer (PSS).

These models consists of nonlinear differential equations, which will be linearized and coupled to the mathematical description of the transmission system. Much of the content in this section comes from [57], as that project and this are interrelated in terms of scope, goals, and members of the project teams.

6.2.1 Generator Models

The classical generator model, and the two-axis generator model can both be found in [1]. In the multi-machine power system model consisting of n generators, it is presumed the first m generators are represented by the two-axis model and equipped with excitation systems, while the remaining $n-m$ generators are represented by the classical model.

6.2.1.1 Classical Generator Model

The classical model ignores excitation control in its formulation and is one of simplest generator models. The loads are represented by a constant impedance and the load nodes and the terminal voltage nodes of the generators are eliminated. The resulting network contains only the internal generator nodes. The generator reactance and the constant impedance loads are included in the bus admittance matrix, Y_{bus}, of the reduced network.

The dynamic equations for the classical generator model are

$$\begin{aligned} M_i\dot{\omega}_i &= P_i - P_{ei} \\ \delta_i &= \omega_i - \omega_S \end{aligned} \tag{6.4}$$

for $i = m+1, m+2, \ldots, n$ where

$$P_i = P_{mi} - E_i^2 G_{ii} \tag{6.5}$$

and

$$P_{ei} = \sum_{j=1, j\neq i}^{n} [E_i E_j B_{ij} \sin(\delta_i - \delta_j) + E_i E_j G_{ij} \cos(\delta_i - \delta_j)]. \quad (6.6)$$

The variables are defined as

$\boldsymbol{E_i}$ Internal bus voltage of generator i

$\boldsymbol{M_i}$ Moment of inertia for generator i

$\boldsymbol{P_{mi}}$ Mechanical power input to generator i

$\boldsymbol{G_{ii}}$ Driving point conductance of node i

$\boldsymbol{G_{ij} + jB_{ij}}$ the transfer admittance between node i and node j in the reduced network

$\boldsymbol{\omega_i}$ rotor speed of generator i (with respect to the synchronous frame)

$\boldsymbol{\omega_S}$ synchronous speed

$\boldsymbol{\delta_i}$ rotor angle of generator i

$\boldsymbol{P_{ei}}$ Electrical power output of generator i

6.2.1.2 Two-Axis Synchronous Machine (SM)

A diagram of a synchronous machine is depicted in figure 6.1. The two-axis model accounts for transient effects of a synchronous machine, but neglects subtransient effects. The subtransient effects are associated with the initial stator flux linkage following a current change. The subtransient time constant is typically less than 5% of the transient time constant, and on the order of 20 milliseconds, hence the reason it is neglected. See [1] or [29] for numerous details regarding these time constants and their relevance to power system models.

In addition to neglecting the subtransients, the two axis model presumes that the system is operating at near synchronous speed. From these assumptions, the generator dynamic equations are

$$
\begin{aligned}
\tau'_{d0i}\dot{E}'_{qi} &= E_{FDi} - E'_{qi} + (x_{di} - x'_{di})I_{di} \\
\tau'_{q0i}\dot{E}'_{di} &= -E'_{di} - (x_{qi} - x'_{qi})I_{qi} \\
M\dot{\omega}_i &= P_{mi} - \tfrac{D_i}{\omega_r}(\omega_i - \omega_S) - (E'_{di}I_{di} + E'_{qi}I_{qi}) + (x'_{qi} - x'_{di})(I_{di}I_{qi}) \\
\dot{\delta}_i &= \omega_i - \omega_r,
\end{aligned}
\tag{6.7}
$$

where the variables are defined as

E'_{di}, E'_{qi} Direct and quadrature axes stator EMFs corresponding to rotor transient flux components, respectively.

I_{di}, I_{qi} The d and q axes stator currents

τ'_{d0i}, τ'_{q0} Open-circuit direct and quadrature axes transient time constants

x'_{qi}, x'_{qi} quadrature axis synchronous and transient reactances

E_{FDi} stator EMF corresponding to the field voltage

D_i damping coefficient of generator i

δ_i rotor angle of generator i with respect to the reference generator or source

ω_r rotational speed of the defined reference generator or source

ω_i rotational speed of the generator i.

6.2.1.3 Reference Angle

Typical analysis of a generator presumes the rotor angle is an independent state variable. In this analysis the rotor angles of each machine are linked via the transmission grid, hence they are not totally independent. However, a new variable for relative rotor angle can be introduced, where the rotor angle of the first generator in the power system (δ_1) can be taken as the reference, and all other relative rotor angles defined from it. In other words

$$
d_{i1} = \delta_i - \delta_1, \tag{6.8}
$$

where $i = 2,3,...,n$. The dynamic equations of section 6.2.1.2 are unchanged in form, however, δ_i is replaced with δ_{i1}, ω_S is replaced by ω_1, and the rotor angle expression in equation 6.7 becomes

$$\dot{\delta}_i = \omega_i - \omega_1 \tag{6.9}$$

for $i = 2,3,\ldots,n$.

6.2.2 ETMSP Type-30 Exciter

The mathematical model representing the excitation system is the ETMSP Type 30 model, and is graphically represented in figure 6.3. The dynamic representation for this type of excitation system can be written as

$$\begin{aligned}
\dot{E}_{FDi} &= \tfrac{K_{Ai}}{T_{Ai}} X_{E2i} - \tfrac{1}{T_{Ai}} E_{FDi} + \tfrac{aK_{Ai}}{T_{Ai}} (V_{REFi} + V_{PSSi} - X_{E1i}) \\
\dot{X}_{E1i} &= -\tfrac{1}{T_{Ri}} X_{E1i} + \tfrac{1}{T_{Ri}} V_{Ti} \\
\dot{X}_{E2i} &= -\tfrac{1}{T_{Bi}} X_{E2i} + \tfrac{1-a}{T_{Bi}} (V_{REFi} + V_{PSSi} - X_{E1i}) \\
V_T &= V_{Tqi} + jV_{Tdi} = (E'_{qi} + x'_{di} I_{di}) + j(E'_{di} - x'_{qi} I_{qi}) \\
i &= 1,2,\ldots,m
\end{aligned} \tag{6.10}$$

where the variables are defined as

V_{Ti} generator terminal voltage

V_{REFi} exciter reference voltage

V_{PSSi} power system stabilizer voltage

a $a = \frac{T_{Ci}}{T_{Bi}}$, where T_{Bi} and T_{Ci} are time constants associated with the excitation system.

i the subscript of the individual generation system on the power system network

6.2.3 ETMSP Type-1 Power System Stabilizer (PSS)

The mathematical model representing the PSS system is the ETMSP Type 1 model, and is graphically represented in figure 6.3.. The dynamic representation for this model can be written as

$$\begin{aligned}
\dot{X}_{S1i} &= -\tfrac{1}{T_{5i}} X_{S1i} + \tfrac{1}{T_{5i}} K_{Si}\omega_{\Delta i} \\
\dot{X}_{S2i} &= -\tfrac{1}{T_{2i}} X_{S2i} + \tfrac{1}{T_{2i}}(1 - \tfrac{T_{1i}}{T_{2i}})[K_{Si}\omega_{\Delta i} - X_{S1i}] \\
\dot{X}_{S3i} &= -\tfrac{1}{T_{4i}} X_{S3i} + \tfrac{1}{T_{4i}}(1 - \tfrac{T_{3i}}{T_{4i}})[X_{S2i} + \tfrac{T_{1i}}{T_{2i}})(K_{Si}\omega_{\Delta i} - X_{S1i})] \\
V_{PSSi} &= -\tfrac{T_{1i}T_{3i}}{T_{3i}T_{4i}} X_{S1i} + \tfrac{T_{3i}}{T_{4i}} X_{S2i} + X_{S3i} + \tfrac{T_{1i}T_{3i}}{T_{3i}T_{4i}} K_{Si}\omega_{\Delta_i}.
\end{aligned} \tag{6.11}$$

where the variables are defined as

$\boldsymbol{T_{1i} - T_{5i}}$ are time constants associated with the power system stabilizer

$\boldsymbol{V_{PSSi}}$ is the output voltage of the PSS

$\boldsymbol{X_{S1i} - X_{S3i}}$ are the state variables in the PSS

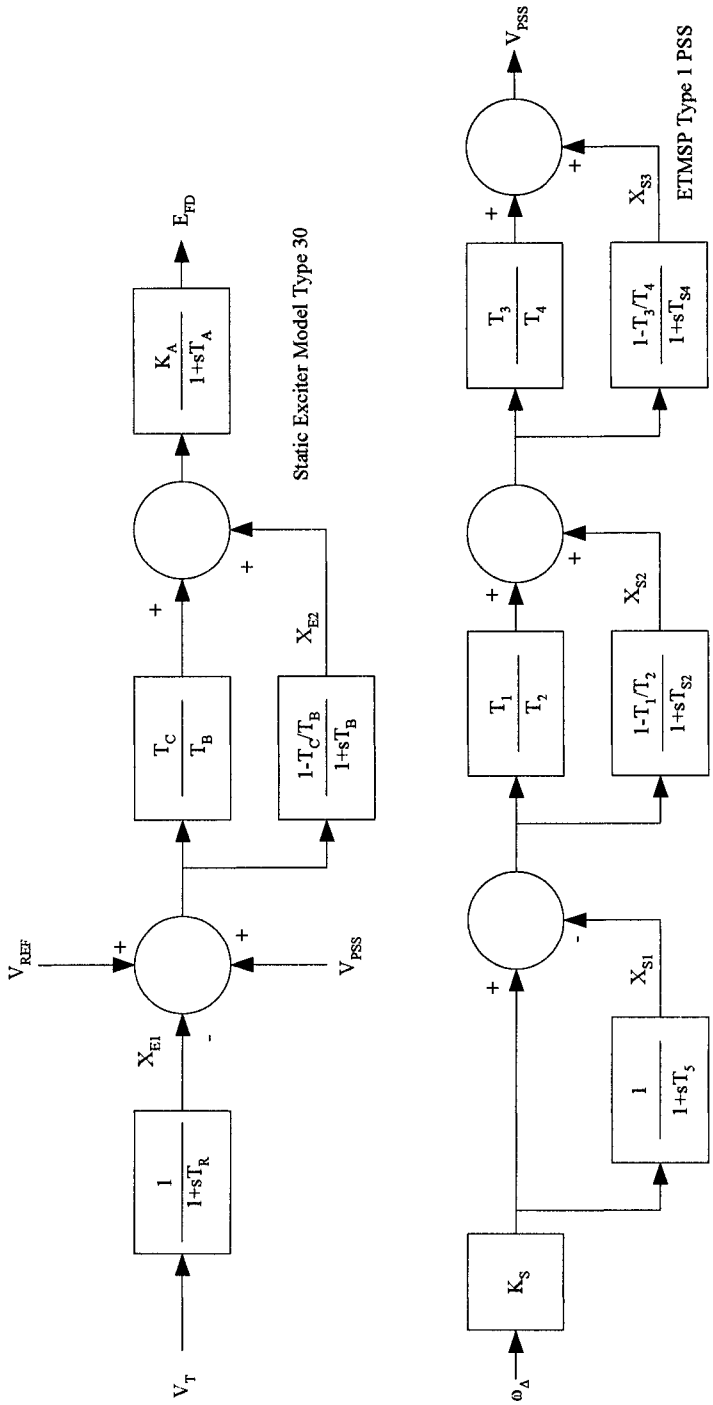
$\boldsymbol{K_{Si}}$ is a system gain factor associated with the PSS

$\boldsymbol{\omega_{\Delta_i}} = \frac{\omega_i}{\omega_r} - 1$, the fractional difference of the system i speed from the reference speed

The PSS is a controller on the system, hence a number of its parameters are *adjustable*. This implies that in a real system application, the settings of the controller would be determined by an "expert" either from the corporate headquarters or in the field. This is the reason a number of the PSS variables appear inspecific in the descriptions.

6.3 Interconnecting Network Equations

The mathematical model representing interconnecting network equations is represented with respect to a synchronously rotating reference frame (in X-Y coordinate system) as shown in figure 6.4.



(a) Type-30 Exciter model

(b) Type-1 PSS model

Figure 6.3: Block diagrams of an ETMSP Type-30 Exciter and Type-1 Power System Stabilizer (PSS)

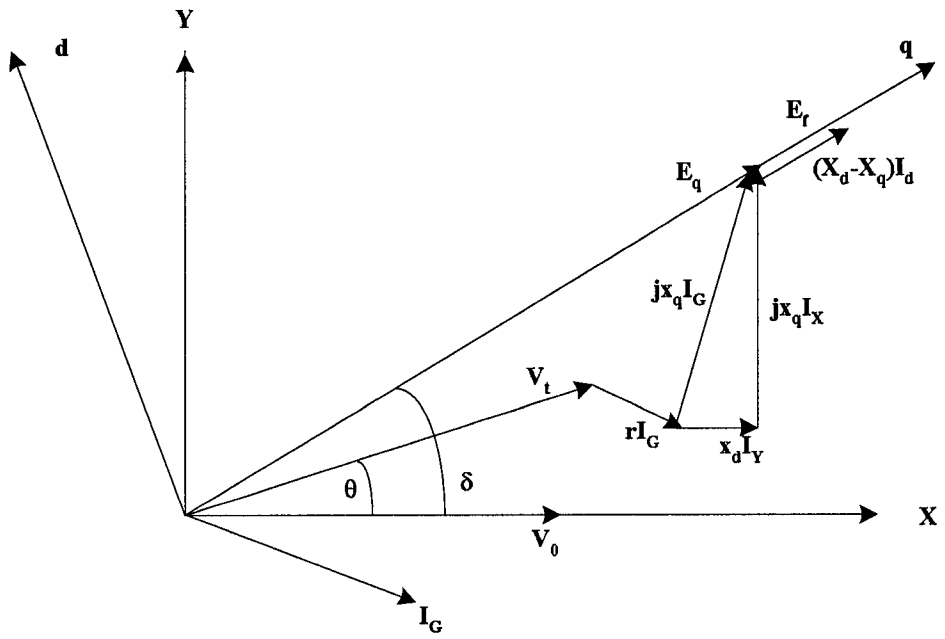


Figure 6.4: Vector diagram of interconnection between d-q and X-Y reference frames for two-axis synchronous generator model

To interface the mathematical model of the network to the dynamic system equations, a transformation from the synchronously rotating reference frame to the fixed reference frame is required. To do this, constant impedance loads are used. By eliminating all the load nodes, the network is reduced to contain only the generator internal buses. The bus admittance matrix Y_{bus} consists of diagonal elements $Y_{ii}\angle\theta_{ij} = G_{ii} + jB_{ii}$, $Y_{ij}\angle\theta_{ij} = G_{ij} + jB_{ij}$. By using the procedure discussed in chapter 9 of [1], the generator currents can be specified in the following form

$$
\begin{aligned}
I_{qi} &= \sum_{j=1}^{m} [F_{G+B}(\delta_{ij})E'_{qj} - F_{B-G}(\delta_{ij})E'_{dj}] + \sum_{k=m+1}^{n} F_{G+B}(\delta_{ik})E_k \\
I_{di} &= \sum_{j=1}^{m} [F_{B-G}(\delta_{ij})E'_{qj} - F_{G+B}(\delta_{ij})E'_{dj}] + \sum_{k=m+1}^{n} F_{B-G}(\delta_{ik})E_k \\
I_k &= \sum_{j=1}^{m} [F_{G+B}(\delta_{ij})E'_{qj} - F_{B-G}(\delta_{ij})E'_{dj}] + \sum_{l=m+1}^{n} F_{G+B}(\delta_{kl})E_l \\
& i = 1,2,\ldots,m \\
& k,l = m+1,m+2,\ldots,n
\end{aligned}
\tag{6.12}
$$

where

$$
\begin{aligned}
F_{G+B}(\delta_{ij}) &= G_{ij}\cos(\delta_{ij}) + B_{ij}\sin(\delta_{ij}) \\
F_{B-G}(\delta_{ij}) &= B_{ij}\cos(\delta_{ij}) - G_{ij}\sin(\delta_{ij}) \\
\delta_{ij} &= \delta_i - \delta_j.
\end{aligned}
\tag{6.13}
$$

6.4 Linearization of Generation Equations and Inclusion of Uncertainty

6.4.1 The Coupled Equations

The combination of the nonlinear differential equations for the generator and the algebraic equations for the transmission system can be coupled into a single group of equations

$$\dot{X} = f(X,Z,u) \tag{6.14}$$

$$0 = g(X,Z), \tag{6.15}$$

where X is the vector of state variables governed by the differential equations, and Z is the vector of network variables.

These equations are linearized to obtain the form

$$\dot{X} = AX + Bu + FZ \tag{6.16}$$

and

$$GZ = HX, \tag{6.17}$$

where X is the linearized vector of state variables, Z is the linearized vector of network variables, and A,F,G, and H are the appropriate coefficient matrices.

A block representation of the coupled system is shown in figure 6.5.

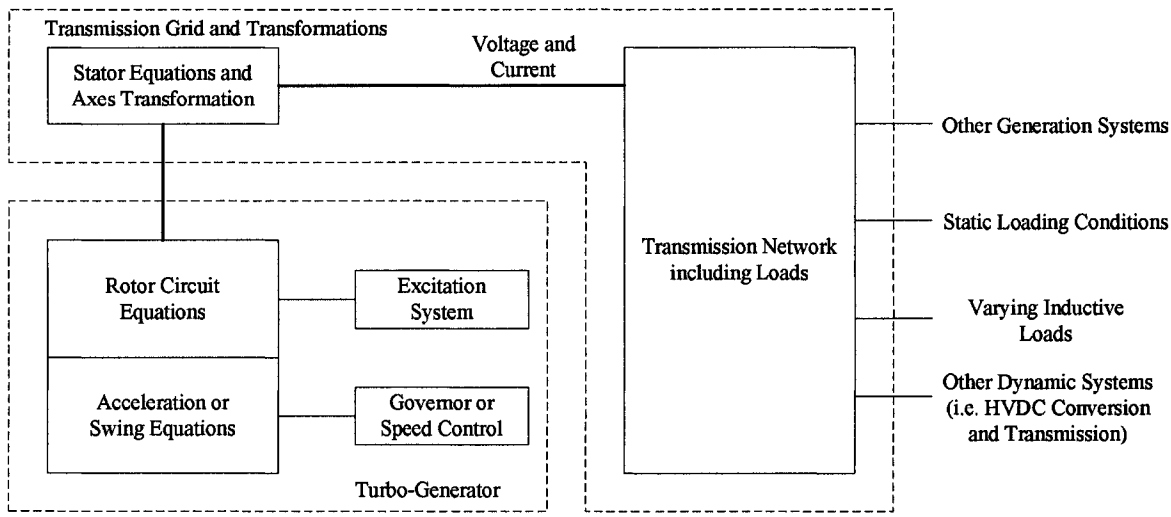


Figure 6.5: Block diagram of dynamic system coupled to an algebraic network

6.4.2 Overall System Equation

The combined dynamic equations governing the generators , exciters, and the PSS have the general form

$$\dot{X} = f(X, Z, u), \tag{6.18}$$

where

$$X^T = \left[X_{SM}^T, X_{PSS}^T, X_{ES}^T\right]^T \tag{6.19}$$

is the vector of state variables. The vector of state variables consists of

$$\begin{aligned} X_{SM} &= \left[E'_{q(1\ldots m)}, E'_{d(1\ldots m)}, \omega_{(1\ldots n)}, \delta_{(2\ldots n)1}\right]^T \\ X_{ES} &= \left[X_{E1(1\ldots m)}, X_{E2(1\ldots m)}, X_{E3(1\ldots m)},\right]^T \\ X_{PSS} &= \left[X_{S1(1\ldots m)}, X_{S2(1\ldots m)}, X_{S3(1\ldots m)}\right]^T. \end{aligned} \tag{6.20}$$

The vector of network variables is

$$Z = \left[I_{q(1\ldots m)}, I_{d(1\ldots m)}, I_{[(m+1)\ldots n]}, V_{T(1\ldots n)}\right]^T, \tag{6.21}$$

and the vector of control inputs is

$$u = \left[V_{REF(1\ldots m)},\right]. \tag{6.22}$$

The variable $\boldsymbol{f}$ is the vector of nonlinear functions

$$f_{1i} = \dot{E}'_{qi} \quad i = 1,\ldots,m$$
$$= \tfrac{1}{\tau'_{d0i}}[E_{FDi} - E'_{qi} + (x_{di} - x'_{di})I_{di}]$$

$$f_{2i} = \dot{E}'_{di} \quad i = 1,\ldots,m$$
$$= \tfrac{1}{\tau'_{q0i}}[-E'_{di} - (x_{qi} - x'_{qi})I_{qi}]$$

$$f_{3i} = \dot{\omega}_i \quad i = 1,\ldots,n$$
$$= \tfrac{1}{M_i}[P_{mi} - \tfrac{D_i}{\omega_S}(\omega_i - \omega_S) - (E'_{di}I_{di} + E'_{qi}I_{qi}) + (x'_{qi} - x'_{di})(I_{di}I_{qi})]$$

$$f_{4i} = \dot{\delta}_{i1} \quad i = 2,\ldots,n$$
$$= \omega_i - \omega_1$$

$$f_{5i} = \dot{E}_{FDi} \quad i = 1,\ldots,m$$
$$= \tfrac{K_{Ai}}{T_{Ai}}X_{E2i} - \tfrac{1}{T_{Ai}}E_{FDi} + \tfrac{aK_{Ai}}{T_{Ai}}(V_{REFi} + V_{PSSi} - X_{E1i})$$

$$f_{6i} = \dot{X}_{E1i} \quad i = 1,\ldots,m$$
$$= -\tfrac{1}{T_{Ri}}X_{E1i} + \tfrac{1}{T_{Ri}}V_{Ti}$$

$$f_{7i} = \dot{X}_{E2i} \quad i = 1,\ldots,m$$
$$= -\tfrac{1}{T_{Bi}}X_{E2i} + \tfrac{1-a}{T_{Bi}}(V_{REFi} + V_{PSSi} - X_{E1i})$$

$$f_{8i} = \dot{X}_{S1i} \quad i = 1,\ldots,m$$

$$= -\tfrac{1}{T_{5i}} X_{S1i} + \tfrac{1}{T_{5i}} K_{Si}\omega_{\Delta i}$$

$$f_{9i} = \dot{X}_{S2i} \quad i = 1,\ldots,m$$

$$= -\tfrac{1}{T_{2i}} X_{S2i} + \tfrac{1}{T_{2i}}(1 - \tfrac{T_{1i}}{T_{2i}})[K_{Si}\omega_{\Delta i} - X_{S1i}]$$

$$f_{10i} = \dot{X}_{S3i} \quad i = 1,\ldots,m$$

$$= -\tfrac{1}{T_{4i}} X_{S3i} + \tfrac{1}{T_{4i}}(1 - \tfrac{T_{3i}}{T_{4i}})[X_{S2i} + \tfrac{T_{1i}}{T_{2i}})(K_{Si}\omega_{\Delta i} - X_{S1i})]$$

Linearization of equation 6.18 leads to

$$\Delta\dot{X} = \frac{\partial f}{\partial X}\Delta X + \frac{\partial f}{\partial Z}\Delta Z + \frac{\partial f}{\partial u}\Delta u \tag{6.25}$$

We also have the network algebraic equation

$$0 = g(X,Z). \tag{6.26}$$

This equation is linearized and organized so that all the terms related to the algebraic variables are put at one side of the equation and those related to the state variables are on the other side of the equation.

We obtain the representation of the whole system in the state space form as

$$\Delta\dot{X} = A\Delta X + F\Delta Z + B\Delta u \tag{6.27}$$

$$G\Delta Z = H\Delta X, \tag{6.28}$$

where $A = \frac{\partial f}{\partial X}$, $F = \frac{\partial f}{\partial Z}$ and $B = \frac{\partial f}{\partial u}$. The procedure to obtain G, H and the detailed expressions for the elements of the coefficient matrices are given in Appendix A of [57].

6.5 Analysis and Examination of Skew μ for Frequency Augmented Systems

Per section 3.4.1 a SS system with parameter variations can be represented as an LFT with a perturbation set due to parameter uncertainty, and additionally augmented with frequency as a perturbation. The frequency range is specified as the range of interest for the system. This formulation has been stated as a two step process, and is briefly demonstrated here.

6.5.1 A State Space System represented as an LFT

Before augmenting a perturbed SS system with frequency, it is beneficial to demonstrate that a SS system can be represented as an LFT.

In figure 6.6, a standard SS drawing is given with the Laplace variable s isolated outside the envelope. By applying the methods of section 2.2.1, the frequency can be represented as a perturbation Δ. The SS system can be represented by an LFT matrix as shown in figure 6.7, where the matrix and the perturbation form the LFT as shown in figure 6.8.

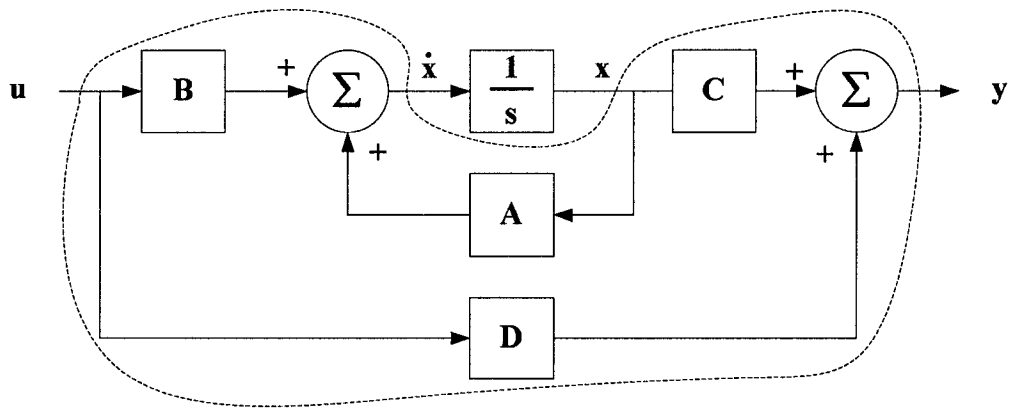


Figure 6.6: State Space system represented as an LFT

In this manner, a standard SS system can be formulated as an LFT where the SS input/output relationship of

$$\dot{x} = Ax + Bu \tag{6.29}$$

$$y = Cx + Du \tag{6.30}$$

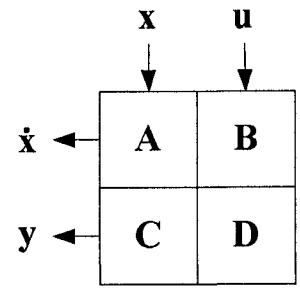


Figure 6.7: State Space system packed into a matrix form for an LFT

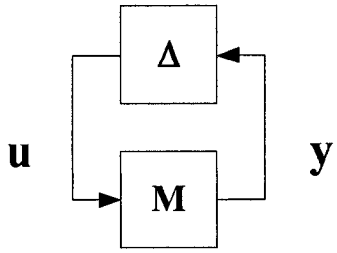


Figure 6.8: SS system forms the standard LFT configuration where Δ is the frequency

is maintained.

6.5.2 A State Space System with Model Uncertainties Represented as an LFT

Since a state space diagram can be transformed to an LFT, then it is relatively easy to add parameter uncertainty in forming an LFT. An example with additive uncertainty for each matrix element of a SS system is shown in figure 6.9. The methodology for rearranging figure 6.9 into an LFT was given in section 2.2.1.

Note that figure 6.9 also includes a perturbation block closing output to input. In this case the closure of output to input with a perturbation block indicates that performance of the system is a perturbation parameter (see [46], among others). The goal of closing the output to the input is to obtain the measurement of stability with respect to system performance as well as system parameter variations.

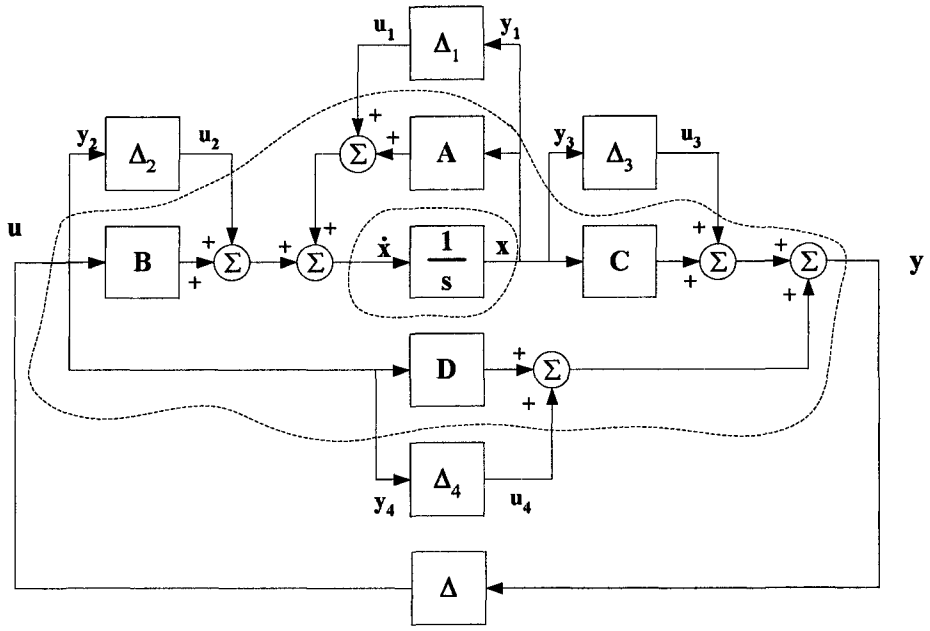


Figure 6.9: State Space system with model uncertainties ready for forming an LFT

A SS system (without other perturbations) can be rearranged into an LFT format where frequency is the fixed range perturbation, and the Δ from output to input is the varying range perturbation. This format for skew μ asks the question "For this fixed frequency range, what level of performance is available?". The random SS test systems of section 6.6 were formed into LFT's in this fashion.

Figure 3.6 in section 3.4.3 provides a more detailed diagram of the example laid out in figure 6.10. The development of equations associated with the figure 3.6 are also found in section 3.4.3. The resulting equations from that development are repeated in equation set 6.31.

$$\begin{aligned} G_{11} &= -Aj\omega_2(I+Aj\omega_0)^{-1}\omega_1 \\ G_{12} &= \omega_2(I+Aj\omega_0)^{-1}B \\ G_{21} &= -Cj\omega_1(I+Aj\omega_0)^{-1} \\ G_{22} &= -Cj\omega_0(I+Aj\omega_0)^{-1}B+D. \end{aligned} \tag{6.31}$$

The matrices A,B,C and D are the SS matrices of the system. The parameter ω_0 is the midpoint of the frequency span of interest, and can be considered an offset. A second

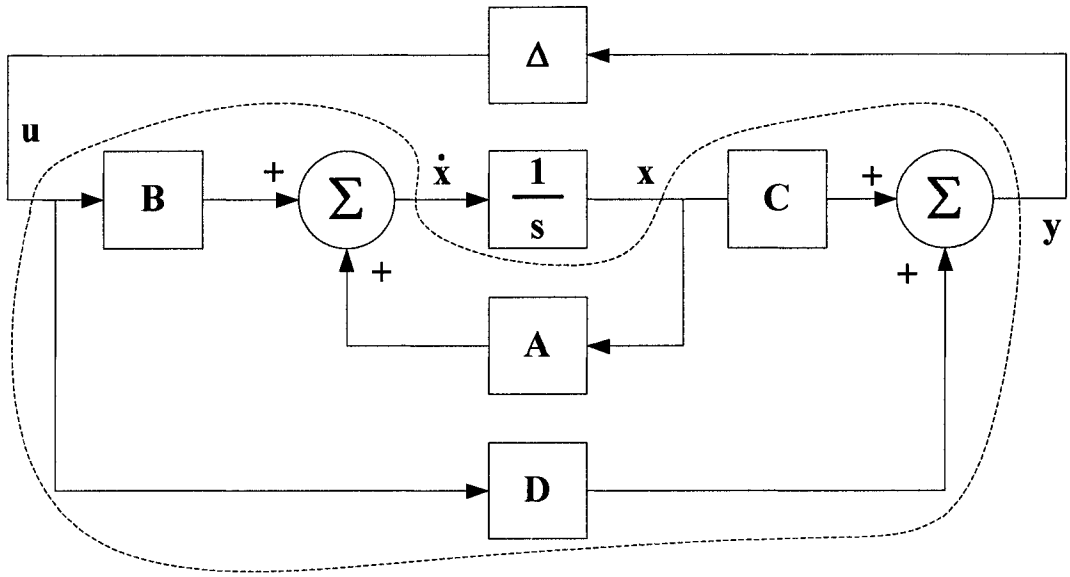


Figure 6.10: State Space system with frequency as fixed range perturbation and input-output performance as varying range perturbation, ready to form an LFT

parameter is required in addition to the offset to span the frequency range of interest. This parameter is chosen based on a real parameter perturbation range of $\|\Delta\| \leq 1$. The resulting parameter that spans the frequency range is split into ω_1 and ω_2, where ω_1 and ω_2 are the square root of the value required to span the frequency range of interest. The parameters ω_1 and ω_2 are chosen as square root values because they enter the matrix equations in a balanced fashion through G_{12} and G_{21}. The advantage of embedding values in matrices in a balanced fashion was demonstrated in section 5.3.1.

A few other comments need to be made about the equation set 6.31. Note that as ω_1 and ω_2 shrink in size (decreasing frequency span), the values in G_{11}, G_{12} and G_{21} shrink in magnitude as well. As the frequency span approaches zero, ω_1 and ω_2 approach zero, the values of G_{11}, G_{12} and G_{21} approach zero, and the calculation of the skew μ bound for the frequency interval essentially becomes a calculation at the frequency point ω_0. This is approximately equivalent to the calculation of the bounds on μ for a SS system *(without frequency augmentation)* at a given frequency point. From this explanation, ones see that if frequency spans become small enough, the results of the skew μ bounds calculation will

approach the results of the gridded frequency analysis.

6.6 Examination of Results for Random Test Systems

Low order randomly generated stable test systems were created using Matlab® support functions. These systems were examined for performance in three different ways.

First, an assessment of the μ upper bound was performed using frequency gridding. This is the defacto standard technique, and gives a reasonable expectation of the system performance, but does not guarantee the upper bound for all frequencies since gridding inherently misses some frequencies.

Next, the systems were frequency augmented to the LFT form compatible to finding skew μ bounds for the system while maintaining a fixed frequency range.

The second and third methods for examining performance were to find the skew μ upper bound in two separate ways. One method was to iteratively apply μ tools to calculate the skew μ upper bound. The other method was to use the skew μ tools created as part of this project to calculate the skew μ upper bound.

In the first example, figure 6.11 is the μ upper bound for a random 2 input, 2 output, 7 state system, formed from a gridded frequency analysis. Ideally, one expects, or would like, the skew μ upper bound result for the frequency augmented systems to approach this graph.

Figure 6.12 shows the result from finding the skew μ upper bound for the test system using an iterative application of μ tools to find the skew μ bound. Although this result shows some of the features of figure 6.11, it provides little detail concerning points where there is rapid variation in the μ upper bound for the gridded analysis shown in figure 6.11.

Figure 6.13 shows the result of finding the skew μ upper bound of the test system using the skew μ upper bound calculation software created as part of this project.

The calculation of the skew μ upper bound over the entire range of interest would only give the maximum value obtained in that interval. In order to provide more information, the frequency interval is broken up into smaller contiguous spans. This methodology is similar

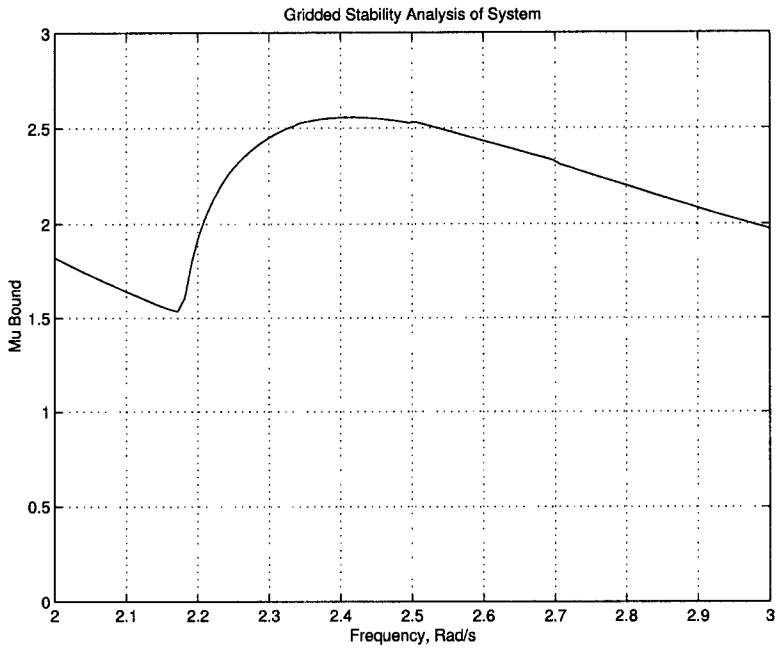


Figure 6.11: Gridded μ upper bound of a test system matrix

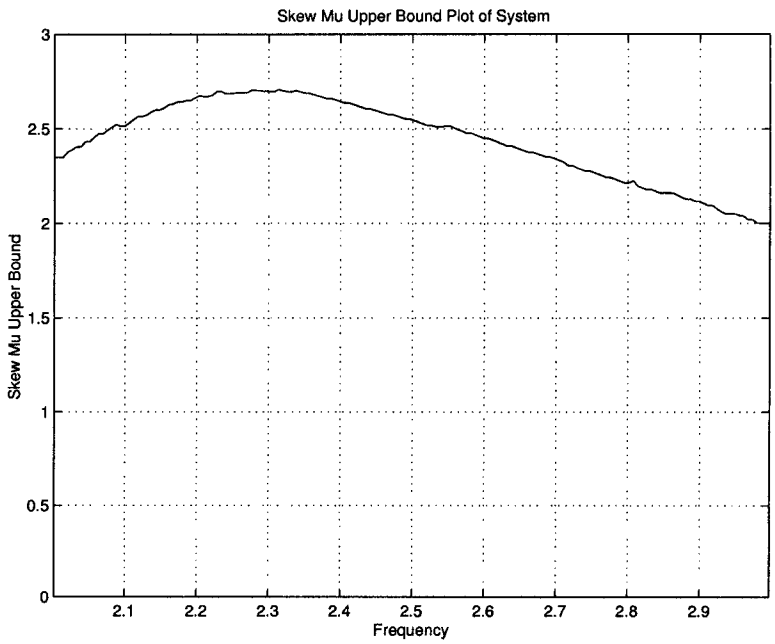


Figure 6.12: Skew μ upper bound for a system found with iterative calculations of μ

to that of chapter 7, however, with skew μ, the spans are guaranteed to be contiguous. Contiguous spans cannot be easily provided with the use of μ upper bound tools. For this test problem, note that the skew μ upper bound calculation reveals much more of the detail around the points where the value of μ appears to be rapidly changing in the frequency gridded problem (i.e. figure 6.11).

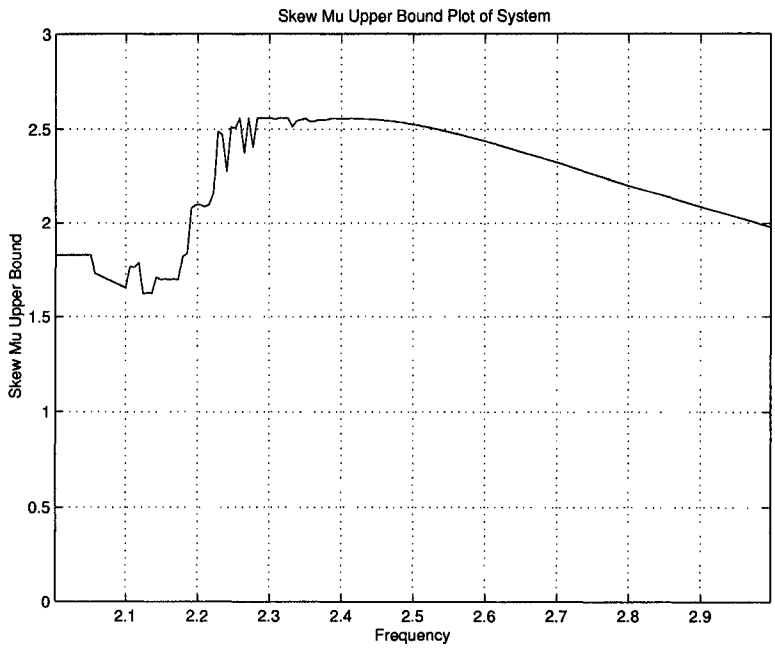


Figure 6.13: Skew μ upper bound for a system with frequency spans of 0.0061 radians, calculated with skew μ software tool

The skew μ upper bound analysis can be pushed closer to the frequency gridded analysis by shrinking the individual frequency spans. In the case of figure 6.14, the frequency spans are 0.0015 radians, where as, in figure 6.13, the spans were 0.0061 radians. The improved accuracy comes with a price. The system in figure 6.13 took 2463.3 seconds to calculate, while the same system using smaller frequency spans in figure 6.14 took 9147.5 seconds to calculate on the same system, a Generic Duron 950 MHz PC.

As is discussed in section 7.2 a small amount of complex uncertainty was added to the system. The system was tested with the skew μ software tool using frequency spans of 0.0061 radians. The result is shown in figure 6.15. The results in figure 6.15 are slightly

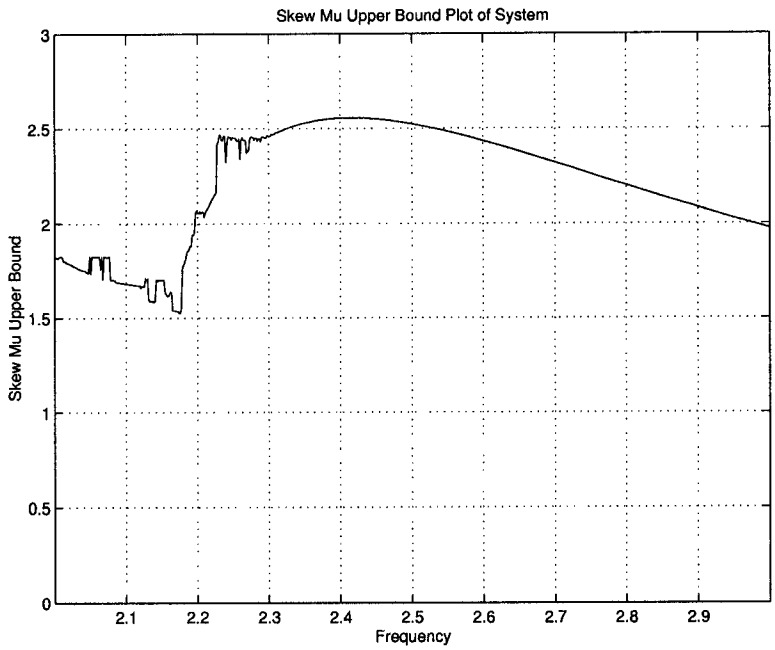


Figure 6.14: Skew μ upper bound for a system with frequency spans of 0.0015 radians, calculated with skew μ software tool

better than the results in figure 6.13, both of which examine the system using contiguous frequency spans of 0.0061 radians. Unfortunately, the addition of complex uncertainty increased the computational complexity such that it required 14,313 seconds to compute the upper bound on a Generic Duron 950 MHz PC using the skew μ software tool. The results of figure 6.15 are on par with those of figure 6.14, however, figure 6.14 took a third less time to calculate. This should not be taken to imply addition of complex uncertainty is not useful, but rather for this problem, it added no particular advantage.

A second example is shown in figures 6.16, 6.17 and 6.18. This example was a stable SS system generated with Matlab® tools to have 2 inputs, 2 outputs and 5 states.

The frequency gridded μ upper bound for this system is less varied than the previous example. Its apparent from figure 6.17 that the skew μ upper bound iteration technique had problems obtaining results similar to the frequency gridding calculation. However, the direct calculation of the skew μ upper bound using the software tool was able to produce

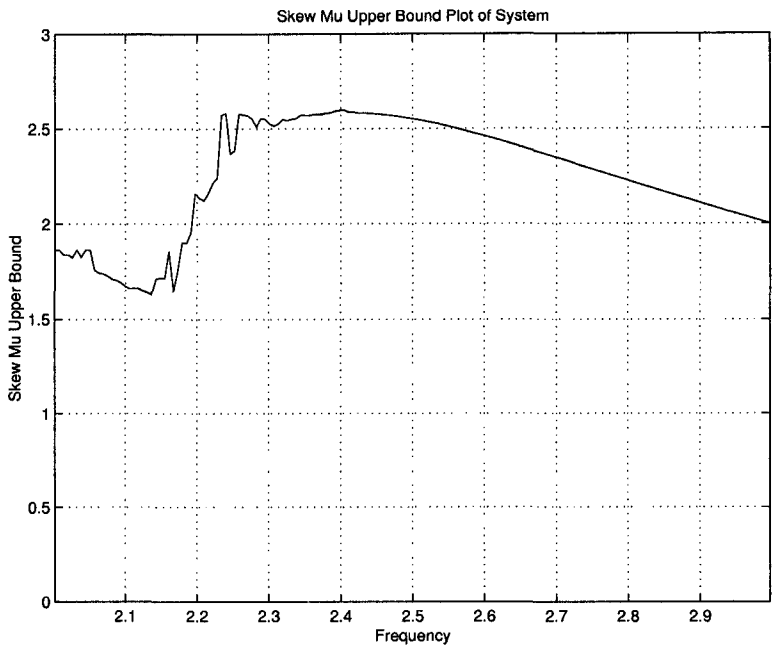


Figure 6.15: Skew μ upper bound for a system with frequency spans of 0.0061 radians and 1% complex uncertainty added, calculated with skew μ software tool

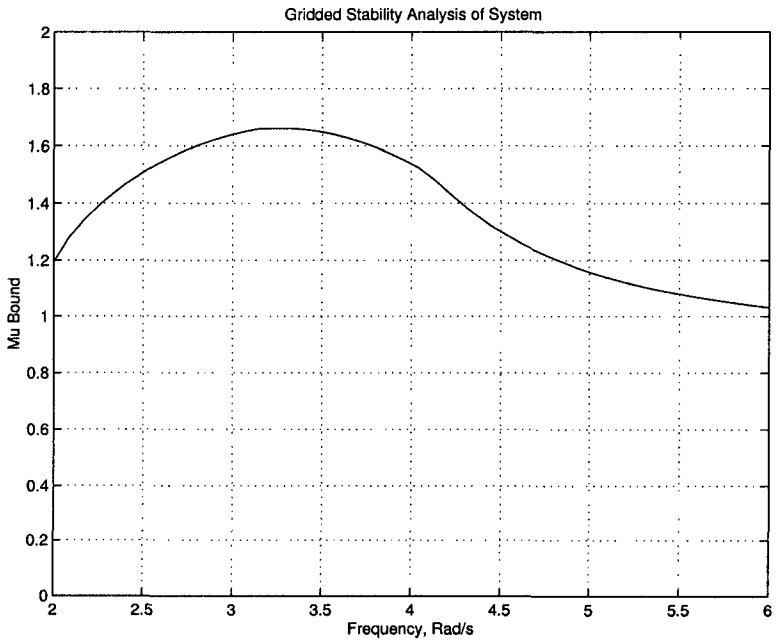


Figure 6.16: μ upper bound for a gridded analysis of the second example

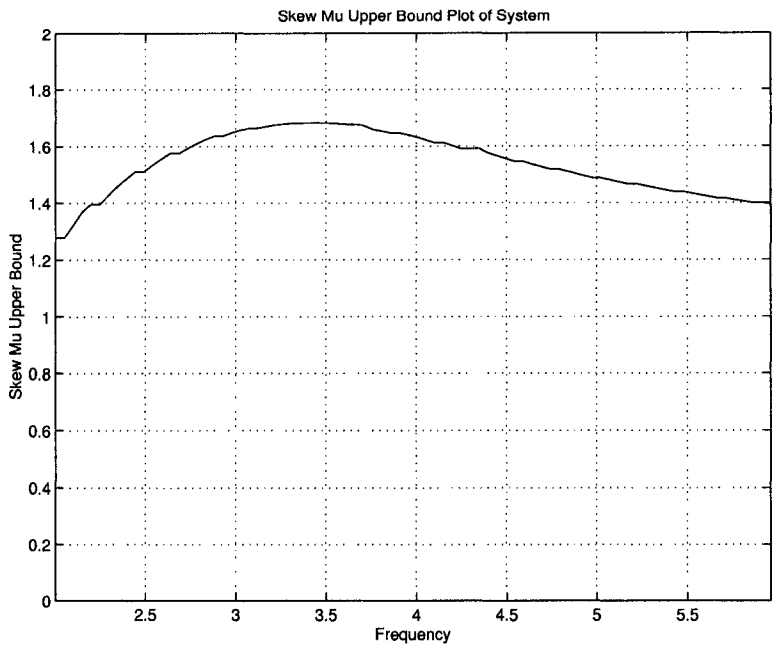


Figure 6.17: Skew μ upper bound for of the second example via the iterative technique

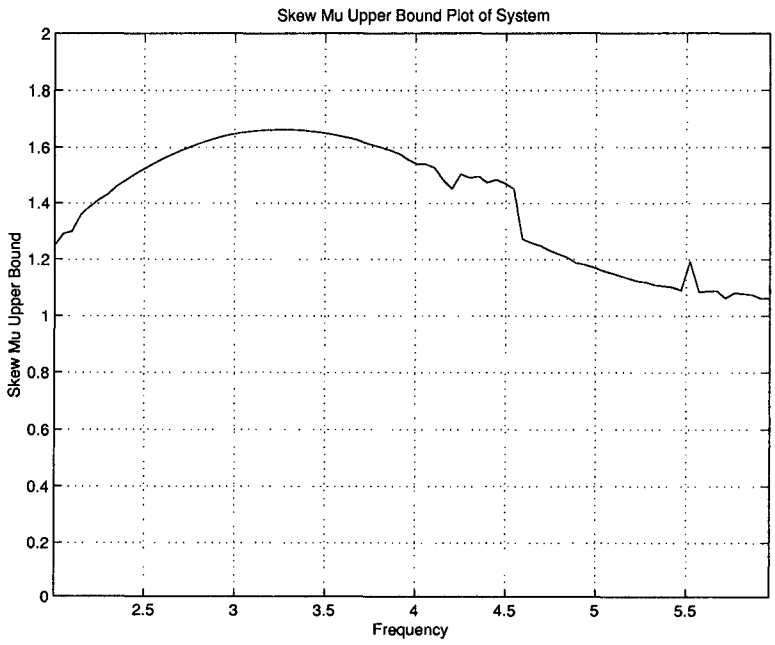


Figure 6.18: Skew μ upper bound for of the second example via direct calculation using the software tool

results similar to those of the gridded analysis. The size of the frequency spans for the iterative and direct skew μ upper bound calculations were .0488 radians.

A few comments need to be made about the small departures from the otherwise smooth line in figure 6.18. These bumps need to be understood in the context of the equation set 6.31. As ω_1 and ω_2 approach 0 with smaller and smaller *frequency spans*, the skew μ upper bound calculation should approach the calculation of the μ upper bound at a single *frequency point*. Note that the μ upper bound at the single frequency point ω_0 is equivalent to the μ upper bound of the equation for G_{22}. If the values contained in the A, B, or C matrices of G_{12} and G_{21} are significantly large, then the calculation of the skew μ upper bound may be significantly affected, as seen in figure 6.18. Eventually, if ω_1 and ω_2 are continually shrunk then the skew μ upper bound values in the bumpy sections of figure 6.18 will have the tendency to approach the values in the gridded frequency μ upper bound calculation.

One may have a tendency to argue that since the frequency augmented formulation approaches the gridded frequency analysis, then there's no need to do a skew μ upper bound calculation. Remember, there's an infinite number of frequency points and not every one can be checked. Frequency gridding shows where the system typically has no problems, and which areas need to be investigated further. The skew μ software tools may then be used for a more detailed analysis of the frequency spans with questionable results.

6.7 Test Results for 4 Machine System

The frequency augmented four machine test system was tested with frequency as the fixed range perturbation, and performance as the varying range perturbation. An iterative μ bound calculation was used to acquire the skew μ upper bound, as well as the skew μ software tools developed in this project. The results of the tests displayed in figures 6.19 and 6.20 require some explanation. As it appears, the results of finding the skew μ bound by iterative techniques seems to far out perform the skew μ software tool, however, this

is only part of the story. The skew μ software tool was not able to take advantage of the LMI upper bound techniques developed in section 5.6. The formulation of the skew μ upper bound LMI equation generated several thousand variables, which was more than the LMI Toolbox®could handle. Hence, the results from the skew μ software tool are only the results of skew μ Osborne balancing techniques.

As previously mentioned, as frequency spans become smaller, the results of the skew μ upper bound calculation should approach those of the μ upper bound calculation at the frequency point. When the internal states of the μ tools toolbox® software were checked against the internal states of the skew μ software tool at a small frequency span, it was found that the Osborne upper bound of the μ tool and the Osborne upper bound of the skew μ tool nearly matched. Additionally, as the internal states of the μ tool were examined more closely, it was found that the significant improvements in the μ upper bound calculations were made in the μ tools gradient descent algorithms and LMI calculations (see [54] for more details). These algorithms were intended to be replaced in the skew μ software tool upper bound calculations by the LMI algorithm of section 5.6 and the GEVP solver from the LMI Toolbox®. In short, the μ tool performed better because the skew μ software tool was not able to use its equivalent LMI tool. In light of this situation, there are two comments that can be made. First, obviously the skew μ theory and software needs to be extended so that algorithms similar to those in the μ tool can be brought into existence. Secondly, since it has been previously demonstrated that the skew μ tool outperforms iterative applications of the μ tool for finding the skew μ upper bound, then one would expect the skew μ tool to perform at least as well as the iterative μ tool technique on the 4 machine system test.

To make an intelligent guess on the computational time improvement the skew μ tool will have over the iterative μ tool method, consider the following: During calculation of the iterative skew μ upper bound, it took the μ tool .2383 seconds per iteration to calculate an upper bound. It is not unreasonable to suggest the skew μ tool will be able to match this time. It will also have the advantage of a fixed range frequency perturbation, meaning it

will not need to iterate. This characteristic should reduce overall computation time to a small fraction of the computation time required for the application of the iterative μ tool method.

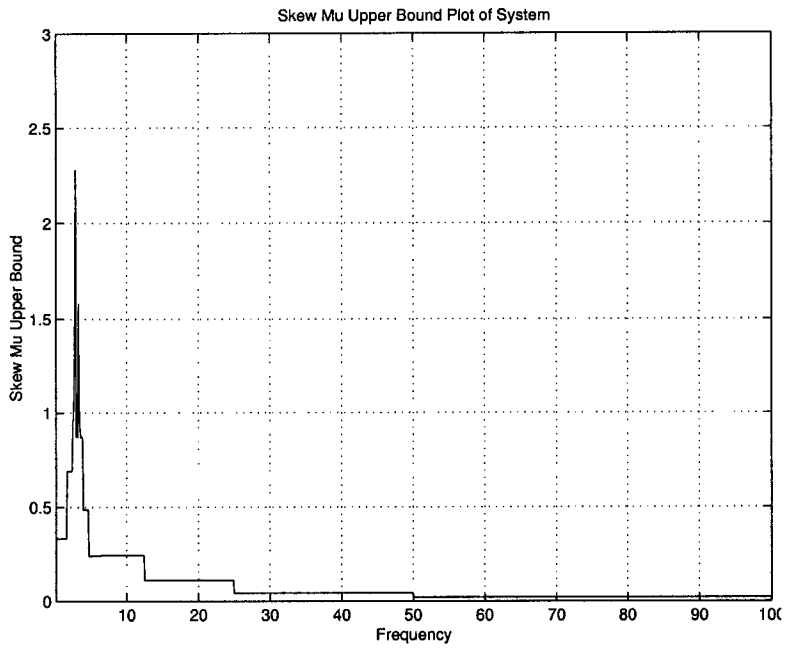


Figure 6.19: Skew μ upper bound for the 4 machine system via iterative calculation of μ tools

6.7.1 Comparison to Known Results for the 4 Machine System

In section 2.3, among others, the potential for system instability when μ/skew μ bounds were greater than 1 was pointed out. It is clear from figures 6.19 and 6.20 that the skew μ upper bound exceeds 1 significantly. In terms of the 4 machine system, to provide a stability guarantee requires that the varying range perturbations be reduced by a factor of at least $\frac{1}{\mu_s}$, where $\frac{1}{\mu_s}$ is the upper bound on skew μ.

In the case of the iterative skew μ calculation in figure 6.19, the peak value of the skew μ upper bound is 2.34, thus the 4 machine system must have its varying range perturbations reduced by the factor $\frac{1}{2.34}$ to guarantee stability. The underlying problem here is distance between the actual value of skew μ and its upper bound. If the distance is large, then the

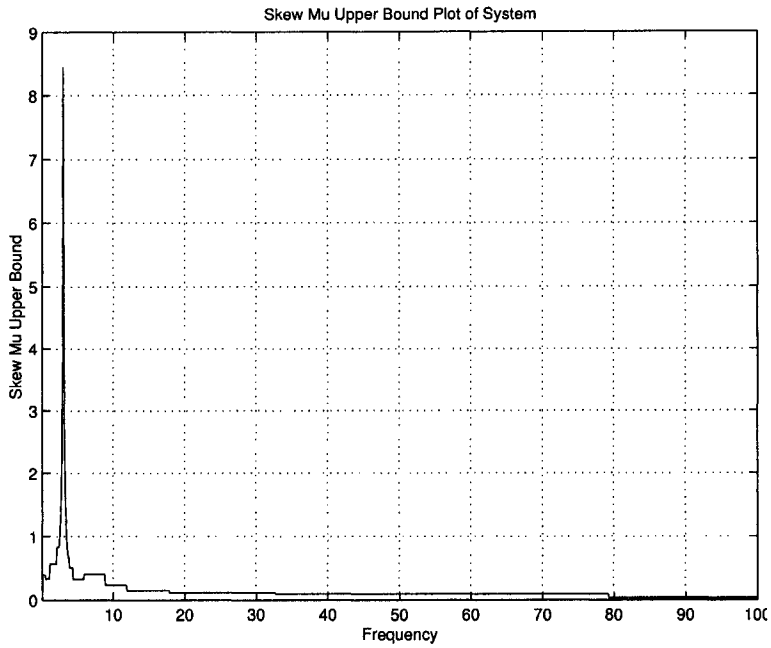


Figure 6.20: Skew μ upper bound for the 4 machine system via direct calculation

upper bound restriction on the size of perturbations is overly conservative. Any recommendations made for, or controls designed around, this case will also be overly conservative. The gridded frequency analysis of the 4 machine system, displayed in figure 7.6, has a peak upper bound μ value of approximately 1.49 at $\omega = 2.73$ rad/s. Even though frequency augmentation is a real perturbation, and thus more difficult to calculate bounds for, one would expect the skew μ upper bound to be closer to the gridded frequency value than the skew μ upper bound indicates[40].

The results of the skew μ techniques applied to the 4 machine system can be compared with those of section 7.3 in this document. Other μ analysis of the 4 machine system can be found in [57] and [55]. However, these thesis apply their own novel analysis techniques to specific conditions surrounding the 4 machine system, thus side-by-side comparisons are not possible.

It is possible to note the gridded analysis and B&B techniques of [57] indicate the frequency at which the peak μ bound result occurs is approximately 2.73 rad/sec. The results of [55] indicates a peak somewhere in the frequency range of 2.73 to 2.92 rad/sec. The skew μ

tool indicates the peak is at a noticeably higher frequency, 3.02 rad/sec, with a peak skew μ value of 8.56. No precise cause has been found for this frequency difference. Currently it is presumed to be associated with the combination of frequency augmentation and skew μ bound calculation. The work in section 7.3, [57] and [55] all use frequency augmentation, but do not use skew μ, either by iteration or direct calculation of the bound.

As previously mentioned, by taking smaller minimum frequency spans of 0.0015 radians, the skew μ tool was able to minimize the skew μ upper bound to a value of 5.25 for the 4 machine system. The results in figure 6.20 where achieved with minimum frequency spans of 0.0977 radians. These are distant from the bounds achieved by skew μ iteration (2.34) and the gridded frequency analysis (1.49). The predominant factor for the difference is believed to be the inability to calculate the skew μ upper bound using the LMI, as stated earlier. The need for future work is concerning this problem is discussed in section 9.2.

6.8 Test Results for the 50 Machine System

As with the 4 machine system, the 50 machine system also has the disadvantage of not being able to use the LMI solver section of the skew μ upper bound software created as part of this project. Hence, the software created in this project only finds the skew μ upper bound for the 50 machine system using the skew μ Osborne upper bound technique.

In a comparison with the results achieved by finding the skew μ upper bound via iterative application of μ tools, the iterative technique outperforms the skew μ upper bound found with the Osborne upper bound technique. Likewise, one can claim that extensions of the μ tools gradient descent and LMI algorithms from the μ tools to the skew μ tools will provide performance on par with that of the current Matlab® μ-tools Toolbox. In this case, one iteration of the μ tools takes approximately 58.16 seconds and recurs a significant number of times to arrive at a skew μ upper bound for a specific frequency range. To arrive at the same result without iterating should take approximately 58.16 seconds using the skew μ software tools with proposed extensions. In the case of figure 6.21 it took 67366 seconds

(18.71 hours) to calculate the results using iterative applications of μ tools. This would be reduced to approximately 5350 seconds (1.48 hours) using the skew μ software tool with proposed extensions to achieve similar or better results. Clearly there is sufficient advantage in reduced computation time to motivate the effort for the proposed gradient descent and LMI extensions from the μ tools to the skew μ tools.

Note, similar calculations for the 4 machine test system yield a reduction from 89.97 seconds for the iterative μ calculation to a potential 10.94 seconds for the skew μ software tools. However, the time savings from the 4 machine problem doesn't generally motivate the effort needed to extend the μ tools to skew μ in the minds of most people. It is the 50 machine test system where the time advantage becomes truly apparent.

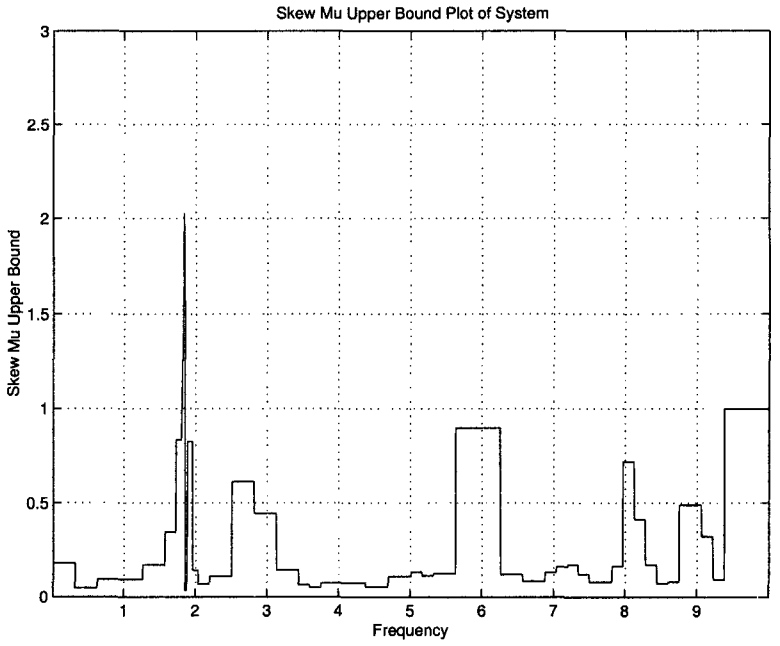


Figure 6.21: Skew μ upper bound for the 50 machine system via iterative calculation of μ tools

6.8.1 Comparison to Known Results for the 50 Machine System

Similarly to the 4 machine system, it is clear from figures 6.21 and 6.22 that the skew μ upper bound for the 50 machine system exceeds 1 significantly.

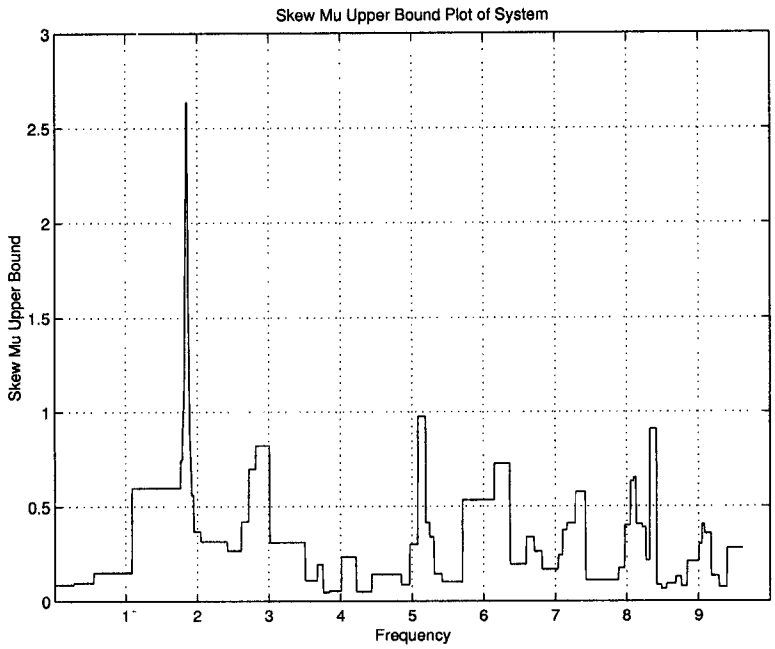


Figure 6.22: Skew μ upper bound for the 50 machine system via direct calculation

In the case of iterative skew μ calculation in figure 6.21, the peak value of the skew μ upper bound is 2.05, thus the 4 machine system must have its varying range perturbations reduced by the factor $\frac{1}{2.05}$ to guarantee stability. As before, if the distance between the actual value of skew μ and its upper bound is large, then the upper bound restriction on the size of perturbations is overly conservative. A gridded frequency analysis of the 50 machine system typically displays has a peak upper bound μ value of approximately .145 at $\omega = 5.82$ rad/s[55]. This would indicate the inaccuracy of the upper bound for any of the μ methods. However, [55] also provides eigen analysis to show that there is a very narrow peak at $\omega = 1.79$ rad/s with a peak μ/skew μ bound greater than 1. Thus, the application of μ/skew μ tools has proved highly beneficial. For the 50 machine system, the value of skew μ at this point is still ambiguous. [55] indicates it to be greater than 1, and the analysis of this document shows it to be less than 2.05. Further work is needed to hone the value to a more desirable level of engineering accuracy.

Prior to development and application of the skew μ software tools, other methods were investigated for μ analysis of power systems . Although those methods did not become the

main thrust of the project, they did provide some useful results and are potential areas for extension of the main results of this project. These topics are covered in the remaining two technical chapters of this document, which follow immediately.

Chapter 7

Branch and Bound Methods Applied to the Power Problem

7.1 Re-Statement of the Problem Characteristics

The size and complexity of the power problem motivates the need for improvements in computational speed. To provide an improvement in computation time for the power problem using existing tools, an algorithm using B&B (Branch and Bound) and some of the characteristics of μ was developed. B&B search improvements are added to the basic algorithm, and results from improved methods are compared to the basic B&B algorithm. The techniques demonstrated in this section can easily be extended beyond the power problem to other large scale problems. In many cases it is also fitting to use B&B techniques in combination with skew μ computational tools to improve the accuracy and speed of the skew μ bounds. This need is driven by a problem dependent characteristic of the skew μ upper bound to be separated by a large band from skew μ . Through the use of B&B techniques along with the skew μ upper bound , a more accurate bound on skew μ can be accomplished while maintaining improved computation speed.

7.1.1 Characteristics of Frequency as a Perturbation

When frequency is included as a perturbation (section 3.4) and the computed bounds indicate $\mu < 1$, this implies the range of the perturbations (including frequency) can be expanded before encountering potential instability. Specifically for frequency, the bounds on μ are stable for at least the frequency range specified, and no further search of that frequency region is necessary to guarantee stability. If the bounds indicate the possibility of $\mu > 1$, then the region must be partitioned into sections for a closer search until the region has been sufficiently mapped so as to provide a reasonable amount of information concerning the bound on μ over that region. This basic description can be constructed as a B&B algorithm for analyzing the stability properties of a system.

As an initial early first step in this project this method was applied. Part of this methodology had been previously done in a project at Iowa State University. This project, being an extension of that original project, started where the previous work stopped. This initial work is documented here because some insights were learned on B&B algorithms, and it provides a reasonable measurement for assessing the current state of computational complexities faced in analyzing the stability of large power systems.

One should note that skew μ is not subject to the restrictions of the previous two paragraphs because frequency is is a fixed range perturbation, and is thus not scaled by skew μ. This should be kept in mind while reading this chapter of the document.

A second issue concerning frequency and B&B searches is choice of parameter. Frequency is a complex variable, however as seen in section 3.4, the complex element (j) can be wrapped into the system matrix, leaving the value of the frequency as a real parameter perturbation. The effect of changing from a complex perturbation arrangement to a real perturbation arrangement has a disadvantage because real perturbations problems are typically more difficult for achieving tight bounds. However, it is much easier to do a B&B search on a real parameter since it has one degree of freedom (real axis), where as a complex perturbation/parameter has 2 degrees of freedom (real and complex axis), as shown in

section 3.4.4. The ability to do a well designed B&B search out weighs the disadvantage of the real parameter problem, hence the real parameter B&B is the preferred choice when using frequency as a perturbation.

7.1.2 A Branch and Bound Algorithm for a Frequency Search

The algorithm described below is a common B&B methodology and can be applied to many different types of problems . This particular case comes from *the initial early first step in this project* and provides detail concerning a basic implementation of robust analysis to power systems. It is not the skew μ methodology, but rather the first step of the project. Indeed, it actually motivated the search for the development of the skew μ methodology. It is provided here so the reader may have a benchmark to compare the results of the algorithm improvements to.

1. Construct μ problem for system and augment with frequency as described in section 3.4.

2. Calculate bounds on μ for the augmented problem with frequency set for a specified range.

   - If bounds indicate $\mu < 1$ then the frequency region passes stability requirements, no need to do further testing of that region. Move to next contiguous region.
   - If bounds indicate the possibility of $\mu > 1$ then the frequency region does not pass stability requirements, and the region should be subdivided into smaller frequency regions for further testing.
   - If bounds indicate the possibility of $\mu > 1$ then the region does not pass stability requirements. However, the region is the smallest frequency subdivision allowed, so the bounds on μ are stored for analysis, and the next contiguous region as identified and stored by the algorithm is analyzed.

A flow chart for this methodology is shown in figure 7.1.

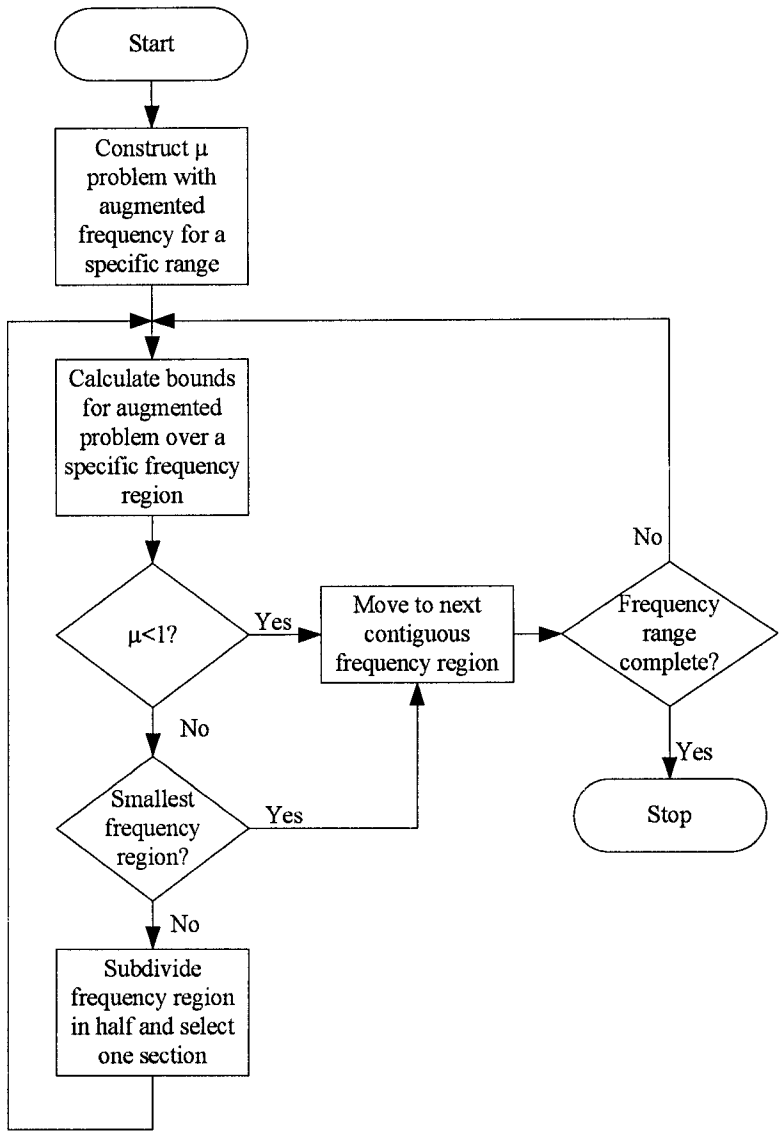


Figure 7.1: Branch and bound algorithm diagram for dissection of frequency intervals

7.1.3 Typical Branch and Bound Philosophy

The typical B&B procedure for testing frequency regions where performance requirements have not been met is subdivision into smaller regions and retesting each region separately. A region where bounds indicate the possibility of $\mu > 1$ is subdivided in half. The first half is tested. If the bounds indicate $\mu < 1$ then the second half is tested. If the bounds indicate possibility of $\mu > 1$ for the first half, then it is subdivided into two halves, and the first of these two halves is tested. This procedure is repeated until eventually either both half sections at a particular branch pass, or the size of the subsections has diminished to a pre-defined minimal section size. Once the minimal section size has been reached, the results from the minimal section size are recorded and its corresponding other half section is tested and recorded. From the minimal branch, the algorithm ascends to the next highest level with untested sections, and proceeds to test and branch as necessary. Data is stored from these sections for examination and plotting at a later time.

In a practical example, any region with bounds $\mu > 1$ is searched on a grid of the minimal section size. If the problem contains a single area where μ peaks at greater than one, this methodology is suitable. If the problem contains large ranges where $\mu > 1$, the criteria for branching and subdividing may need to be increased to capture only the peaks while maintaining reasonable speed of computation.

7.1.4 An Illustrative Example of Branch and Bound

The following explanatory example relates to figure 7.2, and provides a clear example of how the B&B procedure works . Presume that region N contains a small spike where the bounds indicate $\mu > 1$, and all other regions have the property of a bound $\mu < 1$.

1. Region A with a frequency range of ω_{min} to ω_{max} is tested and found to have bounds where $\mu > 1$. A is subdivided into regions B and C.

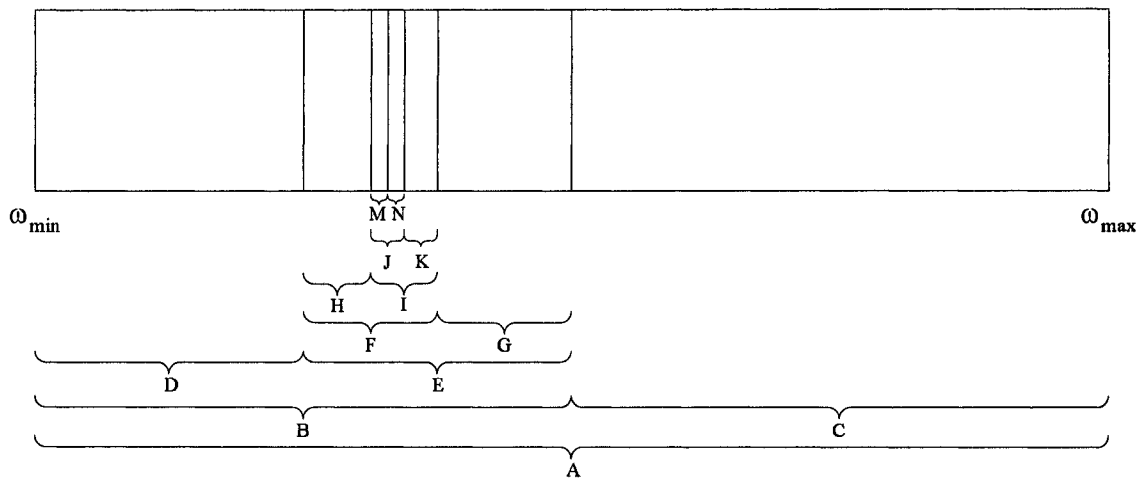


Figure 7.2: Branch and bound diagram for dissection of frequency intervals from illustrative example

2. Region B is tested and found that the bounds indicate $\mu > 1$. B is subdivided into regions D and E.

3. Region D is tested and found that the bounds indicate $\mu < 1$. Testing proceeds to region E.

4. Region E is tested and found that the bounds indicate $\mu > 1$. E is subdivided into regions F and G.

5. Region F is tested and found that the bounds indicate $\mu > 1$. F is subdivided into regions H and I.

6. Region H is tested and found that the bounds indicate $\mu < 1$. Testing proceeds to region I.

7. Region I is tested and found that the bounds indicate $\mu > 1$. I is subdivided into regions J and K.

8. Region J is tested and found that the bounds indicate $\mu > 1$. J is subdivided into regions M and N.

9. Region M is tested and found that the bounds indicate $\mu < 1$. Testing proceeds to region N.

10. Region N is tested and found that the bounds indicate $\mu > 1$. Since N is of minimal size, the value is stored and testing proceeds to region K, the next smallest untested region.

11. Region K is tested. Region G is tested. Region C is tested. All are found to have bounds that indicate $\mu < 1$. Procedure ends.

Note that if the peak for bounds $\mu > 1$ were wider than the single section N, then the algorithm would subdivide region K (for instance) and work down to the minimal division size, and then work back up (and down) the diagram as necessary testing the untested sections and recording the values of μ bounds. For any section where the bounds indicate $\mu < 1$, the value of the bounds is recorded and no further action takes for that section. Obviously this is a very simple case, which is usually different from reality. In reality there are two clear potential pitfalls to this tactic. The first is there could be a significant number of peaks, which drives the algorithm to inspect more of the frequency range on a minute scale, increasing computation time. The 50 machine test system displays some of these characteristics, however, the improved algorithm still performed much faster than the traditional algorithm. The second pitfall is the possibility of conservative bounds, where the upper bound could be continuously over 1, forcing examination of the entire range using the smallest frequency subdivision span. One way around this is to raise the point where subdivision takes place to a μ_{ub} (μ upper bound) value greater than 1. This would speed up both the traditional and improved algorithms since both algorithms are subject to the same problem. A third inherent pitfall to this method is that the values of μ bounds scale the frequency range, hence the result of the μ bounds test is essentially a "yes" or "no" answer. The numerical results for the bounds are not useful since they re-scale the range of

the frequency perturbation. These pitfalls were the motivating factors for using the skew μ approach as a way to get valid numerical bounds in fewer or faster tests.

7.1.5 Improved Selection of Frequency Spans in B& B

The obvious strategy in the selection of B& B frequency spans is the divide-by-two method. Dividing regions in half is easy to implement, but takes no advantage of emerging trends in the calculation of μ_{ub}. If the tested regions continually pass or fail the stability criteria, there is motivation for testing the regions in larger sections.

One method for incrementing test section size is to include a learning rate in the selection of section size. As the sections are tested, the width of the test section is increased via the learning rate if the section test passes ($\mu_{ub} < 1$). If the section test does not pass ($\mu_{ub} > 1$), then the learning rate is reset and the section is subdivided into smaller sections and searched appropriately.

This procedure works in both situations where the trend is for $\mu_{ub} > 1$ or $\mu_{ub} < 1$. The normal situation when the procedure starts is a trend for $\mu_{ub} > 1$. In this case, the procedure decrements the test region size more quickly in order to rapidly find the lower edge where the trend changes to $\mu_{ub} < 1$. Once the lower edge is found, then the algorithm subdivides the sections where $\mu_{ub} > 1$ for a more detailed search, and works its way back up the frequency range. This is the typical observation when running the B& B algorithm.

When continually changing the grid size, particular attention to insuring testing of all frequency regions is needed.

7.1.6 Improved Computation Speed by Re-use of D-scales

When searching frequency regions of relatively small size, there it typically very little change in the D-scales. Since the μ upper bound is $\overline{\sigma}(DMD^{-1}) = \|DMD^{-1}\|_{2i}$, an upper bound may be recalculated immediately by reusing the D-scales. There will be little loss in an optimal bound by reuse of D-scales, but a significant advantage in computational speed. As a precaution, checks may be placed in the code to do a full recalculation of μ bounds and

D-scales after a specific frequency interval, or if the change in $\overline{\sigma}(DMD^{-1})$ is larger than a predefined value from one region to the next. The expectation is that this method would be used on frequency regions where $\mu > 1$, requiring the region to be searched essentially at the minimal frequency division size. One caveat in applying this techniques with other skew μ techniques is to implement any matrix manipulation in a balanced form as referred to in section 5.3.1. Failing to follow this recommendation would most likely invalidate re-use of the D-scales.

7.2 Use of Continuity Results to Improve μ Bounds of Real Problems

In general, the value of μ can be a discontinuous function, as shown in [3]. This is especially characteristic of real parameter perturbation problems, which can cause difficulties in the convergence of μ bounds. Although the natural model may be a real parameter perturbation, it is possible to assume that the perturbations are complex and proceed with analysis. The result from acting in this methodology are typically conservative.

One method for improving the μ bound results for system with real perturbations is via the addition of a small amount of complex uncertainty to the real problem. The inclusion of complex uncertainty provides for continuity of μ and typically improves the results of the μ bounds search. Rather than searching for a value that exists on the real axis, the search occurs on a narrow elongated region with its major axis as the real number line. This region is created by the accumulation of search disks at every point along the specified length of the real number line, as shown in figure 7.3 . From a common sense standpoint, the methodology is well justified since both real world problems and numerical computations are not exact. Models of real parameter perturbations may perhaps be more accurately modeled by the inclusion of a little complex uncertainty, and nearness of numerical results improved by consideration of a slightly complex location.

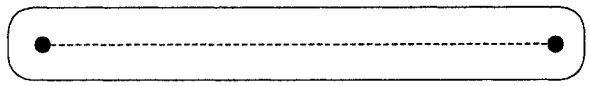


Figure 7.3: Elongated region caused by addition of a small amount of complex uncertainty

The in depth background to the following can be found in [40] with practical implementation information in [2].The following theorem is due to [40], and is reproduced here without proof, to justify the construction for the upper bound.

Theorem 9 *[40] Suppose $P(s)$ is a stable proper, rational $n \times n$ transfer matrix, where $P_\infty \triangleq \lim_{s\to\infty} P(s)$. Define the set $\mathcal{G}$ as follows:*

$$\mathcal{G} \triangleq \{P(j\omega) : \omega \in [0,\infty)\} \cup \{P_\infty\} \tag{7.1}$$

Suppose that $\Delta_2 \subset \mathbb{C}^{n\times n}$ is a complex block structure, and the real block structure $\Delta_1 \subset \mathbb{R}^{n\times n}$ is related to Δ_2 by

$$\Delta_1 = \mathbb{R}^{n\times n} \cap \Delta_2. \tag{7.2}$$

Let $\mathbf{\Delta}$ be the diagonal augmentation of these two sets:

$$\mathbf{\Delta} \triangleq \{diag[\Delta_1, \Delta_2] : \Delta_1 \in \mathbf{\Delta}_1, \Delta_2 \in \mathbf{\Delta}_2\}. \tag{7.3}$$

For any $G \in \mathcal{G}$, define

$$M(\varepsilon, G) \triangleq \begin{bmatrix} G & \sqrt{\varepsilon}G \\ \sqrt{\varepsilon}G & \varepsilon G \end{bmatrix}, \tag{7.4}$$

and the function

$$f(\varepsilon, G) \triangleq \mu_{\mathbf{\Delta}}(M(\varepsilon, G)). \tag{7.5}$$

The function $f : (0,\infty) \times \mathbb{C}^{n\times n} \to \mathbb{R}$ is continuous. For $\varepsilon_1 > \varepsilon_2 > 0$, the following is obtained $f(\varepsilon_1, G) \geq f(\varepsilon_2, G) \geq \mu(G)$ and

$$\lim_{\varepsilon\to 0} f(\varepsilon, G) = \mu(G). \tag{7.6}$$

Further, given and $\delta > 0$, there is an $\varepsilon_* > 0$ such that for ε satisfying $0 < |\varepsilon| \leq \varepsilon_*$,

$$\|M(\varepsilon, G)\|_{\mathbf{\Delta}} = \max_{G\in\mathcal{G}} f(\varepsilon, G) < \|P\|_{\mathbf{\Delta}_1} + \delta. \tag{7.7}$$

The implication is if a system G is augmented to form a system of the original and scaled replicas as in equation 7.4 then $\mu(G)$ is guaranteed to be continuous. This does not guarantee the μ bounds search will be improved, however in practice, this is often the case. In section 7.3 an ε of 0.01 was used to improve the continuity of the real problem by addition of a small amount of complex uncertainty.

7.2.1 Construction of System with Addition of Complex Uncertainty

The following information is mentioned to be explicit concerning the construction of a problem with complex uncertainty added to a real problem. The information is given in the form of a generalized example, however this is exactly the method of construction used in section 7.3.

Presume the system matrix equation has a form

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}. \tag{7.8}$$

Multiply the system matrix by a suitably sized matrix having the structure

$$f = \begin{bmatrix} I \\ \alpha \cdot I \end{bmatrix}, \tag{7.9}$$

where α^2 is a real scalar representing the percentage of the real uncertainty desired to be added to the problem as complex uncertainty.

Thus the matrix

$$G = fMf^T \tag{7.10}$$

is formed, or

$$G = \begin{bmatrix} I \\ \alpha \cdot I \end{bmatrix} \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} I & \alpha \cdot I \end{bmatrix}. \tag{7.11}$$

This can be expanded to

$$G = \left[\begin{array}{cc|cc} M_{11} & M_{12} & \alpha \cdot M_{11} & \alpha \cdot M_{12} \\ M_{21} & M_{22} & \alpha \cdot M_{21} & \alpha \cdot M_{22} \\ \hline \alpha \cdot M_{11} & \alpha \cdot M_{12} & \alpha^2 \cdot M_{11} & \alpha^2 \cdot M_{12} \\ \alpha \cdot M_{21} & \alpha \cdot M_{22} & \alpha^2 \cdot M_{21} & \alpha^2 \cdot M_{22} \end{array}\right]. \tag{7.12}$$

Expressing the perturbation block format for the original M matrix in the form of the Matlab®μ-tools syntax [2]

$$BLK = \begin{bmatrix} -r_1 & 0 \\ -r_2 & 0 \end{bmatrix}, \tag{7.13}$$

where $0 < r \in \mathbb{R}$. This indicates two real perturbation blocks of size r_1 and r_2. To construct the perturbation set for the augmented problem, the form in equation 7.13 is appended with the absolute value of the original block form in equation 7.13 to achieve

$$BLK2 = \begin{bmatrix} BLK \\ |BLK| \end{bmatrix}. \tag{7.14}$$

For the perturbation block associated with G, this can be expanded to

$$BLK2 = \begin{bmatrix} -r_1 & 0 \\ -r_2 & 0 \\ r_1 & 0 \\ r_2 & 0 \end{bmatrix}. \tag{7.15}$$

In the case of equation 7.15, rows 3 and 4 now indicate a complex perturbation of size r_1 and r_2 respectively.

Consider the following property of singular values [46],

$$\overline{\sigma}(AB) \leq \overline{\sigma}(A)\overline{\sigma}(B). \tag{7.16}$$

In the case proposed here, this is specifically

$$\overline{\sigma}(ABA^t) \leq \overline{\sigma}(A)\overline{\sigma}(B)\overline{\sigma}(A) = \overline{\sigma}(A)^2\overline{\sigma}(B). \tag{7.17}$$

Replace A with f in equation 7.16 and assume $\alpha = .1$ then $\overline{\sigma}(f) = 1.005$ and $\overline{\sigma}(f)^2 = 1.01$ Thus the maximum singular value of G is no more than $1.01 \cdot \overline{\sigma}(M)$. Since $\mu(M) \leq \overline{\sigma}(M)$, then it is reasonable to expect the bounds search for $\mu(G)$ to be improved over the bounds search for $\mu(M)$, where as the two robustness problems are close.

Addition of a small amount of complex uncertainty is the final item needed in the construction of the original scheme analyzing power system stability. The results from this scheme are discussed in the next section.

7.3 Results of Method and Comparison

A basic B&B routine as described earlier was applied to the 4-machine and 50-machine test systems. The sequence of iterations for the 4-machine test is presented in table 7.3. An improved B&B routine using the learning rate and D-scale improvements described earlier was also applied to the same system and its iterations are displayed in table 7.4. Graphics of the results from the 4 and 50 machine tests are displayed in figures 7.4, 7.5, 7.7 and 7.8, including a close-up of the region where the μ bound peaks. As a comparison, a gridded frequency analysis is included in figure 7.6

The most apparent difference between the two methodologies is the time cost. The time difference between the two methods is displayed in 7.1. In general, the improved B&B is approximately 3 times as fast as a basic B&B although it took 2 more iterations than the basic B&B. A second important item is that the improved B&B more accurately mapped the μ peak than the basic B&B. In the process, the improved B&B reduced conservativeness of the upper bound by 0.72.

It is evident that the improved B&B did not use evenly spaced frequency regions. However, examining table 7.4 shows that all frequency ranges were covered. In addition, because of the nature of the learning rate implementation, the improved B&B scheme actually covered frequencies up to 140 radians/sec. Most of the improvement over the basic B&B scheme is due to the reuse of the D-scales and computing the norm bound directly, with secondary gains being made by the learning rate.

Clearly the improved B&B scheme has a significant advantage over the basic B&B scheme and has some interesting application possibilities when problems become too large for standard μ or skew μ tools.

Comments should also be made concerning the comparison between the results of the gridded search, figure 7.6, and the branch and bound search, figures 7.4 and 7.5. Examination of the figures reveals the gridded upper bound is much lower than the upper bound of either branch and bound technique. This is due to the difference in the gridded problem versus

the frequency augmented problem. The value for μ of the gridded problem and the value of μ for the frequency augmented problem *are the same*, however, the problems are not the same. Since the problems are not the same, the *bounds* are not necessarily the same, and the bounds are the method used for identifying a potential range for μ. From examination of the gridded system, the upper bound for the frequency augmented problem appears to be significantly larger (> 5 versus < 1.6) for the frequency augmented system. This is correct. Certainly the an upper bound of 5 is valid, but may not be as close as desired to the value of μ. The significance of the frequency augmented system is that no frequency point has been missed. The results indicates there is a potential for instability ($\mu_{ub} > 1$), but the possibility of this instability occuring remains somewhat undetermined because the size of the frequency span itself is affected by the resulting bound on μ. It is reasonable expectation to use a combination of the gridded and frequency augmented methods to gain a more accurate prediction of the frequency at which instability is most likely to occur, and the size of perturbation which is likely to cause instability.

Computer	Basic B&B (sec)	Improved B&B (sec)
HP Kayak Dual 450 MHz Pentium	180.43	63.37
Generic AMD 950 MHz Duron	160.05	54.76
Dell 8100 1.7 GHz Pentium	53.01	18.40

Table 7.1: Time comparisons for basic B&B scheme and improved B&B scheme for three different computers for 4 machine system

Computer	Basic B&B (sec)	Improved B&B (sec)
HP Kayak Dual 450 MHz Pentium	37539 (625.7 min)	15116 (251.9 min)
Generic AMD 950 MHz Duron	70989 (1183.2 min	28203 (470.0 min)
Dell 8100 1.7 GHz Pentium	13006 (216.77 min)	5178 (86.3 min)

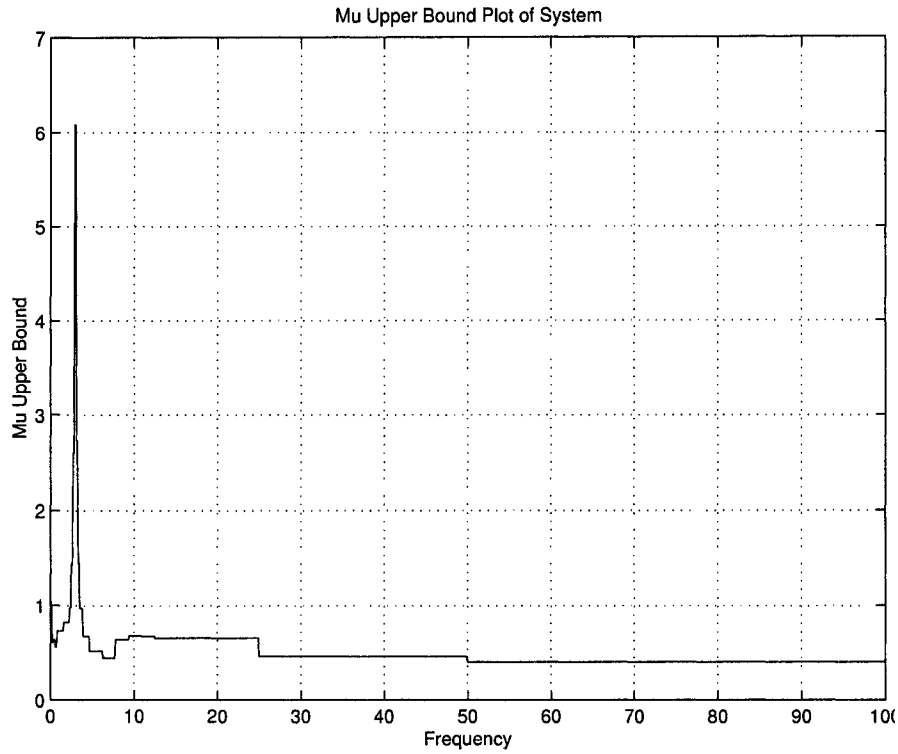
Table 7.2: Time comparisons for basic B&B scheme and improved B&B scheme for three different computers for 50 machine system

Iteration	ω_{min}	ω_{min}	μ Upper Bound
1	0	100	1.27163
2	0	50	1.57835
3	0	25	3.28227
4	0	12.5	2.31274
5	0	6.25	27.84857
6	0	3.125	1.67103
7	0	1.5625	2.68178
8	0	0.78125	1.85057
9	0	0.39063	1.03478
10	0	0.19531	1.04025
11	0.19531	0.39063	0.61117
12	0.39063	0.78125	1.05389
13	0.39063	0.58594	0.64031
14	0.58594	0.78125	0.56301
15	0.78125	1.5625	0.72841
16	1.5625	3.125	1.83883
17	1.5625	2.34375	0.8212
18	2.34375	3.125	2.79529
19	2.34375	2.73438	1.35713
20	2.34375	2.53906	0.97458
21	2.53906	2.73438	1.42994
22	2.73438	3.125	4.80003
23	2.73438	2.92969	2.60127
24	2.92969	3.125	6.08392
25	3.125	6.25	1.24461
26	3.125	4.6875	1.44011
27	3.125	3.90625	1.77247
28	3.125	3.51563	2.23638
29	3.125	3.32031	2.73215
30	3.32031	3.51563	1.441
31	3.51563	3.90625	0.96887
32	3.90625	4.6875	0.67286
33	4.6875	6.25	0.51433
34	6.25	12.5	2.19942
35	6.25	9.375	1.01421
36	6.25	7.8125	0.44478
37	7.8125	9.375	0.64004
38	9.375	12.5	0.67514
39	12.5	25	0.6541
40	25	50	0.46281
41	50	100	0.40348

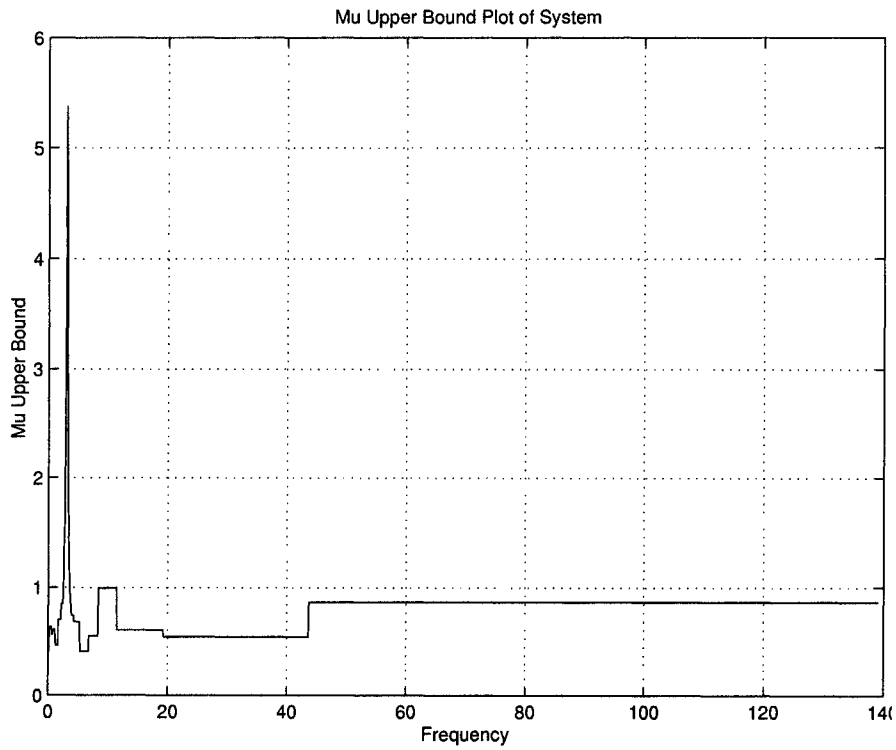
Table 7.3: Iterations for a basic B&B scheme using a divide-by-two format on the 4 machine test problem

Iteration	ω_{min}	ω_{min}	μ Upper Bound
1	0	100	1.2624
2	0	43.4783	1.7142
3	0	16.4379	5.5386
4	0	5.4041	9.2927
5	0	1.5449	2.7024
6	0	0.38405	1.0356
7	0	0.1	1.0279
8	0.1	0.2	0.41866
9	0.2	0.4	0.63746
10	0.4	0.8	1.2053
11	0.4	0.6	0.64873
12	0.6	0.8	0.57285
13	0.8	1.2	0.6193
14	1.2	2.2	1.5103
15	1.2	1.7	0.48024
16	1.7	2.2	0.70787
17	2.2	3.2	3.0984
18	2.2	2.7	1.2512
19	2.2	2.45	0.85626
20	2.45	2.7	1.3114
21	2.45	2.575	1.035
22	2.45	2.55	0.9878
23	2.55	2.65	1.2064
24	2.65	2.75	1.5457
25	2.75	2.85	2.1292
26	2.85	2.95	3.28
27	2.95	3.05	5.3697
28	3.05	3.15	4.5987
29	3.15	3.25	2.7695
30	3.25	3.35	1.8746
31	3.35	3.45	1.4059
32	3.45	3.55	1.1264
33	3.55	3.65	0.9435
34	3.65	3.85	0.84614
35	3.85	4.25	0.74392
36	4.25	5.25	0.69192
37	5.25	8.375	1.3373
38	5.25	6.8125	0.41998
39	6.8125	8.375	0.56089
40	8.375	11.5	0.9911
41	11.5	19.3125	0.61468
42	19.3125	43.7266	0.5519
43	43.7266	139.094	0.8677

Table 7.4: Iterations for an improved B&B scheme using a learning rate and repeated D-scales on small frequency shifts for 4 machine test problem

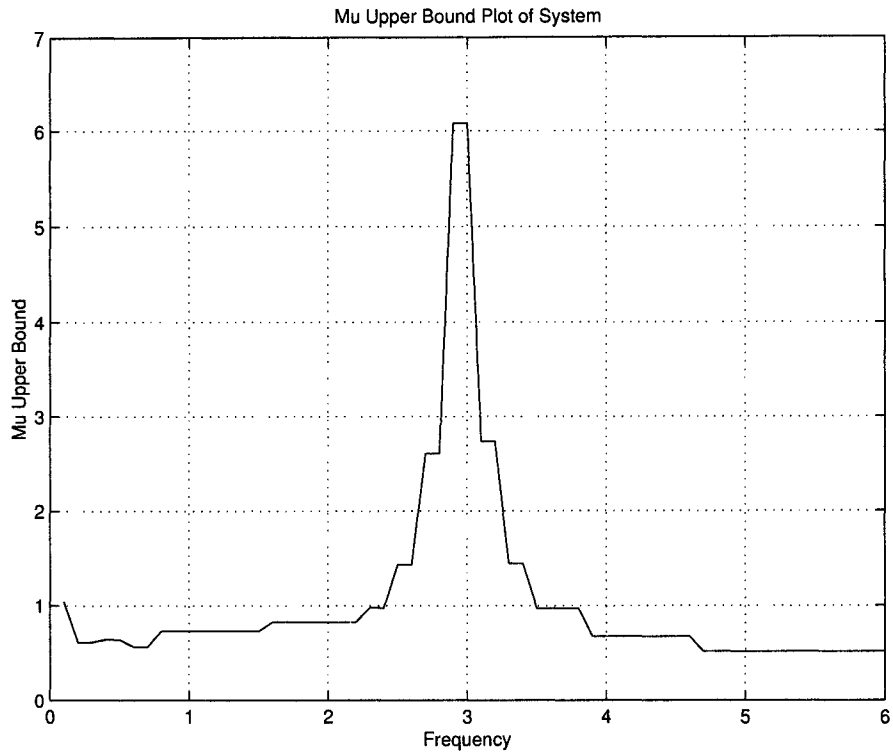


(a) Original B&B

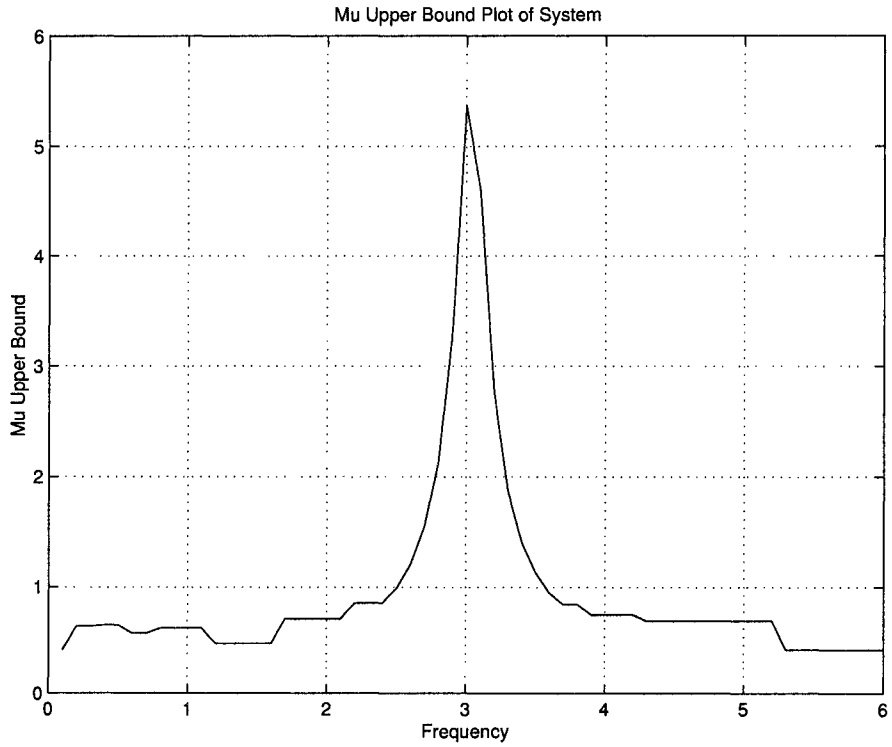


(b) Improved B&B

Figure 7.4: Results from B&B upper bound search algorithms using standard μ tools for 4 machine system-full frequency range view



(a) Original B&B Close up



(b) Improved B&B close-up

Figure 7.5: Results from B&B upper bound search algorithms using standard μ tools for 4 machine system-close up view

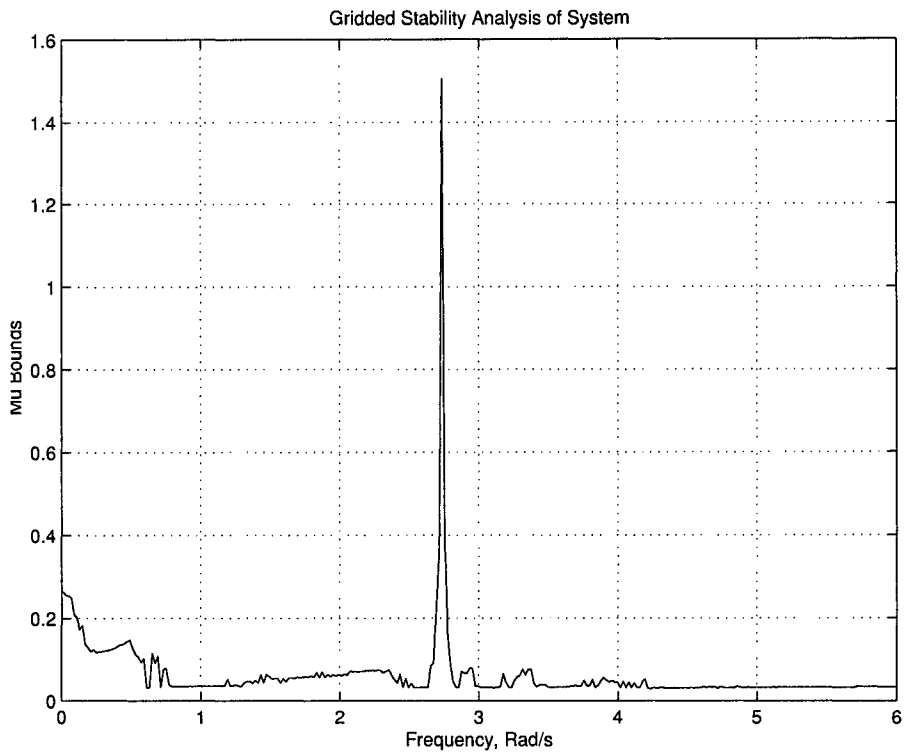
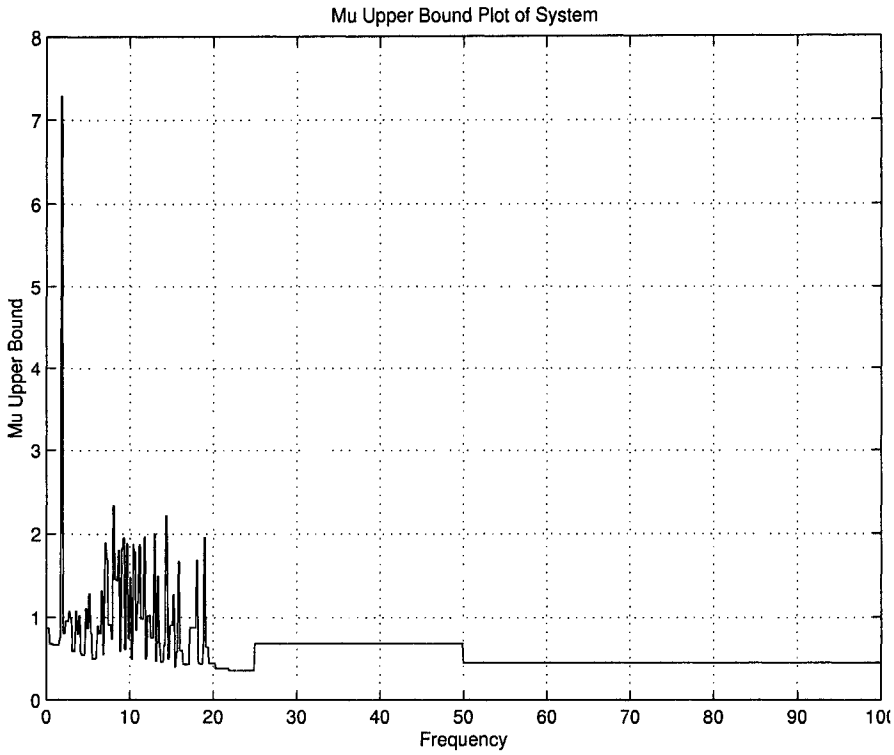
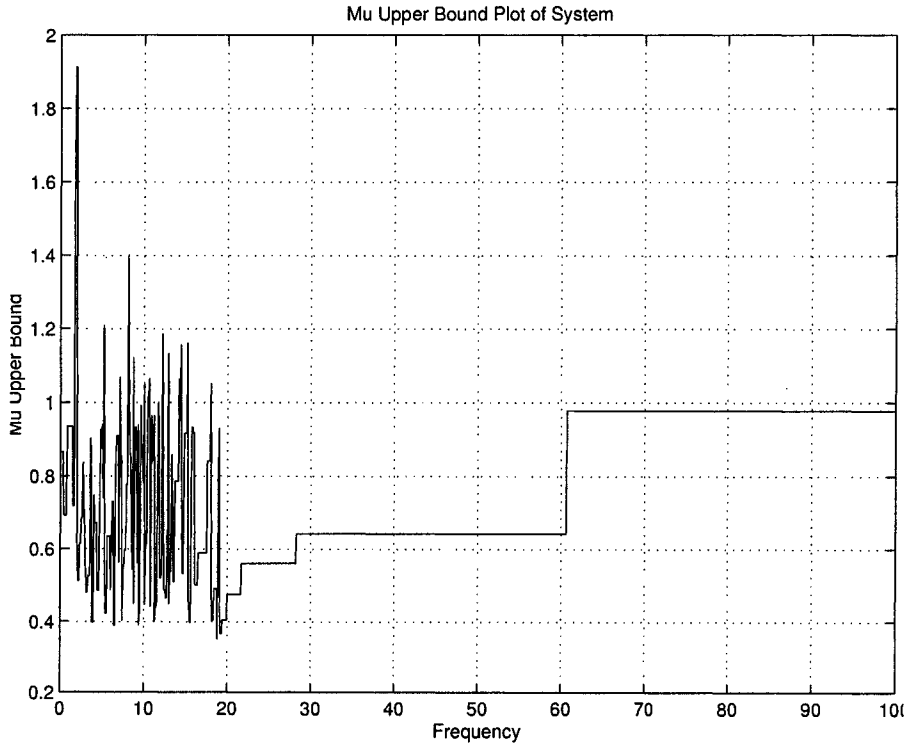


Figure 7.6: Results from a gridded upper bound search using standard μ tools for 4 machine system

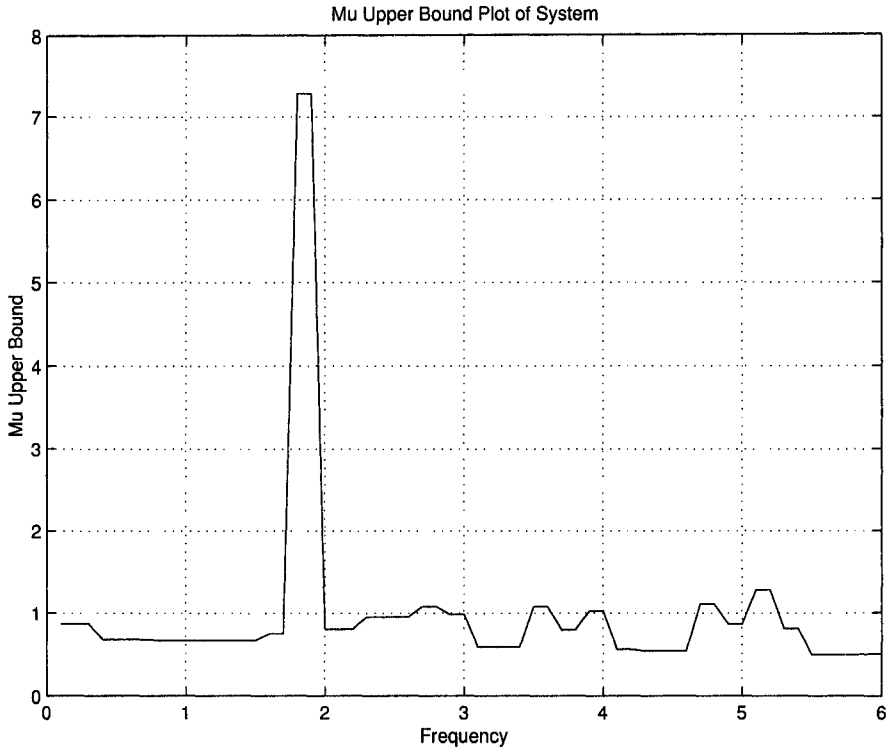


(a) Original B&B

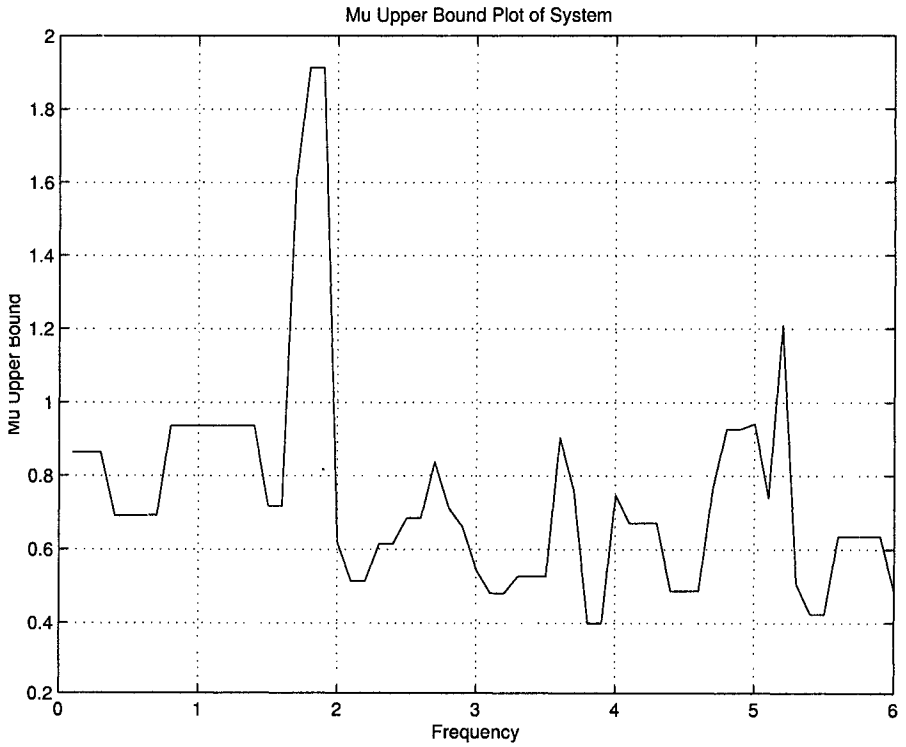


(b) Improved B&B

Figure 7.7: Results from B&B upper bound search algorithms using standard μ tools for 50 machine system-full frequency range view

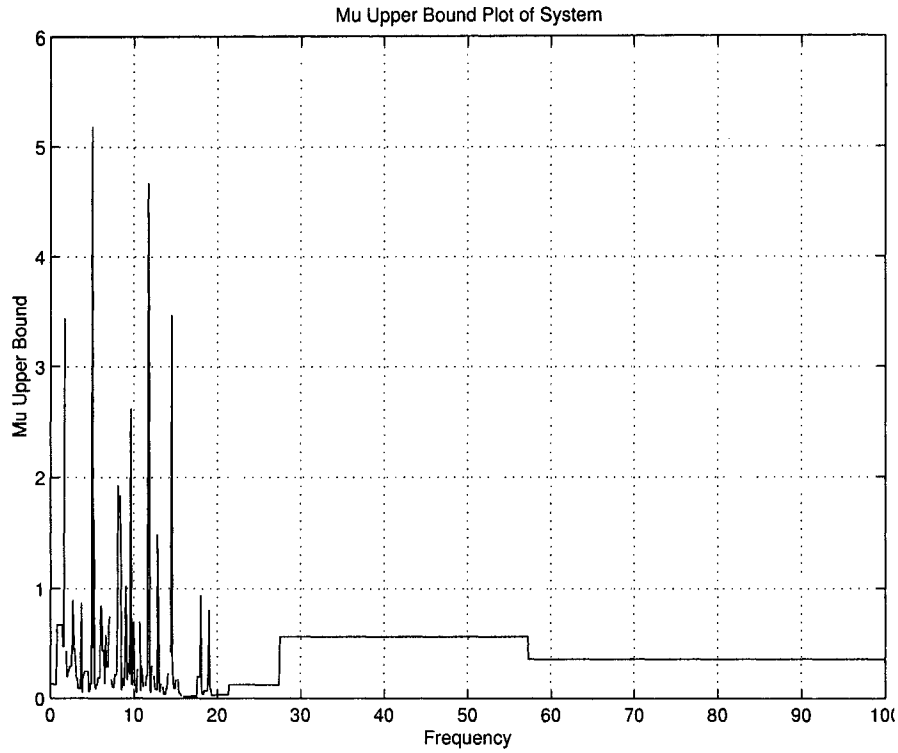


(a) Original B&B

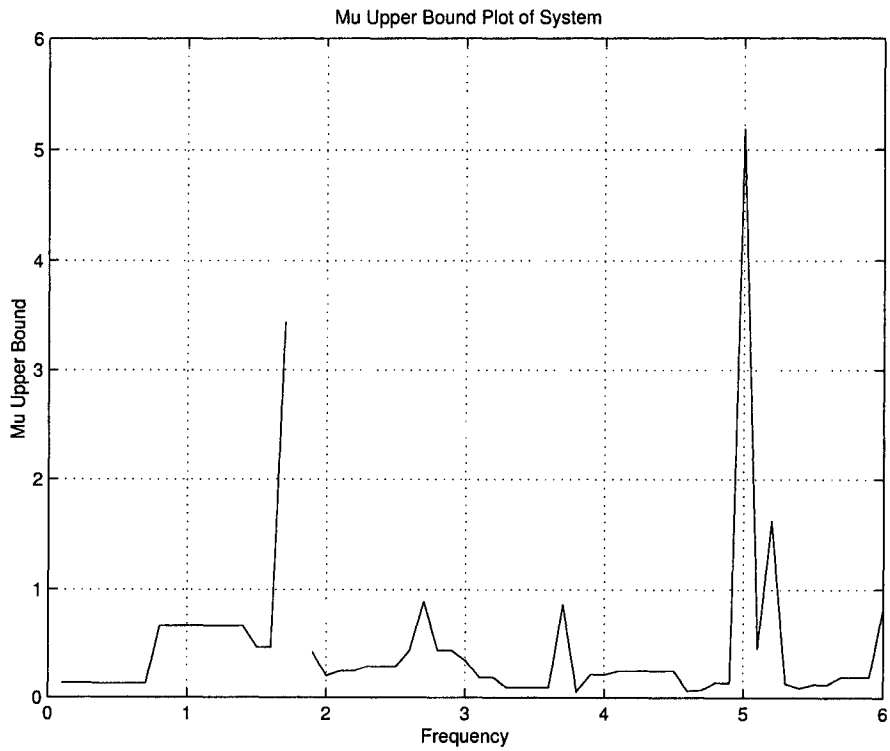


(b) Improved B&B Close-up

Figure 7.8: Results from B&B upper bound search algorithms using standard μ tools for 50 machine system-close up view



(a) 0-100 radian scale



(b) Close-up 0-6 radian scale

Figure 7.9: Results from B&B upper bound search algorithms using direct calculation of upper bound for 50 machine system

Chapter 8

Skew μ Control Synthesis

8.1 The Motivation for Skew μ Control Synthesis

One would like to use the ability to find accurate skew μ bounds to aid in the design of controllers and control systems, such as the one depicted in figure 8.1, where K is the controller that has been added to the system.

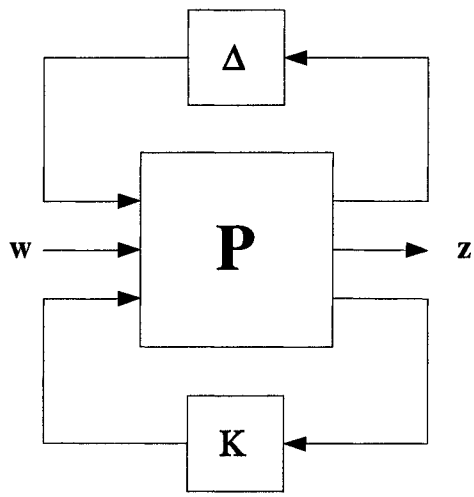


Figure 8.1: A robust control synthesis problem formulated in the LFT format

In particular, there is one specific robust control problem that formulates very well as a skew μ problem, as well as being an important problem in its own right. This problem deals with parameter perturbations as fixed range perturbations, and includes a "fictitious" perturbation block as a varying perturbation which closes the output to the input. This

fictitious block is referred to as the performance block and is typically denoted Δ_p. It is depicted in figure 8.2.

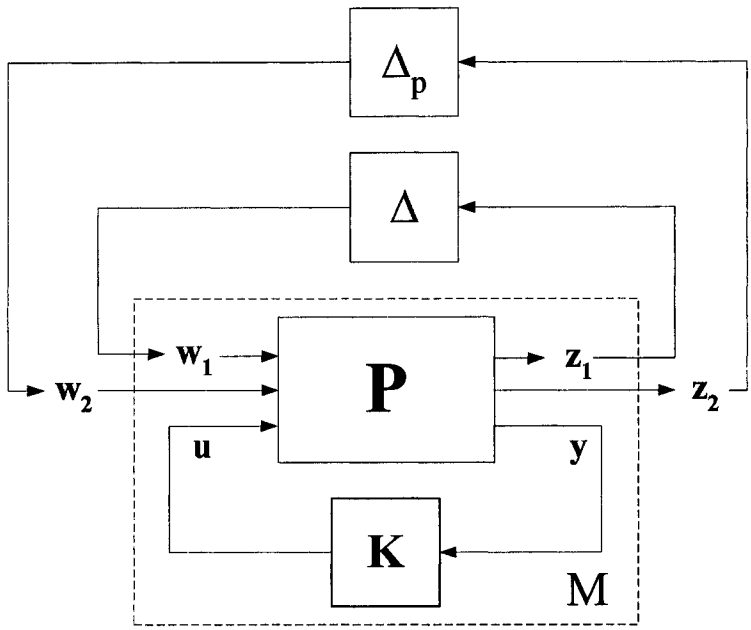


Figure 8.2: LFT diagram of the robust control synthesis framework with performance block

The idea behind closing the output to input with the performance block Δ_p is to find the smallest perturbation (in terms of the 2 norm) that will potentially destabilize the system. When the performance perturbation block is included, the question is asked "Is it possible to potentially destabilize the system by a combination of the parameter perturbations and a disturbance input signal?". As in finding the bounds on μ for previous problems, this problem is analyzed for the smallest perturbation (including performance block) that will set the $\det(I - \Delta M) = 0$.

If skew μ for this situation is a real number (i.e. not infinity), then some level of performance may be obtained. If $\mu_s = 1$ then disturbance rejection performance is exactly as was requested when setting up the range of Δ_p. If skew μ is less than one then the system performance is better (in terms of disturbance rejection) than originally desired. If skew μ is greater than one then the system performance is less than originally specified.

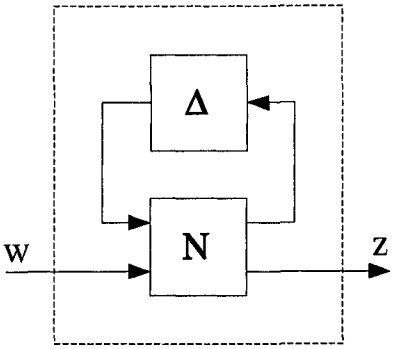


Figure 8.3: A robust analysis problem formulated in the LFT format

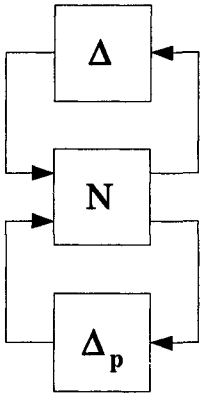


Figure 8.4: Closing the input to output with a performance block

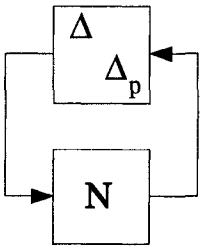


Figure 8.5: Regrouping figure 8.4 using rules of LFT's

The concept can be more easily understood by looking at figures 8.3, 8.4 and 8.5.

In terms of *system gain* from w to z, as in figure 8.3, one would like

$$\|F_u(N,\Delta)\|_\infty \leq 1. \tag{8.1}$$

If the output to input is closed with a perturbation block Δ_p, where $\|\Delta_p\| \leq 1$, then the system looks like figure 8.4. Ultimately this can be simplified with LFT properties to form 8.5. From figure 8.5 one can see that a system gain greater than one could cause a disturbance to be repetitively amplified to the point where it causes instability.

When one probes the characteristics many general problems, one notices a significant concern when finding the μ bounds for those problems. Specifically, most actual application problems have known ranges or expectations for parameter perturbations. The general μ problem formulation expands or shrinks these perturbations as necessary to set $\det(I - \Delta M) = 0$, regardless of the physical relationship of the parameters. Ideally, one would like to fix the variation range of the parameter perturbations, and then find the level of performance available (via Δ_p) that sets $\det(I - \Delta M) = 0$. This formulates the question

> "For perturbations with these specified (fixed) ranges, what level of performance is available from the system?"

This question assumes the controller K, as in figure 8.1, to be known. However, the situation begs that another question be asked:

> "For a system with a specific controller K and level of performance, is there another choice of K that provides better performance?"

At this point the readers should be asking themselves,

> "If there is a *better* controller K, how would one find it?"

Those are the three questions that will be analyzed and answered in this chapter.

The problem proposed in the first question is a skew μ problem where the parameter perturbations are the fixed range perturbations, and the fictitious performance block Δ_p is the varying range perturbation. This makes the problem a prime application for skew μ analysis techniques.

There is no specific deterministic method for finding the controller K that provides optimal performance. The accepted method is somewhat heuristic and consists of three basic steps performed in an iterative sequence. The three steps are:

1. Design a controller K such that for certain fixed range perturbations the input w to output z meets some criteria.

2. Check the skew μ bounds for the system. This implies a choice of D-scales and G-scales, which affects the choice of K.

3. If the performance specification is met, and/or the process appears to have converged then stop, otherwise choose another K and repeat.

8.2 Extending μ Techniques to Skew μ Control Synthesis

Like the other methods in this dissertation, skew μ control synthesis will be developed in terms of existing μ control synthesis. A direct control synthesis method for a μ controller does not exist. However, an iterative procedure called D-K iteration is available (via Matlab®). The D-K iteration procedure couples synthesis of an H_∞ controller with μ analysis methods to create a procedure which returns a controller with a bound on the level of μ and attempts to minimize it.

Synthesis of an H_∞ controller is a convex problem which has a guarantee of finding a global optimum. Likewise, finding an upper bound on μ is also a convex problem with a a

guarantee of finding a global optimum. However, the problems are not jointly convex [46], but one can develop an iteration that is monotonically decreasing. Thus, the procedure for synthesizing a μ controller does not come with a guarantee of finding a global optimum, only converging to a local solution. Having clarified this, a final remark is the D-K iteration procedure works fairly well in practice.

8.2.1 Basics of the D-K Iteration Procedure

First the standard D-K iteration procedure will be explained in terms of μ. Then the procedure will be extended to skew μ. As a general overview, the D-K iteration procedure uses the following steps, which can be referenced to figure 8.7. Alternately, the procedure is also outlined in a flowchart in figure 8.6 which may also be referenced to figure 8.7. Appropriately sized left and right D matrices are indicated by subscripts $_L$ and $_R$. This algorithm will layout the map which will be used through the remainder of this section.

1. Initialize $D(j\omega) = I \ \forall \ \omega$.

2. Find a stable minimum phase SS realization, $D(s)$. This implies synthesizing a transfer function $D(s)$ that fits the series of $D(\omega)$ scales acquired at each frequency point in step 4 (or trivially, step 1), in magnitude only (not phase).

3. Synthesize an H_∞ controller (K) for the scaled problem, $\inf_{K \ stab} \|F_l(\hat{P}, K)\|_\infty$ with the D-scales from step 2 wrapped into P to form $\hat{P}$.

4. Compute the μ upper bound across ω to find $\inf_{D(\omega)\in\mathcal{D}} \sigma(D_L(\omega)M(j\omega)D_R(\omega)^{-1})$. Save the individual D scales from each frequency point.

5. Compare μ upper bound to previous iterations' μ upper bound.

   - If process is converged, or meets specification, stop
   - If process is not converged or greater than specifications,go to 2.

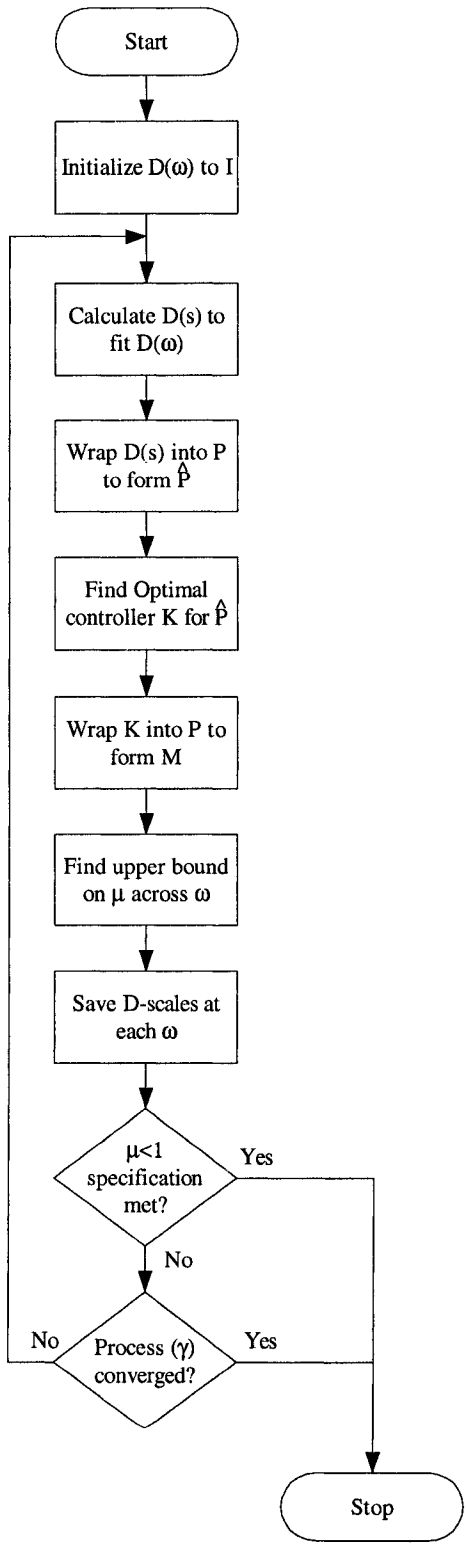


Figure 8.6: Flow chart for D-K iteration procedure

This algorithm requires some relevant remarks:

M is defined as

$$M = F_l(\hat{P}, K). \tag{8.2}$$

Define

$$\hat{D} = \begin{bmatrix} D & 0 \\ 0 & I \end{bmatrix}, \tag{8.3}$$

where D is of the dimension of P and I is of the dimension of K, then $\hat{P} = \hat{D}_L P \hat{D}_R^{-1}$. This implies that

$$D_L F_l(P,K) D_R^{-1} = F_l(\hat{D}_L P \hat{D}_R^{-1}, K) = F_l(P,K). \tag{8.4}$$

The goal here is to prevent scaling of the designed controller K.

There are basically two possible outcomes for this procedure: (1) $\|D_L M D_R^{-1}\|_\infty < 1$, in which case a controller has been created which is stable and meets performance objectives, or (2) the H_∞ norm is no longer decreasing, which indicates controller performance can not be improved and system performance cannot be guaranteed for the perturbation specified. Note that each of these steps are convex, but the combination is not necessarily jointly convex, thus a global minimization is not guaranteed and the process may converge to a local solution instead. Practical experience tends to indicate the procedure works well in most cases.

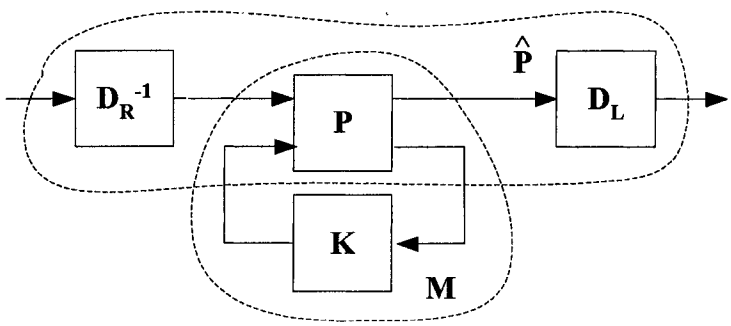


Figure 8.7: Block diagram for D-K iteration procedure

8.3 Control Design for the General μ Problem

The general set up for robust controller synthesis is shown in figure 8.8, with a more detailed depiction for the SS form shown in figure 8.9 . The nominal closed-loop system, M, consists of the open loop plant model P, and controller K. The goal is to design a controller K such that the resulting $M - \Delta$ system has the desired stability and performance properties based on perturbation Δ and performance block Δ_p. If system uncertainties have been accounted for correctly, then the variation in performance between the real application and the robust control model should be within expectation.

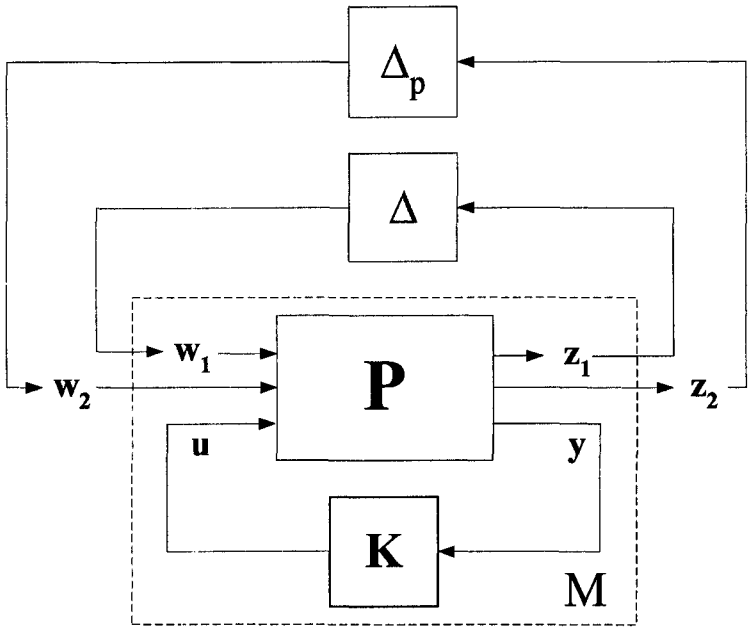


Figure 8.8: LFT diagram of the robust control synthesis framework

8.3.1 H_∞ Optimal Control

H_∞ optimal control deals with worst case disturbance signals in L_2, hence the desire is to minimize $\overline{\sigma}(M(j\omega))$ at the frequency where it peaks through the addition of an appropriate control structure. The result is the square root of energy to square root of energy or square root of power to square root of power gain (i.e. the induced 2 norm) is minimized. The

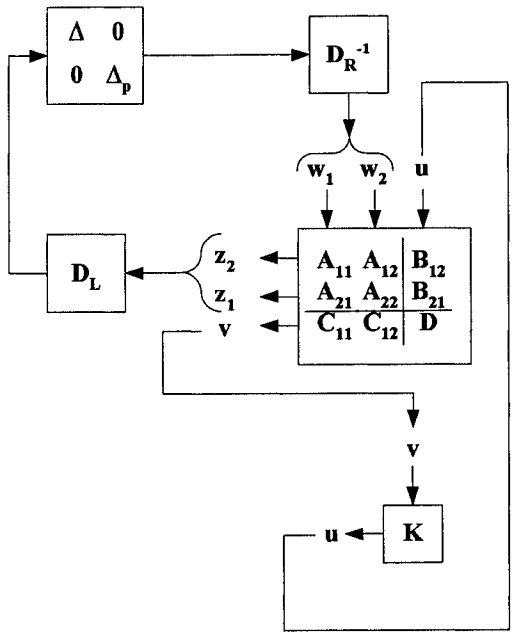


Figure 8.9: Detailed LFT diagram of the robust control synthesis framework with SS Matrix

basics of H_∞ controller synthesis are in the following sections, however, the specifics of H_∞ controller synthesis can be found in [15].

8.3.1.1 H_∞ Formalization

For explaining the basics, figure 8.8 will be used as a reference. It is also presumed that K can be wrapped into P to form M. By doing so, the problem is reduced to the standard μ analysis format.

The input output relationship for a system G can be expressed as

$$\begin{bmatrix} z \\ y \end{bmatrix} = G(s) \begin{bmatrix} w \\ u \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}. \tag{8.5}$$

Additionally, presume G is expressed in SS form

$$G = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right]. \tag{8.6}$$

This form is extremely common since many/most systems are expressed in SS form as a standard convention and any system can be re-expressed in SS form.

The goal is to find a stabilizing K such that minimizes the H-infinity norm of the $G-K$ interconnect, or

$$\inf_{K\ stabil} \|M\|_\infty = \inf_{K\ stabil} \|F_L(G,K)\|_\infty. \tag{8.7}$$

So, for some desired γ, where $\gamma \geq 0$ the desire is to answer to the question "Does there exist an internally stabilizing controller K such that $\|F_L(G,K)\|_\infty \leq \gamma$"?

If the answer is yes, a controller $K(s)$ that does the job can be provided. Typically, a bisection algorithm (γ iteration) is required to solve for the (sub) optimal γ, and is explained in more detail in [15]. The rudiments of the gamma iteration are as follows.

In [15] it is shown that if the system $\|G\|_\infty < \gamma$, then H has no eigenvalues on the imaginary axis, where D is assumed to be 0 and H is defined as

$$H \triangleq \begin{bmatrix} A & \gamma^{-2}BB^H \\ -C^HC & A^H \end{bmatrix}, \tag{8.8}$$

and G is defined in equation 8.6. This implies an iteration between choosing a γ and checking the eigenvalues of H.

In solving for the desired controller, the following assumptions are generally made,

1. (A,B_1) is stabilizable and (C_1,A) is detectable
2. (A,B_2) is stabilizable and (C_2,A) is detectable
3. Matrices D_{11} and D_{22} may be set as

$$D_{11} = 0$$
$$D_{22} = 0. \tag{8.9}$$

4.

$$D_{12}^H \begin{bmatrix} C_1 & D_{12} \end{bmatrix} = \begin{bmatrix} 0 & I \end{bmatrix} \tag{8.10}$$

5.

$$\begin{bmatrix} B_1 \\ D_{21} \end{bmatrix} D_{21}^H = \begin{bmatrix} 0 \\ I \end{bmatrix} \tag{8.11}$$

The requirements on D_{11} and D_{22} are not constraints, but rather convenience. They are not required to be 0, but it does simplify the algorithm needed for finding a stabilizing K.

The fourth and fifth requirements may be met by making

$$D_{12} = \begin{bmatrix} 0 \\ I \end{bmatrix}$$
$$D_{21} = \begin{bmatrix} 0 & I \end{bmatrix}. \tag{8.12}$$

The requirement is that D_{12} and D_{21} are full rank in order to obtain a well posed problem. The use of the identity matrix simplifies the algorithm, and its use is countered by scaling of other parameters elsewhere in the equations.

If the above conditions are met, then a controller, K, is found by solving two Ricatti equations. The viability of the controller can be resolved into five conditions. The conditions are

1. Solve the first Ricatti equation for X:

$$A^T + XA + C_1^T C_1 + X\left(\frac{1}{\gamma^2}B_1 B_1^T - B_2 B_2^T\right)X = 0 \tag{8.13}$$

   with $A + \left(\frac{1}{\gamma^2}B_1 B_1^T - B_2 B_2^T\right)X$ stable.

2. A constraint of the first Ricatti equation solution is that X be PD, i.e.

$$X \geq 0. \tag{8.14}$$

3. Solve the second Ricatti equation for Y:

$$AY + YA^T + B_1 B_1^T + Y\left(\frac{1}{\gamma^2}C_1^T C_1 - C_2^T C_2\right)Y = 0. \tag{8.15}$$

4. A constraint of the second Ricatti equation solution is that X be PD, i.e.

$$Y \geq 0. \tag{8.16}$$

5. A cross coupling constraint of the solution's X and Y is

$$\rho(XY) < \gamma^2. \tag{8.17}$$

If any of these five conditions are not met, then the controller K will not meet the criteria of $\|F_L(G,K)\|_\infty < \gamma$. If these conditions are met, then the controller K can be constructed as

$$K = \left[\begin{array}{c|c} A_\infty & -Z_\infty L_\infty \\ \hline F_\infty & 0 \end{array}\right] \tag{8.18}$$

where

$$\begin{aligned} F_\infty &= -B_2^T X \\ L_\infty &= -Y C_2^T \\ Z_\infty &= (I - \frac{1}{\gamma^2} YX)^{-1} \\ Z_\infty &= A + \frac{1}{\gamma^2} B_1 B_1^T X + B_2 F_\infty + Z_\infty L_\infty C_2 \end{aligned} \tag{8.19}$$

On occasion, performing the H_∞ synthesis procedure until the optimal γ is obtained is counter productive. The underlying problem is that H_∞ synthesis may have problem meeting conflicting objectives over various frequencies. The method for countering this problem is to choose a controller from an iteration several iterations before optimal is obtained. The controllers obtained from the sub-optimal iterations have a tendency to handle objective trade-offs over various frequencies better, even though they are sub-optimal.

This covers the basics of the H_∞ control synthesis procedure. Now this procedure can be extended to skew μ by adapting the procedure in the proper sections.

8.4 Adaptation of the D-K Iteration Method to Skew μ

The controller designed for the standard μ / D-K iteration procedure satisfies the questions "Given certain potential system perturbations, is there a controller that will stabilize the

system?" and "What performance level and perturbation size can the controller be expected to handle?".

The controller designed for the skew μ / D-K iteration procedure answers the question "If a system is subjected to known, fixed size perturbations , is there a controller that will stabilize this system, and what level of performance can be achieved?".

The skew μ case offers the advantage by allowing the specification of fixed size perturbations, which will not be re-scaled by the μ bound. The μ / D-K iteration procedure provides an answer which requires the re-scaling of the perturbations by the bound on μ. Practical problems typically have physical perturbations of a fixed size which cannot be changed. The skew μ / D-K iteration procedure provides a direct solution for these problems, making it a much more advantageous procedure to use.

The D-K iteration procedure is approximately the same for both algorithms. The primary implementation difference between the μ and skew μ algorithms is the peak skew μ bound across frequency,ν, is wrapped into the varying part of the system matrix during each iteration. Mathematically this is expressed as

$$\hat{M} = \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu_i}} \cdot I_\nu \end{bmatrix} \cdot M \cdot \begin{bmatrix} I_f & 0 \\ 0 & \frac{1}{\sqrt{\nu_i}} \cdot I_\nu \end{bmatrix}. \tag{8.20}$$

Because the goal is to design a controller for a system with a varying response over a range of frequency, the value used to wrap into M is the peak value of skew μ at the worst case frequency. Thus, if the peak skew μ value across frequency is $\mu_{s_i} = \nu$ for an individual iteration, then the D-K iteration procedure will converge toward a value of $\mu_{s_i} = 1$ for an individual iteration i. When $\mu_{s_i} \approx 1$, the resulting performance level in terms of a skew μ controller design will then be the cumulative product of μ_{s_i} for each iteration of the D-K iteration procedure, or

$$\begin{aligned} \mu_s(M) &= \nu_1 \nu_2 \ldots \nu_i \ldots \nu_n \\ i &= 1 \ldots n, \end{aligned} \tag{8.21}$$

where i is the iteration number.

This is depicted in figure 8.11, where the uncertainty Δ_p is assumed to be varying perturbation parameter, and the other perturbation, Δ is assumed to be fixed in range.
A flow chart of the process is given in figure 8.10.

8.4.1 Probable Use for the Skew μ D-K Iteration Method

Figure 8.11 gives a hint concerning the most likely use for the skew μ D-K iteration procedure. Presume Δ_p indicates the uncertainty depiction of the performance block, and Δ refers to all other perturbations and is assumed to be fixed in range. This defines the question "If there exist uncertainties of a given range, then what level of performance is achievable?". Repeating this slightly differently; many problems have a known range of uncertainty which can not be changed. If this is so, then what level of performance can be obtained from a controller, and what is the structure of the controller.
This is the immediate appeal of the skew μ D-K iteration procedure; however, other uses will emerge as the software and applied examples becomes more widely available.

8.5 A Design Example Using the Skew μ D-K Iteration Method

This is a skew μ performance example where the original example was taken from [46]. In [46], the example is given as a "μ" controller design example. In this document it will be performed as a skew μ controller design example to answer the question, "Given parameter perturbations with these fixed ranges, what level of performance is available from the system, and what controller will give this level of performance?". Many of the techniques applied here have been discussed earlier in this document, thus one may want to consult sections of chapter 2 or [46] while reviewing this example.
Consider the case where there is multiplicative input uncertainty and a weighted performance sensitivity. Additionally, consider a simplified distillation process represented as

$$G(s) = \frac{1}{75s+1}\begin{bmatrix} 87.8 & -86.4 \\ 108.2 & -109.6 \end{bmatrix}. \tag{8.22}$$

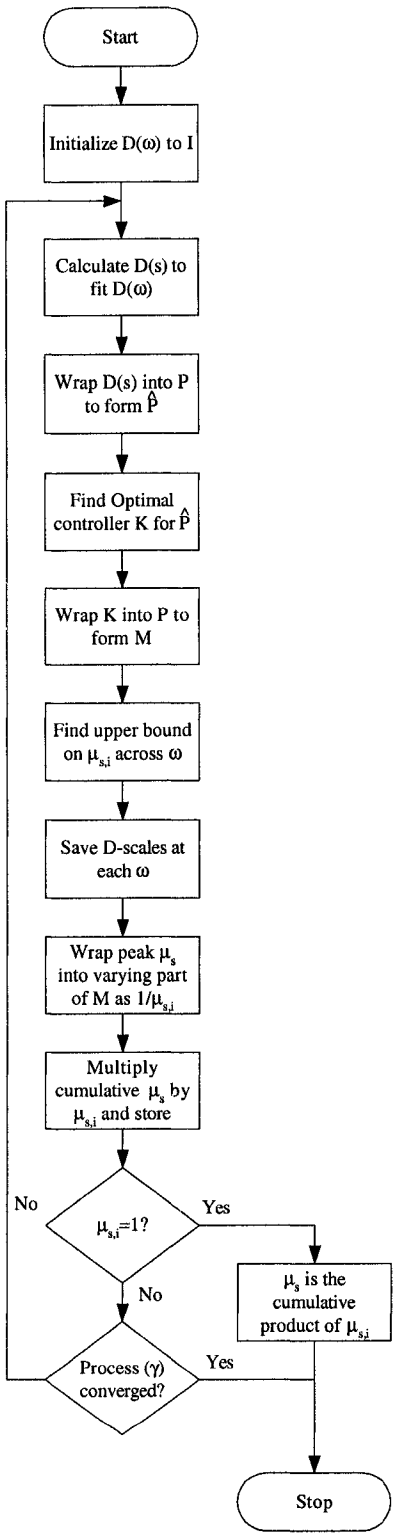


Figure 8.10: Flow chart for the skew μ D-K iteration procedure

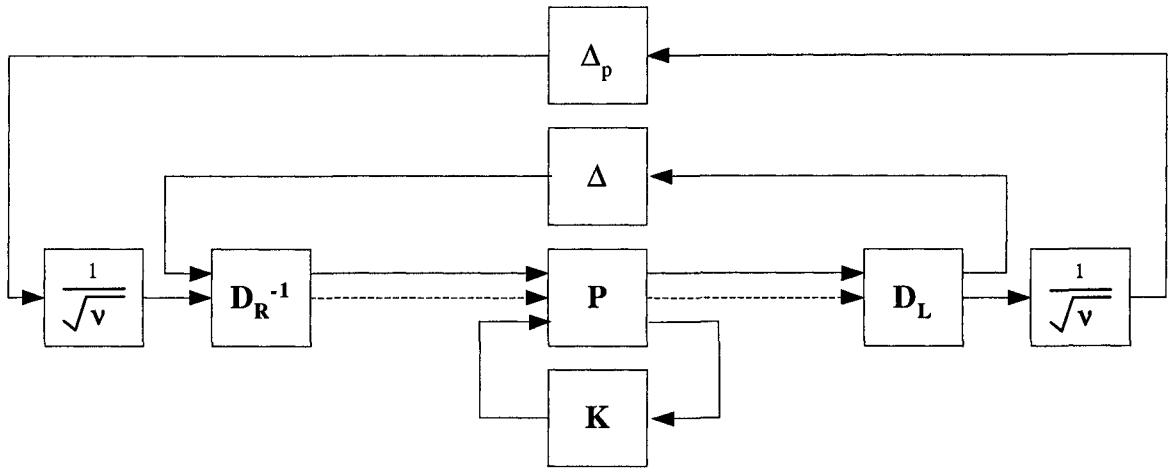


Figure 8.11: LFT diagram of the robust control synthesis framework with skewing included

$G(s)$ is embedded in the system as shown in the block diagram of figure 8.12.

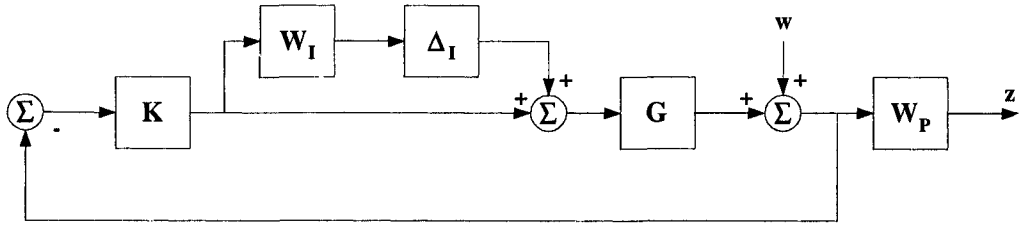


Figure 8.12: Simplified distillation process block diagram

Figure 8.12 can be rearranged into the simplified LFT form of figure 8.13.

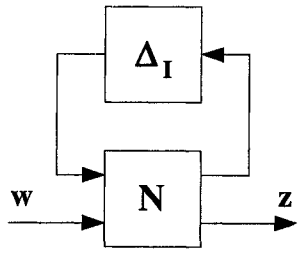


Figure 8.13: Distillation example reduced to an LFT

When the performance block is included, then the system can be depicted as in figure 8.14. This is the format needed for performing skew μ controller synthesis.

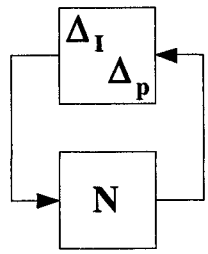


Figure 8.14: Reduced LFT with performance block

Note that in figures 8.13 and 8.14, the system N is actually

$$N = F_l(P,K), \tag{8.23}$$

where P is constructed as

$$P = \begin{bmatrix} 0 & 0 & w_I \\ w_pG & w_p & w_pG \\ -G & -I & -G \end{bmatrix}. \tag{8.24}$$

This includes the plant model, the uncertainty weight and the performance weight, but not the controller which is to be designed.

The uncertainty weight w_II and performance weight w_PI are defined as follows,

$$w_I(s) = \frac{s+0.2}{0.5s+1} \tag{8.25}$$

and

$$w_P(s) = \frac{\frac{s}{2}+0.05}{s}. \tag{8.26}$$

The objective is to minimize the peak value of $\mu_s(N)$ for the perturbation structure $\hat{\Delta}$, where $\hat{\Delta}$ is defined as

$$\hat{\Delta} = \begin{bmatrix} \Delta_I & 0 \\ 0 & \Delta_p \end{bmatrix}. \tag{8.27}$$

Diagonal input uncertainty will be considered, thus Δ_I is a 2-by-2 diagonal matrix. The performance block Δ_p is a full 2-by-2 matrix representing the performance specification. This is sufficient to define the perturbation block structure as

$$\hat{\Delta} = \begin{bmatrix} \Delta_{I_1} & 0 & 0 & 0 \\ 0 & \Delta_{I_2} & 0 & 0 \\ 0 & 0 & \Delta_{p11} & \Delta_{p12} \\ 0 & 0 & \Delta_{p21} & \Delta_{p22} \end{bmatrix}. \tag{8.28}$$

When the perturbation block structure is defined, it implies D-scales with the structure

$$D = \begin{bmatrix} d_1 & 0 & 0 & 0 \\ 0 & d_2 & 0 & 0 \\ 0 & 0 & d_3 & 0 \\ 0 & 0 & 0 & d_3 \end{bmatrix}. \tag{8.29}$$

Initially, the D-scales are set to identity, i.e. $d_1 = d_2 = d_3 = 1$.

The skew μ DK iteration procedure can now be used in an attempt to obtain the skew μ optimal controller for this example.

8.5.1 Iteration 1

Step 1:

With the D-scales initialized to $D = I$, the H_∞ algorithm produced a 6 state controller with an H_∞ norm of $\gamma = 1.277$.

Step 2:

The skew μ upper bound gave the skew μ curve "iter. 1" in figure 8.15, corresponding to a peak skew μ value of $\mu_s = 1.269$.

Step 3:

The frequency-dependent $d_1(\omega)$ and $d_2(\omega)$ from Step 2 were each fitted using a 4^{th} order transfer function. $d_1(\omega)$ and the 4^{th} order transfer function (dotted line) are shown in figure 8.17 and labelled 'iter. 2". d_2 is not shown because it was found that $d_1 \approx d_2$.

8.5.2 Iteration 2

Step 1:

The H_∞ software provided a 22^{nd} order controller with an H_∞ norm of .911.

Step 2:

The controller gave a peak skew μ for this iteration of 0.887.

However, this is not the cumulative value of the skew μ bound. This is the skew μ bound for the matrix M_s, where the previous bound on skew μ has been wrapped into the varying section of M to form M_s. Thus the cumulative value of skew μ is the value sought. In this case, the cumulative product of the peak skew μ bounds is 1.126. This is skew μ at this iteration. As more iterations are done, the closer skew μ for the iteration approaches 1. Eventually more iterations will not affect the cumulative value of skew μ and the level of performance for the skew μ controller can not be significantly improved.

Step 3:

The resulting D-scales were only slightly changed from the previous iteration as can be seen from figure 8.17, for the line denoted as "iter. 3".

8.5.3 Iteration 3

Step 1:

The use of the D-scales from the previous iteration did not improved the H_∞ norm. It fell back slightly to .956. Given the small change in D-scales and the increase in the H_∞ norm, it was decided to end the iterations at this point. The skew μ performance for this system is 1.0665. This implies the system has about 7% less disturbance than originally specified for the performance perturbation, with system parameter perturbations as originally specified at the beginning of the problem.

8.5.4 The Advantage Over μ Designs

The difference between the results of this example and the original example is that skew μ says:

> "For system perturbations as defined, the system performance is about 7% less than originally specified. Reduce the performance expectation by 7% to guarantee that instability cannot occur ."

The μ bound (assuming it gave the same numerical value) would be saying:

> "For system perturbations as defined, one must reduce the original expectation on performance *and the size of the system perturbations (i.e. all perturbations)* by about 7% to guarantee stability ."

As previously pointed out. Many/most times system parameters are are not arbitrarily changeable. In the case of the distillation column example, one would sincerely doubt that system parameter perturbation could be re-scaled to satisfy some desired performance object. Thus, one can see that the skew μ controller design tool potentially provides more benefit than the μ controller design tool for real world applications.

As the skew μ control design tool becomes available to the control design engineers, it is reasonable to expect that it will encounter a wide range of applications. This example has been specifically chosen because the example is a well known and easily verifiable by other engineers looking to use skew μ tools.

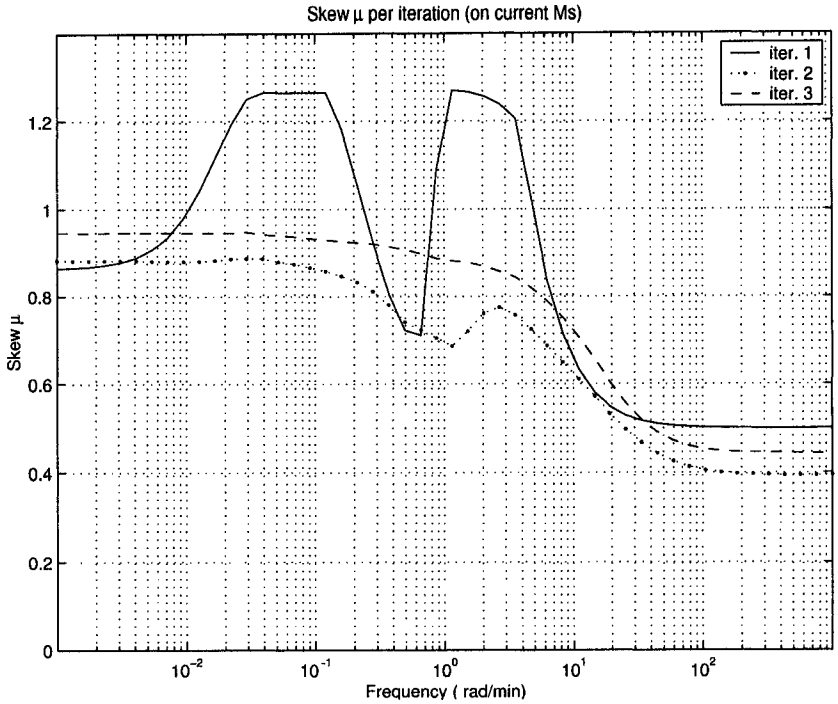


Figure 8.15: Skew μ from each iteration-note that M_s is not the same for each iteration

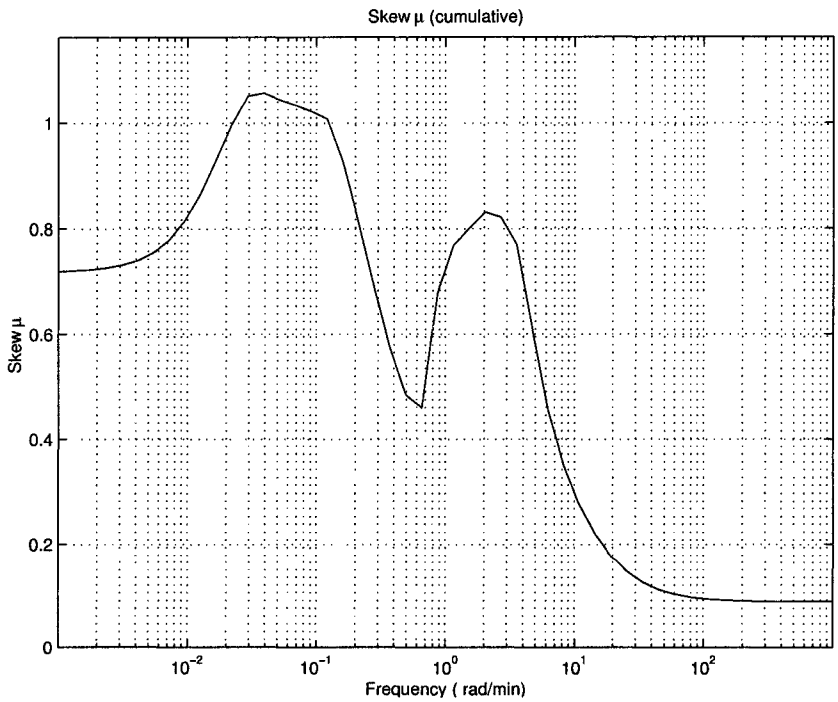


Figure 8.16: Skew μ for the system (cumulative result of 3 iterations)

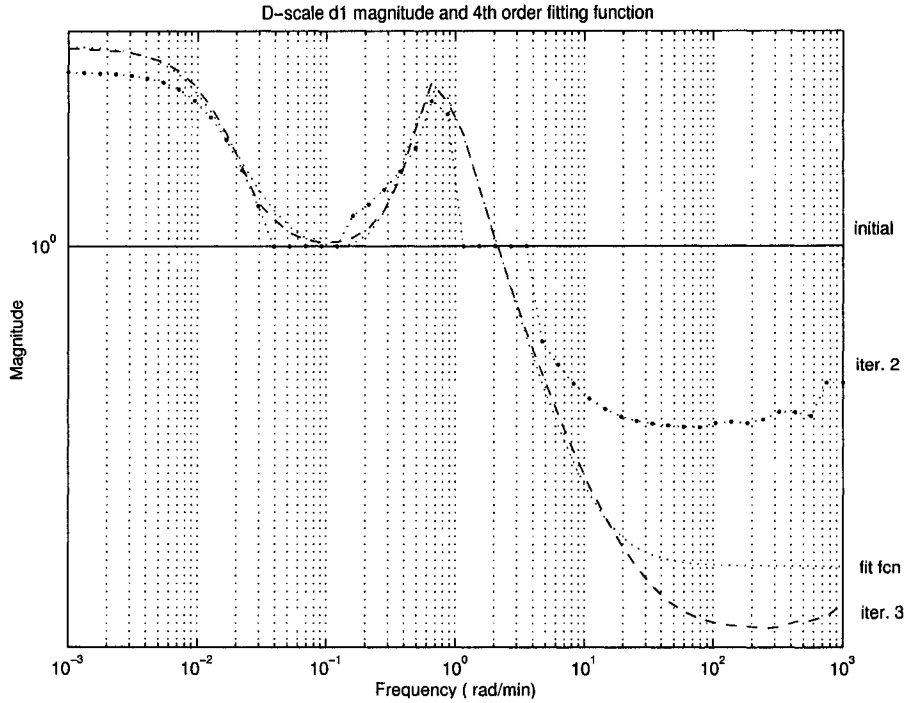


Figure 8.17: D-scale d_1 for the 3 iterations, with a 4^{th} order fit

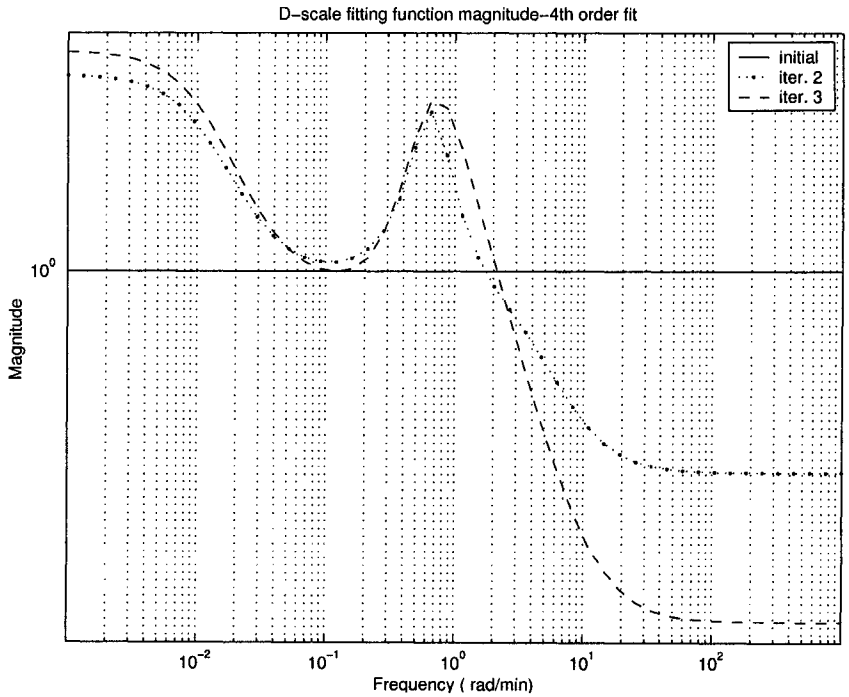


Figure 8.18: D-scale d_1 4^{th} order fitting functions for the iterations

8.5.5 Comments on the Controller Design of the Example

The system of interest was a closed loop system with its controlled output feeding back to its summing junction, as shown in figure 8.12. The input-output pair w, z represents an extraneous disturbance injected into the system (w) versus the error it causes (z) in the output. The desire in designing this controller is to minimize the gain from w to z while maintaining the range of perturbations for the other uncertainties. This is the definition of the "performance" desired from the example. The objective of the design, to guarantee stable performance under all specified conditions, was not met.

The results of this example can be stated in a succinct list:

- A controller has been designed using a systematic procedure.
- For the perturbations given and the performance specification requested, the controller (as designed) can not meet the specification.
- If the controller performance (disturbance rejection) specification is reduced by approximately 7%, then the designed controller will be able to guarantee stability under all other assumptions given.

It is certainly possible for one to "poke around" and "design" a better controller. However, this methodology provides a systematic way to obtain a *reasonable* controller in a finite amount of design time.

One can see in figures 8.17 and 8.18 that the change in D-scales from one iteration to the next becomes very similar to the fitting function, hence the room for improvement in the D-scales was decreasing rapidly.

Additionally, in figure 8.15 one can see the second iteration provides significant improvement. However, as iterations continue, the skew μ bound based on each subsequent M_s will move towards a value of 1. This implies that more iterations will not improve the controller performance. From the second to the third iteration, the distance between the bound and

the constant value '1' has been greatly reduced. From the cumulative plot in figure 8.16 compared to figure 8.15 one can see the cumulative peak occurs in an area where the third iteration result is already near to one. Increased iterations would have minimal impact in reducing the peak value in figure 8.16.

Ultimately these results reduce down to a recommendation to accept the skew μ controller achieved at iteration 3, and try to reduce the potential for disturbance as a way to guarantee stability, rather than continuing the controller iteration procedure.

Although the example problem is a small representation of the capability of the tool, it serves to illuminate the tools usefulness by highlighting its primary characteristic of improving the disturbance rejection of a control system design in the presence of uncertainty.

Chapter 9

Concluding Remarks

This dissertation concludes with a summary of major points, association with related works, and a brief discussion of potential future directions.

9.1 Summary

The skew μ problem has existed in stasis since the early years of robust controls, in part because the concept could easily be understood once the concept of μ was grasped. Secondly, skew μ research was ignored because partial results could be achieved by iteratively applying μ tools and some variable manipulation. This method was cumbersome and time consuming, at best.

As the need for skew μ solutions became more critical, some interest in skew μ surfaced ([22], [25]). These works implied solutions to the skew μ problem, but did not provide detailed solutions or computational methods. The work represented by this dissertation builds on the previous works through the addition of new theory, as well as computationally through the development of new software tools.

It has been shown that skew μ inherits the properties of μ in most aspects, and is therefore computationally hard. Like μ, the numerical approximation to skew μ comes by the way of bounds. The bounds of skew μ bear a similarity to those of μ in their derivation, but depart significantly in their computation. To this end software routines have been developed in Matlab® which integrate into the existing μ analysis package to compute bounds for skew

μ. A code listing for these routines exist in a separate Colorado State University document to prevent them from being lost.

To itemize the developments of this project that have made a positive contribution to the body of robust control knowledge, this list is given.

- Developed a theoretical skew μ lower bound.
- Implemented a numerical computation for the skew μ lower bound.
- Developed a theoretical skew μ upper bound.
- Implemented a numerical computation for the skew μ upper bound using an iterative procedure.
- Developed/enhanced the theoretical skew μ upper bound as an LMI.
- Provided schemes for converting scaling matrices between those used for calculating the upper bound via maximal singular value and those used in calculating the upper bound via an LMI.
- Implemented a numerical computation for the skew μ upper bound using an LMI.
- Developed an improved form of B&B as applied to the frequency augmentation problem in robust control.
- Power system analysis improvements:
  - Improved B&B techniques as applied to power system
  - Improved computational speed of analysis for power systems couched in the robust control format.
  - Provided H_∞ control design method for PSS via skew μ synthesis.
  - Provided a one-shot method for identifying a point where worst case stability problems are likely to occur.

- Developed a D-K iteration style procedure for a skew μ controller.
- Implemented the D-K iteration skew μ controller concept in software.

Of this itemized list, the most significant developments are the theoretical and computational skew μ upper and lower bounds. It is hoped that the computational tools will enjoy a wide range of deployment in practical applications. As the computational tools experience higher usage, we believe the interest in the theoretical results will blossom also.
Looking forward in time this is where the development of skew μ is expected to advance.

9.2 Future Direction

9.2.1 Branch and Bound Improvements

In this work two improvements of B&B as applied to μ analysis have been applied. The first being the reuse of D-scales and the $\|\cdot\|_2$ norm to provide faster calculation of an upper bound. The second was application of an adaptive learning rate to the branching algorithm. The inclusion of a learning rate is the simplest of adaptive techniques. Certainly there is room for other adaptive techniques in the area of B&B applied to μ analysis. Beyond these, intelligent observation coupled with heuristic algorithms may provide some yet unforseen result of an advantageous nature.

9.2.2 LMI Upper Bound

The skew μ LMI upper bound has been derived and software created for its computation using the Matlab® LMI Toolbox. Unfortunately, for large problems such as power systems problems, the LMI solver becomes overwhelmed. More powerful LMI solvers exist, in particular the μ LMI solver contained in the μ Tools Toolbox seems to perform quite well. Since the skew μ LMI computation was implemented directly to a general LMI solver, there is probably significant room for exploiting the properties of skew μ in order to obtain a more efficient, purpose built LMI solver.

9.2.3 Power Systems

This project has made some inroads on the robust analysis of power systems. However, the enormity of power systems problems leaves significant room for improvement. Within each individual generation system (i.e. one generator) there are many possible parameter variations and unmodeled dynamics. These systems are well known and the accepted models are unlikely to change. Some reduction in complexity is acceptable. However, beyond acknowledged means of reduction, the governing bodies of power generation will not accept the methods. Advancement in robust power generation modeling and analysis must be developed in either the way perturbations are quantified for the system, or in the computational scheme used for computing the analysis. Beyond this, modifications to power analysis must be rigorously demonstrated before they will be accepted.

9.2.4 Improved Bounds Computation

Skew μ computations were developed out of μ computations. Part of this was necessary to maintain the framework of the existing Matlab® product, however, it is entirely likely that the algorithms could be improved if developed on their own, apart from the existing routines.

For that matter, algorithms are continually evolving (as long as there are mathematicians) and the current routines existing in the μ analysis toolbox may be the subject of improvement in the future.

9.2.5 More Design Examples

Providing a superior product does not insure success in the market place. If the results of this dissertation are not distributed to the mainstream, then skew μ may continue to languish until another researcher decides to do something about it, again.

For the techniques of this dissertation to advance requires that they be applied to mainstream problems (with success) and published. When the results become apparent within

the mainstream of control engineering, then the techniques will flourish to the point of commercialization.

This is the next serious point for skew μ.

9.2.5.1 Integration into Matlab®Toolbox

Integration into the Matlab®μ tools toolbox requires two things to happen. First, significant testing and bug correction needs to take place on the current versions of skew μ software to bring it up to commercial standards. Secondly, the commercial requirements for submitting and integrating the skew μ tools into the Matlab®μ tools toolbox need to be explored. The long term vision for this project (i.e. completion of this document is not seen as the endpoint) is to bring the software tools of this project to a sufficient state such that they can be included in a future version of the Matlab®μ tools toolbox.

Appendix A

Useful Definitions and Information

A.1 Definitions

Affine Function A function $f : \mathcal{S} \rightarrow \mathcal{T}$ is *affine* if $f(x) = f_0 + T(x)$ where $f_0 \in T$ and $T : \mathcal{S} \rightarrow \mathcal{T}$ is a linear map, i.e.,

$$T(\alpha_1 x_1 + \alpha_2 x_2) = \alpha_1 T(x_1) + \alpha_2 T(x_2) \tag{A.1}$$

for all $x_1, x_2 \in \mathcal{S}$ and $\alpha_1, \alpha_2 \in \mathbb{R}$

B&B (Branch and Bound) A B&B algorithm is an advanced form of gridding a parameter space. It is a general iteration technique that is applicable to optimization problems, where domain dependant upper and lower bounds are computed. B&B partitions the domain to get better bounds. The finer the partition of the domain of optimization, the smaller the gap between the upper and lower bounds.

Ball, B Let $\|\Delta\|$ be a vector norm on the vector space Δ, with $r > 0$. The ball of radius r is the set $\mathbf{B}\Delta \triangleq \{x \in \Delta : \|x\| \leq r\}$

Convex Sets A set $\mathcal{S}$ in a linear vector space is said to be *convex* if

$$\{x_1, x_2 \in \mathcal{S}\} \Rightarrow \{x \triangleq \alpha x_1 + (1-\alpha)x_2 \in \mathcal{S} \; \forall \; \alpha \in (0,1)\}. \tag{A.2}$$

Detectable This is a weaker form of the notion of observable. A system is said to be detectable if any un-observable modes are stable.

Froebenius Matrix Norm (Euclidean Norm) This is the square root of the sum of the squared element magnitudes

$$\|A\|_F = \sqrt{\sum_{i,j} |a_{ij}|^2} = \sqrt{tr(A^H A)}, \tag{A.3}$$

where tr is the trace, the sum of the diagonal elements, and A^H is the complex conjugate transpose of A.

In what follows, let A be an $n \times n$ matrix with complex elements and with eigenvalues $\lambda_i(A)$ $(i = 1,2,\ldots,n)$. Two matrix norms are employed. The first is the Frobenius norm. The other matrix norm is the spectral norm:

$$\|A\|_s = [\rho(AA^H)]^{\frac{1}{2}} = \overline{\sigma}(A) \tag{A.4}$$

where $\rho(A)$ is the spectral radius

$$\rho(A) = max_i|\lambda_i(A)| \tag{A.5}$$

Because of the relation

$$\frac{1}{\sqrt{n}}\|A\|_F \leq \|A\|_s \leq \|A\|_F, \tag{A.6}$$

where $\|\cdot\|_s$ is the spectral norm, a substantial reduction of either norm will involve a reduction of the other. Clearly it is desirable to scale a matrix prior to performing computations.

Hermitian Matrices From [28] and [47], the following are the definition and some properties of Hermitian matrices:

A matrix is Hermitian if $A = A^H$, where $A^H = \bar{A}^T$.

A matrix is skew-Hermitian if $A = -A^H$.

$A+A^H, AA^H$, and A^HA are all Hermitian for all $A_n \in \mathbb{C}$.

If $A = A^H$, then for all complex vectors x, the number x^HAx is real.

If A, B are Hermitian, then $aA+bB$ is Hermitian for all real scalars a, b.

$A - A^H$ is skew-Hermitian for all $A_n \in \mathbb{C}$.

If A is Hermitian, then jA is skew-Hermitian.

Every eigenvalue of a Hermitian matrix is real.

Induced Norm-L_{2i} Consider the equation

$$y = Ax. \tag{A.7}$$

The gain of the equation may be defined as $\frac{\|y\|_p}{\|x\|_p}$. The maximum gain for all possible input directions is given by the induced norm, which is defined as

$$\|A\|_{ip} \triangleq \max_{x \neq 0} \frac{\|Ax\|_p}{\|x\|_p}, \tag{A.8}$$

where x_p is the vector p norm. For the case of the $2i$ norm,

$$\|A\|_{i2} = \overline{\sigma}(A) = \sqrt{\rho(A^HA)}, \tag{A.9}$$

also known as the singular value or spectral norm.

Linear Matrix Inequality (LMI) A linear matrix inequality has the form

$$F(x) \triangleq F_0 + \sum_{i=1}^{m} x_i F_i > 0, \tag{A.10}$$

where the vector $x \in \mathbb{R}^m$ is the variable and the symmetric matrices $F_i = F_i^T \in \mathbb{R}^{n \times n}, i = 0, \ldots, m$ are specified. The function $F(x)$ is an affine function mapping a finite dimensional vector space $\mathbb{V}$ to the set $\mathbb{S} \triangleq \{G | \exists\, n > 0 \textit{ such that } G = G^T \in \mathbb{R}^{n \times n}$. The inequality symbol in A.10 implies that $F(x)$ is positive-definite, or

$$u^T F(x) u > 0 \tag{A.11}$$

for all nonzero $u \in \mathbb{R}^n$.

Matrix Norm A norm on a matrix $\|A\|$ is a matrix norm if it obeys the following properties.

The following properties are true for any norm on a matrix.

- Non-negative, i.e. $\|A\| \geq 0$
- Positive, i.e. $\|A\| = 0 \Leftrightarrow A = 0$
- Homogenous, i.e. $\|\alpha \cdot A\| = |\alpha| \cdot \|A\|$ for all complex scalars α
- Triangle inequality $\|A + B\| \leq \|A\| + \|B\|$

In addition, the following property makes a norm on a matrix a matrix norm.

- Sub-Multiplicative property $\|AB\| \leq \|A\| \cdot \|B\|$

Performance The degree to which an item functions as intended. In the case of robust control, robust performance requires nominal plant stability, and that some performance condition (typically $\mu < 1$) be met for all possible plants, including the worst case condition. Possible plants are quantified in terms of their uncertainty condition.

Perturbation As used in this dissertation is interchangeable with *uncertainty*. The differences between the model and the actual system are termed model perturbations. Typically these take place in one of two forms; parameter variations, or real uncertainty; unmodeled dynamics or complex uncertainty.

Robust System stability is said to have the property of being *robust*, if it holds for the nominal model and the nominal model plus perturbations within the stated range. Alternatively, when applied to controls, a control system is said to be robust if it is insensitive to differences between the actual system and the model of the system which was used to design the controller.

Singular Value Decomposition Any matrix $A \in \mathbb{C}_{m \times n}$ can be factored into a singular

value decomposition

$$A = U\Sigma V^H, \tag{A.12}$$

where the $m \times m$ matrix U and the $n \times n$ matrix V are unitary. The $m \times n$ matrix Σ contains a diagonal matrix Σ_A of real, non-negative singular values, σ_i, arranged in a descending order, and a complementary 0 matrix to complete the dimension requirements. Thus,

$$\Sigma = \begin{bmatrix} \Sigma_A \\ 0 \end{bmatrix}, m \geq n \tag{A.13}$$

or

$$\Sigma = \begin{bmatrix} \Sigma_A & 0 \end{bmatrix}, n \geq m \tag{A.14}$$

and

$$\Sigma_A = diag\{\sigma_1, \sigma_2, \ldots, \sigma_k\}; k = \min(m,n). \tag{A.15}$$

Additionally,

$$\overline{\sigma} \triangleq \sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_k \triangleq \underline{\sigma}, \tag{A.16}$$

where $\overline{\sigma}$ is called the maximal singular value and $\underline{\sigma}$ is called the minimal singular value.

An SVD is not unique (i.e. different choices of U and V). However, the non-zero SV's are unique, and may be found as the square root of the non-zero eigenvalues of A^HA or AA^H, i.e.

$$\sigma_i(A) = \sqrt{\lambda_i(A^HA)} = \sqrt{\lambda_i(AA^H)}. \tag{A.17}$$

Skew μ See page 39 for details.

The skewed structured singular value $\mu_s(M)$ of a matrix $M \in \mathbb{C}^{m \times n}$ with respect to a block structure $\mathcal{K}_f(m_{r_f}, m_{c_f}, m_{C_f})$ and $\mathcal{K}_v(m_{r_v}, m_{c_v}, m_{C_v})$ is defined as:

$$\mu_s(M) = \frac{1}{\min\limits_{\Delta \in W_{\mathcal{K}_f,\mathcal{K}_v}} \{\overline{\sigma}(\Delta_v) |\ det(I - \Delta M) = 0 for\ structured\ \Delta\}}$$

with $\mu_s(M) = 0$ if no $\Delta \in W_{\mathcal{K}_f,\mathcal{K}_v}$ solves $det(I - \Delta M) = 0$, and $\mu_s(M) = \infty$ if $\exists$ a $\begin{bmatrix} \Delta_f & 0 \\ 0 & 0 \end{bmatrix} \in W_{\mathcal{K}_f,\mathcal{K}_v}$ solving $\det(I - \Delta M) = 0$.

spectral radius This is the eigenvalue with largest magnitude from the set of eigenvalues of a given matrix. In particular for a matrix A, $\rho(A)$ denotes the spectral radius, and can be defined mathematically as

$$\rho(A) = max|\lambda(A)|. \tag{A.19}$$

The real spectral radius is defined as the real valued eigenvalue with largest magnitude from the set of real eigenvalues of a given matrix. This is defined mathematically as

$$\rho_R(A) = max\{|\lambda(A)| : \lambda \; is \; a \; real \; eigenvalue \; of \; A\}. \tag{A.20}$$

In the case where no real valued eigenvalue exists, then

$$\rho_R(A) = 0. \tag{A.21}$$

stabilizable This is a weaker form of the notion controllable which includes the requirement that uncontrollable modes are stable.

Structured Singular Value (SSV)- μ μ, is the stability requirement, where the need is to find the smallest structured Δ (measured in terms of $\overline{\sigma}(\Delta)$) which makes $det(I - M\Delta) = 0$; then $\mu(M) = \frac{1}{\overline{\sigma}(\Delta)}$. Formally stating this:

$$\mu(M) = \frac{1}{\min\limits_{\Delta \in X_{\mathcal{K}}} \{\overline{\sigma}(\Delta) |\; det(I - \Delta M) = 0 for \; structured \; \Delta\}}$$

where values of $\mu(M) \geq 1$ provide the possibility of becoming unstable, and values of $\mu(M) < 1$ are stable, for $\overline{\sigma}(\Delta) \leq 1$ (and $\mu(M) = 0$ if no Δ solves $det(I - M\Delta) = 0$).

Uncertainty See *Perturbation*

REFERENCES

[1] P.M. Anderson and A.A. Fouad. *Power System Control and Stability.* IEEE Press, 1994.

[2] Gary Balas, John Doyle, Keith Glover, Andrew Packard, and Roy Smith. The μ analysis and synthesis toolbox. MathWorks and MUSYN, 1991.

[3] B. Barmish, P. Khargonekar, Z. Shi, and R. Tempo. Robustness margin need not be a continuous function of the problem data. *Systems & Control Letters*, 15:91–98, 1989.

[4] Carolyn Beck and John C. Doyle. Mixed μ upper bound computation. In *Proceedings of the 31st Conference on Decision and Control*, pages 3187–3192, 1992.

[5] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM, 1994.

[6] Richard D. Braatz, Peter M. Young, John C. Doyle, and Manfred Morari. Computational complexity of μ calculation. *IEEE Transactions on Automatic Control*, 39:1000–1002, 1994.

[7] Ron W. Daniel, Basil Kouvaritakis, and Haniph Latchman. Principal direction alignment: A geometric framework for the complete solution to the μ problem. *IEE Proceedings, Part D*, 133(2):45–56, March 1986.

[8] M. Djukanovic, M. Khammash, and V. Vittal. Structured singular value theory based stability robustness of power systems: Parti-robust analysis and part ii controller sysnthesis. In *Proceedings of the 36th Conference on Decision and Control*, pages 2702–2707, 1997.

[9] M. Djukanovic, M. Khammash, and V. Vittal. Applicaiton of the structured singular value theory for robust stability and control analysis in multimachine power systems, part i: Framework development. *IEEE Transactions on Power Systems*, 13:1311–1316, 1998.

[10] M. Djukanovic, M. Khammash, and V. Vittal. Applicaiton of the structured singular value theory for robust stability and control analysis in multimachine power systems, part ii: Numerical simulation and results. *IEEE Transactions on Power Systems*, 13:1317–1322, 1998.

[11] John Doyle. Analysis of feedback systems with structured uncertainty. *IEE Proceedings, Part D*, 129(6):242–250, November 1982.

[12] John C. Doyle. Lecture notes on advances in multivariable control. ONR/Honeywell Workshop, Minneapolis, October 1984.

[13] John C. Doyle. Structured uncertainty in control system design. In *Proceedings of the 24th Conference on Decision and Control*, pages 260–265. IEEE, 1985.

[14] John C. Doyle, Bruce A. Francis, and Allen R. Tannenbaum. *Feedback Control Theory*. Macmillan, New York, 1992.

[15] John C. Doyle, Keith Glover, Pramod Khargonekar, and Bruce Francis. State space solutions to $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control problems. *IEEE Transactions on Automatic Control*, 8:831–847, 1989.

[16] Micheal K. H. Fan and André L. Tits. Characterization and efficient computation of the structured singular value. *IEEE Transactions on Automatic Control*, AC-31:734–743, 1986.

[17] Micheal K. H. Fan, André L. Tits, and John C. Doyle. Robustness in the presence of joint parametric uncertainty and unmodeled dynamics. In *Proceedings of the American Control Conference*, pages 1195–1200, 1988.

[18] Micheal K. H. Fan, André L. Tits, and John C. Doyle. Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics. *IEEE Transactions on Automatic Control*, AC-36:25–38, 1991.

[19] G. Ferreres and V. Fromion. Robustness analysis using the ν tool. In *Proceedings of the 35th Conference on Decision and Control*, pages 4566 –4570, 1996.

[20] G. Ferreres, V. Fromion, and G. Scorletti. Advanced computation of the robustness margin. In *Proceedings of the 35th Conference on Decision and Control*, pages 4580–4584, 1996.

[21] G. Ferreres and M. M'Saad. Direct computation of the maximal ssv over the frequency range using the ν tool. In *Proceedings of the 33rd Conference on Decision and Control*, pages 2147 –2148, 1994.

[22] Gilles Ferreres. *A Practical Approach to Robustness Analysis with Aeronautical Applications*. Kluwer Academic/Plenum Publishers, New York, 1999.

[23] P. Gahinet, A. Nemirovski, A. Laub, and M. Chilali. *LMI Control Toolbox*. The Math Works Inc, Natick, Mass., 1995.

[24] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP Completeness*. W. H. Freeman, New York, 1979.

[25] Sonja Glavaški and Jorge E. Tierno. Advances in worst case h_∞ performance computation. *Proceedings of the 1998 IEEE Conference on Control Applications*, 1:668–673, 1998.

[26] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 1987.

[27] Rod Holland and Peter Young. A skew μ lower bound. In *Proceedings of the American Control Conference*, pages 2753–2758, 2002.

[28] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, New York, 1985.

[29] Marija Ilic and John Zaborsky. *Dynamics and Control of Large Electric Power Systems*. Wiley, New York, New York, 2000.

[30] Mustafa Khammash. Private Communication, 2001.

[31] P. Kundur. *Power System Stability and Control*. McGraw-Hill, 1993.

[32] P. Kundur, G.J. Rogers, and D.Y. Wong. *Extended Transient-Midterm Stability Program Package: Version 2.0 Users Manuals*. EPRI EL-6648, New York, New York, 1989.

[33] A. Nemirovski and P. Gahinet. The projective method for solving linear matrix inequalities. In *Proceedings of the American Control Conference*, pages 840–844, 1994.

[34] Nesterov, Yu, and A. Nemirovski. *Interior Point Polynomial Methods in Convex Programming: Theroy and Applications*. SIAM Books, Philidelphia, Pennsylvania, 1994.

[35] Matthew P. Newlin and Peter M. Young. Mixed μ problems and Branch and Bound techniques. *International Journal of Robust and Nonlinear Control*, 7:145–164, 1997.

[36] E. E. Osborne. On preconditioning of matrices. *Journal of the Association for Computing Machinery*, 7:338–345, 1960.

[37] Andrew Packard and John C. Doyle. Robust control of multivariable and large scale systems. Technical report, Honeywell Systems and Research Center, 1988.

[38] Andrew Packard and John C. Doyle. Structured singular value with repeated scalar blocks. In *Proceedings of the American Control Conference*, pages 1213–1218, 1988.

[39] Andrew K. Packard and John C. Doyle. The complex structured singular value. *Automatica*, 29:71–109, 1993.

[40] Andrew K. Packard and Pradeep Pandey. Continuity properties of the real/complex structured singular value. *IEEE Transactions on Automatic Control*, AC-38:415–428, 1993.

[41] Andy Packard, Micheal K. H. Fan, and John C. Doyle. A power method for the structured singular value. In *Proceedings of the 27^{th} Conference on Decision and Control*, pages 2132–2137, 1988.

[42] R. Redheffer. Inequalities for a matrix riccati equation. *Journal of Mathematics and Mechanics*, 8(3), 1959.

[43] Michael Safonov. Stability margins for diagonally perturbed multivariable feedback systems. *IEE Proceedings, Part D*, 129:251–256, 1982.

[44] Carsten Scherer and Siep Weiland. *Linear Matrix Inequalities in Control V3.00.* Self Published, WWW, 2000.

[45] A. Sideris. Elimination of frequency search from robustness tests. *IEEE Transactions on Automatic Control*, 37:1635–1640, 1992.

[46] Sigurd Skogestad and Ian Postlethwaite. *Multivariable Feedback Control.* John Wiley & Sons, New York, 1996.

[47] Gilbert Strang. *Linear Algebra and Its Applications.* Harcourt Brace Jovanovich, San Diego, 1988.

[48] Vijay Vittal. Private Communication, 2001.

[49] Peter M. Young. Private Communication, 2001.

[50] Peter M. Young. *Robustness with Parametric and Dynamic Uncertainty.* PhD thesis, California Institute of Technology, 1993.

[51] Peter M. Young. Controller design with real parametric uncertainty. *International Journal of Control*, 65:469–509, 1996.

[52] Peter M. Young. Structured singular value approach for systems with parametric uncertainty. *International Journal of Robust and Nonlinear Control*, 7:653–680, 2001.

[53] Peter M. Young and John C. Doyle. A lower bound for the mixed μ problem. *IEEE Transactions on Automatic Control*, 42:123–128, 1997.

[54] Peter M. Young, Matthew P. Newlin, and John C. Doyle. Computing bounds for the mixed μ problem. *International Journal of Robust and Nonlinear Control*, 5:573–590, 1995.

[55] Xuechun Yu. Robustness analysis and controller design for static var compensators in power systems. Technical report, Iowa State University Electric Power Research Center, 1999.

[56] Kemin Zhou, John Doyle, and Keith Glover. *Robust and Optimal Control.* Prentice Hall, New Jersey, 1996.

[57] Chuanjiang Zhu. *Robustness Analysis for Power Systems Based on the Structured Singular Value Tools and the* ν *Gap Metric.* PhD thesis, Iowa State University, 2001.