Extensions to the Structured Singular Value

Thesis by
Sven H. Khatri

In Partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy

California Institute of Technology
Pasadena, California
1999
Submitted October 2, 1998

To My Parents

Acknowledgements

First of all, I would like to thank my advisor, Professor John C. Doyle, for giving me the opportunity to pursue my doctoral degree with such an exciting and dynamic group at CalTech. Additionally, his insight, perspective of the big picture, and drive to look for new problems have opened my eyes forever. I can honestly say that I know less now than I thought I knew when I first came to CalTech.

I would also like to thank my fellow group mates during my stay for all of the interesting conversations, technical discussions, and widening my scope of the world. I am particularly grateful to Pablo Parrilo for all of our meanderings through research, our common terminology, endless hours of Truco, the beauty of MATLAB proofs, the art of bicycle repair, and most of all, helping me find the thank God bucket on the climb through grad school. I owe much to Sanjay Lall for all of the pep talks through the home stretch. I am indebted to Matt Newlin for his unique perspective, flair for details, and his patience. We have shared the mountains, valleys, and an outrageous obsession for gear. I would like to thank Xiaoyun Zhu for her effort and help in dealing with deadlines.

There have been many people outside of work with whom I have shared the lighter and darker sides of life. Many thanks to my many roommates at 165N for making sure I didn't go hungry, to Ben for making me carry heavy packs to the Sierra, to Tom for the countless dinners at B.C., to Jan for being Jan, to Dale for giving me something for which to strive, to my surgeons for trying to keep me in one piece, to the Latins for accepting me as one of their own, and to the Grinch for all his stories.

My deepest gratitude goes to my family, Mom, Dad, Sunil, Sheila, Moti, Dada, PadmaPhoi, Smita, Sammy, and Nina, for your constant support, unconditional love, and never ending abuse.

This work was sponsored by the U.S. Department of Engery Office of Energy Research through their Computational Science Graduate Fellowship program, under contract W-7405-Eng-82.

Abstract

There are two basic approaches to robustness analysis. The first is Monte Carlo analysis which randomly samples parameter space to generate a profile for the typical behavior of the system. The other approach is fundamentally worst case, where the objective is to determine the worst behavior in a set of models. The structured singular value, μ, is a powerful framework for worst case analysis. Where $\frac{1}{\mu}$ is a measure of the distance to singularity using the ∞-norm.

Under the appropriate projection, the uncertainty sets in the standard μ-framework that admit analysis are hypercubes. In this work, μ and the computation of the bounds is extended to spherical sets or equivalently measuring the distance to singularity using the 2-norm. The upper bound is constructed by converting the spherical set of operators into a quadratic form relating the input and output vectors. Using a separating hyperplane or the S-procedure, a linear matrix inequality (LMI) upper bound can be constructed which is tighter than and consistent with the standard μ upper bound. This new upper bound has special structure that can be exploited for efficient computation and the standard power algorithm is extended to compute lower bounds for spherical μ. The upper bound construction is further generalized to more exotic regions like arbitrary ellipsoids, the Cartesian product of ellipsoids, and the intersection of ellipsoids. These generalizations are unified with the standard structures. These new tools enable the analysis of more exotic descriptions of uncertain models.

For many real world problems, the worst case paradigm leads to overly pessimistic answers and Monte Carlo methods are computationally expensive to obtain reasonable probabilistic descriptions for rare events.

A few natural probabilistic robustness analysis questions are posed within the μ-framework. The proper formulation is as a mixed probabilistic and worst case uncertainty structure. Using branch and bound algorithms, an upper bound can be computed for probabilistic robustness. Motivated by this approach, a purely probabilistic μ problem is

posed and bounds are computed. Using the existing machinery, the branch and bound computation cost grows exponentially in the average case for questions of probabilistic robustness. This growth is due to gridding an n-dimensional surface with hypercubes.

A motivation for the extensions of μ to other uncertainty descriptions which admit analysis is to enable more efficient gridding techniques than just hypercubes. The desired fundamental region is a hypercube with a linear constraint. The motivation for this choice is the rank one problem. For rank one, the boundary of singularity is a hyperplane, but the conventional branch and bound tools still result in exponential gridding growth.

The generalization of the μ-framework is used to formulate an LMI upper bound for μ with the linear constraint on uncertainty space. This is done by constructing the upper bound for the intersection of an eccentric ellipsoid with the standard uncertainty set. A more promising approach to this computation is the construction of an implicit μ problem where the linear constraints on the uncertainty can be generically rewritten as an algebraic constraint on signals. This may lead to improvements on average to the branch and bound algorithms for probabilistic robustness analysis.

Table of Contents

List of Figures

Chapter 1

Introduction

Nearly all of the modelling of phenomenon in engineering systems is based in the language of mathematics. There is a limitation in the precision with which this language is able to describe reality. Real phenomenon are extremely complex and much of the microscopic phenomenon and dynamics must be ignored or lumped together to construct useful low order tractable engineering models. For example, thermodynamics is an average case description of microscopic phenomenon described more precisely by statistical mechanics. Thermodynamics provides a description of system level phenomenon while ignoring the fine scale details. These details are generally not of engineering interest and have little or no impact on system performance. As a result of the number particles involved and the scale of the fine dynamics is that thermodynamics provides an accurate and simple model of various aggregate phenomenon.

The problem is even worse for systems which do not admit useful first principles models and the models are based upon black box identification methodologies. The models are forced to be imprecise because they are based on finite observation, noisy data, and are restricted to a particular mathematical language like finite dimensional linear time invariant systems. The benefit of this approach is there is a strong connection with reality and can provide low order models for engineering without a deep understanding of the underlying dynamics. The success of this type of modelling depends on the predictive power of the constructed models. For systems that operate in a linear regime, it is much easier to explore and predict the behavior of the system using black box methodologies. This is a major limitation to this paradigm for modelling. If the system is linear and the model is constructed in this manner, then it is less likely to predict spurious phenomenon because of the connection with the physical world.

These simplified models are the basis for system design and prediction.

As systems become more complex, there is an increased reliance upon a synthetic or virtual environment to reduce the cost and time associated with prototypes and physical testing. Although these are necessary stages, the advances in computation and algorithm development shifts the focus of research and development (R&D) towards computer based experiments. An example of this is the interplay between computational fluid dynamics (CFD) and wind tunnel testing. The advances in CFD and computation have outpaced the developments in wind tunnels. Which leads to more of the design and testing stages occur in the synthetic environment rather than physical wind tunnel models. Although the synthetic will not replace the physical for R&D, it plays an increasingly important role.

One of the reasons that the synthetic prototype hasn't replaced the physical prototype is that synthetic experiments can produce erroneous results which are artifices of the mathematical model and not behavior of the actual system. Additionally, physical prototypes are a powerful validation of synthetic predictions. The physical experiments are also useful tool for improving synthetic experiments. Further, physical tests can explore various environments which are not computationally tractable like actual flight testing which may involve aeroelastic phenomenon during radical maneuvers. For this problem there is no hope to make accurate synthetic predictions.

In some disciplines, the synthetic environment has been tremendously successful and essential to the engineering process. The quintessential example is digital VLSI design. Using the Mead-Conway design rules, amazingly complex integrated circuits have been designed with relative ease. Fundamental to the success of digital circuit design is the decoupling of the dynamics of the underlying analog phenomenon for the subcircuits, which extends the predictive power of the simple high level digital models used in design. This synthetic approach hasn't been nearly as successful in analog VLSI where the analog phenomenon are to be exploited. Analog VLSI is not nearly as complex as its' digital counterparts, but there is still great difficulty in getting the designed circuit to perform the desired function. The main problem in predicting the system behavior is the uncertainty in the transistor parameters. By adopting the digital protocol, digital VLSI is relatively insensitive to such variations.

Within the discipline of control design, the fundamental principle is feedback. Feedback is a very powerful and sensitive tool with which a system can achieve performance levels that are otherwise impossible. For complex systems, feedback almost generically drives the system unstable. Great care must be taken when designing controllers to exploit the real

dynamics of the system rather than artifices of the mathematics; otherwise system performance may be degraded or even destabilized by the application of feedback.

For most disciplines approximate models are good enough. When using feedback there is a strong coupling of the dynamics of the system with the dynamics of the controller. This strong coupling is both a blessing and a curse because of the potential for significant modification to the system's characteristics. As a result of the strong coupling, the performance of the controlled system may be sensitive to the gap between the model and reality if the controller is designed poorly. Engineering judgement and experience are critical in guiding the design of controllers.

It is not uncommon for a controller to work in simulation, but fail miserably in practice. This is due to inaccuracies of the model, from sensor and actuator limitation to unmodelled dynamics. If a plant has an unstable pole and the controller is designed for an unstable pole-zero cancellation as the method to stabilize the system, then the stability of the controlled system is infinitely sensitive to the location of the unstable pole. An arbitrarily small error in the location of the unstable pole will result in an unstable system. This is a classic example of poor controller design. There is too much faith in the model which is fundamentally imprecise. For a complicated system, these issues aren't so obvious and a rigorous mathematical approach is required.

These sensitivity issues are at the heart of robust control theory. The general philosophy is to design the controller for a set of plants rather than a specific plant. If the controller works for every system within the set, and the real system is in the set, then the controller will work. Even if the actual system is not in the set, one is more inclined to believe the robustly designed controller will "work" rather than one designed for a single model.

A simpler problem is that of robustness analysis. Given a controller and a set of models, does the controller stabilize and/or meet the performance requirement for every model in the set. This is a simplification of the robust design problem. If this question cannot be answered, then how does one know if the controller designed meets the performance requirement? A further simplified but equivalent robustness analysis problem is: Given a set of models, determine whether the entire set is stable and meets the performance requirement. It turns out that the robustness analysis problem is hard in general but a great deal of progress has been made in the computation of some mathematical formalizations of robustness analysis questions.

In order to ask a robustness question there is a need for a precise mathematical description for a set of models. This set should be rich enough to include real systems. For the classical approaches to robustness analysis, these sets have very simple descriptions. For example, the small gain theorem [49] guarantees stability for the feedback interconnection of two systems if both are stable and the loop gain is less than one. If the gain of the nominal system is γ and the uncertainty is written as a feedback interconnection with the nominal system, then for all stable perturbations with gain less than $\frac{1}{\gamma}$ the system is stable. This is a very crude description for a set of models. This set is likely to be much larger than necessary to cover the real system.

From an engineering perspective, the uncertainty in a system has a remarkable amount of structure. Given a resistor in a circuit, there is uncertainty in the nominal value of the resistor and the associated parasitics because of the imprecision of the manufacturing process. Further, the properties of this resistor may vary as the environment changes like sensitivity to temperature and aging of the component. If the uncertainty of a system was just parametric variation, then the standard small gain approach would provide a very conservative estimate to the robustness properties of the system.

The structured singular value, μ or equivalently $\frac{1}{K_m}$, introduced by Doyle [16] and Safonov [40], respectively, is a generalization and unification of the spectral radius and maximum singular value for matrices. The μ-framework has proven to be a powerful setting for robustness analysis. The μ-framework is sufficiently general to include other robustness analysis approaches as special cases. Further there is enough inherent structure to the framework to admit computation for a variety of uncertainty types. The μ-framework is a powerful and useful generalization and unification of the small gain theorem and state space representations. Further this extension of state space representations provides a natural and powerful language for the description of uncertain systems.

Although computing μ is NP-hard [10], computationally tractable upper and lower bounds exist [47, 30, 34, 45, 48, 6]. NP-hardness of the computation is a property of the problem and not the algorithm. It is a fundamental question that is open in the field of computational complexity to determine the computational consequences of a problem being NP-complete. For a thorough treatment of the subject the reader is referred to Garey and Johnson [19]. It is generally accepted that if a problem is NP-complete, then it cannot be computed in polynomial time as a function of the problem size. Usually the computation time grows exponentially. In

practical terms, the exact computation of μ is not tractable in the worst case and practical computation will have to rely upon bounds which are computationally tractable.

For some problems, the gap between the upper and lower bounds may be large. Strategies for reducing the gap are based upon branch and bound (B&B) techniques [32]. The B&B algorithm involves dividing a μ problem into two μ problems and repeating this process until the gap between the bounds is acceptable. Critical to success of these techniques is preventing exponential growth in the number of active μ problems, which is not possible in the worst case. Practical experience indicates that this refinement of the bounds is reasonable for the average case.

The standard μ-framework is a fundamentally worst case paradigm, and the robustness analysis questions that can be answered are also worst case formulations. The resulting answer either guarantees performance for every model in the described set or says that at least one model in the set doesn't meet the performance requirement. Although this worst case paradigm provide interesting theoretical and tractable computational formulations for robustness analysis, ultimately the problems to be addressed are grounded in engineering reality where the worst case view of the world is not necessarily a viable option and forces over engineering, waste, and inefficient design.

Often, real world problems cannot be satisfactorily answered with guarantees; this will lead to overly conservative results. If it is unlikely that the set of destabilizing parameter choices would occur in the age of the universe, then it is good enough from an engineering perspective.

Many robustness analysis questions are probabilistic in nature. Trying to determine the probability of failure of a system or the yield of a manufacturing process are fundamentally probabilistic. Two natural questions arise. First, what is the probability that the system is stable? Second, what is the distribution of performance. Monte Carlo methods can be used to answer these types of questions.

Monte Carlo methods work well for determining the distribution of average events, but do not provide useful information about the tails of the distribution and bounding rare events without excessive computation. The number of Monte Carlo trials necessary to describe rare events is intractable, and even then there are only probabilistic guarantees in the form of confidence levels. For rare events a large number of Monte Carlo simulations are necessary to get reasonable statistical significance for the computed probability for these rare events.

The μ-paradigm is designed for the guaranteed analysis of *extremely*

rare events, the μ framework can be extended to "probabilistic μ," which is not strictly worst case. The strength of this probabilistic formulation of μ is in bounding the tail of the distribution for rare/bad events.

In Chapter 2, probabilistic formulations of μ are motivated from the system context. These formulations are inherently a mixture of probabilistic and worst case uncertainty descriptions. This is a manifestation of guaranteeing performance and/or stability of a system for a particular choice of parameters. Answering nominal stability and performance questions fall within the worst case μ paradigm. From a computation and representation perspective, probabilistic distributions are only permitted for real parametric uncertainty. These formulations will be simplified to a "purely probabilistic μ" problem, for which a μ function is defined and upper and lower bounds on this probability distribution are computed.

In the process of applying branch and bound (B&B) algorithms to μ, additional local information can be extracted about the system. Specifically regions of parameter space can be classified as stable or a bound on the performance can be computed, and a system has a probability of being in this region of parameter space. This results in a bound on the probability that the system is stable or meets a certain performance level. By extending the B&B algorithm, one arrives at a guaranteed bound of a function on parameter space as presented by Zhu [50]. The function presented in [50] is an approximation to probabilistic μ as defined in this work.

An extension of the B&B algorithm leads to bound computation for probabilistic μ . Probabilistic μ seems to be computationally intractable even for low dimensional random problems (≥ 4); the bounds are very conservative for any reasonable computation time. Even though the B&B and probabilistic μ algorithms are trivial to parallelize, there are still fundamental improvements that can be made in the algorithms. The basic problem is dealing with the exponential growth associated with gridding the boundary of singularity.

In an effort to deal with the generic exponential growth of the B&B computation, it is necessary to develop a richer class of worst case uncertainty descriptions for which bounds analogous to the bounds for standard μ can be computed. Traditionally, the enriching of the uncertainty description has meant adding to the possible uncertainty structures, like real parameters and repeated scalars.

The main contribution of this work is the development of mathematical machinery to guarantee non-singularity for regions which are more exotic than hypercubes. Structures like spheres, ellipses, hypercubes with a corner removed, and so on. These tools may be useful in addressing some

of the computation barriers encountered in applying B&B to probabilistic robustness analysis.

Chapter 3 addresses spherically described regions. Given a matrix and a block structure for the uncertainty, $\frac{1}{\mu}$ defines a distance to singularity of $(I - M\Delta)$ in the space of allowable Δ. For the standard μ problem this distance is measured using the ∞-norm. It turns out that an analogous upper bound can be derived if the distance is measured using the 2-norm. This will be referred to as spherical μ. The upper bound is constructed by converting the spherical uncertainty constraint into a quadratic signal constraint and applying a separating hyperplane or S-procedure [27] argument. This basic construction is the foundation for the ensuing generalizations. The consistency of the standard and spherical μ upper bound is proven. Additionally, a small gain interpretation to the bound is developed.

Chapter 4 presents a number of generalizations of spherical μ for which the upper bound also generalizes. These generalizations include the standard μ constructions like repeated scalars, real parameters, and full blocks. It is further extended to ellipsoidal regions, Cartesian products of spheres, inverse spheres, and the intersection of regions. The end result is a generalization of the μ framework for which the standard μ and generalized μ [29] are special cases.

Chapter 5 addresses some computational issues associated with the upper and lower bound computations for spherical μ. The structure of the upper bound computation is very special. Making use of this structure using a generalization of the Perron-Frobenius theory, it is possible to compute this upper bound efficiently without using the standard numerical LMI solver.

Additionally, the necessary conditions for optimality associated with the lower bound are derived. This is to show that much of the work that has been done on the lower bound can be extended but it is not sufficiently interesting to warrant further investigation. It is straightforward to extend to the generalizations of Chapter 4, but is not done here.

Chapter 6 investigates the operator description of the inverse problem for which the spherical μ upper bound is exact. The motivation for this investigation is to understand the gap between two possible methods for constructing upper bounds for spherical constraints on full blocks. These bounds are based upon two different but equivalent μ constructions. For the case where the original full block problem consists of only scalar blocks, one upper bound reduces to the spherical μ upper bound and the other reduces to the standard μ upper bound. This gap is ex-

plained using the relaxation of the uncertainty set for which the upper bound is exact.

Chapter 7 is a return to the problem of exponential growth due to gridding the boundary of singularity which was one of the motivations for the generalizations of μ in the earlier chapters. The goal is to develop new tools for the analysis of regions which are not axially aligned hypercubes, specifically, hypercubes with a linear cut. These regions are motivated by the rank one probabilistic case. Three methods for computing upper bounds are presented, the first is using the spherical μ and the extensions, the second is based upon the implicit formulation of μ, and the third uses a change of variables to construct a conservative μ problem.

Chapter 8 addresses the hierarchical modelling endeavor from a reductionist approach. The motivation for hierarchical modelling is that the model for a system depends on the question it will be used to answer or equivalently, to what it is interconnected. Some of the issues addressed are the distribution of the hierarchy, mathematical language for such modelling, uncertainty representations, and choosing a model from such a hierarchy. Some of the drawbacks, difficulties, and flaws associated with this approach are addressed.

Chapter 9 presents an approach for implicit identification of a system where it is not possible to actively drive any of the interconnection variables. The interconnection point can be driven passively by attaching a known system. In order to construct a model general enough to predict the model of the system for a known arbitrary system attached, it is necessary to have information about half of the interconnection variables.

1.1 Notation

The notation is largely standard. If a and b are real numbers such that $a < b$, then (a, b) is the open interval, $[a, b]$ is the closed interval, and $\{a, b\}$ is a set of two elements containing a and b. The set of real numbers is denoted by $\mathbb{R}$, the set of complex numbers is denoted by $\mathbb{C}$, and the set of integers is denoted by $\mathbb{Z}$.

The *adjoint* $\mathcal{L}^*$ of a linear operator $\mathcal{L}$ is the unique linear operator that satisfies $\langle x, \mathcal{L}(y)\rangle = \langle \mathcal{L}^*(x), y\rangle$, for all x and y. For the special case when $\mathcal{L}(x) = Mx$ where M is a matrix, M^* is the conjugate transpose of M. M^T is the transpose of a matrix. The $n \times n$ identity matrix will be denoted by I_n, and I is used when the dimension follows from the context. $\mathbb{1}_n$ is an $n \times n$ matrix of 1's. $diag[A_1, \cdots, A_n]$ is a block diagonal matrix with A_i

being the i^{th} block on the diagonal. $vec(M)$ is the columns of a matrix M stacked into a vector.

$\rho(\mathcal{L})$ is the spectral radius of a finite dimensional linear operator $\mathcal{L}$ and is defined to be the nonnegative number $\rho(\mathcal{L}) = \max\{|\lambda| : \mathcal{L}(x) = \lambda x, x \neq 0\}$. The largest real eigenvalue and the largest magnitude real eigenvalue for a matrix are denoted by $\overline{\lambda}_r(M)$ and $\rho_r(M)$, respectively. The maximum and minimum singular values of a matrix M are denoted by $\overline{\sigma}(M)$ and $\underline{\sigma}(M)$ respectively.

$Tr(M)$ is the trace of M. $det(M)$ is the determinant of M. The Frobenius norm of a matrix M is defined as $\|M\|_F \triangleq \sum_{ij} |m_{ij}|^2$. A Hermitian matrix $M(= M^* \in \mathbb{C}^{n\times n})$ is said to be positive (semi) definite if $x^*Mx > 0 (\geq 0)$ for all $x(\neq 0) \in \mathbb{C}^n$.

The Kronecker product of two matrices, A and B, is denoted by $A \otimes B$. The Hadamard (or Schur) element by element product of two compatibly dimensioned matrices $A = [a_{ij}]$ and $B = [b_{ij}]$ is denoted by $A \circ B \triangleq [a_{ij}b_{ij}]$. In this paper the Hadamard product is sometimes used to specify matrix structure. For example, to specify that a matrix D is diagonal, the notation $I \circ D = D$ is used.

The Hilbert space of square summable vector-valued signals over $\mathbb{Z}$ is denoted l_2^m, with associated inner product $\langle u, v\rangle = \sum_{n=-\infty}^{\infty} u^*[n]v[n]$. The norm of a signal v is defined as $\|v\| = \sqrt{\langle v, v\rangle}$. The outer product is defined as $\Lambda(u, v) = \sum_{n=-\infty}^{\infty} u[n]v^*[n]$ and is a matrix.

1.2 Mathematical Machinery

When dealing with the Hadamard product of matrices, the standard matrix manipulation intuition often fails. Two useful properties of the Hadamard product are presented in Lemmas 1.1 and 1.2.

Lemma 1.1 *If* $A = diag[a_1, \cdots, a_n]$ *and* $B = diag[b_1, \cdots, b_n]$, *then*

$$AXB = X \circ (ab^T).$$

Lemma 1.2 (Schur Product Theorem [22]) *If* $X_i > (\geq) 0$ *and* $X = X_1 \circ X_2$, *then* $X > (\geq) 0$.

The following is a well known result that is useful for converting various nonlinear (convex) optimization problems into linear problems.

Lemma 1.3 (Schur Complement) *[8] The matrix inequality,*

$$\begin{bmatrix} Q & S \\ S^* & R \end{bmatrix} > 0 \tag{1.1}$$

with $Q = Q^*$ and $R = R^*$ is equivalent to

$$Q - SR^{-1}S^* > 0, \; R > 0. \tag{1.2}$$

Solving a problem often entails converting a new problem to a collection of problems which are considered to be solved. The notion of solved has changed in time from closed form solutions to include numerical solutions. Within optimization and control, many problems can be reduced to linear matrix inequalities (LMIs) [8]. The LMI framework is a powerful general optimization tool. Rather than developing unwarranted specialized solvers for problems, general solvers are adequate for proof of concept and algorithm development. Further, if a problem can be reduced to an LMI, it is considered solved. An LMI is defined as

$$F(x) \triangleq F_0 + \sum_{i=1}^{m} x_i F_i > 0,$$

where $x \in \mathbb{R}^m$ is the variable and $F_i \in \mathbb{R}^{n \times n}$ are given symmetric matrices. The existence of a feasible solution can be interpreted as the nonempty intersection of the set given by the affine function $F(x)$ and the self-dual cone of positive definite matrices. A generalized eigenvalue problem (GEVP) takes the form

$$\min_x \{\lambda : \lambda B(x) - A(x) > 0, B(x) > 0, C(x) > 0\} \tag{1.3}$$

where A, B and C are symmetric matrices that depend affinely on x. This is a quasi-convex optimization problem.

LMIs are convex optimization problems that can be solved efficiently in polynomial time. The most effective computational approaches use projective or interior-point methods [28] to compute the optimal solution.

1.3 Linear Fractional Transformation

Within every robust control paradigm there is a class or set of models that are associated with the system being investigated. This set has to be sufficiently rich to cover the behavior of the actual system. The associated

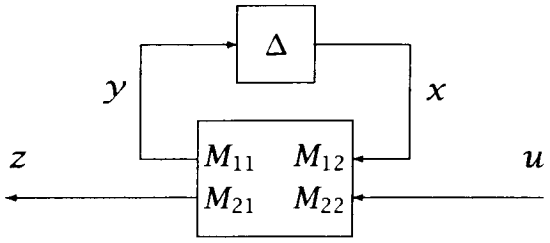

Figure 1.1: Linear Fractional Transformation

mathematical description must be simple and admit computational analysis for the framework to be useful. In this treatment, the set of models are described by a linear fractional transformation (LFT).

A LFT is shown in Figure 1.1. This block diagram representation is equivalent to the equations

$$\begin{aligned} y &= M_{11}x + M_{12}u \\ z &= M_{21}x + M_{22}u \\ x &= \Delta y. \end{aligned}$$

In solving for the map from u to z, x and y are eliminated and the map can be expressed in terms of M and Δ. The resulting map is given by

$$z = (M_{22} + M_{21}\Delta(I - M_{11}\Delta)^{-1}M_{12})u. \tag{1.4}$$

(1.4) is denoted $z = (\Delta \star M)u$ as a shorthand notation. Similarly a lower LFT is defined by $M \star \Delta = M_{11} + M_{12}\Delta(I - M_{22}\Delta)^{-1}M_{21}$.

The LFT $\Delta \star M$ is said to be well-posed if and only if there is a unique solution to the loop equations shown in Figure 1.1. Well-posedness reduces to the invertibility of $(I - M_{11}\Delta)$. If $\Delta \star M$ is well-posed, then the only solution to $(I - M_{11}\Delta)y = 0$ and $(I - \Delta M_{11})x = 0$ is $x = y = 0$. If the system is not well-posed, then there are infinitely many solutions to the loop equations with $\|x\|$ and $\|y\|$ arbitrarily large. In some sense the system is unstable. This is the motivation for the analysis of the loop shown in Figure 1.3. Investigations into this loop are the main emphasis of this work.

Typically, Δ is an element of a set, and guaranteeing that the map is well-posed for the entire set implies some notion of stability and/or robustness. If Δ is the delay operator $z^{-1}I$, then the system shown in Figure 1.1 reduces to the standard discrete time state space representation. Guaranteeing that the LFT is well-posed for the closed unit disk $\subset \mathbb{C}$ is equivalent to $\rho(M_{11}) < 1$ which is the standard stability result.

LFT systems are a natural generalization of state space representations. By allowing Δ to represent more general structures, the LFT system formulation provides a convenient framework to add various types of uncertainty operators, like arbitrary bounded operators, nonlinearities, linear time varying systems, linear time invariant systems, time varying parameters, and static parameters [34] [46] [14], for which essentially all the major state space results can be generalized [5].

The standard additive and multiplicative uncertainty are special cases of the LFT framework. The interconnection of LFT systems can be rewritten as a single LFT system which lends to the generality of the framework. The nature of constructing a single uncertainty block for a LFT model and the distributed nature of uncertainty leads to a block diagonal structure for Δ (see [34] and [49] for examples). In addition to the structure, the uncertainty is also described by a norm constraint which limits the size of the perturbation.

Implicit LFT

An implicit LFT system is described by $0 = (\Delta \star M)w$ as shown in Figure 1.2, where w contains all the system variables. There is no distinction between inputs and outputs. Implicit LFT systems are a generalization of the behavioral framework proposed by Willems [44].

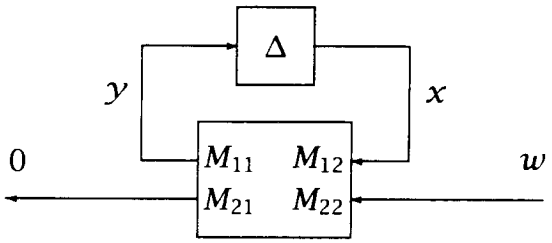

Figure 1.2: Implicit LFT System

Any LFT system can be converted into an implicit LFT system by rewriting $z = M_{21}x + M_{22}u$ as

$$0 = M_{21}x + \begin{bmatrix} M_{22} & -I \end{bmatrix} \begin{bmatrix} u \\ z \end{bmatrix}.$$

The motivation for the behavioral approach is from first principles modeling. For example $F = Ma$, neither F nor a is assumed to be an

input or an output. Once either F or a is defined, both are defined. There is no notion of signal flow. These interconnections can get awkward for input-output models when the signal space of two systems are partitioned into inputs and outputs incompatibly for interconnection. The choice is to avoid the partitioning until necessary.

Interconnection of Implicit LFT Systems

Within the implicit LFT framework, the interconnection of systems is simple. The implicit description of a system describes the equations a system must satisfy,

$$0 = (\Delta_i \star M_i)w_i \text{ where } M_i = \begin{bmatrix} A_i & B_i \\ C_i & D_i \end{bmatrix}.$$

So if two systems are connected, then they still satisfy the same equations, and the interconnection must be defined ($T_1 w_1 + T_2 w_2 = 0$) which defines the intersection of behaviors [44]. So the implicit LFT model of the interconnected system is given by $0 = (\Delta \star M)w$ where:

$$M = \begin{bmatrix} A_1 & 0 & B_1 & 0 \\ 0 & A_2 & 0 & B_2 \\ C_1 & 0 & D_1 & 0 \\ 0 & C_2 & 0 & D_2 \\ 0 & 0 & T_1 & T_2 \end{bmatrix} \tag{1.5}$$

$$\Delta = \begin{bmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{bmatrix} \tag{1.6}$$

Implicit representations can be interconnected, manipulated and reduced without committing to a particular input-output form, which is only relevant to certain applications, and can be derived *a posteriori* if necessary as shown by D'Andrea and Paganini [13].

Integral Quadratic Constraints

The implicit LFT framework also allows our model to include integral quadratic constraints (IQCs)[27]. IQCs are inequalities involving a quadratic form in signal space:

$$\langle \Pi w, w \rangle = \int_0^{2\Pi} w(\omega)^* \Pi w(\omega) d\omega \leq 0 \tag{1.7}$$

where Π is a self adjoint LTI operator. IQCs can be used to define sets in which signals must exist, like defining sets of allowable noise. Given $\Pi(e^{j\omega}) = \Pi(e^{j\omega})^* \in L_\infty \Rightarrow \exists k > 0 \ni kI + \Pi > 0$. By doing a spectral factorization [49] of $kI + \Pi$, we find a $Q : \Pi = kI - Q^*Q$. Defining $P = k^{1/2}I \Rightarrow \Pi = P^*P - Q^*Q$. Equation 1.7 becomes $\| Qw \|_2 \geq \| Pw \|_2$. This can be written as an uncertain implicit equation of the form $(P + \Delta_c Q)w = 0$ where Δ_c is an arbitrary contractive operator [35].

1.4 μ-Framework

The fundamental picture in the μ-paradigm [34] is shown in Figure 1.3. This is motivated by the question of well-posedness of LFT systems. For the purposes of this treatment, M is a matrix and Δ is a matrix in a set of allowable matrices. M can be generalized to a linear operator between signal spaces. Similarly, Δ can be extended to include nonlinearities, arbitrary operators, linear time invariant operators, time varying parameters, and static parameters. The set of allowable Δ is described by block diagonal structure, and a size constraint, $\overline{\sigma}(\Delta) < 1$. For the standard μ problem description, a block of Δ can be full, a repeated complex scalar, or a repeated real scalar. These structures readily admit computation and are commonly encountered as uncertainty descriptions for real systems. The fundamental question in the μ-paradigm is: Are there any nontrivial solutions to the loop equations shown in Figure 1.3 or equivalently is the LFT system well-posed? $\mu(M)$ answers that question.

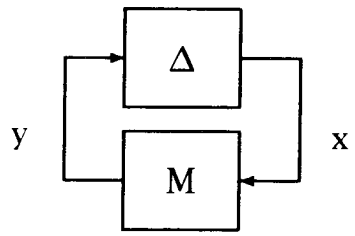

Figure 1.3: Standard Interconnected System

Definition 1.4 *The uncertainty structure is defined by*

$$\Delta \triangleq \{diag[\delta_{r1}I_{k_{r1}}, \cdot, \delta_{rn_R}I_{k_{rn_R}}, \delta_{c1}I_{k_{c1}}, \cdot, \delta_{cn_C}I_{k_{cn_C}}, \Delta_{f1}, \cdot, \Delta_{fn_F}]: \delta_{ri} \in \mathbb{R}, \delta_{ci} \in \mathbb{C}, \text{ and } \Delta_{fi} \in \mathbb{C}^{n_{fi} \times n_{fi}}\} \tag{1.8}$$

and

$$\boldsymbol{B\Delta} \triangleq \{\Delta : \Delta \in \boldsymbol{\Delta}, \overline{\sigma}(\Delta) \leq 1\}. \tag{1.9}$$

With the appropriate projection the uncertainty set in (1.9) defines a hypercube of dimension $n_R + n_C + n_F$.

Definition 1.5 *[49] For $M \in \mathbb{C}^{n\times n}$, $\mu_{\boldsymbol{\Delta}}(M)$ is defined as*

$$\mu_{\boldsymbol{\Delta}}(M) \triangleq \frac{1}{\min\{\overline{\sigma}(\Delta) : \Delta \in \boldsymbol{\Delta}, \det(I - M\Delta) = 0\}} \tag{1.10}$$

unless no $\Delta \in \boldsymbol{\Delta}$ makes $I - M\Delta$ singular, in which case $\mu_{\boldsymbol{\Delta}}(M) \triangleq 0$.

Alternatively (1.10) can be written as

$$\mu_{\boldsymbol{\Delta}}(M) = \sup_{\Delta\in\boldsymbol{B\Delta}} \rho_r(M\Delta) = \sup_{\Delta\in\boldsymbol{B\Delta}} \overline{\lambda}_r(M\Delta) \tag{1.11}$$

For special structures of $\boldsymbol{\Delta}$, the following commonly used properties of matrices follow as special cases:

$$\boldsymbol{\Delta} = \delta_c I \Rightarrow \mu_{\boldsymbol{\Delta}}(M) = \rho(M) \tag{1.12}$$

$$\boldsymbol{\Delta} = \delta_r I \Rightarrow \mu_{\boldsymbol{\Delta}}(M) = \rho_r(M)$$

$$\boldsymbol{\Delta} = \Delta_f \Rightarrow \mu_{\boldsymbol{\Delta}}(M) = \overline{\sigma}(M), \tag{1.13}$$

hence the name, structured singular value.

Given two uncertainty structures $\boldsymbol{\Delta}_1$ and $\boldsymbol{\Delta}_2$, then a new structure can be defined as

$$\boldsymbol{\Delta}_3 \triangleq \left\{ \begin{bmatrix} \Delta_1 & \\ & \Delta_2 \end{bmatrix} : \Delta_1 \in \boldsymbol{\Delta}_1, \Delta_2 \in \boldsymbol{\Delta}_2 \right\}. \tag{1.14}$$

Theorem 1.6 (Main Loop Theorem [49])

$$\mu_{\boldsymbol{\Delta}_3}(M) < 1 \Leftrightarrow \begin{cases} \mu_{\boldsymbol{\Delta}_1}(M_{11}) < 1 \\ \sup_{\Delta_1\in\boldsymbol{B\Delta}_1} \mu_{\boldsymbol{\Delta}_2}(\Delta_1 \star M) < 1 \end{cases}$$

Imagine that $\boldsymbol{\Delta}_1$ is used to parameterize a set whose elements are uncertain models, $\boldsymbol{B\Delta}_2 \star (\boldsymbol{\Delta}_1 \star M)$. Every model in all of the sets is well-posed if and only if $\mu_{\boldsymbol{\Delta}_3}(M) < 1$. Nominal stability is a μ problem (1.12). As a result of Theorem 1.6, μ exactly answers the problem of robust stability which addresses the guaranteed stability for a set of models.

The problem of robust performance is to guarantee that a set of models achieve a certain level of performance. Looking back at Figure 1.1, assuming the system is well-posed and the l_2-induced gain from u to z is less than one, this is equivalent to saying

$$\overline{\sigma}(\Delta \star M) < 1 \ \forall \Delta \in \boldsymbol{B\Delta}.$$

Using (1.13) this can be rewritten as

$$\sup_{\Delta \in \boldsymbol{B\Delta}} \mu_{\Delta_f}(\Delta \star M) < 1.$$

Combining this with the well-posedness assumption and using Theorem 1.6, this robust performance question can be converted to the following standard μ problem:

$$\text{Robust Performance} \Leftrightarrow \mu_{\hat{\Delta}}(M) < 1$$

where

$$\hat{\Delta} \triangleq \begin{bmatrix} \Delta & \\ & \Delta_f \end{bmatrix}.$$

μ exactly answers a number of robustness questions, but a tractable method to compute μ is not obvious from the definition (1.10). However, it is easy to construct crude bounds,

$$\rho(M) \leq \mu_{\Delta}(M) \leq \overline{\sigma}(M). \tag{1.15}$$

These bounds can be refined.

Computing $\mu_{\Delta}(M)$ is NP-hard, but computationally tractable upper and lower bounds exist [47, 30, 34, 45, 48, 6]. The lower bound computation is based upon the spectral radius definition of μ (1.11). A necessary condition for optimality can be derived [34]; this necessary condition is a system of algebraic equations. The standard power algorithm [29] is an iterative approach to finding a solution to these algebraic equations. The problem is nonconvex, and there are no convergence guarantees for this iteration. Even if the iteration converges, it may converge to a local maximum.

An LMI upper bound can be constructed by refining (1.15). Assume that $\Delta \in \boldsymbol{B\Delta}$ and $D\Delta = \Delta D \ \forall \Delta \in \boldsymbol{B\Delta}$; it follows that

$$\begin{aligned} \rho(\Delta M) &\leq \overline{\sigma}(\Delta M) \\ &\leq \overline{\sigma}(D^{-1}D\Delta M) \\ &\leq \overline{\sigma}(D^{-1}\Delta D M) \\ &\leq \overline{\sigma}(\Delta D M D^{-1}) \\ &\leq \overline{\sigma}(D M D^{-1}). \end{aligned}$$

Finding the D that minimizes $\overline{\sigma}(DMD^{-1})$ is a GEVP and is constructed as follows:

$$\begin{aligned}
&\overline{\sigma}(DMD^{-1}) < \gamma \\
\Leftrightarrow\ &\|DMD^{-1}x\|_2^2 < \gamma^2\|x\|_2^2\ \forall x \neq 0 \\
\Leftrightarrow\ &D^{-*}M^*D^*DMD^{-1} - \gamma^2 I < 0 \\
\Leftrightarrow\ &M^*XM - \gamma^2 X < 0,\ X = D^*D > 0
\end{aligned}$$

The last expression is an LMI in X, and finding the infimum for γ for which the LMI is feasible is a GEVP (1.3). A tighter bound that exploits the special structure of real parameters [48] can be constructed and the associated LMI follows:

$$\mu_{\Delta}(M) \leq \overline{\mu}_{\Delta}(M) \triangleq \inf_{D\in\Delta, G\in\mathbf{G}} \{\gamma : M^*DM + j(GM - M^*G) - \gamma^2 D < 0\} \tag{1.16}$$

is a convex optimization problem where

$$\mathbf{D} \triangleq \{Diag\left[D_{r1}, \cdots, D_{rn_R}, D_{c1}, \cdots, D_{cn_C}, d_1 I_{n_f 1}, \cdots, d_{fn_F} I_{n_f F}\right] > 0\} \tag{1.17}$$

and

$$\mathbf{G} \triangleq \{Diag\left[G_{r1}, \cdots, G_{rn_R}, 0, \cdots, 0\right] : G_{ri} = G_{ri}^*\}. \tag{1.18}$$

1.5 Skew μ, $\check{\mu}$

If μ is less than one, then the system achieves the robust performance specified. If μ is greater than one, then there is a model in the set which doesn't achieve the performance specified. A more natural question is: Given a set of models, what is the performance level achieved? This can be answered by asking an iterative set of μ problems where the performance weight is being adjusted until μ is exactly equal to one.

Alternatively, $\check{\mu}$ can be defined to answer the question more directly. Implicit in the above procedure is that the set of models described by the LFT does not change and only the performance block is scaled. This is the motivation for the following definition.

Definition 1.7 *For $M \in \mathbb{C}^{n\times n}$ and using (1.14) to define Δ_3, $\check{\mu}_{\Delta_3}(M)$ is defined as*

$$\check{\mu}_{\Delta_3}(M) \triangleq \frac{1}{\min\{\overline{\sigma}(\Delta_2) : \Delta \in \Delta_3, \Delta_1 \in B\Delta_1, \det(I - M\Delta) = 0\}} \tag{1.19}$$

unless no $\Delta = \begin{bmatrix} \Delta_1 & \\ & \Delta_2 \end{bmatrix} \in \begin{bmatrix} B\Delta_1 & \\ & \Delta_2 \end{bmatrix}$ *makes* $I - M\Delta$ *singular, in which case* $\check{\mu}_{\Delta_3}(M) \triangleq 0$.

The associated upper bound is identical to the standard upper bound (1.16) with γ^2 replaced by $\begin{bmatrix} I & \\ & \gamma^2 I \end{bmatrix}$.

1.6 Implicit μ

The fundamental question is still the existence of nontrivial solutions, but an additional constraint is added. The loop equations are shown in Figure 1.4. The loop equations are identical to Figure 1.3, but the constraint

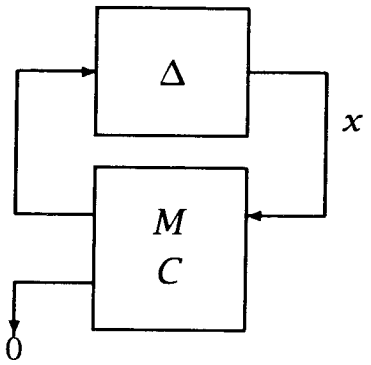

Figure 1.4: Standard Implicit System

that $Cx = 0$ is added to the possible outputs of Δ.

A simplified definition of the implicit μ and the upper bound presented in [35] follow.

Definition 1.8 *For the implicit system in Figure 1.4, the structured singular value* $\mu(C,M)$ *is defined as*

$$\mu(C,M) \triangleq \frac{1}{\min\left\{\overline{\sigma}(\Delta) : \Delta \in \mathbf{\Delta}, Ker\begin{bmatrix} I - \Delta M \\ C \end{bmatrix} \neq 0\right\}}, \tag{1.20}$$

unless $Ker\begin{bmatrix} I - \Delta M \\ C \end{bmatrix} = 0, \forall \Delta \in \mathbf{\Delta}$, *in which case* $\mu(C,M) \triangleq 0$.

Let $C_\perp$ be the matrix whose columns form a basis for the kernel of C. Then the following is an upper bound of $\mu(C,M)$,

$$\overline{\mu}(C,M) \triangleq \inf_{I \circ D = D > 0} \{\gamma : C_\perp^*(M^*DM - \gamma^2 D)C_\perp < 0\}. \tag{1.21}$$

This bound can be generalized to the structures presented for standard μ like real parameters.

Chapter 2

Probabilistic Robustness Analysis (PRA)

Robustness analysis in the μ-framework is a worst case paradigm, and μ answers a particular type of robustness analysis question. If μ is less than 1, then the set of models contained in the uncertainty description are guaranteed to be stable and meet the performance requirement specified. On the other hand, if μ is greater than 1, then there is a model in the set which does not meet the performance specified.

For many problems the worst case paradigm is overly conservative. What if the expectation that a system coming off an assembly does not achieve the desired performance is a once in the age of the universe event? From a practical real world perspective, the product and the manufacturing process should not be redesigned because the end result is good enough. Additionally, some robustness analysis questions are fundamentally probabilistic in nature. To assess the risk of a particular mission, or approximate the yield of a fabrication process, the desired analysis question is a question of probability or distribution.

Traditionally, these probabilistic robustness analysis questions have been "answered" using Monte Carlo methods by sampling parameter space to approximate the probability [38, 42]. To mix the probabilistic formulation with the worst case uncertainty in a unified framework, it is convenient to extend the μ-framework to include probabilistic uncertainty descriptions. The natural extension of the Monte Carlo methods presented in [38] is to perform a Monte Carlo analysis over μ problems as parameterized by the probabilistic parameters.

The drawback with Monte Carlo methods is that a large number of Monte Carlo samples are required to get an accurate description of rare events. Monte Carlo methods are good for approximating the left part of the cartoon in Figure 2.1. The bounds for μ are useful for describing the

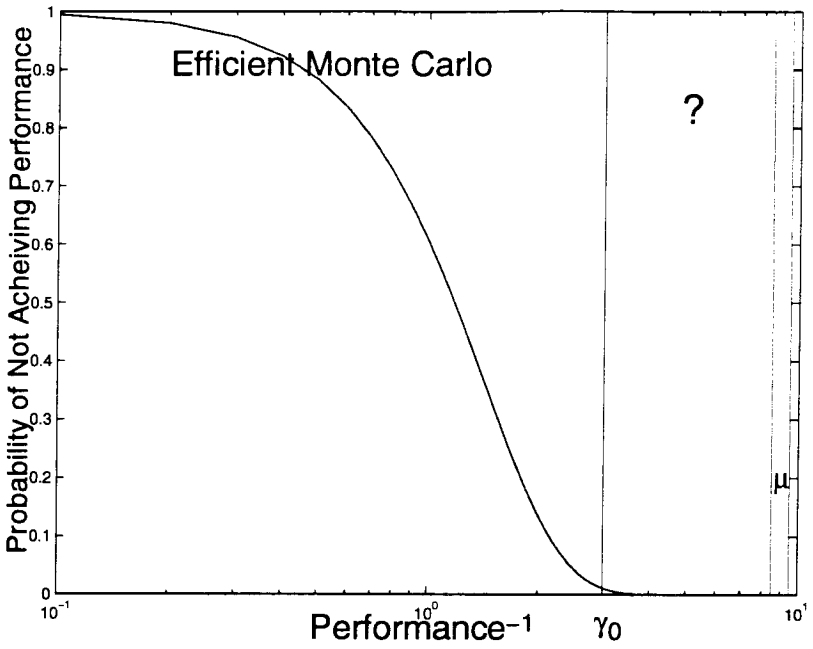

Figure 2.1: Monte Carlo vs. Worst Case Methods

right most non-zero point on the curve in Figure 2.1. In order to describe the region in between, the μ formulation will be extended to probabilistic μ, and ultimately leveraging this extension with Monte Carlo methods will provide a better description of the curve for rare events. Actually, probabilistic μ can be used to answer questions that are traditionally in the realm of Monte Carlo methods but would not be computationally efficient by comparison.

In this development, probabilistic formulations of μ are motivated from the system context. These formulations are inherently a mixture of probabilistic and worst case uncertainty. This is a manifestation of guaranteeing performance and/or stability of a system for a particular choice of parameters as well as incorporating unmodeled dynamics and time varying parameters for which there seems to be no convenient probabilistic extension but are natural in the μ-framework. The static parameters can be characterized by a probability distribution and admit analysis.

In the process of applying branch and bound algorithms to μ, additional information can be extracted about the system. Specifically regions of parameter space can be classified as stable or a bound on the performance can be computed, and a system has a probability of being in this region of parameter space. This results in a bound on the probability that

the system is stable and/or a bound on the distribution of achievable performance.

These formulations will be simplified to a "purely probabilistic μ" problem, for which a μ-function is defined and upper and lower bounds on the probability distribution of the μ-function are computed using a modification of the standard branch and bound algorithm developed by Newlin [32]. The function presented by Zhu in [50] is an approximation to purely probabilistic μ as defined in this paper.

For the mixed probabilistic and worst case formulation, it is easy to extend the branch and bound algorithm to construct an upper bound to the distribution of μ-function. Further, Monte Carlo methods can be leveraged with this upper bound information to construct a more efficient Monte Carlo approximation of the distribution. This is the same idea as the "soft" lower bound presented in [50].

2.1 Branch and Bound (B&B)

The fundamental idea of B&B optimization methods is to divide and conquer. Given a region over a finite parameter space, an objective function whose optimum is to be computed, and computationally tractable bounds for this optimization problem: the optimization problem can be solved. If the gap between the bounds is too large, then the problem can be divided into two subproblems by splitting the parameter space. The bounds for the two new problems can be used to construct better bounds to the original optimization problem.

This process can be repeated for each subproblem until the gap between the bounds is acceptable. This methodology can result in exponential growth of the number of problems to be solved as a function of the dimension of the parameter space. Therefore, pruning the set of subproblems is an important step of the algorithm. In the process of creating subproblems and computing bounds, some regions of parameter space can be ruled out for the location of the optimum. If the upper bound on a region is less than the lower bound on another, there is no reason to continue dividing such regions because it increases computation cost without improving performance.

In the μ context, branching improves the quality of the bounds, but the computational complexity of the bounds on the subdivided region is the same as the original problem. Although pruning is critical to the computational effectiveness, computing real/mixed μ is NP-hard so in the worst

case there is still exponential growth in the computation required. The B&B approach has been applied to the computation of real/mixed μ. The reader is referred to [32], [1], and [48] for specific details.

2.2 PRA Background

In order to discuss a distribution for μ, a μ-function must be defined on the space of Δ. This μ-function combined with a probability distribution in Δ-space results in a probabilistic definition of μ, hence the term "probabilistic μ." Some investigations in this direction were made in [50]; the approach taken by Zhu was using

$$\mu(M,\Delta) \triangleq \overline{\lambda}_r(M\Delta) \tag{2.1}$$

as the definition of the μ-function. Although not explicitly defined in this manner, that was the spirit of the work. This function is motivated by (1.11) and the fact this is the same function that is used in the B&B algorithms presented in [32]. As a result, the existing B&B methodologies can be extended to computing an upper bound to the tail of the distribution for the μ function in (2.1). This extension is presented in [50]. Some other worst-case work on probabilistic robustness analysis has been done by Barmish in [3]. This work investigates the worst-case distribution of uncertain paramters.

The idea here is to use local information which has been extracted while applying the B&B methods to obtain guaranteed bound on the distribution of $\mu(M,\Delta)$. If $\mu_\Delta(M) < \gamma_0$ for a region $\mathcal{A}$ in Δ-space, then region $\mathcal{A}$ can be ignored. No useful information about rare events will be obtained by further dividing $\mathcal{A}$ or doing Monte Carlo analysis with sample points in $\mathcal{A}$. Further, the probability that $\mu(M,\Delta) \geq \gamma_0$ is less than $1 - p(\mathcal{A})$. This would be a hard upper bound to the tail of the function. If Monte Carlo methods are applied to the regions for which $\mu_\Delta(M)$ is not guaranteed to be below γ_0, a better approximation to probabilistic μ can be determined for a specified number of Monte Carlo samples. This is the soft lower bound presented in [50].

The B&B algorithm in [32] is good for computing μ. From a system context, the exact value of μ is not important but rather is $\mu < 1$. In an effort to address probabilistic robustness analysis from a system context, it is necessary to modify the B&B algorithm and define a new μ-function which is useful for answering these probabilistic robustness questions.

2.3 Definition of Probabilistic μ

This development will **not** use (2.1) as the definition for the μ function. Some preliminaries are needed to motivate the definition used in this presentation.

There are two natural problems in the domain of probabilistic μ: Computing the probability of stability and the distribution for guaranteed performance for an uncertain system. The resulting formulation is a problem of the μ loop form shown in Figure 1.3. Where

$$\mathbf{\Delta} = \begin{bmatrix} \mathbf{\Delta}_1 & 0 \\ 0 & \mathbf{\Delta}_2 \end{bmatrix},$$

and $\mathbf{\Delta}_i$ may have additional structure. There is a probability distribution on $\mathbf{\Delta}_1$, and $\mathbf{\Delta}_2$ is a worst case block. So for the above problems for which probabilistic μ is appropriate, $\mathbf{\Delta}_2 = z^{-1}I$ for probabilistic stability in discrete time and $\mathbf{\Delta}_2 = \begin{bmatrix} z^{-1}I & 0 \\ 0 & \mathbf{\Delta}_3 \end{bmatrix}$ where $\mathbf{\Delta}_3$ is full and norm bounded by γ for probabilistic performance in the discrete time case.

For the stability problem, the probability to be evaluated is given by

$$P(\mu_{\mathbf{\Delta}_2}(\mathbf{\Delta}_1 \star M) < 1), \tag{2.2}$$

which is the probability that the system is stable. For the performance problem, the distribution to be evaluated is given by

$$P\left(\mu_{\mathbf{\Delta}_2}\left(\begin{bmatrix} I & 0 \\ 0 & \frac{1}{\gamma}I \end{bmatrix}(\mathbf{\Delta}_1 \star M)\right) < 1\right) \quad \forall \gamma > \gamma_0, \tag{2.3}$$

which is a probabilistic skew-μ problem. The idea in this formulation, is that the cumulative distribution function for guaranteeing stability and achieving a performance level $\gamma(> \gamma_0)$ is to be computed. This is the motivation for the following definitions.

Definition 2.1 *The set*

$$\mathbb{S}_{M,\gamma,\mathbf{\Delta}_2} \stackrel{\triangle}{=} \{\mathbf{\Delta}_1 \in \mathbf{\Delta}_1 : \mu_{\mathbf{\Delta}_2}(\mathbf{\Delta}_1 \star M) < \gamma\}$$

The definition of probabilistic μ is based upon Definition 2.1.

Definition 2.2 *The Probabilistic μ-function, $\mu_p(M, \mathbf{\Delta}, k)$ is defined as the value of γ for which the probability that $\mathbf{\Delta} \in \mathbb{S}_{M,\gamma,\mathbf{\Delta}_2} = 1 - k$.*

From the system context, the distribution function means nothing. Ultimately the concern is where $\mu < 1$, so for the generalization to probabilistic μ, the question is: what is the probability that $\mu_p < 1$? The distribution function only makes sense if it is the distribution of performance. From a practical point of view, it only makes sense from the system context to consider the skew version of Definitions 2.1 and 2.2 where only the performance block scales with γ. This is exactly analogous to answering the question: Given an uncertain system, what is the guaranteed performance? This question is addressed by skew μ.

Within this presentation, only the computation of the purely probabilistic case will be explored, rather than the mixed probabilistic/worst case formulation. For purely probabilistic μ, $\mathbf{\Delta}_2 = [\,]$. It is easy to extend some of the basic instruments for the computation of purely probabilistic μ to the mixed formulation; some ideas become easier to implement and some do not extend at all. This will be described during the development.

In the purely probabilistic case, there is no notion of stability and $\mu_{\mathbf{\Delta}_2}(\mathbf{\Delta}_1 \star M)$ means nothing because there are no loop equations associated with $\mathbf{\Delta}_2$. The only issue is the existence of nontrivial solutions to the loop equations for $\mathbf{\Delta}_1 \star M$ or equivalently the singularity of $I - M\mathbf{\Delta}_1$. For the purely probabilistic case, $\mathbf{\Delta}_1$ is divided two regions by the surface of singularity of $I - M\mathbf{\Delta}_1$ around the origin. The style of branching useful for computing bounds to the mixed case motivates the following definition for purely probabilistic μ for which (2.1) is an approximation.

Definition 2.3 *The set $\mathbb{S}_{M,\gamma}$ for $M \in \mathbb{R}^{n\times n}$ is the largest connected region containing the origin in $\mathbf{\Delta}$-space ($= \mathbb{R}^n$) for which $(I - M\mathbf{\Delta})$ is invertible,*

$$\mathbb{S}_{M,\gamma} \triangleq \{\Delta \in \mathbf{\Delta} : \det(\gamma I - M\Delta) \neq 0,\ \exists \text{ path from } \Delta \to 0 : \det(\gamma I - M\Delta_p) \neq 0 \text{ along the path.}\} \tag{2.4}$$

The definition for purely probabilistic μ used in this development is given in Definition 2.2 using $\mathbb{S}_{M,\gamma}$ instead of $\mathbb{S}_{M,\gamma,\mathbf{\Delta}_2}$.

To simplify the accounting, only uniform distributions over $[-1, 1]^n$ will be considered for $\mathbf{\Delta}$. Additionally for the remainder of this chapter, the term probabilistic μ will be used for purely probabilistic μ.

It follows immediately that $\mu_{\mathbf{\Delta}}(M) = \mu_p(M, \mathbf{\Delta}, 0)$. As a result, computing probablistic μ is NP-hard because computing one of the points on the curve is NP-hard. It is not clear how to approximate $\mu_p(M, \mathbf{\Delta}, k)$ by using Monte Carlo analysis; this is because there is no nominal notion of stability for a randomly chosen point in $\mathbf{\Delta}$-space for the purely probabilistic formulation.

It is conceptually easy to extend the basic methods of probabilistic μ to the mixed case. To extend the probabilistic formulation to the mixed:

1. Assume the problem is purely probabilistic.
2. Compute $\mathbb{S}_{M,\gamma}$.
3. Do some accounting and find the worst case over some parameters/full blocks.
4. Compute bounds on the probability of stability (2.2) or for a point on the performance curve (2.3).

In practice, this will not be the algorithm used. The first problem is that for actual problems, M is not necessarily real. As a result, $\mathbb{S}_{M,\gamma}$ is the entire space excluding a set of measure zero. Additionally, the method of computing $\mathbb{S}_{M,\gamma}$ involves B&B algorithms; applying B&B algorithms to complex and particularly full blocks results in tremendous growth in the dimension of the problem without significant performance improvements and is also an accounting nightmare. A better way to perform the computation for the mixed is to apply the B&B algorithm to Definition 2.1 directly and branch only on the probabilistic and static parameters.

Additionally, computing $\mathbb{S}_{M,\gamma}$ exactly is NP-hard, and like other μ computations it is necessary to compute upper bounds to μ_p or equivalently, sets contained within $\mathbb{S}_{M,\gamma}$. If the problem formulation has only one complex stability block and one full performance block in $\mathbf{\Delta}_2$, as the regions in $\mathbf{\Delta}_1$-space become arbitrarily small. The resulting problem converges to a μ-problem with one full block, and one complex repeated scalar block in $\mathbf{\Delta}_2$-space. For this uncertainty structure, μ is equal to the upper bound [49]. It would not be necessary to reduce every region in $\mathbf{\Delta}_1$ to be arbitrarily small, for some larger blocks the upper bound would be sufficient to guarantee the performance required.

2.4 Another Look at μ

μ can be interpreted as the supremum of a function, $\overline{\lambda}_r(M\Delta)$, on a region in $\mathbf{\Delta}$. This is the motivation for using (2.1) as the μ-function, but this isn't consistent with the mixed probabilistic framework. There is no clear way to extend (2.1) to determine stability for a particular point in $\mathbf{\Delta}_1$. Ultimately singularity of $I - M\Delta$ is the primary issue. Using (1.10) leads to the interpretation that $\frac{1}{\mu_{\mathbf{\Delta}}(M)}$ is the distance to singularity of $I - M\Delta$. Where

the distance is measured using $\|\cdot\|_\infty$ in Δ-space. So $\frac{1}{\overline{\mu}_\Delta(M)}$ can be used to construct a hypercube on which $I - M\Delta$ is guaranteed to be invertible. This hypercube is defined by $\Delta \in Box\left(0, \frac{1}{\overline{\mu}_\Delta(M)}\right)$. Where $Box(\Delta_0, v)$ is used to denote the axially aligned hyper-rectangle centered at Δ_0 and the vector $v, v_i > 0$, is the half length of the sides of the hyper-rectangle.

Using Figures 2.2 and 1.1, Δ_0 can be wrapped into M to create a new μ problem with

$$\overline{M} \triangleq \Delta_0(I - M\Delta_0)^{-1} \text{ and } \overline{\Delta}. \tag{2.5}$$

Computing $\overline{\mu}_{\overline{\Delta}}(\overline{M})$ is no more difficult than for the original problem, and $Box(\Delta_0, \frac{1}{\overline{\mu}_{\overline{\Delta}}(\overline{M})})$ defines a region in the Δ-space of the original problem for which the original problem is guaranteed to be nonsingular. This follows from Theorem 2.4.

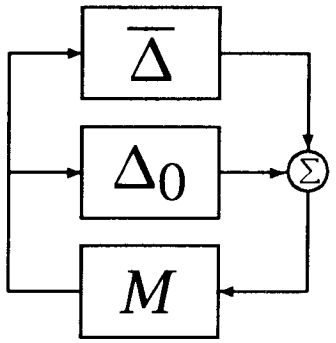

Figure 2.2: Construction of $\overline{M}$

For example, let

$$M = \begin{bmatrix} .8309 & .9677 \\ 2.7697 & -1.0533 \end{bmatrix}. \tag{2.6}$$

Figure 2.3 shows the boxes in Δ on which $I - M\Delta$ is guaranteed to be invertible. These boxes were constructed by chosing random points Δ_O in $B\Delta$ and computing the μ upper bound for $\overline{M}$. This upper bounds defines the associated nonsinuglar box. Notice that the regions in gray do not intersect the regions in black. In fact, the surface on which $I - M\Delta$ drops rank defines the boundary between these regions.

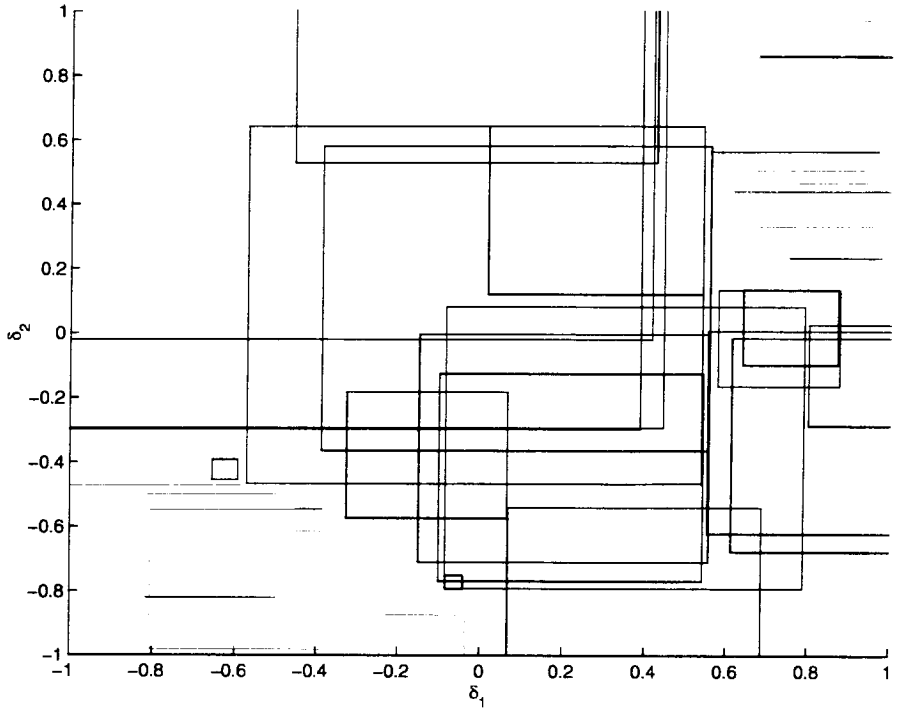

Figure 2.3: Squares in Δ on which $I - M\Delta$ is invertible for the entire interior. The centers are chosen randomly and the size is determined by the distance signularity.

2.5 Bounds on Probabilistic μ

In order to compute bounds for $\mu_p(M, \Delta, k)$, bounds must be computed for $\mathbb{S}_{M,\gamma}$. Assuming $(I - M\Delta_0)^{-1}$ exists, define

$$\hat{M} \triangleq \Delta_0(I - M\Delta_0)^{-1} diag(v)^{-1}.$$

Theorem 2.4 *Assuming* $\overline{\mu}(\hat{M}) < 1$ *then*

$$\Delta_0 \in \mathbb{S}_{M,1} \Leftrightarrow Box(\Delta_0, v) \subset \mathbb{S}_{M,1}$$

and

$$\Delta_0 \in (\Delta - \mathbb{S}_{M,1}) \Leftrightarrow Box(\Delta_0, v) \subset (\Delta - \mathbb{S}_{M,1})$$

Proof: By contradiction, assume $\Delta_0 \in \mathbb{S}_{M,1}$ and $Box(\Delta_0, v) \not\subset \mathbb{S}_{M,1}$. $\Rightarrow$ $\exists \Delta_1 \in Box(\Delta_0, v)$ such that $det(I - M\Delta_1) = 0$. $\Rightarrow$ $\exists \Delta_2 = (\Delta_1 - \Delta_0) diag(v) \in Box(0, 1)$ and $(I - \hat{M}\Delta_2)$ is singular. $\Rightarrow$ $\mu(\hat{M}) \geq 1$, contradiction. The arguments for the other cases are similar. $\square$

The black regions in Figure 2.3 are guaranteed to be in $\mathbb{S}_{M,1}$, and the gray regions are guaranteed to be in $\mathbf{\Delta} - \mathbb{S}_{M,1}$.

Bounds at a Point

The results shown in Figure 2.3 could be used to compute bounds for k such that

$$\mu_p(M, \mathbf{\Delta}, k) = 1,$$

but computing the volume of a union of hypercubes is intractable. A systematic method consistent with standard B&B algorithms is chosen instead.

A sufficient test for whether $\Delta_0 \in (\mathbf{\Delta} - \mathbb{S}_{M,1})$ is if the number of positive real eigenvalues greater than 1 of $M\Delta_0$ is odd. This follows for the simplified version of the μ paradigm presented. A sufficient test for whether $\Delta_0 \in \mathbb{S}_{M,1}$ is if $\overline{\lambda}_r(M\Delta_0) < 1$. Otherwise, it can be tested/checked if any adjacent regions have been classified. If there is such a region, then if there exists a $Box(\Delta_t, \epsilon)$ in $\mathbf{\Delta}$-space that contains a subset of both adjacent regions such that $I - M\Delta$ is invertible on the entire box, then the unclassified region should be assigned to the same region as its neighbor. This will be referred to as similarly classifiable neighbors. This follows from Theorem 2.4

The algorithm for probabilistic μ B&B to be presented is inherently recursive. There exists three types of regions: undecided, unclassified, and classified. The undecided are regions which might straddle the boundary of singularity. The unclassified are regions which do not straddle the boundary, but it is not known whether they belong to $\mathbb{S}_{M,1}$ or its complement. The classified are regions for which it is known to be in $\mathbb{S}_{M,1}$ or its complement. All the regions are hyperrectangles.

The algorithm:

1. Choose an undecided region to be active.
2. Divide the active region into 2 active regions, chopping on the longest edge.
3. Test: $\overline{\mu}(\hat{M}) < 1$ for both active regions. If false, then deactivate region and assign to undecided.
4. Apply Sufficient Tests to active regions.
5. If the active region is not classifiable, check for an adjacent classified region. If there are none then deactivate and assign to unclassified;

otherwise the region is classifiable with the same classification as the adjacent classified region.

6. If the active region is classifiable, look for adjacent regions among the unclassified regions. If one is found, then activate these new classifiable regions and repeat step 6. If none are found, then deactivate classifiable regions and assign to the set of classified regions with classification information.

7. Is the desired classification accuracy achieved? No ⇒ Goto Step one, Yes ⇒ Stop.

The checking of neighbors is an artifice of the purely probabilistic formulation. The interpretation of this assignment in the system context is determining whether the nominal system is stable and achieves the desired performance. This follows from Definition 2.1. If the nominal system is unstable and/or doesn't meet the performance requirement, then it is not possible to assign the entire region to being in the complement of $\mathbb{S}$. As a result only any upper bound can be computed for the mixed case.

Applying the above algorithm to M from (2.6) results in Figure 2.4. Where the construction of the boxes is identical to Figure 2.3 except Δ_0 are chosen systematically and the regions of nonsingularity may be rectangles rather than squares. The resulting bounds are

$$\text{gray volume} = .1873 < \{k\colon \mu_p(M, \boldsymbol{\Delta}, k) = 1\} < .2192 = 1 - \text{black volume}.$$

The black boxes are classified to $\mathbb{S}_{M,1}$, and the gray boxes are classified to the complement. The regions in between are undecided, and there were no unclassified regions.

One way to construct bounds for other points on the curve would be to replace M by $\frac{M}{\gamma}$ and repeat the above algorithm. The resulting bounds would apply to $\mu_p(M, \boldsymbol{\Delta}, k) = \gamma$. This is an inefficient method for computing bounds on $\mu_p(M, \boldsymbol{\Delta}, k)$, because there is information in the boundary that has been computed that can be recycled. A more efficient method for computing bounds on the curve follows.

Bounds for the Curve

Suppose that we wish to compute bounds of the curve relating k and γ in

$$\mu_p(M, \boldsymbol{\Delta}, k) = \gamma$$

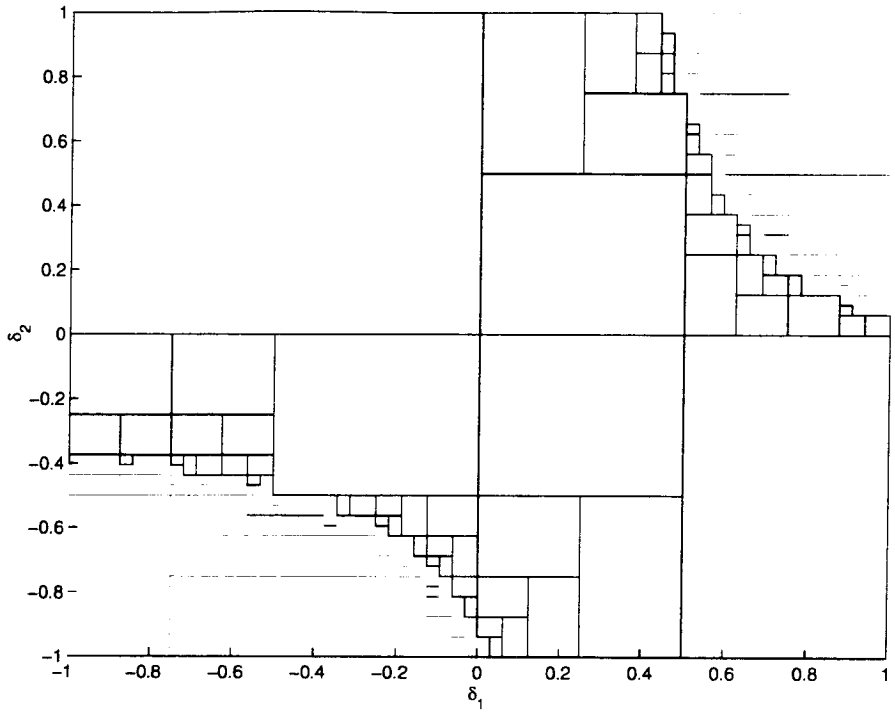

Figure 2.4: Computing Bounds of $\mathbb{S}_{M,1}$

for $\gamma \geq \gamma_0$, using Definition 2.2. This problem is equivalent to computing bounds of the curve relating k and γ in

$$\mu_p\left(\frac{M}{\gamma_0}, \Delta, k\right) = \gamma \tag{2.7}$$

for $\gamma \geq 1$. To convert from one bound to the other only requires a scaling by γ_0.

Using the algorithm presented in Section 2.5, upper and lower bounds can be computed for the value of k at $\gamma = 1$ for (2.7). In the process of doing this computation, $Box(0, 1)$ has been carved into a region contained within $\mathbb{S}_{M,1}$ and a region contained in its complement as shown in black and gray, respectively, in Figure 2.4 for example. To compute bounds for the value of k and $\gamma > 1$ for (2.7), it is necessary to compute bounds on $\mathbb{S}_{M,\gamma}$. Using Definition 2.3, bounds for $\mathbb{S}_{M,\gamma}$ can be computed by drawing $Box\left(0, \frac{1}{\gamma}\right)$ on Figure 2.4. Computing the gray volume contained in the box and the black volume contained in the box and multiplying both by $\left(\frac{\gamma}{2}\right)^n$, where n is the dimension of Δ-space, determine guaranteed bounds for k at γ. This can be done for all $\gamma > 1$, hence generating guaranteed bounds

for parts of the curve.

This method is applied to the following example:

$$M = \begin{bmatrix} .4155 & .4838 \\ 1.3849 & -.5266 \end{bmatrix},$$

which is just (2.6) scaled by a factor of 2. The resulting curve is shown in Figure 2.5. The lower curve in Figure 2.5 is the lower bound of μ_p,

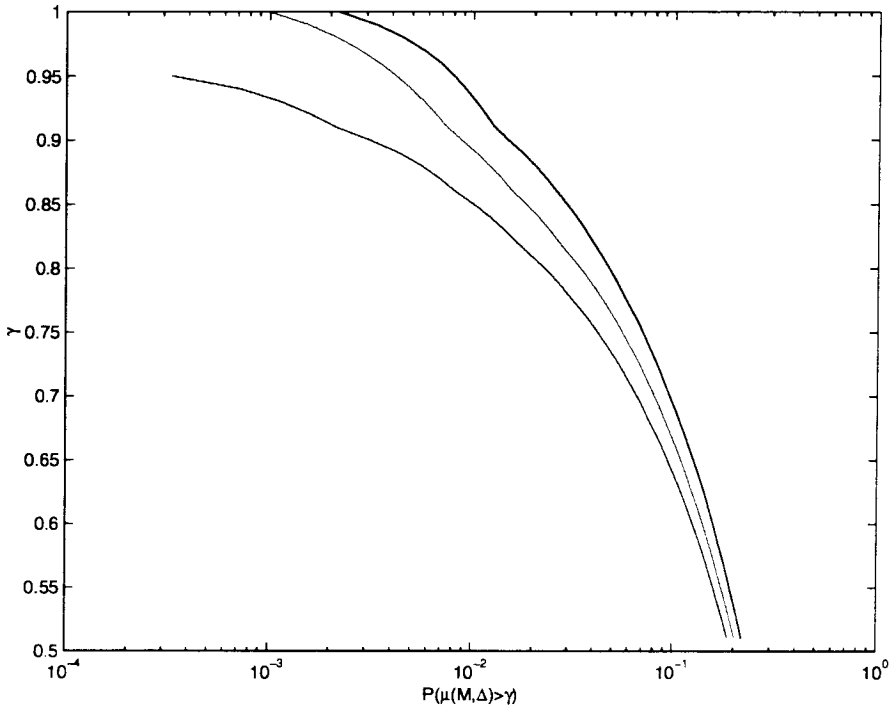

Figure 2.5: Upper, Lower, and Soft Bounds of μ_p

the upper curve is the upper bound, and the middle curve is the "soft" lower bound and is computed by applying the sufficient conditions for assignment from Section 2.5 to the centers of the unassigned regions. The results of the sufficient conditions are used to assign these regions with no guarantee on the resulting curve. This would be a Monte Carlo technique leveraged with the guaranteed bounds as no Monte Carlo points are placed in regions for which the answer has been guaranteed.

This trick for computing the curve will not extend to the mixed case where a probabilistic description of the achieved performance is desired. The problem is that the variation of γ is in the worst case block, so we are not branching over that parameter. Scaling by γ would describe a

conservative region for which the system is not singular. This can be used as a starting point for carving parameter space because some regions can be excluded.

2.6 Combining Monte Carlo and μ

The Monte Carlo methodology can be leveraged with the information from the B&B schemes for the mixed case. In the purely probabilistic, it is much harder because there are only sufficient conditions for assigning a point to being inside or outside of $\mathbb{S}$ and finding a path in Definition 2.3 for each Monte Carlo point is out of the question. On the other hand the Definition 2.1 indicates that the assignment of a point is a μ computation.

The method for combining these two methodologies is essentially the same as "soft" lower bound presented by Zhu [50]. The B&B guaranteed analysis identifies regions of Δ_1-space that are benign and further analysis is not necessary, so it would be a waste to carry out Monte Carlo analysis on these regions. This elimination increases the statistical significance of the resulting Monte Carlo simulations for assessing the likelihood of rare events. If 99 percent of the Δ_1-space is benign and all of the Monte Carlo simulations come from the remaining regions, each simulation is equivalent to 100 simulations had the simulations been chosen from all of Δ_1.

2.7 Computational Issues

The B&B techniques work well for computing μ. In the worst case the problem is still NP-hard, but in the average case the methodology works well. In applying the B&B techniques for computing bounds to the probabilistic formulation of μ results in bad dimensional growth not only in the worst case but also in the average case. The fundamental problem is the need to grid the boundary of singularity. This problem is the motivation for the analysis of more exotic regions in uncertainty space in the following chapters.

For the 2-d problem presented here, this algorithm works reasonably well. Applying to same methodology to slightly larger problems, (4-d) leads to tremendous increases in computation. To the point where computation on a Sun Ultra 1 is not practical. The gridding problem is shown in Figure 2.6. So it is necessary to grid the region on one side of the curve to an accuracy of ϵ for the general PRA problem. Where ϵ is the volume

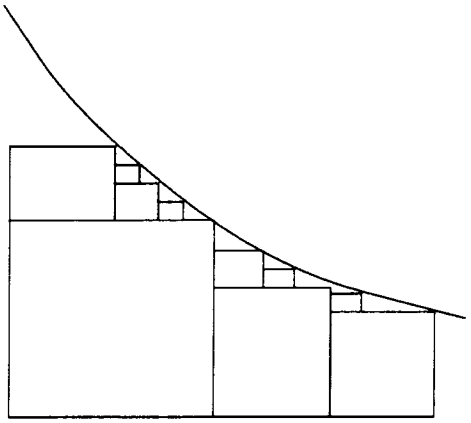

Figure 2.6: Gridding the Boundary

left uncovered by the gridding. The number of regions required grows like $\left(\frac{1}{\epsilon}\right)^{(n-1)}$ where n is the dimension of the space to be gridded. As a result the computation time grows exponentially as a function of the dimension of the problem.

Intuition and Figure 2.6 hide the problems that are generically encountered in high dimensions. In high dimensions, nearly all of the volume of a hypercube is near the boundary. The volume of the unit hypercube is 1, and the volume of a hybercube with side length γ is γ^n. For $\gamma < 1$, $\gamma^n \to 0$ as $n \to \infty$. If a hypercube is found that is on one side of a separating surface then the volume of that hypercube is minuscule compared to the total volume isolated by the surface.

In high dimensions, the B&B approach carves up the unit hypercube into extremely small volumes. In fact, even small increases in the size of the gridding elements would lead to large increases in their volumes which would drastically reduce the number of required regions.

Chapter 3

Spherical μ

Fundamentally, $\frac{1}{\mu}$ defines the distance to singularity for a matrix over a space of perturbations. The natural choice for the distance measurement is the ∞-norm. This is based on engineering motivation. The problem of guaranteeing the nominal l_2-gain for a system is less than 1 is a μ problem for $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ and $\Delta = \begin{bmatrix} \delta_c I & \\ & \Delta_f \end{bmatrix}$ where A, B, C, and D are the state space representation of the system. It follows that $\mu_\Delta(M) < 1 \Leftrightarrow$ the l_2-gain is less than 1. Further, if there are independent uncertain parameters contained within an interval and bounded unmodeled dynamics and the performance is to be guaranteed in the presence of this uncertainty, then the problem is naturally extended to a μ problem using the Main Loop Theorem 1.6. This independence of uncertain, stability, and performance blocks in Δ makes the ∞-norm the natural choice for measuring distance to singularity.

For the probablistic formulations of μ, these hypercubic regions in Δ-space lead to generic exponential growth in the computation. It is necessary to describe more exotic regions to grid the boundary of singularity. To be of use these exotic regions must admit computationally tractable bounds. In particular, the goal is to describe standard regions with linear cuts which is further investigated in Chapter 7.

More generally, it is also important to know how the μ-framework can be generalized and still admit tractable analysis. An extension of the μ-framework is generalized μ as posed by Chen [11]. For this generalization, Chen [12] presents a closed form solution for generalized μ for when M is rank one.

So as an initial foray into more exotic regions, what happens if the 2-norm is used to measure the distance to singularity in Δ-space? This is a special case of Chen's generalized μ. Can the upper bound be extended? The lower bound will be addressed in Chapter 5. Note that the standard

measurement in Δ-space, $\overline{\sigma}(\Delta)$ for block diagonal Δ is an ∞-norm over induced 2-norms.

3.1 Definition of Spherical μ, μ_S

For simplicity in derivation and presentation it is assumed that the uncertainty consists of only complex scalars. There are no full blocks, repeated scalars, or real parameters, and the uncertainty set is described by a Frobenius norm constraint rather than an induced norm constraint like $\overline{\sigma}(\Delta) \leq 1$.

Definition 3.1 *The default uncertainty structure is*

$$\Delta_{\mathbf{s}} := \{diag\,[\delta_1, \cdots, \delta_n] : \delta_i \in \mathbb{C}\}. \tag{3.1}$$

The level curves of $\|\Delta\|_F$ are defined by

$$\sum_{i=1}^{n} |\delta_i|^2 = K, \tag{3.2}$$

which describes hyperspheres in $\Delta_{\mathbf{s}}$ space, hence the name spherical μ (μ_S).

Definition 3.2 *For $M \in \mathbb{C}^{n\times n}$, Spherical μ, μ_S is defined as*

$$\mu_S(M) \triangleq \frac{1}{\min\{\|\Delta\|_F : \Delta \in \Delta_{\mathbf{s}}, \det(I - M\Delta) = 0\}} \tag{3.3}$$

unless no $\Delta \in \Delta_{\mathbf{s}}$ makes $I - M\Delta$ singular, in which case $\mu_S(M) \triangleq 0$.

Alternatively, (3.3) can be written as

$$\mu_S(M) = \sup_{\Delta\in\Delta_{\mathbf{s}}, \|\Delta\|_F<1} \rho(M\Delta) \tag{3.4}$$

Much like the standard μ problem, computing spherical μ exactly seems to be intractable. As a result, one is resigned to computationally tractable upper and lower bounds as in the standard μ case. Computing the exact value of $\mu_S(M)$ is NP-hard in the case where $\Delta_{\mathbf{s}}$ is restricted to be real. This follows from the bounds relating μ and μ_S in [24] and the approximability results in [17]. The implication is that except for special cases, like rank one analyzed in [9] where an explicit solution is given, obtaining the exact value is computationally infeasible especially for medium to large sized problems.

Naturally there are simple bounds relating μ and μ_s; this is based on the equivalence of $\|\cdot\|_2$ and $\|\cdot\|_\infty$. The graphical intuition is show in Figure 3.1. The relationship is presented in Theorem 3.3.

Theorem 3.3 $\mu_s(M) \leq \mu(M) \leq \sqrt{n}\mu_s(M)$

Proof: The statement follows immediately from

$$\{\Delta: \Delta \in \Delta_{\mathbf{s}}, \|\Delta\|_F < 1\} \subseteq \{\Delta: \Delta \in \Delta_{\mathbf{s}}, \overline{\sigma}(\Delta) < 1\} \tag{3.5}$$

$$\{\Delta: \Delta \in \Delta_{\mathbf{s}}, \overline{\sigma}(\Delta) < 1\} \subseteq \{\Delta: \Delta \in \Delta_{\mathbf{s}}, \|\Delta\|_F < \sqrt{n}\} \tag{3.6}$$

and the associated spectral radius definitions for μ_s and μ, (3.4) and (1.11) respectively. □

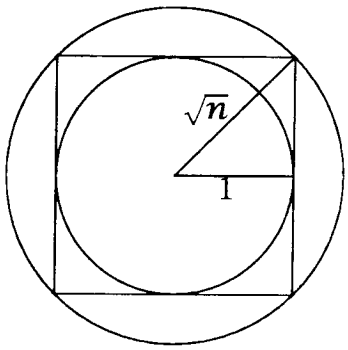

Figure 3.1: Cube, Inscribed, and Superscribed Spheres

As a consequence of Theorem 3.3, any upper bound of $\mu(M)$ is also an upper bound of $\mu_s(M)$, additionally, any upper bound of μ_s can be scaled to determine an upper bound of μ. In general, these would be conservative bounds which scale badly with dimension.

3.2 An Upper Bound to Spherical μ

As is the case for μ, an upper bound to μ_s is a lower bound on the distance to singularity, because $\frac{1}{\mu_s}$ defines the distance. As a result, the $\overline{\mu}_s$ specifies a region in uncertainty space for which $det(I - M\Delta) \neq 0$ on the interior of the region as shown in Figure 3.2.

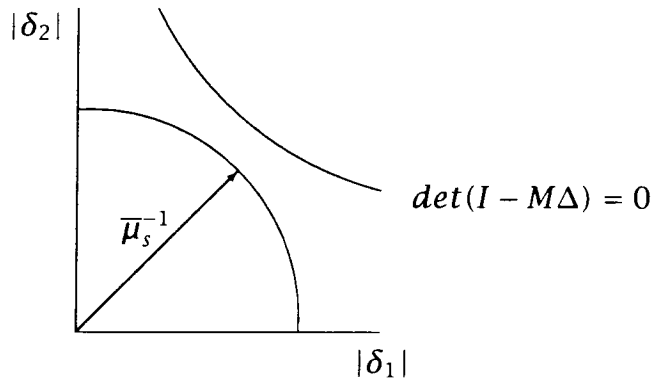

Figure 3.2: Guaranteed Region of Non-Singularity

The traditional derivation of the μ upper bound is based upon small gain arguments. If the gain of $M\Delta$ is guaranteed to be less than 1, it follows that the associated loop equations cannot have a non-zero solution. A set of scalings, which do not change the existence of non-zero solutions to the loop equations, are used to find an equivalent system with a smaller loop gain. These scalings are analogous to the similarity transformation for state-space representations. Finding the optimal scaling and the associated loop gain is a convex optimization problem.

Using this methodology, it is not clear how to proceed in deriving an analogous upper bound. The formulation here is based on converting the spherical constraint on the operator Δ into a signal constraint on x from the standard μ loop in Figure 1.3, and a separating hyperplane argument.

A $\overline{\mu}_s(M)$, which is not necessarily an upper bound of $\mu(M)$, can be computed by solving an LMI similar to the standard μ upper bound (1.16). The derivation of the LMI follows, but first a critical tool is presented.

The following Lemma is used in a separating hyperplane argument to construct a sufficient LMI condition for there being no nontrivial solutions to a quadratic form or equivalently no nontrivial solutions to the loop equations.

Lemma 3.4 *If* $A > 0$ *and* $B > (\geq)0$ *then* $Tr(AB) > (\geq)0$.

Proof:

$$Tr(AB) = Tr(AB^{\frac{1}{2}}B^{\frac{1}{2}}) = Tr(B^{\frac{1}{2}}AB^{\frac{1}{2}})$$

$$B^{\frac{1}{2}} = (B^{\frac{1}{2}})^* \Rightarrow B^{\frac{1}{2}}AB^{\frac{1}{2}} > (\geq)0 \Rightarrow Tr(B^{\frac{1}{2}}AB^{\frac{1}{2}}) > (\geq)0$$

□

The converse of Lemma 3.4 is the separating hyperplane or S-procedure [27] argument. If $Tr(AB) < 0$ and $A > 0$ it follows that B is not positive semidefinite.

If B is an element of a set of matrices, $\mathbb{B}$ and a matrix $A > 0$ is found so that $Tr(AB) < 0$ for all values of B, then the intersection between the set $\mathbb{B}$ and the set of positive matrices is empty. The matrix A defines a hyperplane that separates the cone of positive definite matrices from the set $\mathbb{B}$. This condition for separating $\mathbb{B}$ from the positive cone is only sufficient.

Theorem 3.5 (Spherical μ Upper Bound)

$$\mu_s(M) \leq \overline{\mu}_s(M) \triangleq \inf_{D>0} \{\gamma : M^*(I \circ D)M - \gamma^2 D < 0\} \tag{3.7}$$

Proof: x and y are the output and input vectors of Δ, respectively.

$$\sum_{i=1}^{n} |\delta_i|^2 < \gamma^{-2} \tag{3.8}$$

$$\Leftrightarrow \delta^*\delta < \gamma^{-2} \text{ for } \delta = [\delta_1 \ \delta_2 \cdots \delta_n]^T \tag{3.9}$$

$$\Leftrightarrow \begin{bmatrix} \gamma^{-2} & \delta^* \\ \delta & I \end{bmatrix} > 0 \text{ Applying Lemma 1.3} \tag{3.10}$$

Using $x_i = \delta_i y_i = y_i \delta_i$ and a congruence transformation with

$$C = \begin{bmatrix} 1 & & & \\ & y_1 & & \\ & & \ddots & \\ & & & y_n \end{bmatrix}, \tag{3.11}$$

$$\Rightarrow \begin{bmatrix} \gamma^{-2} & x^* \\ x & I \circ (yy^*) \end{bmatrix} \geq 0 \text{ Applying Lemma 1.1} \tag{3.12}$$

$$\Leftrightarrow I \circ (yy^*) - \gamma^2 xx^* \geq 0 \text{ Applying Lemma 1.3} \tag{3.13}$$

$$\mu_s(M) < \gamma \text{ if } y = x = 0 \text{ is the only solution to the quadratic form (3.13) with } y = Mx. \tag{3.14}$$

Using the converse of Lemma 3.4,

$$\text{If } \exists D > 0\text{: } Tr(D(I \circ (yy^*) - \gamma^2 xx^*)) < 0, \forall x \in \mathbb{C}^n,$$
$$\text{then the quadratic form (3.13) has no nontrivial solutions.}$$

Using $y = Mx$,

$$Tr(D(I \circ ((Mx)(Mx)^*) - \gamma^2 xx^*)) < 0, \forall x(\neq 0) \in \mathbb{C}^n$$

Using $Tr(AB) = TR(BA)$,

$$\Leftrightarrow x^* M^* (I \circ D) Mx - \gamma^2 x^* Dx < 0, \forall x(\neq 0) \in \mathbb{C}^n \tag{3.15}$$

$$\Leftrightarrow M^* (I \circ D) M - \gamma^2 D < 0 \tag{3.16}$$

□

The separating hyperplane argument can be used to derive the standard μ upper bound. The main difference between this derivation and the $\overline{\mu}(M)$ formulation is in deriving the quadratic form in (3.13). Rather than have 1 large quadratic form, the result is n independent scalar quadratic constraints.

3.3 Properties of $\overline{\mu}_s$

The LMI optimization in Theorem 3.5 can be solved by using general purpose LMI solvers, but it can be solved more efficiently,

$$\inf_{D>0} \{\gamma : M^*(I \circ D)M - \gamma^2 D < 0\} = \sqrt{\rho(M^T \circ M^*)}. \tag{3.17}$$

The reader is referred to Chapter 5 for more details.

The spherical μ upper bound LMI in (3.7) is related to the standard μ upper bound LMI, but it is unclear whether the new LMI is *consistent* with the standard LMI. Corollary 3.6 and Theorem 3.8 are presented to prove that the two bounds are in fact consistent.

Corollary 3.6 $\overline{\mu}_s(M) = \overline{\mu}_s(TMT^{-1})$ *where* $I \circ T = T > 0$

Proof: Replacing M by TMT^{-1} and assuming that the associated μ_s upper bound LMI is feasible for a value γ, then LMI in (3.16) becomes

$$T^{-*} M^* T^* (I \circ D) TMT^{-1} - \gamma^2 D < 0 \tag{3.18}$$

and is feasible. The following construction shows that for the same value of γ, then, by construction, the μ_s upper bound LMI is feasible for the original matrix M.

$$M^*T^*(I \circ D)TM - \gamma^2 T^*DT < 0$$

Let $T^* = diag[t_1, \cdots, t_n]$,

$$M^*(I \circ D \circ (tt^*))M - \gamma^2(D \circ (tt^*)) < 0 \text{ Applying Lemma 1.1}$$

Let $\overline{D} = D \circ (tt^*)$,

$$M^*(I \circ \overline{D})M - \gamma^2\overline{D} < 0$$

$\overline{D} > 0$ follows from Lemma 1.2 $\Rightarrow$ for a given γ; if the LMI in (3.18) has a solution, then the LMI in (3.16) has a solution.

$$\Rightarrow \overline{\mu}_s(M) \leq \overline{\mu}_s(TMT^{-1}) \leq \overline{\mu}_s(T^{-1}TMT^{-1}T) = \overline{\mu}_s(M)$$

□

In Corollary 3.6, T enters in the same manner as the D-scales from the standard μ upper bound. This implies that $\overline{\mu}_s(M)$ has exploited all of the structure in the uncertainty set that $\overline{\mu}(M)$ has and that the additional freedom in the $\overline{\mu}_s(M)$ LMI is related to the structure which results from the Frobenius set description. In the standard case, the hypercube does not provide additional structure to exploit.

The following lemma is a useful tool in relating $\overline{\mu}_s(M)$ and $\overline{\mu}(M)$.

Lemma 3.7 *$D \leq nI \circ D$ for $D > 0$*

Proof:

$$\overline{\sigma}(\mathbb{1}_n) = n \text{ and } \underline{\sigma}(nI) = n$$

$$\Rightarrow nI - \mathbb{1}_n \geq 0$$

$$(nI \circ D) - D = (nI - \mathbb{1}_n) \circ D$$

Applying Lemma 1.2,

$$(nI \circ D) - D \geq 0$$

□

Theorem 3.3 presents a relationship between μ and μ_s. Theorem 3.8 shows that the LMI upper bounds share the same relationship.

Theorem 3.8 $\overline{\mu}_s(M) \leq \overline{\mu}(M) \leq \sqrt{n}\overline{\mu}_s(M)$

Proof: The standard upper bound (1.16) and the spherical upper bound (3.16) can be rewritten as

$$\overline{\mu}(M) = \inf_{D_1>0}\{\gamma: M^*(I \circ D_1)M < \gamma^2(I \circ D_1)\}$$

and

$$\overline{\mu}_s(M) = \inf_{D_2>0}\{\gamma: M^*(I \circ D_2)M < \gamma^2 D_2\}$$

respectively. Setting $D_2 = I \circ D_2 = I \circ D_1$

$$\Rightarrow \overline{\mu}_s(M) \leq \overline{\mu}(M)$$

Applying Lemma 3.7:

$$M^*(I \circ D_2)M < (\overline{\mu}_s(M) + \epsilon)^2 D_2 \leq n(\overline{\mu}_s(M) + \epsilon)^2(I \circ D_2)\ \forall \epsilon > 0$$

Setting $D_1 = D_2$ and taking the limit as $\epsilon \to 0$:

$$\Rightarrow \overline{\mu}(M) \leq \sqrt{n}\overline{\mu}_s(M)$$

□

Theorems 3.3 and 3.8 indicate further consistency between the spherical μ upper bound, the standard μ upper bound, and the set containment shown in Figure 3.1. If the relationship between $\mu_s(M)$ and $\mu(M)$ was not the same as the relationship between their respective upper bounds, then there would be instances where one of the upper bounds could be trivially improved. For example, assume that the relationship from Theorem 3.8 was replaced by $\overline{\mu}_s(M) \leq \overline{\mu}(M) \leq n\overline{\mu}_s(M)$. If both of these upper bounds were computed with $\overline{\mu}_s(M) = 1$ and $\overline{\mu}(M) = n$, tHen latter bound would be useless, because the former implies $\mu_s(M) \leq 1$. This in turn implies that $\mu(M) \leq \sqrt{n}$ from Theorem 3.3, which would be a tighter upper bound for $\mu(M)$.

$\mu(M)$ is equal to the upper bound in the rank one case. The exactness of the upper bound for the rank one case is a direct consequence of convexity. In fact, in this case the determinant is linear in the uncertain parameters. The same properties hold for $\mu_s(M)$ in the rank one case.

Theorem 3.9 $\overline{\mu}_s(M) = \mu_s(M)$ *for* $rank(M) = 1$

Proof: Consider the rank one matrix M. Applying the explicit solution given in (3.17), the optimal value of γ is $\overline{\mu}(M) = \sqrt{\sum_{i=1}^{n} |m_{ii}|^2}$. An explicit perturbation that makes $(I - M\Delta)$ singular is $\Delta = (I \circ M^*)/\gamma^2$. Since the Frobenius norm of Δ is $1/\gamma$, then γ is also a lower bound for μ_s. Therefore, $\mu_s = \overline{\mu_s}$ in the rank one case. □

This is consistent with Chen's results [12] which show that the rank one problem is easy for a larger class of uncertainty descriptions. So an easy problem is still easy.

The computation presented here is an upper bound. A question critical to assessing the quality of this upper bound is: how large can the gap between μ_s and $\overline{\mu}_s$ be? The worst ratio between $\mu_s(M)$ and $\overline{\mu}_s(M)$ is at least $\sqrt{n}$. This ratio is achieved by

$$M = \begin{bmatrix} 0 & 1 \\ I & 0 \end{bmatrix}, \tag{3.19}$$

for which $\mu_s(M) = \frac{1}{\sqrt{n}}$ and $\overline{\mu}_s(M) = 1$.

Corollary 3.10 *If $n \leq 3$ (3 scalar blocks), then $\frac{\overline{\mu}_s(M)}{\mu_s(M)} \leq \sqrt{n}$*

Proof: For $n = 3$ it has been shown that

$$\mu(M) = \overline{\mu}(M)$$

by Packard in [34]. Applying Theorems 3.3 and 3.8,

$$\overline{\mu}_s(M) \leq \overline{\mu}(M) = \mu(M) \leq \sqrt{n}\mu_s(M)$$

$$\Rightarrow \frac{\overline{\mu}_s(M)}{\mu_s(M)} \leq \sqrt{n}$$

(3.19) shows that this bound is tight. □

3.4 Small Gain Formulation

Note the similarity between the μ and μ_s upper bound LMIs. This hints that there may be a small gain interpretation for the μ_s LMI upper bound in (3.16). By reversing the operations in the standard construction of the μ LMI upper bound from the small gain condition, the LMI upper bound for for μ_s can be converted to the following gain condition:

$$\mu_s(M) \leq \inf_{D>0} \left\{ \gamma : \overline{\sigma}((I \circ D)^{\frac{1}{2}} M D^{\frac{-1}{2}}) < \gamma \right\} \tag{3.20}$$

If this gain condition is satisfied, why does it imply that there are no non-trivial solutions to the canonical loop equations for any uncertainty in a particular spherical set?

For the standard μ LMI upper bound, the D-scales are added because they can change the gain of the loop, but the existence of a non-trivial solution is invariant to these D-scales because they are constructed to commute with the uncertainty structure.

For the spherical μ LMI upper bound, there is additional freedom in the D-scales interpreted from (3.20). Nominally, the standard D-scales could be chosen, but in this problem there is additional structure to exploit. The spherical constraint on Δ is not an induced norm constraint from $\|\cdot\|_2$ to $\|\cdot\|_2$. In other words, the elements on the boundary of the unit ball in $\mathbf{\Delta}$-space do not have the same induced 2-norm. So this uncertainty structure can also absorb scalings Q that preserve $\|Q\Delta\|_2 < 1$, for this sort of scaling the resulting small gain upper bound would be

$$\mu_s(M) \leq \overline{\sigma}(MQ^{-1}). \tag{3.21}$$

Note that structure and the set of allowable Q has not been specified.

The small gain derivation for the spherical μ upper bound is a combination of these two types of scaling. This combination is represented as a single scaling.

Alternate Proof to Theorem 3.5:

$$\delta^*\delta \leq \gamma^2$$

$$\gamma^2 I - \delta\delta^* \geq 0$$

Apply Lemma 1.2:

$$(\gamma^2 I - \delta\delta^*) \circ D \geq 0 \text{ for } D > 0$$

Using Lemma 1.1:

$$\gamma^2 I \circ D \geq \Delta D \Delta^*$$

$$\gamma^2 I \geq (I \circ D)^{-\frac{1}{2}} \Delta D \Delta^* (I \circ D)^{-\frac{1}{2}})$$

$$\Rightarrow \overline{\sigma}((I \circ D)^{-\frac{1}{2}} \Delta D^{\frac{1}{2}}) \leq \gamma \ \forall D > 0$$

Using a small gain argument and setting $X^*X = D$, there are no nontrivial solutions to the loop equations if:

$$\exists X > 0 : \ \overline{\sigma}\left((I \circ X^*X)^{\frac{1}{2}} M X^{-1}\right) < \frac{1}{\gamma}$$

$$\Leftrightarrow \exists D > 0 \colon M^*(I \circ D)M - \gamma^2 D < 0$$

□

Although the upper bound can be derived in this manner, it is less clear how to begin. There is a little black magic involved in knowing where to get started, which is the hard part of the derivation. In this case, reverse engineering the answer makes this derivation straightforward.

This bound can be derived without using the additional freedom from (3.21). An alternate derivation is based on the fact that the spherical constraint is an induced norm constraint. The sets described by $\{\Delta: \Delta \in \mathbf{\Delta}_s, \|\Delta\|_{\infty,2} < 1$ and $\{\Delta: \Delta \in \mathbf{\Delta}_s, \|\Delta\|_{2,1} < 1$ are the same as the set described by the spherical constraint. Alternate upper bounds for $\mu_s(M)$ are $\|DMD^{-1}\|_{2,\infty}$ and $\|DMD^{-1}\|_{1,2}$ where D has the standard commuting structure. Although these bounds are different for a particular D, the infimum over all D for both gains is the same, and this optimization problem can be converted to the same spherical μ upper bound (3.16).

The natural LMI upper bound formulations which are derived from the alternate induced norm formulations do not extend to some of the critical generalizations presented in Chapter 4.

Using the separating hyperplane formulation, the difficulty resides in finding a quadratic signal space description for the operator constraint, where the small gain formulation addresses the operators directly and implicitly addresses the signals satisfying the loop equations. Conceptually, the small gain formulation is easier to understand, but adding structure like real uncertainty and the intersection of regions are straightforward using the more abstract separating hyperplane argument. These constructions will be covered in more detail in Chapter 4.

Chapter 4

Generalizations of μ_S

The particular formulation for spherical μ from Chapter 3 is very restrictive and many of the uncertainty structures of interest within the generalized μ formulation by Chen [11] are more general but fall within a special class with ellipsoidal descriptions. These ellipsoidal uncertain parameter descriptions are obtained from parameter estimation procedures [25]. The problems that have been investigated are for the rank one case, because of tractability of the computation [9].

It is easy to define new formulations of μ, but those of interest are the ones that admit computable *tight* bounds. By extending the computational tools to the non rank one ellipsoidal μ problem, more general analysis can be performed for systems with such uncertainty representations.

Another reason to extend the machinery from spheres to ellipses is for the purpose of gridding the boundary of singularity. Spheres are not appreciably better than cubes for gridding a boundary. The advantage of ellipses is that by choosing them to be highly eccentric and aligned with the boundary, the act of gridding a hyperplane doesn't result in exponential growth.

The LMI upper bound computation for spherical μ (3.16) can be generalized from an uncertainty constrained to lie within a hypersphere to a hyperellipsoid. Further, the upper bound can be generalized to include other traditional uncertainty structures like repeated scalars, real parameters, and full blocks. These generalizations can be mixed and matched to form additional generalizations of the LMI in equation (1.16). Further, this new formulation will be unified with the standard μ formulation and generalized μ as developed by Newlin [31]. Further, there are some insights on the relationship between the non-repeated and repeated scalar upper bounds.

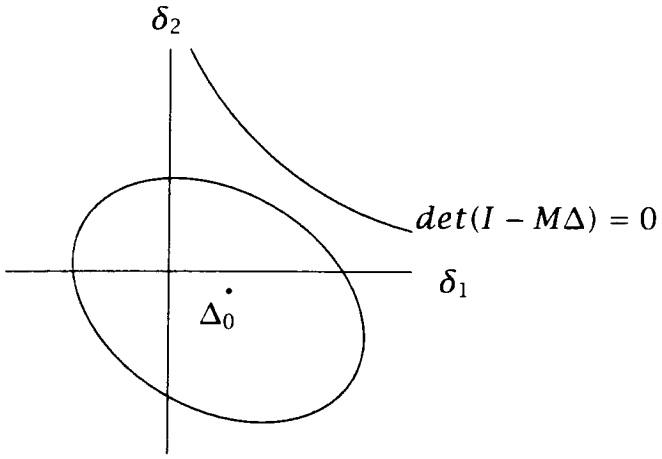

Figure 4.1: Ellipsoidal Region of Guaranteed Non-Singularity

4.1 Off-center HyperEllipsoids

The natural generalization of the hypersphere is the hyperellipsoid centered at some arbitrary point in space,

$$(\delta - \delta_0)^* P(\delta - \delta_0) < 1, P > 0. \tag{4.1}$$

μ for an off-center hyperellipsoid is defined by

$$\mu_e(M) \triangleq \frac{1}{\min\{\sqrt{(\delta - \delta_0)^* P(\delta - \delta_0)} : \Delta \in \mathbf{\Delta_s}, \det(I - M\Delta) = 0\}} \tag{4.2}$$

The graphical implication of this formulation of μ is shown in Figure 4.1.

It is trivial to generalize $\overline{\mu}_s$, equation (3.7), when P is a diagonal matrix to compute an upper bound, $\overline{\mu}_e(M)$, of elliptical μ, $\mu_e(M)$. For this case, the principle axes of the hyperellipsoid are aligned with the coordinate axes in $\mathbf{\Delta}$-space. As a result, the uncertainty and the matrix M can be rescaled and a linear fractional transformation performed so that the uncertainty satisfies a spherical constraint and is centered at the origin; the rescaling formulation follows. Assume that V is a diagonal matrix that has the associated radii of the hyperellipsoid as each diagonal element. It follows that $V^{-1}\Delta$ satisfies a spherical constraint. This implies that $\mu_e(M) = \mu_s(MV)$ and that the LMI upper bound for μ_s can be used to compute an upper bound for μ_e if the hyperellipsoid is axially aligned.

For the general hyperellipsoid (4.1), the proof in Theorem 3.5 immediately extends and an upper bound LMI solution can be constructed. The main modification to the proof of Theorem 3.5 is to replace the spherical

description (3.9) by the hyperellipsoid description (4.1). The implications of this modification is to replace δ by $\delta - \delta_0$, I by P^{-1}, and x by $x - \Delta_0 y$ in each of the subsequent lines of the proof.

The GEVP for an upper bound for μ_e for the general hyperellipsoid is given by

$$\overline{\mu}_e(M) \triangleq \inf_{D>0} \{\gamma : M^*(P^{-1} \circ D)M - \gamma^2(I - \Delta_0 M)^* D(I - \Delta_0 M) < 0\}. \tag{4.3}$$

For $\overline{\mu}_s(M)$, $P = P^{-1} = I$ and $\Delta_0 = 0$ and (4.3) reduces to the spherical μ upper bound (3.7) as would be expected from the derivation. Theorems 3.3, 3.8, and 3.9 and Corollaries 3.6 and 3.10 can be appropriately generalized for the elliptical formulation. Additionally, the structure of the GEVP in (4.3) can be cast as a cone-preserving LMI, and Theorem 5.4 can be applied to solve this GEVP efficiently.

4.2 Repeated Scalars

Often it is necessary to have multiple copies of a parameter in the LFT representation of a system. For example, if the square of the parameter is necessary or the parameter enters in two independent parts of the system, then two copies of the parameter are required. From a linear systems perspective, this is analogous to the need for a state vector for a system rather than a single scalar.

Within the standard μ-framework an uncertainty structure which admits analysis is the repeated scalar, where Δ contains repeated copies of δ_i on the diagonal. A fundamental block in the uncertainty structure is $\delta_i I$.

$$\Delta_{rs} \triangleq \{diag\,[\delta_1 I_{c_1}, \cdots, \delta_n I_{c_n}] : \delta_i \in \mathbb{C}\}. \tag{4.4}$$

This uncertainty structure can be fit into a spherical μ context. The only change to Definition 3.2 is that $\|\Delta\|_F$ is replaced by $\delta^*\delta$ where $\delta = \begin{bmatrix} \delta_1 \\ \vdots \\ \delta_n \end{bmatrix}$.

For this formulation, the LMI upper bound extends. The derivation from Theorem 3.5 must be modified to include this generalization. Remember that the spherical operator constraint (3.8) must be converted to a signal constraint. The repeated uncertainty structure means that multiple copies are needed in an inequality like (3.10) to properly apply the

congruence transformation which converts the set of operators into a signal space description of the uncertainty.

The solution to this problem is to embed the operator matrix inequality into a larger space. By applying the following to (3.10),

$$K^* \begin{bmatrix} \gamma^{-2} & \delta^* \\ \delta & I \end{bmatrix} K \geq 0, \tag{4.5}$$

and choosing

$$K = \begin{bmatrix} 1 & & & \\ & \mathbb{1}_{1\times c_1} & & \\ & & \ddots & \\ & & & \mathbb{1}_{1\times c_n} \end{bmatrix}$$

implies that (4.5) becomes

$$\begin{bmatrix} \gamma^{-2} & \hat{\delta}^* \\ \hat{\delta} & Q \end{bmatrix} \geq 0, \tag{4.6}$$

where $\hat{\delta}$ is the vector containing the main diagonal of Δ meaning that it contains the appropriate number of repeated copies of each δ_i and

$$Q \triangleq diag(\mathbb{1}_{c_1}, \cdots, \mathbb{1}_{c_n}). \tag{4.7}$$

After replacing (3.10) by (4.6), the remainder of the derivation of the upper bound is unchanged.

The resulting upper bound GEVP is given by

$$\mu_{s,\Delta_{rs}}(M) \leq \overline{\mu}_{s,\Delta_{rs}}(M) \triangleq \inf_{D>0} \{\gamma : M^*(Q \circ D)M < \gamma^2 D\}. \tag{4.8}$$

In the repeated scalar upper bound (4.8), Q is not an invertible matrix. Q enters the LMI as a term of a Hadamard product; as a result the invertibility of Q is irrelevant.

For $\overline{\mu}_s(M)$, $c_i = 1$ implies that $Q = I$ and subsequently the upper bound (4.8) reduces to $\overline{\mu}(M)$ (3.7). Theorems 3.3, 3.8, and 3.9 and Corollary 3.6 can be appropriately generalized for the repeated complex scalar formulation. Additionally, the structure of the GEVP in (4.8) can be cast as a cone-preserving LMI and Theorem 5.4 can be applied to solve this GEVP efficiently.

Investigating the obvious similarity of the (4.3) and (4.8) when $\Delta_0 = 0$, Q is the limit of a sequence of P_n^{-1}'s. In fact, P^{-1} can be thought of as a relaxation of the repeated structure in $\boldsymbol{\Delta}$-space as shown in Figure 4.2. As one of the axes of P_n^{-1} tends to 0, the M and Δ can be rescaled to construct an equivalent elliptical μ problem with a repeated scalar.

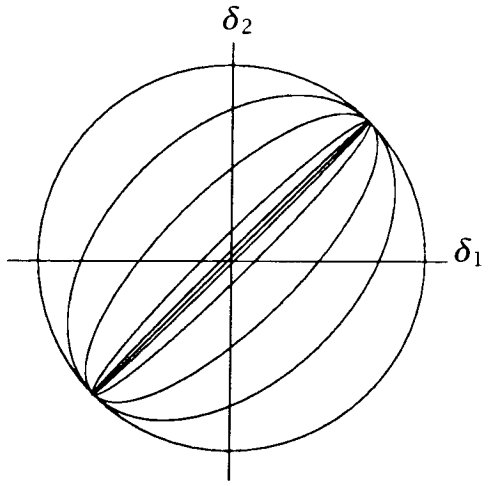

Figure 4.2: Ellipsoid to Repeated Parameter

4.3 Real Parameters

For real systems, representing parameters as being only norm bounded is conservative. If the parameter is physically motivated as a mass or a spring constant, it has the additional structure of being real. To reduce the conservatism of the analysis, this additional structure must be exploited as is done in the standard μ formulation.

The uncertainty structure for purely real non-repeated parameters is described by

$$\mathbf{\Delta_{rp}} \stackrel{\Delta}{=} \{diag\,[\delta_1, \cdots, \delta_n] : \delta_i \in \mathbb{R}\}.$$

The definition of μ_s is trivially extended.

The upper bound derivation in Theorem 3.5 can also be extended. The first observation is that spherical description for Δ, (3.9) and (3.10), describe a larger set than those of interest. They do not specify that the uncertainty is real. The constraint that

$$\Delta = \Delta^* \tag{4.9}$$

must be added to the quadratic form (3.10) and remains an additional constraint. Much like for the basic problem, this constraint must be converted to a signal constraint. Using

$$Y = \begin{bmatrix} y_1 & & \\ & \ddots & \\ & & y_n \end{bmatrix},$$

it follows that (4.9) is equivalent to

$$\begin{aligned} Y^*(\Delta - \Delta^*)Y &= 0 \ \forall Y \\ \Leftrightarrow I \circ \left((y^T)^* x^T - (x^T)^* y^T\right) &= 0 \\ \Leftrightarrow I \circ (yx^* - xy^*) &= 0 \\ \Leftrightarrow I \circ (Mxx^* - xx^*M) &= 0 \end{aligned}$$

The operator constraint due to the real structure of the parameters (4.9) has been converted to an equivalent signal constraint above. The separating hyperplane argument needs to be modified, because the signal constraint is an equality constraint rather than an inequality.

To apply the separating hyperplane argument appropriately to this problem, it is necessary to generalize Lemma 3.4. Specifically, algebraic constraints must be added to exploit the new equality constraint quadratic form. A trivial corollary of Lemma 3.4 follows.

Corollary 4.1 *Given* $A_{11} > 0, B \geq 0$ *then* $Tr\left(A\begin{bmatrix} B & 0 \\ 0 & 0 \end{bmatrix}\right) \geq 0.$

Applying the converse of Corollary 4.1 to

$$\begin{bmatrix} (I \circ (Mx)(Mx)^* - y^2 xx^*) & 0 \\ 0 & I \circ (Mxx^* - xx^*M) \end{bmatrix},$$

A_{21} and A_{12} have no effect and can be set to 0 without loss of generality. Setting $D = A_{11}$ and $G = A_{22}$ results in the following upper bound to $\mu_{s,\Delta_{rs}}(M)$,

$$\overline{\mu}_{s,\Delta_{rp}}(M) \stackrel{\Delta}{=} \inf_{D>0, G=G^*} \{\gamma : M^*(I \circ D)M + j((I \circ G)M - M^*(I \circ G)) < \gamma^2 D\}. \tag{4.10}$$

The additional freedom in the construction of the standard μ upper bound LMI in (1.16) associated with the G-scales corresponds to the constraint that δ_i is real and that the input and output signals of the block must be aligned in $\mathbb{C}$. The G-scales are independent of the magnitude and relationship between separate blocks. This observation will become clearer when the intersection of regions are considered.

4.4 Full Blocks

The uncertainty structure for only full block uncertainty is

$$\Delta_f := \{diag\,[\Delta_{fn_1}, \cdots, \Delta_{fn_F}] : \Delta_{fi} \in \mathbb{C}^{n_{fi} \times n_{fi}}\}.$$

Computing an upper bound for μ with spherical constraints on $\overline{\sigma}(\Delta_i)$ which includes non-scalar Δ_i, with

$$\boldsymbol{B\Delta_f} = \left\{ \Delta : \sum_{i=1}^{n} |\overline{\sigma}(\Delta_i)|^2 < 1 \right\}, \tag{4.11}$$

is not a natural extension of the LMI for diagonal uncertainty. For diagonal uncertainty in the standard μ problem, the structure of the D-scales can be specified by $Q \circ D$ where D is full. For full blocks the resulting structure in the D-scales cannot be specified using the Hadamard product of a constant matrix and a full D matrix.

Reduction

One method to construct an upper bound is to convert the original spherical $\mu_{s,\Delta_f}(M)$ problem with full blocks into a new problem, $\mu_s(\overline{M})$ where $\mu_{s,\Delta_f}(M) \leq \mu_s(\overline{M})$. $\overline{M}$ is constructed from M. M must be partitioned into M_{ij} which are compatible with the block structure of the uncertainty. What is meant by this is that M_{ij} maps the output of the j^{th} block of Δ_f to the input of the i^{th} block of Δ_f. $\overline{M}$ can be constructed as

$$\overline{M}_{ij} = \overline{\sigma}(M_{ij}).$$

The new uncertainty representation has a diagonal complex scalar structure which is the same as (3.1) with spherical level sets. The new problem has a smaller dimension matrix than the original problem and the upper bound has a closed form expression, but this bound may be conservative because $\mu_{s,\Delta_f}(M) \leq \mu_s(\overline{M})$ and a gap will introduce additional conservativeness.

Expansion

Another method is to construct a larger problem, with twice the matrix dimension and twice as many blocks in Δ, for which $\sqrt{\mu_{s,\Delta_f}(M)} = \mu_{g,\Delta_{new}}(\hat{M})$ and an LMI upper bound to the augmented problem, $\overline{\mu}_{g,\Delta_{new}}(\hat{M})$, can be constructed. The original problem is given by the standard μ loop in Figure 1.3, and has a spherical constraint on Δ. This is equivalent to the problem shown in Figure 4.3, where

$$\boldsymbol{\Delta}^1 = \{diag(\delta_1 I_{N_1}, \cdots, \delta_n I_{N_n}), \delta_i \in \mathbb{C}\}$$

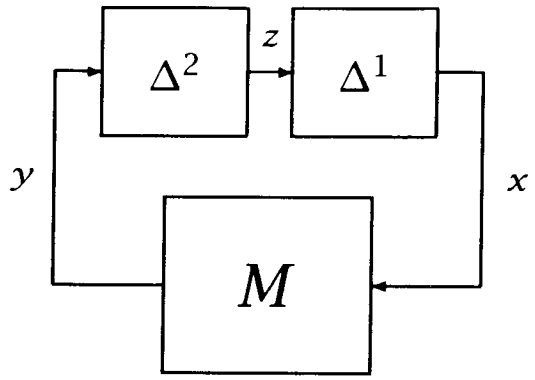

Figure 4.3: Expanded System

with the standard hypercube constraint that

$$|\delta_i| \leq 1, \forall i \tag{4.12}$$

and

$$\boldsymbol{\Delta}^2 = \boldsymbol{B}\boldsymbol{\Delta}_f.$$

The spherical constraint can be moved from Δ^2 to Δ^1 to form an equivalent system. So the hypercube constraint (4.12) is replaced by

$$\sum_{i=1}^{n} |\delta_i|^2 \leq 1$$

and the structure of $\boldsymbol{B}\boldsymbol{\Delta}$ is unchanged but the spherical constraint (4.11) is replaced by

$$\overline{\sigma}(\Delta_i^2) \leq 1 \ \forall i.$$

The system shown in Figure 4.3 can be put into the standard μ loop form using $\hat{M}$ and $\hat{\Delta}$ in (4.13) and (4.14). Let

$$\hat{\boldsymbol{\Delta}} = \left\{ \begin{bmatrix} \Delta^1 & 0 \\ 0 & \Delta^2 \end{bmatrix} : \Delta^1 \in \boldsymbol{\Delta}^1, \Delta^2 \in \boldsymbol{\Delta}^2 \right\} \tag{4.13}$$

and

$$\hat{M} = \begin{bmatrix} 0 & I \\ M & 0 \end{bmatrix}. \tag{4.14}$$

By construction,

$$\mu_{\mathcal{G},\hat{\Delta}}(\hat{M}) = \sqrt{\mu_{s,\Delta_f}(M)}. \tag{4.15}$$

To compute an upper bound for (4.15), the separating hyperplane argument can be extended to this construction. The end result is that full blocks with the standard constraints induce the standard D-scales for full blocks, and the spherical constraint on the repeated scalars induce the D-scales associated with (4.8). The resulting upper bound of $\mu_{\mathcal{G},\hat{\Delta}}(\hat{M})$ is given by

$$\inf_{D_1>0,D_2\in\boldsymbol{D}_2}\left\{\gamma : \hat{M}^*\begin{bmatrix} Q\circ D_1 & 0 \\ 0 & D_2 \end{bmatrix}\hat{M} - \gamma^2\begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix}\right\} \tag{4.16}$$

where Q is similar to (4.7), and $\boldsymbol{D}_2$ is the standard μ D-scales for full blocks; it consists of $d_i I_{n_i}$ blocks for $d_i > 0$.

Using simple manipulation, (4.16) can be rewritten as

$$\inf_{D_1>0,D_2\in\boldsymbol{D}_2}\left\{\gamma: \begin{array}{c} Q\circ D_1 < \gamma^2 D_2 \\ M^*D_2M < \gamma^2 D_1 \end{array}\right\}.$$

If Δ^2 consists of only scalar blocks and $Q = I$, then (4.16) is a cone-preserving LMI. It follows that at the infimum $I \circ D_1 = \gamma^2 D_2$, so D_2 can be eliminated and the spherical μ upper bound is recovered. See Chapter 5 for details.

4.5 Spheres in Δ^{-1}-space

Generalized μ, as coined by Newlin [31], has the same fundamental picture as standard μ as shown in Figure 1.3. What changes is the description of the set of Δ_i for some i. Rather than a small gain constraint like $\overline{\sigma}(\Delta_i) < 1$, there is a minimum gain constraint like $\underline{\sigma}(\Delta_i) > 1$. $\mu_{\mathcal{G}} < 1$ if and only if there are no non-zero solutions to the loop equations for the generalized set of allowable Δ.

The natural generalization of this new uncertainty construction to the spherical context is a sphere in Δ^{-1}-space,

$$\sum_{i=1}^{n}\left|\frac{1}{\delta_i}\right|^2 < \gamma^2.$$

To construct an upper bound for this problem, the derivation from Theorem 3.5 is largely unchanged. δ is replaced by δ^{-1} which denotes the element by element inverse. This forces the congruence transformation of (3.11) to be modified; specifically y_i is replaced by x_i and largely x and y swap roles in the remainder of the proof.

The resulting upper bound is

$$\overline{\mu}_{s,\Delta^{-1}}(M) := \inf_{D>0}\{\gamma : I \circ D - \gamma^2 M^* D M < 0\}. \tag{4.17}$$

An alternate way of looking at the problem is to reverse the direction of signal flow around the loop in Figure 1.3. The sphere in Δ^{-1}-space becomes a sphere in Δ-space, and M is replaced by M^{-1}. If M is not invertible, it immediately follows that $\mu_g(M) = \infty$ by choosing $x_0 (\neq 0)$ in $\mathcal{N}(M)$ which is a non-zero solution to the loop equation for all $\gamma > 0$. Hence the loop equations always have a non zero solution. The upper bound (4.17) verifies this by

$$x_0^* (I \circ D - \gamma^2 M^* D M) x_0 > 0 \ \forall \gamma.$$

4.6 Cartesian Product of Spheres

The central idea with new definitions of μ involve defining level sets in Δ-space. Again the definition of μ is trivially extended to $\mu_{cp,h}$ for which the level sets of interest are the cartesian product of h hyperspheres.

Associated with a spherical constraint in Δ-space there is a quadratic form. If there are multiple independent spherical constraints, then it follows that it can be represented as one large quadratic constraint. Subsequently, an LMI upper bound can be constructed.

The construction for the two independent spheres follows, and the extension to an arbitrary number of spheres is straightforward. Assume that Δ-space is partitioned into two subspaces so that

$$\delta = \begin{bmatrix} \overline{\delta} \\ \hat{\delta} \end{bmatrix},$$

and there is a spherical constraint on each subspace

$$\overline{\delta}^* \overline{\delta} < \gamma^2, \ \hat{\delta}^* \hat{\delta} < \gamma^2.$$

Converting this into two sets of signal constraints and rewriting it as a one signal constraint leads to the following quadratic form,

$$\begin{bmatrix} I \circ (\overline{y}\,\overline{y}^*) - \gamma^2 \overline{x}\,\overline{x}^* & 0 \\ 0 & I \circ (\hat{y}\hat{y}^*) - \gamma^2 \hat{x}\hat{x}^* \end{bmatrix} \geq 0. \tag{4.18}$$

Applying the separating hyperplane argument, and generalizing for h hyperspheres yields the following GEVP,

$$\overline{\mu}_{cp,h}(M) := \inf_{D>0} \left\{ \gamma : M^*(I \circ D)M - \gamma^2 \begin{bmatrix} \mathbb{1}_{s_1} & & \\ & \ddots & \\ & & \mathbb{1}_{s_h} \end{bmatrix} \circ D < 0 \right\}. \tag{4.19}$$

The standard μ level sets described by hypercubes is the cartesian product of n 1-d spheres. This formulation recovers the standard μ LMI upper bound when the uncertainty consists of only scalar blocks. This construction immediately generalizes to other uncertainty structures.

4.7 General Quadratic Descriptions

Some of the LMIs derived, for spherical μ (3.16), elliptical μ for $\Delta_0 = 0$ (4.2), repeated scalars (4.8), spheres in Δ^{-1} (4.17), and cartesian products of spheres (4.19), have a similar structure involving the Hadamard product. This structure is exploited in making the connection between elliptical μ and repeated scalars in Section 4.2. More generally, this similarity can be used to morph between a much larger class of problems.

The feasibility LMIs for the problems listed above are special cases of

$$M^*(P \circ D)M < (Q \circ D). \tag{4.20}$$

For the form in (4.20), the only restriction is that P and Q are symmetric. By reversing the construction of the LMI, an uncertainty description can be constructed for which (4.20) is a sufficient condition for testing if $\mu < 1$. This can be extended to an upper bound GEVP with the appropriate scaling of γ inserted into (4.20).

The reverse constructed uncertainty set description is

$$P \geq \Delta Q \Delta^*. \tag{4.21}$$

This quadratic description requires that Δ be strictly diagonal and complex, which may include repeated scalar blocks. The uncertainty sets for spherical μ and the generalizations which fall into this class aren't initially written in this form, but they are equivalent to constraints that can be written in this form.

$P = Q = I$ corresponds to the standard formulation of the upper bound for μ with non-repeated scalar uncertainty. $P = I$ and $Q = \mathbb{1}$ corresponds to the spherical μ upper bound. As Q is relaxed from I to $\mathbb{1}$ and using the continuum from spheres to repeated scalars in Figure 4.2, a continuum of problems is formulated from standard μ to standard μ with repeated scalars for which the upper bound extends. The level curves of the uncertainty sets associated with the relaxation from $Q = I$ to $\mathbb{1}$ are shown in Figure 4.4 with

$$Q = \begin{bmatrix} 1 & \epsilon \\ \epsilon & 1 \end{bmatrix}$$

for ϵ = 0, .1, .4, .7, .9, and1. Alternatively, the hypercube can be directly morphed to repeated scalars as shown in Figure 4.4 for which $P = Q$ using the values of Q from above.

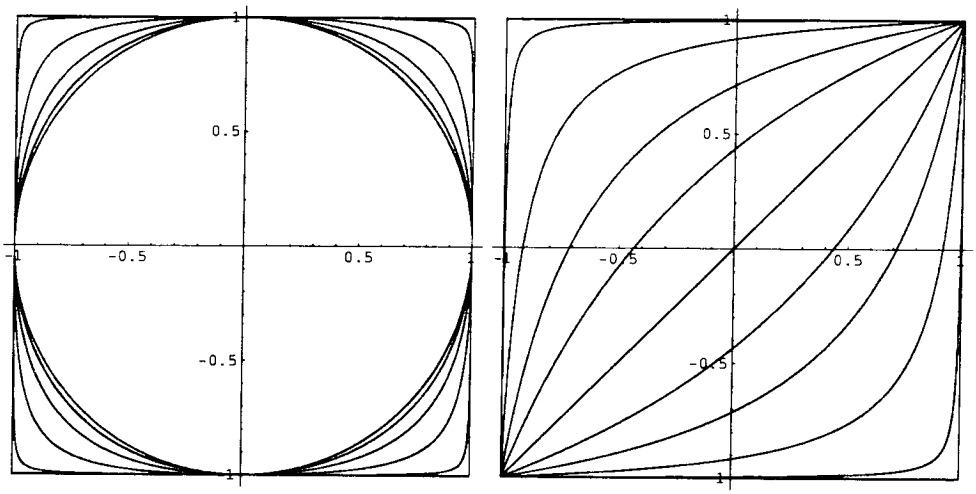

Figure 4.4: Morphing Hypercubes

Although more general quadratic descriptions for Δ-space can be defined, it is not always possible to convert these descriptions to quadratic forms on the signals x and y. This is necessary to apply the separating hyperplane argument to construct a sufficient LMI condition.

4.8 Intersection of Regions

The construction of a large quadratic form from smaller quadratic forms is an intersection of constraints. The construction presented in Section 4.6 for the cartesian product of spheres is the intersection of independent constraints. For the real parametric uncertainty construction in Section 4.3, the constraints are not independent in terms of the signals involved but do exploit independent aspects of the structure.

Representing the signal constraint as a single matrix inequality and the construction of the LMI upper bound does not require the independence of the constraints. So given two elliptical sets parameterized by y as shown in Figure 4.5, an LMI upper bound can be constructed that is tighter than the bounds associated with each elliptical constraint separately.

First, μ must be defined for the intersection of off center hyperellipsoid regions, and it will be called $\mu_{ic}(M)$ as a shorthand notation. Note that the

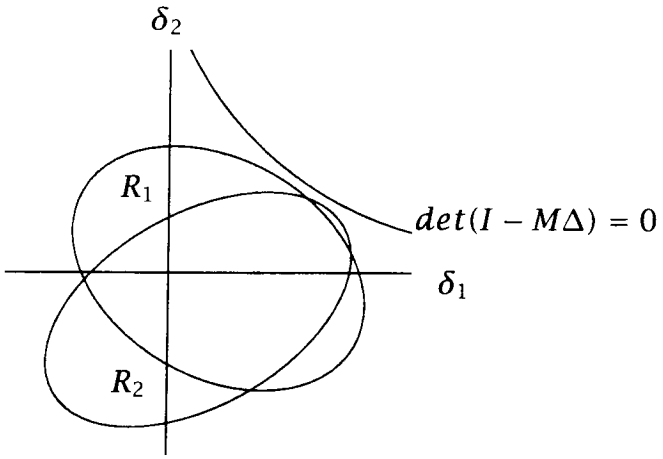

Figure 4.5: Intersection of Regions

regions haven't been specified in the notation. μ for the intersection of off center ellipsoid regions is defined by

$$\mu_{\cap}(M) \triangleq \frac{1}{\min\{\gamma\colon (\delta - \delta_{0_i})^* P_i(\delta - \delta_{0_i}) < \gamma^2 \ \forall i, \det(I - M\Delta) = 0\}}. \tag{4.22}$$

It follows that the upper bound using any one of the ellipses, $\overline{\mu}_e(M)$, is also an upper bound for $\mu_{\cap}(M)$ because by construction $\mu_{\cap}(M) \leq \mu_e(M)$, but a tighter LMI upper bound can be constructed.

The method of construction is essentially the same as those previously presented. The quadratic forms are written as a single structured quadratic form like the generalization of 4.18. The separating hyperplane argument is applied and an LMI upper bound can be constructed. The constructed LMI can be specified using the LMIs from each constraint independently.

Associated with γ and each off center ellipse is a feasibility LMI,

$$\{D_i \in \boldsymbol{D_i}\colon L_i(D_i, \gamma^2) < 0\}, \tag{4.23}$$

where L_i is a matrix valued linear function of a matrix D_i. For each value of γ an LMI can be constructed whose feasibility guarantees no solutions to the loop equations over the space of $\boldsymbol{\Delta}$ specified by the intersected regions. To compute the infimum of the value of γ for which the LMI is feasible is a GEVP. The upper bound GEVP associated with the intersection of the regions is given by

$$\overline{\mu}_{\cap}(M) \triangleq \inf_{D_i \in \boldsymbol{D_i}} \left\{ \gamma\colon \sum_i L_i(D_i, \gamma^2) < 0 \right\}. \tag{4.24}$$

Note that the LMIs in (4.23) must be compatibly constructed for (4.24) to be a valid upper bound.

For the purpose of constructing the LMI upper bound, the compatible construction requires that the same signals x and y are used in constructing the signal constraints. Otherwise the signals constraints are incompatible for applying the separating hyperplane argument. This forces each constraint to use the same Δ-space or equivalently M is the same for relating x and y for all of the constraints. This is the motivation for originally considering off-center hyperellipsoids of Section 4.1.

Chapter 5

Computation of the Bounds of Spherical μ

In a previous chapter, an upper bound for μ_e is derived. This bound is a convex optimization problem in the form of an LMI, specifically a GEVP. For problems of this form there exist numerical solvers that perform the optimization in polynomial time. These solvers are general LMI solvers and do not exploit all the special structure of a problem to optimize the numerical efficiency. It turns out that the upper bound LMI for μ_e has special structure that can be exploited. This structure converts the LMI optimization problem into the spectral radius of an associated linear operator.

Additionally, it was stated that a lower bound computation for μ_s could be constructed that is analogous to the standard μ lower bound computation. This computation involves the non-convex optimization problem from (1.11). This optimization is intractable, so in practice one must settle for local optimization. For μ_s, an alignment condition associated with a local extremum and the modification to the standard power algorithm are presented. This is a natural extension of the standard power algorithm μ lower bound computation.

5.1 Upper Bound Computation

The upper bound for μ_e derived is in the form of an LMI. The LMI formulation is a powerful tool; if a problem can be converted to an LMI, then computational cost of the optimization has polynomial growth and is generally considered tractable. The idea is that the problem has been reduced to a known "solved" problem.

An observation that can be made is that general LMI solvers are not the

most computationally efficient methods for all LMI problems. Often there exist reformulations of the problem that offer more computationally algorithms. In some sense, one can view these algorithms as exploiting special structure of an LMI to construct computationally efficient numerical solutions to a special class of LMIs.

An example of this is the LMI associated with solving the spectral radius of a matrix,

$$\rho(M) = \inf_{D>0} \left\{ \gamma : M^*DM - \gamma^2 D < 0 \right\}. \tag{5.1}$$

For this example, D is a full matrix and the answer to this optimization problem is the spectral radius of M. D being full means that there are $n(n+1)/2$ free variables in the LMI optimization, and the resulting optimization is much slower than computing the spectral radius using more specialized methods. There is a connection between the spectral radius computation and the structure of the LMI in (5.1); presumably there is special structure in this LMI that can be exploited in the computation to make the problem *easier* to compute.

An exact solution for the GEVP upper bound for elliptical μ (4.3) with $\Delta_0 = 0$ is presented. The formulation presented immediately extends to include repeated scalar blocks (4.8). In the remainder of this chapter, this class of problems will be referred to as elliptical μ or μ_e. By using a generalized version of the classical Perron-Frobenius theorem, the optimal value is shown to be equal to the spectral radius of an associated linear operator. This enables more efficient computation of the optimal solution, using other algorithms like the power iteration method. For a more thorough treatment of a broader class of cone-preserving LMIs for which the computation simplifies, the reader is referred to [36].

Background

All of the elements of a nonnegative matrix are real and nonnegative. Let $S \subset \mathbb{R}^n$, and define S^G as the set of all finite nonnegative scalings of the elements of S. Then, a set $\mathcal{K}$ is a *cone* if $\mathcal{K} = \mathcal{K}^G$. The *dual* of a set S is $S^* = \{y \in \mathbb{R}^n : x \in S \Rightarrow \langle x, y \rangle \geq 0\}$. A cone $\mathcal{K}$ is *pointed* if $\mathcal{K} \cap (-\mathcal{K}) = \{0\}$, and *solid* if the interior of $\mathcal{K}$ is not empty. A set S is *convex* if $x_1, x_2 \in S$ implies $\lambda x_1 + (1-\lambda)x_2 \in S \ \forall \lambda \in [0,1]$. A cone that is convex, closed, pointed and solid is called a *proper* cone.

In the vector space of symmetric real matrices, it is easy to show that the set of positive semidefinite matrices,

$$\mathcal{P} \triangleq \{D : D \in \mathbb{R}^{n \times n}, D = D^T, D \geq 0\},$$

defines a proper cone. The inner product of two matrices is given by

$$\langle A, B \rangle = Tr(AB).$$

This purpose of introducing this vector space is to think about LMIs as linear operators which act on matrices. These matrices are the variable over which to optimize. Additionally, this formulation makes it easier to exploit the special properties of the linear operators of interest, rather than a baroque description for the structure of the LMIs. $\mathcal{P}$ is crucial to describing the special structure of the LMIs which will be exploited to simplify the computation.

Problem Statement

The basic structure of the problem to be addressed is as follows: the LMI upper bound for elliptical μ can be written as

$$\gamma_0 = \inf_{D>0} \left\{ \gamma : \mathcal{L}(D) \triangleq M^*(Q \circ D)M < \gamma^2 D \right\}, \tag{5.2}$$

where $Q \geq 0$. Note that the spectral radius LMI (5.1) is a special case of the above operator with $Q = \mathbb{1}$.

A linear operator $\hat{\mathcal{L}}$ preserves the proper cone $\mathcal{P}$ if

$$D \geq 0 \Rightarrow \hat{\mathcal{L}}(D) \geq 0. \tag{5.3}$$

$\mathcal{L}$ satisfies the cone-preserving structure described in (5.3) as shown in Lemma 5.1.

Lemma 5.1 *Let Q be positive semidefinite. Then, the proper cone $\mathcal{P}$ of positive semidefinite matrices is preserved by the operator $\mathcal{L}(D) = M^*(Q \circ D)M$.*

Proof: $\mathcal{L}$ is the composition of the two operators $\mathcal{L}_1(D) = Q \circ D$ and $\mathcal{L}_2(D) = M^*DM$. The first operator is cone-preserving by Lemma 1.2. The second operator has the same property, because $D \geq 0$

$$\Rightarrow y^*Dy \geq 0 \ \forall y$$

choosing $y = Mx$,

$$\Rightarrow x^*M^*DMx \geq 0 \ \forall x \ \Rightarrow \mathcal{L}_2(D) \geq 0.$$

Therefore,

$$D \geq 0 \Rightarrow \mathcal{L}_1(D) \geq 0 \Rightarrow \mathcal{L}(D) = \mathcal{L}_2(\mathcal{L}_1(D)) \geq 0.$$

□

Perron-Frobenius

Thus far the LMI upper bound for μ_e has been modestly altered and the cone-preserving property of $\mathcal{L}$ has been identified. The purpose of verifying the cone-preserving structure of $\mathcal{L}$ is to exploit the spectral properties of such operators.

The motivation for the study of cone-preserving operators arose from investigations of nonnegative matrices. A matrix is nonnegative if it is square and each element is real and nonnegative. This nonnegative structure to matrices arises in Markov chains, Leontief economic models, as well as other domains. For a matrix M with nonnegative structure, the set of all vectors in the nonnegative orthant is a proper cone that is preserving under multiplication by M. This is the setup for the classical Perron-Frobenius theorem [22]. The implication is that the spectral radius of M is an eigenvalue, and there is a nonnegative eigenvector associated with the eigenvalue $\rho(M)$. The classical Perron-Frobenius theorem has been generalized significantly. For the problem at had, a particular finite dimensional version is stated which is sufficient for the GEVP problems of interest.

Theorem 5.2 (Perron-Frobenius [7]) *Assume that the linear operator $\hat{\mathcal{L}}$ maps the proper cone $\mathcal{K}$ into itself. Then*

1. *$\rho(\hat{\mathcal{L}})$ is an eigenvalue.*
2. *$\mathcal{K}$ contains an eigenvector of $\hat{\mathcal{L}}$ corresponding to $\rho(\hat{\mathcal{L}})$.*
3. *$\mathcal{K}^*$ contains an eigenvector of $\hat{\mathcal{L}}^*$ corresponding to $\rho(\hat{\mathcal{L}})$.*

Solving Cone-Preserving GEVPs

In the following, the optimal solution for the GEVP (5.2) and an associated optimal D are recharacterized, meaning the GEVP is converted to another optimization problem which readily admits more efficient computational techniques.

The GEVP problem (5.2) is relaxed by allowing the LMI variable D to be semidefinite. The new problem is

$$\hat{\gamma}_0 \triangleq \inf_{D \geq 0} \{\gamma : M^*(Q \circ D)M < \gamma^2 D\}. \tag{5.4}$$

For this problem, Lemma 5.3 defines a lower bound to $\hat{\gamma}_0$.

Lemma 5.3 *The optimal solution of (5.4) has as a lower bound*

$$\hat{\gamma}_0^2 \geq \rho(\mathcal{L}). \tag{5.5}$$

Proof: $\mathcal{L}$ preserves the proper cone $\mathcal{P}$, so Theorem 5.2 can be applied. Let $Y \geq 0$ be the eigenvector of $\mathcal{L}^*$ associated with the eigenvalue $\rho(\mathcal{L})$. It follows that

$$\begin{aligned} \mathcal{L}(D) &< \gamma^2 D \\ \Rightarrow \langle \mathcal{L}(D), Y \rangle &< \gamma^2 \langle D, Y \rangle \\ \Rightarrow \langle D, \mathcal{L}^*(Y) \rangle &< \gamma^2 \langle D, Y \rangle \\ \Rightarrow \langle D, \rho(\mathcal{L}) Y \rangle &< \gamma^2 \langle D, Y \rangle \\ \Rightarrow \rho(\mathcal{L}) \langle D, Y \rangle &< \gamma^2 \langle D, Y \rangle \\ \Rightarrow \rho(\mathcal{L}) &< \gamma^2. \end{aligned}$$

Therefore, γ^2 has to be strictly greater than the spectral radius of $\mathcal{L}$ for (5.2) to have a feasible solution. □

For a particular value of γ, if D is a feasible solution to the original problem (5.2), then D is a feasible solution to the relaxed problem (5.4). This implies that $\gamma_0 \geq \hat{\gamma}_0$.

Theorem 5.4 *The optimal solution of (5.2) has*

$$\gamma_0^2 = \rho(\mathcal{L}). \tag{5.6}$$

Proof: $\mathcal{L}$ preserves the cone $\mathcal{P}$, so Theorem 5.2 can be applied. Let $X \geq 0$ be the eigenvector of $\mathcal{L}$ associated with the eigenvalue $\rho(\mathcal{L})$.

Case 1: , $X > 0$: Using Lemma 5.3, $\gamma_0^2 \geq \rho(\mathcal{L})$.
Claim: There exist solutions for $\gamma^2 > \rho(\mathcal{L})$
choose $D = X > 0$,

$$\gamma^2 D - \mathcal{L}(D) = (\gamma^2 - \rho(\mathcal{L}))D > 0$$

$\Rightarrow \gamma_0^2 \leq \rho(\mathcal{L}) \Rightarrow \gamma_0^2 = \rho(\mathcal{L})$.

Case 2: , $X \in$ boundary $\mathcal{P}$: This case corresponds to the concept of cone-preserving operators not being *irreducible* [7]. Using Lemma 5.3, $\gamma_0^2 \geq \rho(\mathcal{L})$.
Claim: There exist solutions for $\gamma^2 > \rho(\mathcal{L})$.
If there exists a solution to

$$\mathcal{L}(D) - \gamma^2 D = -R < 0, \tag{5.7}$$

then $\gamma_0 < \gamma$. For $\gamma^2 > \rho(\mathcal{L})$, the system of linear equations in (5.7) has a unique solution.

$$(5.7) \equiv \frac{\mathcal{L}(D) + R}{\gamma^2} = D.$$

$$\rho\left(\frac{\mathcal{L}(D)}{\gamma^2}\right) < 1 \Rightarrow D_{k+1} = (\mathcal{L}(D_k) + R)/\gamma^2$$

with $D_0 = 0$ is a convergent iteration.

$$D_i \geq \frac{R}{\gamma^2} > 0 \ \forall i \geq 1$$

$$\Rightarrow D = \lim_{k \to \infty} D_k > 0 \text{ satisfies (5.7)}$$

$$\Rightarrow \gamma^2 \leq \rho(\mathcal{L}) \Rightarrow \gamma_0^2 = \rho(\mathcal{L})$$

□

An example of an operator which is not *irreducible* is the operator associated with the spherical μ upper bound when

$$M = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}.$$

The spectral radius of the operator is 1 and the associated eigenvector is $D = \mathbb{1}$. This eigenvector is only semi-definite and the construction of Case 2 is needed.

All of the "vectors" in the preceding setting are matrices. Note that the matrix representation may result in a more compact representation and more efficient computation than the standard vector notation. It is a way of taking advantage of the underlying structure of the linear vector space.

For the case where Q is the identity which corresponds to the spherical μ upper bound, the result from Theorem 5.4 can be simplified. First, an equivalent problem is constructed.

Lemma 5.5 *There exists a* $D_1 (\neq 0) \geq 0$ *such that*

$$M^*(I \circ D_1)M = \gamma^2 D_1 \tag{5.8}$$

if and only if there exists a $D_2 (\neq 0) \geq 0$ *such that*

$$I \circ (M^* D_2 M) = \gamma^2 D_2 \text{ and } I \circ D_2 = D_2. \tag{5.9}$$

Proof: $\Rightarrow$: Assume $\exists D_1 \geq 0$ satisfying equation (5.8), by choosing $D_2 = I \circ D_1$, it follows that $D_2 \geq 0$ and it satisfies equation (5.9).
$\Leftarrow$: Assume $\exists D_2 \geq 0$ satisfying equation (5.9), choosing $D_1 = \frac{M^* D_2 M}{\gamma}$ it follows that $I \circ D_1 = D_2$, $D_1 \geq 0$, and it satisfies equation (5.8). □

Using the equivalent formulation (5.9), a new but equivalent LMI upper bound for μ_s can be constructed. The linear operator, $L_2(D) = I \circ (M^* D M)$, preserves the proper cone of positive semidefinite diagonal matrices. As a result, this formulation can be used to construct a lower-dimensional linear operator for solving the μ_s LMI upper bound. The following corollary describes this construction.

Corollary 5.6 *Let γ_0 be the optimal solution of the GEVP:*

$$\gamma_0 = \inf_{D>0} \{\gamma : M^*(I \circ D)M - \gamma^2 D < 0\}.$$

Then,

$$\gamma_0^2 = \rho(M^T \circ M^*).$$

Proof: Using Lemma 5.5 and the cone-preserving structure of $L_2(D) = I \circ (M^* D M)$ defined over the linear vector space of diagonal matrices, it follows that $\gamma_0^2 = \rho(L_2)$. L_2 can be rewritten as a matrix N operating on $\mathbb{R}^n$ where the basis vector e_i is chosen to be the matrix of 0's with a 1 as the ii-entry. As a result

$$N = C^T (M^T \otimes M^*) C,$$

where

$$C = [c_1, \cdots, c_n]$$

and

$$c_i = (n(i-1) + i)^{th} \text{ column of } I_{n^2}.$$

C acts like a sieve on $M^T \otimes M^*$, and $N = M^T \circ M^*$. It follows that $\rho(N) = \rho(L_2)$; therefore, $\gamma_0^2 = \rho(M^T \circ M^*)$. □

So for the particular structure of the μ_s LMI upper bound, the optimization problem can be converted to computing the spectral radius of n-dimensional matrix. This is a far simpler computation than the original n-dimensional matrix inequality with $\frac{n(n+1)}{2}$ free variables. Also note the structure of N. All of the elements of N are real and nonnegative. This is the form of the problem for the original Perron-Frobenius Theorem. The eigenvector associated with the spectral radius is in the nonnegative orthant. This is the D matrix in the proper cone of positive semi-definite diagonal matrices.

Computation

In Theorem 5.4, the solution to elliptical μ upper bound was shown to be equivalent to the spectral radius of a related finite dimensional cone-preserving linear operator. In some sense the problem has been simplified, but this spectral radius still must be computed efficiently.

One possible method to compute the spectral radius is to construct a matrix representation of the operator and compute the eigenvalues. This solution computes more information than is necessary, and as a result it is inefficient for the optimization problem at hand.

Under the mild hypothesis of primitivity, which is a subset of irreducibility, it is possible to use power iteration-type methods to compute the spectral radius. Primitivity is equivalent to requiring the spectral radius of $\mathcal{L}(\cdot)$ to be strictly greater than the modulus of any other eigenvalue [7]. It is always possible to obtain a primitive operator by arbitrarily small perturbations of a non primitive operator.

Under this assumption, the simple iteration

$$D_{k+1} = \frac{\mathcal{L}(D_k)}{\|\mathcal{L}(D_k)\|} \tag{5.10}$$

is guaranteed to converge to the eigenvector associated with the spectral radius and its norm to the optimal value, for every initial value $D_0 > 0$. If primitivity is not satisfied, then this iteration will not necessarily converge.

An example of a non-primitive operator associated with $\mathcal{L}$ is given by

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

for spherical μ. For this operator, 1 and -1 are eigenvalues. The associated eigenvectors are I and

$$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

In fact, the spectral radius of $\mathcal{L}$ can be computed using a power iteration method even if the primitivity assumptions are eliminated. The solution is similar to the proof for the second case in Theorem 5.4. A cone-preserving operator is constructed such that the spectral radius of the new operator is related to the original and the new operator satisfies the primitivity assumption.

A possible choice for the new cone-preserving operator is

$$\hat{\mathcal{L}}(D) \triangleq \mathcal{L}(D) + D \text{ for } D \succ 0,$$

the eigenvalues of $\hat{\mathcal{L}}$ are the eigenvalues of $\mathcal{L}$ with real part increased by one and the associated eigenvectors are unchanged. As a result, $\rho(\hat{\mathcal{L}}) = \rho(\mathcal{L}) + 1$ and $\rho(\hat{\mathcal{L}})$ is only achieved by the eigenspace associated with the eigenvalue $\rho(\mathcal{L})$. So, if $\hat{\mathcal{L}}$ is used in the iteration (5.10), then the iteration converges and $\rho(\hat{\mathcal{L}})$ is computed.

There exist more sophisticated versions of the power iteration methodology, that result in faster convergence, as well as upper and lower bounds on the optimal value. This power iteration procedure is similar to the power-type algorithms usually used in computing μ lower bounds in 5.2, but is a much simpler problem because the problem is convex. Although the problems are similar, what is being computed is very different. The spectral radius computed for the μ_s upper bound is unrelated to the positive real spectral radius computed for the μ_s lower bound. They are eigenvalues of different operators.

Suboptimal solutions of LMIs

The cone-preserving requirement for the LMI is strict, since the implication is that in the limit the optimal solution actually satisfies an equality. The LMIs appearing in control problems are not necessarily satisfied by an equality at optimality. An example is the standard μ upper bound LMI, where the LMI variable D is not full, but structured. In other words, the partial order induced by the inequality is not the order induced by the variable D.

However, the methodology presented above can be used as an efficient method for computing suboptimal feasible solutions for certain problems. These suboptimal values can often be used as starting points for more general LMI solvers.

For example, for the standard μ upper bound LMI (1.16)

$$M^*(I \circ D)M - \gamma^2(I \circ D) < 0, \qquad D > 0,$$

it is possible to compute an approximate solution by using the following procedure:

1. Compute the exact solution γ_1^2, D_1 of the spherical μ upper bound LMI (3.7).

2. Compute the smallest η that satisfies (5.11)

$$D_1 \leq \eta^2(I \circ D_1). \tag{5.11}$$

This is a generalized eigenvalue problem that can be reduced to the computation of the maximum eigenvalue of a hermitian matrix. Because D is positive definite, it follows that $\eta^2 \leq n$ from Lemma 3.7.

3. A suboptimal solution of the LMI is given by $I \circ D_1$, and the optimal value is $\gamma = \eta\gamma_1 \leq \sqrt{n}\gamma_1$.

Effectively, we have

$$M^*(I \circ D_1)M \leq \gamma_1^2 D_1 \leq \eta^2\gamma_1^2(I \circ D_1).$$

It is possible to get arbitrarily close to the worst case gap of $\sqrt{n}$ between the optimal and suboptimal solutions for the upper bound of μ. For example, for the matrix

$$M = \begin{bmatrix} 1 & \varepsilon & \cdots & \varepsilon \\ \varepsilon & \varepsilon & \cdots & \varepsilon \\ \vdots & \vdots & \ddots & \vdots \\ \varepsilon & \varepsilon & \cdots & \varepsilon \end{bmatrix},$$

with ε arbitrarily small, the optimal value of the LMI (1.16) is 1, but the fast upper bound approaches $\sqrt{n}$ as $\varepsilon \to 0$. Note that for $\varepsilon = 0$, the fast upper bound is in fact exact (equal to 1). This is a consequence of the discontinuity of the generalized eigenvalue problem (5.11).

Another procedure for computing fast solutions of the μ LMI is the one developed by Osborne [33]. A preliminary comparison made with random, normally distributed matrices gives a slight advantage to the Osborne procedure. However, the algorithm proposed can give better upper bounds (the opposite is also possible), as the following example shows. For the matrix

$$M = \begin{bmatrix} 0 & -9 & -4 \\ 2 & 6 & 6 \\ -3 & -1 & 6 \end{bmatrix}$$

the μ upper bound computed by Osborne preconditioning is 10.321, and the bound of the proposed procedure is 9.69 (the value of the LMI upper bound is 9.6604, and is in fact equal to μ since there are three blocks).

Computational Advantage

As a simple example of the computational benefit of the proposed computation, the time required to compute the solution of the spherical μ LMI upper bound (3.7) is compared for a given problem.

Choosing M to be a 16×16 complex matrix, randomly generated, the computation of the optimal value of the LMI (3.7) with a general purpose LMI solver for MATLAB [18] and a tolerance set to 10^{-4} requires (on a Sun Ultra 1 140) approximately 160 seconds. Using the procedure presented here, either by power iteration or explicitly computing the eigenvalues, the answer can be obtained in less than one second.

5.2 μ_S Lower Bound

For standard μ, a lower bound is essential to assess the quality of the upper bound for a particular problem. The situation is no different for spherical μ. In the development presented here, the Standard Power Algorithm (SPA) [34] [29] with be extended to spherical constraints in Δ-space. The basis of SPA is in the conditions for local optimality of $\rho(M\Delta)$. As a result the extension of the SPA to computing a lower bound to μ_s hinges on the optimality condition for a different $\boldsymbol{B\Delta}$. In fact, this condition for μ_s is a natural extension for the standard condition. The other lower bound power type algorithms (SIA, WRA, ROA) presented in [29] are designed to improve the computation for the real and mixed μ. These algorithms can be extended to spherical μ, but these extensions are not interesting and will not be developed further.

Optimality Conditions

The local solutions of the optimization problem in (3.4) are to be characterized. This characterization of the necessary conditions for a local maximum will be key in the formulation of the iterative algorithm.

The following lemma is useful in the derivation for necessary conditions for optimality.

Lemma 5.7 *Let* $a, b, x \in \mathbb{C}^n$. *Assume that* $\mathrm{Re}(x^*b) \leq 0$ *for every* x *satisfying* $\mathrm{Re}(x^*a) \leq 0$. *Then* $b = \eta a$, *for some* $\eta \geq 0$.

Proof: Given that $\mathrm{Re}(x^*b) \leq 0$ for every x satisfying $\mathrm{Re}(x^*a) \leq 0$, it follows that $\mathrm{Re}(x^*b) \geq 0$ for every x satisfying $\mathrm{Re}(x^*a) \geq 0$. Therefore, $\exists \eta$ which is a nonnegative real valued function of x such that

$$\mathrm{Re}(x^*b) = \eta(x)\mathrm{Re}(x^*a) \quad \forall x \in \mathbb{C}^n.$$

Using

$$\mathrm{Re}(y^*z) = \frac{y^*z + y^*z}{2},$$

it follows that

$$x^*b + b^*x = \eta(x)(x^*a + a^*x) \ \forall x \in \mathbb{C}^n,$$

which can be rewritten as

$$x^*(b - \eta(x)a) + (b - \eta(x)a)^*x = 0 \ \forall x \in \mathbb{C}^n.$$

$$\Rightarrow b - \eta(x)a = 0$$

$$\Rightarrow \eta(x) = \text{ constant } \geq 0.$$

□

The optimization problem (3.4) is in general nondifferentiable at certain points. As usual, some generic nondegeneracy conditions are imposed. Specifically that the maximum eigenvalue of $M\Delta$ has multiplicity one at optimality, this ensures differentiability. Under these assumptions, Theorem 5.8 presents a necessary condition for local optimality.

Theorem 5.8 *Let* $\Delta_0 = \text{diag}(\delta), \delta = (\delta_1, \delta_2, \ldots, \delta_n), \delta_i \in \mathbb{C}$ *satisfy*

$$\sum_{i=1}^{n} |\delta_i|^2 \leq 1,$$

and x *and* y *be the right and left eigenvectors of* $M\Delta_0$*, with the normalization condition* $y^*x = 1$*. Then if* $\rho(M\Delta)$ *has a local maximum at* Δ_0*, the following conditions are satisfied:*

$$(M\Delta_0)x = \lambda x \tag{5.12}$$

$$y^*(M\Delta_0) = \lambda y^* \tag{5.13}$$

$$y_i^* x_i = |\delta_i|^2 \tag{5.14}$$

Proof: (5.12) and (5.13) follow immediatedly. Consider any curve $\Delta(t) \in \mathbf{B}\Delta \ \forall t$, with $\Delta(0) = \Delta_0$. By assumption, the maximum eigenvalue of $M\Delta(t)$ is simple at $t = 0$, so differentiating with respect to t yields:

$$\dot{\lambda}(0) = y^*M\dot{\Delta}(0)x = y^*M\text{diag}(v)x$$

where v satisfies $\text{Re}(v^*\delta) \leq 0$. Since the eigenvalue has maximum absolute value at Δ_0,

$$\text{Re}(y^*M \text{ diag}(v)x) = \text{Re}(y^*M \text{ diag}(x)v) \leq 0.$$

Apply Lemma 5.7 to obtain

$$y^* M \mathrm{diag}(x) = \eta \delta^*,$$

for some $\eta \geq 0$. Postmultiplying by δ to construct

$$y^* M \mathrm{diag}(x)\delta = y^* M \Delta_0 x = \lambda y^* x = \lambda = \eta \delta^* \delta = \eta.$$

Postmultiply by $\Delta_0 = \mathrm{diag}(\delta)$ to obtain

$$\begin{aligned} y^* M \mathrm{diag}(x)\, \Delta_0 &= \eta \delta^* \Delta_0 \\ y^* M \Delta_0\, \mathrm{diag}(x) &= \eta \delta^* \Delta_0 \\ \lambda y^* \mathrm{diag}(x) &= \lambda \delta^* \Delta_0 \end{aligned}$$

or equivalently, $y_i^* x_i = |\delta_i|^2$. □

The problem has been reduced to an algebraic nonlinear system. The approach is to solve this system of nonlinear equations by an iterative (or fixed point) procedure.

Changing variables, (5.12)-(5.14) (assuming $\|\bar{a} \circ w\| \neq 0$) can be rewritten in the form:

$$\beta a = Mb, \qquad z = \frac{\bar{w} \circ a}{\|\bar{w} \circ a\|} \circ w \tag{5.15}$$

$$\beta w = M^* z, \qquad b = \frac{\bar{a} \circ w}{\|\bar{a} \circ w\|} \circ a. \tag{5.16}$$

where $\bar{x} = (a^T)^*$ or equivalently the element by element complex conjugate. The equations have been written in this particular form for several reasons. First, they are similar to the those in [34, (7.14)]. More importantly, they can be used to define an iteration procedure to obtain candidate solutions of the optimization problem.

In particular, given initial values for b and w, a and z can be obtained from (5.15), and then update b and w using (5.16) (these vectors can be constrained to have unit norm). The repetition of this process gives rise to a sequence of vectors a_k, b_k, w_k, z_k; hopefully this iteration converges.

If the algorithm converges to a solution, a lower bound is obtained. Letting

$$\Delta_0 := \mathrm{diag}\left(\frac{\bar{a} \circ w}{\|\bar{a} \circ w\|}\right)$$

it follows that $(I - M\frac{1}{\beta}\Delta_0)$ is singular, and the associated μ_s lower bound is β.

Computational considerations

It is important to remark the fact that the characterization of candidate optimal solutions (i.e., that satisfy the alignment conditions), and the existence of an iteration that has those solutions as fixed points, does not imply anything about the convergence properties of the iterative procedure.

Ideally, the optimal solution would be a stable fixed point of the iteration, and would have a large basin of attraction. However, these properties depend strongly on the exact details of the procedure employed.

Numerical tests with the proposed iteration, starting from random initial conditions, usually give very fast convergence to a local maximum for most matrices. The computational requirements are very reasonable, since the algorithm only performs matrix and vector multiplications.

However, it has been observed that the algorithm *does not* perform satisfactorily in the class of matrices which have a strong antidiagonal component.

Chapter 6

Spherical μ Upper Bound - the Inverse Problem

In understanding the relationship between a μ computation and the associated upper bound, it is useful to determine the inverse problem for the upper bound. The term inverse problem refers to a μ problem which is a relaxation of the original problem for which the upper bound is exact. By a relaxation it is meant that $\mathbf{\Delta} \subset \mathbf{\Delta}_{new}$.

Within the standard μ-framework and the Integral Quadratic Constraint (IQC) framework, there have been a number of important results and insights regarding inverse problems, [41], [26], [35], and [37] in describing the uncertainty for which the standard μ upper bound in (3.7) is exact.

Using the results of Shamma [41], Megretski [26], and the extensions by Paganini [35], the μ upper bound, $\overline{\mu}_{\Delta}$, is exact if Δ is relaxed from a structured constant matrix operating on a vector to a structured operator acting on a vector valued signal $y[n]$. The matrix induced norm, $\overline{\sigma}(\Delta)$, is relaxed to the operator induced norm $\|\Delta\|_{l_2^m - ind}$.

To better understand the relationship between spherical μ (3.3) and the associated upper bound (3.16), these relaxation ideas are extended to spherical μ and are connected to observations regarding the spherical full block uncertainty upper bound construction in Section 4.4.

6.1 Motivation

In the extension of spherical μ to include full blocks as developed in Section 4.4, there is a spherical constraint on the norms of the full block matrix operators. The expanded upper bound presented in Section 4.4 relies on the construction of a related augmented μ_g problem and moving the spherical constraint from the full blocks to the new repeated scalar

blocks as shown in Figure 4.3.

Figure 6.1. Alternatively, the order of Δ^1 and Δ^2 can be reversed to

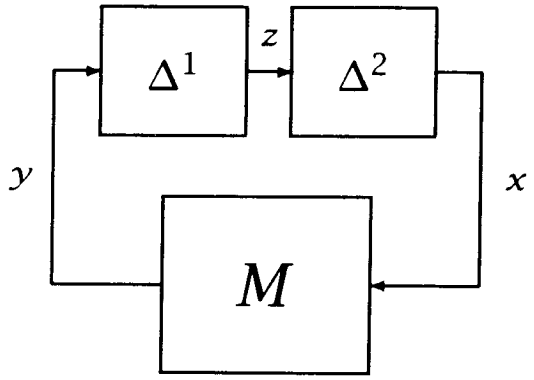

Figure 6.1: Another Expanded System

construct another equivalent μ_g problem as shown in Figure 6.1. Both of these new representations can be rewritten as μ_g problems and used to construct an upper bound LMI. With both of the uncertainty descriptions in Figures 4.3 and 6.1, it is possible to swap the spherical and standard constraints without altering the admissible relationships between the vectors x and y. Using $\hat{\mathbf{\Delta}}$ from (4.13), $\boldsymbol{B\hat{\Delta}}$ is defined with a spherical constraint on $\mathbf{\Delta^1}$ and a standard constraint, $\|\Delta_i^2\| \leq 1$, on $\mathbf{\Delta^2}$ to define new equivalent uncertainty representations.

$\boldsymbol{B\hat{\Delta}}$ in the new uncertainty descriptions are equivalent to the unit ball in the original uncertainty description as shown in Lemma 6.1.

Lemma 6.1

$$\bigcup_{\Delta \in \boldsymbol{B\Delta}_f} \Delta = \bigcup_{diag(\Delta^1,\Delta^2)\in \boldsymbol{B\hat{\Delta}}} \Delta^2\Delta^1 = \bigcup_{diag(\Delta^1,\Delta^2)\in \boldsymbol{B\hat{\Delta}}} \Delta^1\Delta^2$$

Proof: By construction, if $\Delta \in \boldsymbol{B\Delta}_f$ then choose $\delta_i = \|\Delta_i\|$ and $\Delta_i^2 = \frac{\Delta_i}{\|\Delta_i\|}$. If $\|\Delta_i\| = 0$ then set $\Delta_i^2 = 0$. This implies that the left-hand set above is contained within the other two sets. Similarly, if $diag(\Delta^1, \Delta^2) \in \boldsymbol{B\hat{\Delta}}$, then choose $\Delta_i = \delta_i^1 \Delta_i^2$, which implies that the right two sets above are contained within the left set. The right equality follows by transitivity. □

So rather than the canonical μ loop as shown in Figure 1.3, a new μ_g problem has been constructed as shown in Figures 4.3 and 6.1. These problems can be converted to a canonical μ loop as shown in Figure 6.2.

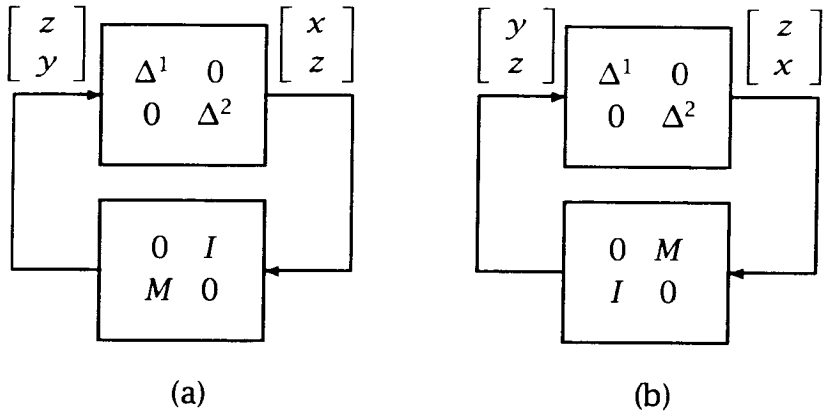

Figure 6.2: Conversion to Standard Interconnection

Theorem 6.2 *The original loop has a nontrivial solution on $B\Delta_f$ if and only if the loops shown in Figure 6.2 have a nontrivial solution on $B\hat{\Delta}$.*

Proof: This follows immediately from Lemma 6.1 and the respective equivalence of Figures 6.2, 4.3, and 6.1. □

The relationship between μ_s for the original problem and μ_g for this new constructions is given by

$$\sqrt{\mu_{s,\Delta}(M)} = \mu_{g,\hat{\Delta}}\left(\begin{bmatrix} 0 & I \\ M & 0 \end{bmatrix}\right) = \mu_{g,\hat{\Delta}}\left(\begin{bmatrix} 0 & M \\ I & 0 \end{bmatrix}\right).$$

The square root is needed to relate the uncertainty descriptions as parameterized by the scale factor acting on the associated unit balls. The uncertainty in the original problem has been described by the product of two new uncertainties, and the square root is used to counteract the effect of how the product scales with γ. This is the correction needed to generalize Lemma 6.1 for all balls (not just the unit ball) in Δ-space.

Using the LMI upper bound for μ_g which follows from the results in Chapter 4, an LMI upper bound can be constructed for the problems shown in Figure 6.2. For the first equivalent construction in Figure 6.2a, the resulting upper bound is given by

$$\mu_{g,\hat{\Delta}}\left(\begin{bmatrix} 0 & I \\ M & 0 \end{bmatrix}\right) \le \inf_{D_1>0, D_2 \in \mathbf{D}_2} \left\{ \gamma : \begin{array}{c} P \circ D_1 < \gamma^2 D_2 \\ M^* D_2 M < \gamma^2 D_1 \end{array} \right\}, \tag{6.1}$$

where $\mathbf{D}_2$ is the standard μ D-scales for full blocks,

$$D_2 = \begin{bmatrix} \delta_1 I_{n_1} & & \\ & \ddots & \\ & & \delta_n I_{n_n} \end{bmatrix},$$

and P is the commuting structure associated with the repeated structure of Δ^1,

$$P = \begin{bmatrix} \mathbb{1}_{k_1} & & \\ & \ddots & \\ & & \mathbb{1}_{k_n} \end{bmatrix}.$$

For the second equivalent construction shown in Figure 6.2b, the resulting upper bound is given by

$$\mu_{g,\Delta^2}\left(\begin{bmatrix} 0 & M \\ I & 0 \end{bmatrix}\right) \leq \inf_{D_1>0, D_2 \in \mathbf{D}_2} \left\{ \gamma : \begin{array}{c} D_2 < \gamma^2 D_1 \\ M^*(P \circ D_1)M < \gamma^2 D_2 \end{array} \right\}, \tag{6.2}$$

with P and D_2 being the same as for the first construction. What is the relationship between (6.1) and (6.2)?

To investigate this relationship, choose $\mathbf{\Delta}_f$ to be composed of n non-repeated complex scalar blocks. For this case, the original problem reduces to spherical μ from Definition 3.2, and an upper bound is given by (3.7). For this choice of $\mathbf{\Delta}_f$, (6.1) becomes

$$\sqrt{\mu_{s,\mathbf{\Delta}_f}(M)} \leq \inf_{D_1>0, D_2>0} \left\{ \gamma : \begin{array}{c} I \circ D_1 < \gamma^2 I \circ D_2 \\ M^*(I \circ D_2)M < \gamma^2 D_1 \end{array} \right\}.$$

This is a cone-preserving LMI as introduced in Chapter 5, so at optimality for the infimum the inequality is satisfied by an equality. The resulting upper bound is

$$\mu_s(M) \leq \min_{D_2 \geq 0, D_2 \neq 0} \left\{ \gamma^2 : M^*(I \circ D_2)M = \gamma^4 D_2 \right\},$$

which results in exactly the same solution as the spherical μ upper bound from (3.7). On the other hand, (6.2) becomes

$$\sqrt{\mu_s(M)} \leq \inf_{D_1>0, D_2>0} \left\{ \gamma : \begin{array}{c} I \circ D_2 < \gamma^2 D_1 \\ M^*(I \circ D_1)M < \gamma^2 I \circ D_2 \end{array} \right\}.$$

This reduces to

$$\mu_s(M) \leq \inf_{D_1>0, D_2>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma I \circ D_2 < \gamma^2 D_1 \right\}. \tag{6.3}$$

In the following, this upper bound (6.3) will be shown to be equivalent to the standard μ LMI upper bound (1.16). Choosing

$$D_2 = \gamma I \circ D_1 \text{ and } D_1 = I \circ D_1,$$

it follows that

$$\begin{aligned} \inf_{D_1>0, D_2>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma I \circ D_2 < \gamma^2 D_1 \right\} &\leq \\ \inf_{D_1>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma^2 I \circ D_1 \right\} &= \overline{\mu}(M). \end{aligned}$$

Alternatively, if

$$\begin{aligned} X &> I \circ Y \\ \Rightarrow I \circ (X - I \circ Y) &> 0 \quad \text{using Lemma 1.2} \\ \Rightarrow I \circ X &> I \circ Y \end{aligned}$$

Applying this to the inequality $\gamma I \circ D_2 < \gamma D_1$, it follows that

$$\begin{aligned} \inf_{D_1>0, D_2>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma I \circ D_2 < \gamma^2 D_1 \right\} &\geq \\ \inf_{D_1>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma^2 I \circ D_1 \right\}. \end{aligned}$$

Hence,

$$\inf_{D_1>0, D_2>0} \left\{ \gamma : M^*(I \circ D_1)M < \gamma I \circ D_2 < \gamma D_1 \right\} = \overline{\mu}(M).$$

Therefore, this LMI upper bound (6.3) is equivalent to the standard μ upper bound in (1.16).

Why does commuting Δ^2 and Δ^1 result in a different upper bound? The implication of the above observation is in the proper relaxation of the problem which makes the upper bound exact; Lemma 6.1 does not hold. For the scalar block case presented, the left equality holds, but the set on the right is the standard ball in Δ-space as defined by $\| \cdot \|_\infty$.

6.2 The Inverse Problem

In trying to understand the reason for the difference between the two upper bounds developed (6.1) and (6.2), it is necessary to know for what problem is the spherical μ upper bound exact. The problem for which the upper bound is exact is of the standard μ loop form as shown in Figure 1.3. M is still a matrix, but Δ will be relaxed from an element in a set of constant

matrices to an element of a set of operators on signals. Subsequently, x and y switch from vectors to vector valued signals. The solution of the inverse problem is a description of the set of operators for which the upper bound is exact.

For the standard μ formulation without spherical constraints, the upper bound inverse problem for full blocks was answered by Shamma and Megretski in [41] and [26], respectively. This was extended to the repeated complex scalar case by Paganini [35]. For the standard case, the solution to the inverse problem is to replace $\|\Delta\| \leq 1$ by $\|\Delta\|_{l_2^m - ind} \leq 1$. Additionally, if Δ is constrained to be a nonlinear time-varying (NLTV) operator or linear time varying (LTV) operator, the upper bound is still exact. The reason is that the signal space description of the uncertainty is unchanged with this added structure.

Leveraging off the results of [26] and [35] for IQCs and matrix valued IQCs, the upper bound can be shown to be exact for the integral version of the quadratic form associated with the spherical uncertainty set (3.13). The difference between this construction and those of Megretski and Paganini is that for 1 quadratic form their LMI upper bounds are exact for the vector form. As a result, the methodology used to close the gap for multiple quadratic forms is used to close the gap for the one quadratic form in this presentation. The extension to the generalizations in Chapter 4 is straightforward.

The possible values of the quadratic form are given by

$$\boldsymbol{S} = \bigcup_{x \in \mathbb{X}} I \circ \Lambda(Mx, Mx) - \Lambda(x, x) \; (= Q(x)). \tag{6.4}$$

If $\mathbb{X} = \mathbb{C}^n$ then $\boldsymbol{S}$ is a cone. If the LMI upper bound is feasible, then there is a hyperplane which strictly separates, except for the $x = 0$, $\boldsymbol{S}$ from the cone of positive semi-definite matrices. For this case $\boldsymbol{S}$ is not necessarily convex, which leads to the gap between μ_s and its upper bound.

If $\mathbb{X}$ is relaxed to the set of all l_2^n signals, then the closure of $\boldsymbol{S}$, $\overline{\boldsymbol{S}}$, is convex. The construction that follows is identical to that presented in [35]. Given

$$Q(x_1) \text{ and } Q(x_2) \in \boldsymbol{S}$$

then it follows that

$$Q\left(\frac{z^{-n}x_1 + z^n x_2}{2}\right) \in \boldsymbol{S} \; \forall n$$

and

$$\lim_{n \to \infty} Q\left(\frac{z^{-n}x_1 + z^n x_2}{2}\right) = \frac{Q(x_1) + Q(x_2)}{2} \in \overline{\boldsymbol{S}}$$

where z^{-1} is the standard shift operator. If there is a non-trivial intersection between the cone of positive semi-definite matrices and $\overline{S}$, then there is still an intersection if $\mathbb{X}$ is further restricted to $\|x\|_{l_2} < 1$ because without this restriction S is a cone. With this choice of $\mathbb{X}$, $\overline{S}$ is convex and compact. The set of positive semi-definite matrices is convex. It follows that the separating hyperplane defined by the LMI variable D in (3.16) is necessary and sufficient for the infimum [39].

The spherical uncertainty quadratic form (3.13) is a signal based description of the uncertainty operator $B\Delta_s$ for which the upper bound is exact. The following theorem relates the signal space description to the operator space description.

Theorem 6.3 *The set of input-output signals consistent with*

$$\left\| \left(c_1 \delta_1 \frac{1}{c_1} \right), \cdots, \left(c_n \delta_n \frac{1}{c_n} \right) \right\|_{l_2^n, l_2^1 - ind} \leq 1 \ \forall c_i \in \mathbb{C} \text{ where } \delta_i y_i = x_i \tag{6.5}$$

is exactly the set described by the following IQC:

$$I \circ \Lambda(y, y) - \Lambda(x, x) \geq 0 \tag{6.6}$$

Proof:

$$\left\| \left(c_1 \delta_1 \frac{1}{c_1} \right) \cdots \left(c_n \delta_n \frac{1}{c_n} \right) \right\|_{l_2^n, l_2^1 - ind} \leq 1 \ \forall c_i \in \mathbb{C}$$

by definition:

$$\Leftrightarrow \sup_{\overline{y} \in l_2^n} \frac{\| \sum_{i=1}^n c_i \delta_i \frac{1}{c_i} \overline{y}_i \|^2}{\sum_{i=1}^n \| \overline{y}_i \|^2} \leq 1 \ \forall c_i \in \mathbb{C}$$

$$\Leftrightarrow \sup_{\overline{y} \in l_2^n} \left(\sum_{i,j=1}^n c_i^* c_j \left\langle \delta_i \frac{1}{c_i} \overline{y}_i, \delta_j \frac{1}{c_j} \overline{y}_j \right\rangle - \sum_{i=1}^n \| \overline{y}_i \|^2 \right) \leq 0 \ \forall c_i \in \mathbb{C}$$

applying a change of variables: $y_i = \frac{1}{c_i} \overline{y}_i$;

$$\Leftrightarrow \sup_{y \in l_2^n} \left(\sum_{i,j=1}^n c_i^* c_j \langle \delta_i y_i, \delta_j y_j \rangle - \sum_{i=1}^n \langle c_i y_i, c_i y_i \rangle \right) \leq 0 \ \forall c_i \in \mathbb{C}$$

using $x_i = \delta_i y_i$;

$$\Leftrightarrow \sum_{i,j=1}^n c_i^* c_j \langle x_i, x_j \rangle - \sum_{i=1}^n \langle c_i y_i, c_i y_i \rangle \leq 0$$
$$\forall c_i \in \mathbb{C}, \ \forall \text{ admissible signals } x_i \text{ and } y_i$$

using $c^T = [c_1, \cdots, c_n]$;

$$\Leftrightarrow c^*(\Lambda(x,x) - I \circ \Lambda(y,y))c \leq 0 \ \forall c_i \in \mathbb{C}$$

$$\Leftrightarrow I \circ \Lambda(y,y) - \Lambda(x,x) \geq 0.$$

□

In the LTV case, the operator description for the uncertainty can be simplified.

Corollary 6.4 $\|\delta_1, \cdots, \delta_n\|_{l_2^n, l_2^1 - ind} \leq 1$ *for* $\delta_i \in \delta_{LTV}$ *where* $\delta_i y_i = x_i$

$$\Leftrightarrow I \circ \Lambda(y,y) - \Lambda(x,x) \geq 0.$$

Proof: Theorem 6.3 holds if δ_i is restricted to be LTV. For LTV uncertainty, $c_i \delta_i \frac{1}{c_i} = \delta_i$, this simplifies $\|(c_1\delta_1\frac{1}{c_1}) \cdots (c_n\delta_n\frac{1}{c_n})\|_{l_2^n, l_2^1 - ind} \leq 1 \ \forall c_i$ to $\|\delta_1 \cdots \delta_n\|_{l_2^n - l_2^1 - ind} \leq 1$. □

If the operator conditions in Theorem 6.3 and Corollary 6.4 are restricted to matrices, the result is the vector constraint

$$\|\delta_1 \cdots \delta_n\|_2 \leq 1$$

which is equivalent original spherical constraint from (3.2).

6.3 Revisiting Full Blocks

With the conditions for exactness of the upper bound, the full block problem is revisited. To understand the gap between the two bounds for the spherical full block structured uncertainty case, only the signal representation of the uncertainty is used. Referring back to Figure 4.3 for the case where all the full blocks are scalar, the set of matrix valued IQCs describing the uncertainty in signal space are

$$I \circ \Lambda(x,x) \geq I \circ \Lambda(z,z) \geq \Lambda(y,y).$$

By choosing $x = z$, this reduces to

$$I \circ \Lambda(x,x) \geq \Lambda(y,y),$$

which is exactly the IQC associated with spherical uncertainty or the integral version of (3.13).

For Figure 6.1, the set of matrix valued IQCs describing the uncertainty are

$$\begin{array}{rcl} I \circ \Lambda(x,x) & \geq & \Lambda(z,z) \\ I \circ \Lambda(z,z) & \geq & I \circ \Lambda(y,y). \end{array} \tag{6.7}$$

The relationship between x and y is not clear. Theorem 6.5 shows that this relationship is equivalent to the IQC related to the standard μ uncertainty description.

Theorem 6.5 *The constraint on allowable signal pairs x,y imposed by (6.7) is equivalent to the constraint imposed by*

$$I \circ \Lambda(x,x) \geq I \circ \Lambda(y,y).$$

Proof:

$\Leftarrow$:

Choose

$$z_i[l] = \begin{cases} 0 & l \neq kn+i, \forall k \in \mathbb{Z} \\ \left(\sum_{j=kn+1}^{(k+1)n} x_i^*[j]x_i[j]\right)^{\frac{1}{2}} & \exists k \in \mathbb{Z} : \; l = nk+i \end{cases}$$

For this choice of z:

$$\langle z_i, z_j \rangle = \begin{cases} 0 & i \neq j \\ \langle x_i, x_i \rangle & i = j \end{cases}$$

Given:

$$I \circ \Lambda(x,x) \geq I \circ \Lambda(y,y)$$

Using the above construction for z,

$$\Rightarrow \exists z : \; I \circ \Lambda(x,x) = \Lambda(z,z) = I \circ \Lambda(z,z) \geq I \circ \Lambda(y,y)$$

$\Rightarrow$:

$$I \circ \Lambda(x,x) \geq \Lambda(z,z) \Rightarrow I \circ \Lambda(x,x) \geq I \circ \Lambda(z,z)$$

$$\Rightarrow I \circ \Lambda(x,x) \geq I \circ \Lambda(y,y)$$

□

The signal space description

$$I \circ \Lambda(x,x) \geq I \circ \Lambda(y,y)$$

is the signal space description for the standard μ upper bound inverse problem. So the representation shown in Figure 6.1 with the spherical constraint on the left uncertainty is equivalent to Δ with a standard uncertainty structure in the operator case.

By solving the inverse problem, it is clear why there is a gap between the upper bounds for the problems shown in Figure 6.2. Conceptually, what is happening is that the admissible signals for the spherical uncertainty structure are almost the same as the standard case, but there are additional constraints on the output of the spherical block from terms like $\langle z_i, z_j \rangle$. When the output of the spherical uncertainty goes directly into a standard uncertainty, only the energy into the various channels of the standard uncertainty are important. So with a particular choice of the signals $z_i[n]$, the additional constraints are not active. If the output of the spherical uncertainty directly enters M, then there is no way to get around the constraints on $\langle z_i, z_j \rangle$. Hence the apparent inconsistency between the two formulations has been resolved.

Chapter 7

μ with Linear Cuts

Although the ellipsoidal regions may be sufficient to deal with the generic exponential growth encountered in the probabilistic formulation of μ presented in Chapter 2, the accounting involved in tracking the resulting regions in uncertainty space and the associated probability computation become intractable. The natural choice for the fundamental region or unit ball in Δ-space is the standard hypercube with an additional linear constraint. This linear constraint need not be axially aligned and should be chosen to be aligned with the boundary of singularity to efficiently grid this boundary.

Further motivation for linear cuts is from the probabilistic μ problem when M has rank-one. For this problem the boundary of singularity is a hyperplane. Armed with the knowledge that M is rank-one, it is easy to answer the probabilistic μ problem. Using the approach in Chapter 2, this computation still experiences exponential growth in computation generically, because of the need to grid the boundary of singularity. The B&B algorithm isn't exploiting the special structure of the problem.

For the purposes of the upper bound computations presented here, the linear constraint is given *a priori* and need not be chosen to be aligned with the boundary of singularity.

Three different methods for constructing μ LMI upper bounds for hypercubic regions with a linear cut are developed. One of the methods is based upon the generalizations of spherical μ and the associated upper bound computations presented in Chapter 4, specifically elliptical μ and the intersection of regions.

The next method involves constructing an equivalent implicit μ problem; by equivalent it is meant that implicit problem generically has a nontrivial solution if and only if the originally linear cut problem has a nontrivial solution. Effectively, the signal constraints of the implicit framework are used to enforce the linear cut in Δ-space.

The final method converts the linear cut problem to a standard μ problem for which μ is greater than or equal to μ for the linear cut problem. It follows that the standard upper bound for this problem is also an upper bound for μ with a linear cut. The conversion involves a change of variables in Δ-space and the associated matrix M is higher dimensional.

The quality and computational complexity of each method is discussed. Exact bounds are achieved for rank-one problems with all the methods under certain assumptions. Comparison of the three approaches for random matrices are given through numerical examples.

7.1 μ with Linear Cuts

Only nonrepeated scalar uncertainty structures are considered here. Ultimately branching is only intended for real scalar uncertainties, and it is easy to extend the results to the repeated scalar case. The standard μ region of a hypercube is generalized by the addition of a linear constraint on the real uncertain parameters to the level sets in uncertainty space. These level sets define the unit ball $\boldsymbol{B\Delta}$ in Δ-space. The non-axially aligned constraints are of the form

$$|c_0 + c^T\delta| \leq \gamma, \tag{7.1}$$

where $c = [c_1, c_2, \cdots, c_n]^T \in \mathbb{R}^n$, and $\delta = [\delta_1, \delta_2, \cdots, \delta_n]^T$ corresponds to the parameters in Δ. This constraint defines the set in $\mathbb{R}^n$ between the two hyperplanes $\{\delta : c_0 + c^T\delta = \pm 1\}$. If $\sum_{i=0}^{n} |c_i| > 1$, then this is an active constraint in shaping $\boldsymbol{B\Delta}$.

Definition 7.1 *For $M \in \mathbb{R}^{n\times n}$, μ with linear cuts is defined as*

$$\mu_{lc,\Delta}(M) \triangleq \frac{1}{\min\{\max\{|c_0 + c^T\delta|, \|\Delta\|\} : \Delta \in \boldsymbol{\Delta}, \det(I - M\Delta) = 0\}} \tag{7.2}$$

unless $det(I - M\Delta) \neq 0, \forall \Delta \in \boldsymbol{\Delta}$, in which case $\mu_{lc,\Delta}(M) \triangleq 0$.

For application to probabilistic robustness analysis, specifically gridding the boundary of stability, this formulation for linear cuts is not very useful. The problem is that the entire region in Δ-space is scaling with γ. For this to be a practical gridding methodology, the region must have a fixed hypercubic substrate and allow the boundary of the non-axially aligned constraint to be chosen to maximize the region of non-singularity.

The resulting reformulation is similar to skew μ, $\check{\mu}$ from Section 1.5. The basic idea is that for the GEVP associated with the upper bound computation to remain convex, γ must parameterize a nested family of regions in Δ-space. Associated with each value of γ is an LMI. The nesting insures that if the LMI associated with γ_0 is feasible, then all of the associated LMIs are feasible for $\gamma > \gamma_0$. Similarly, if the LMI associated with γ_0 is infeasible, hen all of the associated LMIs are infeasible for $\gamma < \gamma_0$. For skew μ, only one of the uncertainty blocks scales by γ. The remaining blocks have the standard unity gain constraint. Assuming γ only scales the final block of Δ, the result is that γ^2 only scales the part of the LMI variable D associated with the last block scales, and the upper bound becomes

$$\check{\mu}(M) \leq \inf_{D \in \boldsymbol{D}} \left\{ \gamma : M^* D M - \begin{bmatrix} I & 0 \\ 0 & \gamma^2 \end{bmatrix} D < 0 \right\}.$$

For the problem of probabilistic robustness analysis and the gridding of the boundary of singularity, the appropriate modification of μ is to only scale the linear cut with γ. The remaining constraints are fixed.

Definition 7.2 *For $M \in \mathbb{R}^{n \times n}$, $\check{\mu}_{lc}$ is the skew version of μ with linear cuts and is defined as*

$$\check{\mu}_{lc,\Delta}(M) \triangleq \frac{1}{\{\min |c_0 + c^T \delta| : \Delta \in \boldsymbol{B\Delta}, \det(I - M\Delta) = 0\}} \tag{7.3}$$

unless $det(I - M\Delta) \triangleq 0, \forall \Delta \in \mathbf{B\Delta}$, in which case $\check{\mu}_{\mathbf{lc},\Delta}(M) := 0$.

7.2 $\check{\mu}_{lc}$ Upper Bounds

Much like in the spherical case, it is easy to formulate new variations of μ. These variations are normally computationally intractable and their potential utility depends upon the existence of computationally tractable bounds. The primary concern here is the computation of an upper bound. Note that it is simple to extend the following upper bound formulations and equivalent constructions to $\mu_{lc,\Delta}$.

Ellipsoidal Cut

The linear constraints defined in (7.1) are the limit of a sequence of off-center hyperellipsoids in parameter space as the eccentricity tends to infinity. Using the upper bound for the intersection of regions presented

in Chapter 4 and the appropriate modification for the skew nesting of uncertainty sets, an LMI upper bound for $\check{\mu}_{lc,\Delta}$ can be computed. The construction follows.

Let $T = \begin{bmatrix} c^T \\ C_\perp^T \end{bmatrix}$, where c is the vector normal to the hyperplanes to be approximated, $\|c\| = 1$, and $C_\perp$ is the matrix whose columns form an orthonormal basis for the kernel of c^T. Then T is a unitary matrix. Let

$$P = T^{-1}\Sigma T = T^{-1} \begin{bmatrix} 1 & & & \\ & \frac{1}{\sigma_2^2} & & \\ & & \cdots & \\ & & & \frac{1}{\sigma_n^2} \end{bmatrix} T. \tag{7.4}$$

Then the level set $L_\gamma = \left\{\delta : \sqrt{\delta^T P \delta} = \frac{1}{\gamma}\right\}$ describes the hyperellipsoid with $\frac{1}{\gamma}, \frac{\sigma_2}{\gamma}, \ldots, \frac{\sigma_n}{\gamma}$ being the lengths of the axes. When $\sigma_i \to \infty (i = 2, \ldots, n)$, L_γ approaches the hyperplanes $\{\delta : c^T\delta = \pm\frac{1}{\gamma}\}$. To account for the asymmetry due to c_0, the center of the ellipse must be moved. The appropriate shift is $\Delta_0 = -c_0 diag[c]$. This γ and Δ_0 are the same as in the elliptical μ upper bound LMI (4.3). Combine this bound with the standard μ feasibility LMI, which is the standard upper bound and can guarantee that $\mu < 1$ or equivalently that $det(I - M\Delta) \neq 0$ on $\boldsymbol{B\Delta}$, to construct an upper bound on $\check{\mu}_{\Delta,lc}(M)$ using Section 4.8,

$$\inf_{D_1 \in \boldsymbol{D}_1, D_2 > 0} \{\gamma : M^*(D_1 + P^{-1} \circ D_2)M < D_1 + \gamma^2 (I - \Delta_0 M)^* D_2 (I - \Delta_0 M)\}. \tag{7.5}$$

For the construction with nonrepeated real scalars and $M \in \mathbb{R}^{n \times n}$, the G-scales for exploiting the real structure of the uncertainty do not improve the bounds. For more general problems, (7.5) should include the G-scales.

However, with the high eccentricity of the ellipsoid, the upper bound achieved is very conservative. And the conservativeness decreases when the eccentricity is reduced, at the cost of worse approximation of the hyperplanes. The method implemented intersects a high eccentricity ellipse with a less eccentric ellipse to try to improve the performance. As a result, this optimization is not convex, because of the freedom in the choice of the ellipses. There do exist schemes for finding a local optimum. This leads to even more potential conservativeness.

Implicit Method

Within the implicit framework shown in Figure 1.4, the new feature in this framework is the freedom to add algebraic constraints to the standard

loop equation. Specifically, the signal constraint that $Cx = 0$ is added. For μ with a linear cut, (7.1) represents a constraint in the operator space $\mathbf{\Delta}$. This operator must be converted to a signal constraint to cast $\check{\mu}_{lc}$ as a standard implicit μ problem.

As one of the alternatives to the elliptical method, the following implicit system in Figure 7.1 can be constructed to perform linear cuts on $\mathbf{B\Delta}$. The construction is similar to the implicit formulation for spherical μ presented in [24].

Theorem 7.3 *For generic k,*

$$\check{\mu}_{lc,\mathbf{\Delta}}(M) < 1 \iff \check{\mu}_{\tilde{\mathbf{\Delta}}}(\tilde{C}, \tilde{M}) < 1,$$

where

$$\tilde{\mathbf{\Delta}} = \left\{ \begin{bmatrix} \Delta & & \\ & \Delta & \\ & & \delta_0 \end{bmatrix} : \Delta \in \mathbf{\Delta}, \delta_0 \in \mathbb{R} \right\},$$

$$\tilde{M} = \begin{bmatrix} M & 0 & 0 \\ K & 0 & 0 \\ k^T & 0 & 0 \end{bmatrix}, \tilde{C} = [c_0 k^T \; c^T \; 1].$$

Where $k^T \in \mathbb{C}^n$ *is an arbitrary row vector, and* $K = \begin{bmatrix} k^T \\ \vdots \\ k^T \end{bmatrix} \in \mathbb{C}^{n \times n}$.

Proof: The structure of the implicit system in Figure 7.1 is the graphical construction associated with $\tilde{M}$, $\tilde{C}$, and $\tilde{\mathbf{\Delta}}$. This construction imposes the following relationship in addition to the original interconnection between Δ and M, using $v_1 = k^T x$:

$$v = \begin{bmatrix} v_1 \\ \vdots \\ v_1 \end{bmatrix}, \; u = \Delta v = \begin{bmatrix} \delta_1 v_1 \\ \vdots \\ \delta_n v_1 \end{bmatrix}.$$

The imposed algebraic constraint on the signal is

$$\delta_0 v_1 = c^T u = \left(\sum_{i=1}^{n} c_i \delta_i \right) v_1.$$

Generically, $v_1 \neq 0$ over the space of possible choices for k, hence,

$$1 \geq |\delta_0| = \left| \sum_{i=1}^{n} c_i \delta_i \right| = |c^T \delta|.$$

$$\Delta \in \mathbf{B}\Delta_{lc} \Longleftrightarrow \tilde{\Delta} \in \mathbf{B}\tilde{\Delta}.$$

The remainder of the proof follows from the definitions of $\check{\mu}_{lc,\Delta}(M)$ and $\check{\mu}_{\tilde{\Delta}}(\tilde{C},\tilde{M})$. □

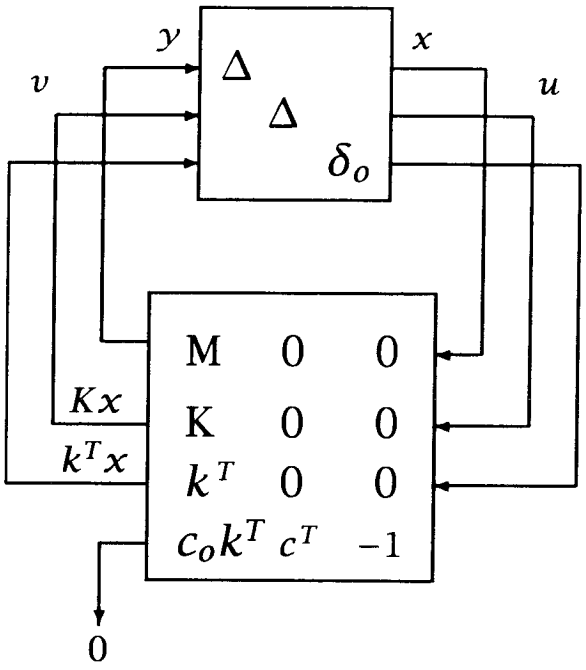

Figure 7.1: Implicit System Constructed for Linear Constraints

Figure 7.1 is the block diagram formulation for the construction in Theorem 7.3.

The computation of the upper bound involves the following issues:

- Permutations on the rows and columns of $\tilde{M}$ are needed such that $\tilde{\Delta}$ consists of n 2×2 repeated real scalar blocks and one real scalar. The upper bound formula actually used in the computation is a bit more involved than (1.21) due to the existence of the repeated scalar blocks. The proper generalization involves the G-scales to exploit the structure of the real uncertainty.

- The scalar γ^2 in the upper bound (1.21) should be replaced by the matrix $S_\gamma = \begin{bmatrix} I_{2n} & \\ & \gamma^2 \end{bmatrix}$. Then only the linear constraint $|c_0 + c^T\delta| \leq \frac{1}{\gamma}$ is scaled, while the unit ball for the original parameters remain fixed during the γ iteration.

- The quality of the upper bound depends largely on the choice of the vector k in $\tilde{M}$. The optimization problem including k is non-convex, which is also an interesting research problem. Right now as a heuristic we pick k to be the input vector of M corresponding to its maximum singular value to make $\tilde{M}$ more like a rank-one matrix, due to the fact that the upper bound is exact when $\tilde{M}$ is rank-one. And the performance of the upper bounds achieved is much better than those with random chosen k.

- It may be useful to have redundant constraints for the upper bound computation.

Parallelogram Method

To simplify the presentation of the third method, it is assumed the $c_0 = 0$. When $c_0 \neq 0$ the construction is even more conservative and the numerical experience indicate that this isn't the approach to take. For the examples presented $c_0 = 0$.

The parallelogram method is similar to the implicit formulation, in that both convert the linear constraint $|c^T\delta| \leq 1$ into a norm constraint on a scalar δ_0, and the matrix dimension is increased in both cases. In this method, only the standard μ upper bound computation is involved. The cost is some extra regions outside $\mathbf{B}\Delta_{lc}$ have to be included when checking for nonzero solutions. This may lead to conservativeness and forces μ for this new problem to be an upper bound for $\breve{\mu}_{lc}$.

δ_0 is defined to be

$$\delta_0 = c_1\delta_1 + c_2\delta_2 + \cdots + c_n\delta_n.$$

Therefore

$$|c^T\delta| \leq 1 \Longleftrightarrow |\delta_0| \leq 1.$$

Let $\hat{\Delta} = diag[\delta_0, \delta_2, \delta_2, \cdots, \delta_n, \delta_n]$, then

$$\Delta = P_L\hat{\Delta}P_R, \tag{7.6}$$

where

$$P_L = \begin{bmatrix} 1 & -c_2 & 0 & -c_3 & 0 & \cdots & -c_n & 0 \\ 0 & 0 & 1 & & & & & \\ & & & 0 & 1 & & & \\ & & & & & \ddots & & \\ & & & & & & 0 & 1 \end{bmatrix},$$

and

$$P_R = \begin{bmatrix} \frac{1}{c_1} & & & & \\ \frac{1}{c_1} & 0 & & & \\ 0 & 1 & & & \\ \frac{1}{c_1} & & 0 & & \\ 0 & & 1 & & \\ \vdots & & & \ddots & \\ \frac{1}{c_1} & & & & 0 \\ 0 & & & & 1 \end{bmatrix}.$$

But the unit ball $\mathbf{B}\hat{\Delta}$ doesn't map to $\mathbf{B}\Delta_{lc}$ using (7.6). The mapping of $\mathbf{B}\hat{\Delta}$ contains $\mathbf{B}\Delta_{lc}$.

Setting $\hat{M} = P_R M P_L$. Then the skew μ upper bound on $\check{\mu}_{\hat{\Delta}}(\hat{M})$ is also an upper bound on $\check{\mu}_{lc,\Delta}(M)$. This follows from the definition of μ and the existence of non-zero solutions. As the simple example, Figure 7.2 shows mapping of the unit norm bounded set in $\hat{\Delta}$ to the left, while the region for a linear cut is the one to the right. The potential conservativeness of this bound is obvious.

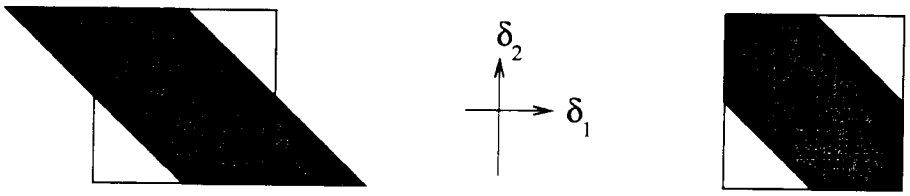

Figure 7.2: $\mathbf{B}\hat{\Delta}$ and $\mathbf{B}\Delta_{lc}$

The reason for the conservativeness is in performing the change of variables the norm constraint on δ_1 no longer exists in the reformulation. δ_1 is not in $\hat{\Delta}$ and has been replaced by δ_0 which is used to define the linear cut. Alternatively, a different δ_i can be left off. By leaving out a different δ_i, the associated added regions are different. If the δ_i with the maximum c_i is chosen, the volume of the extra region is minimized, and this will be a heuristic used to, hopefully, minimize the conservativeness of the bound. This heuristic is supported by the results of numerical experience. The excess regions can be excluded by adding an implicit constraint as presented in Section 7.2 but will not be pursued.

7.3 Numerical Examples

Rank-one problems are the motivation for doing linear cuts. So random rank-one matrices were used to test the effectiveness of the above methods. It turns out, just like the standard μ upper bound, the upper bounds achieved on $\check{\mu}_{lc,\Delta}(M)$ with all the three methods are exact when M is rank-one, which is not surprising because essentially the elliptical cut and implicit methods are just extensions of the standard μ upper bound, and the parallelogram method employs the standard μ computation directly, and the conservativeness caused by extra regions doesn't exist in the rank-one case, assuming the cuts are appropriately aligned with the level sets. If the level sets aren't appropriately aligned with the cut, then the parallelogram method will be conservative.

For general random matrices where each element is random, the relative performance of the different methods varies with the problem. Figures 7.3 and 7.4 show the results for two random matrices. For Figure 7.3, the elliptical cut method works better than the implicit method, while the parallelogram method doesn't achieve any bound because of its conservativeness.

For Figures 7.3 and 7.4, $\check{\mu} = 2$, $c_0 = 0$ and the normal vector c is chosen to be the gradient of the maximum real eigenvalue function at the vertex p where the worst case is achieved. The curve marked by "x"'s is the boundary where the singularity of $I - M\Delta$ occurs.

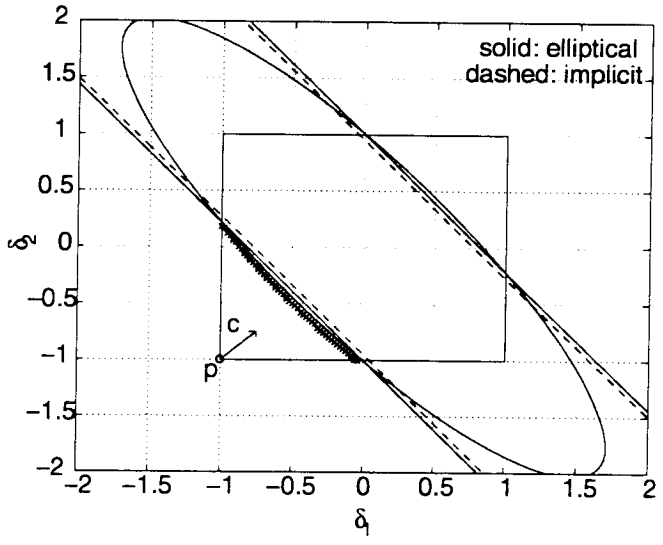

Figure 7.3: Example 1

In Figure 7.4, both the implicit method and the parallelogram method achieve the exact bound while the performance of the elliptical cut is very poor.

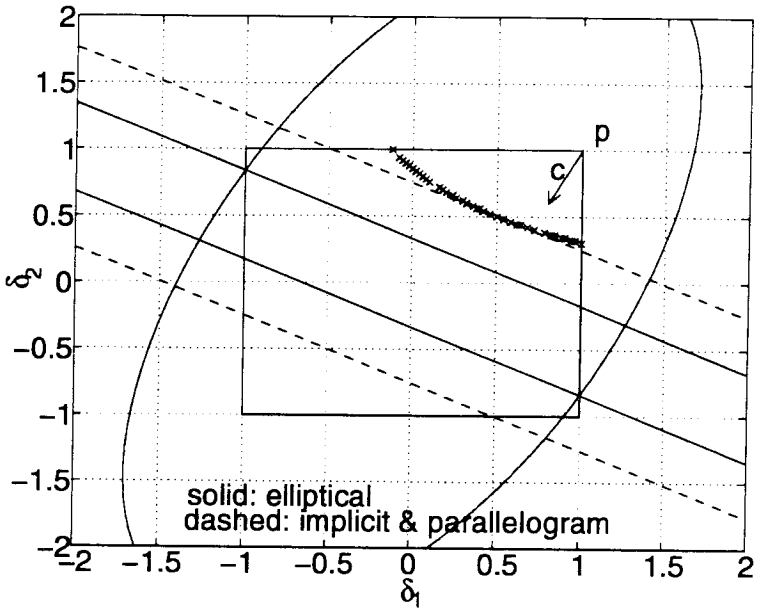

Figure 7.4: Example 2

The table in Figure 7.5 shows the results of applying the three methods for computing upper bounds to 25 random matrices of dimension 2, 3, and 4. The problems are constructed so that 1 is a guaranteed lower bound to $\mu_{lc,\Delta}(M)$, but the actual value is unknown. For some problems, the parallelogram and ellipsoid method are unable to find any bound to $\mu_{lc,\Delta}(M)$. The entries in the column labeled # indicate the number of problems for which that method was able to compute an upper bound. The entries in the column labeled CPU indicate the CPU time in seconds on a Sun Ultra 1 using SDPSOL for solving LMIs. The entries in the column labeled avg. bnd is the average upper bound computed for the problems that the associated method was able to compute a bound. The CPU time is left blank for the ellipsoid method because the bounds computed used methods to try to solve the nonconvex optimization problem which resulted in significantly longer computation time. The implicit method gives by far the best bounds, especially as the dimension increases. Additionally the implicit and parallelogram methods are almost identical in terms of computation time. The implicit is hands down the best of the three methods.

Dimension	$n = 2$			$n = 3$			$n = 4$		
Method	#	avg. bnd	CPU	#	avg. bnd	CPU	#	avg. bnd	CPU
Implicit	25	1.0163	.7572	25	1.2559	2.5392	25	1.5963	2.904
Parallelogram	25	1.1534	.7864	25	1.5031	2.5324	23	2.0485	2.900
Ellipsoid	25	1.0811		14	1.5137		12	5.7061	

Figure 7.5: Numerical Upper Bound Results for Random Problems

7.4 Choosing the Linear Cut

To use the linear cut formulation for computing $\mathbb{S}$ for purely probabilistic μ from Definition 2.3, the normal vector c must be computed. What is desired is a vector normal to the boundary of singularity at the singular point closest to some point in Δ-space. From a practical point of view, there are a number of approximate methods to compute this linear constraint to be aligned with the boundary of singularity.

A possible method to compute this gradient is from power algorithm methodologies [29] for computing the μ lower bound. If the algorithm finds a local minimum rather than a global minimum, then the associated gradient would yield a conservative non-singular region and doesn't help the cause against exponential growth. Alternatively, a pseudo-gradient can be constructed by computing upper bounds for a number of μ problems. The different μ problems are closely related; they use the same M and uncertainty structure Δ but use different points in Δ-space as the origin. There are no guarantees on the quality of the pseudo-gradient. These issues will not be addressed further.

Chapter 8

Hierarchical Uncertain Modelling

For the modelling of complex systems, the natural reductionist approach is to divide a system into subsystems and model each subsystem separately. The traditional reductionist view is that if the model of each component is accurate, then the behavior predicted by the interconnection of the subsystem models will accurately predict the behavior of the real composite system.

One of the lessons from control theory is that the above statement is not true. The problem is feedback. For a poorly designed system, the performance of the conglomerate system will be highly sensitive to the gap between reality and the subsystem models. Often subsystems are buffered from each other to restrict the coupling of the dynamics of the subsystems which simplifies the design process to just component issues rather than system issues. It is a protocol of sorts. This has been highly successful in digital VLSI. The downside of this overhead is conservatism in achievable performance.

There is a tradeoff between model fidelity and complexity. Ultimately, the model used to answer a question should be the least complex model which is sufficient to answer the question at hand. For a complex system, it is unlikely that a single model will suffice. What is needed is a family of models from which to choose an appropriate model. Robustness analysis presents a powerful tool not only for making predictions about the performance of complex systems, but also determining if an uncertain model is sufficient to guarantee performance.

A possible modelling paradigm is to have a single high order best model and perform systematic model reduction with different weights as necessary. This formulation for hierarchical modelling brings up the issue of uncertain model reduction in a behavioral setting, which is a hard problem and some work has been done by Beck in [4]. Although further progress may be made in behavioral model reduction, it is unlikely that the com-

putation will progress to the point where it is tractable to model reduce the interconnection of phenomenon described by partial differential equations (PDEs). As a result, it is necessary to use hand crafted family of preconstructed uncertain models within the paradigm presented here. These points will not be addressed further.

The issues which will be addressed are the choices made and a flaw of the reductionist approach.

8.1 Choices

A modelling framework involves choices. The hierarchical modelling paradigm chosen is a byproduct of reductionist approach to modelling. In the case of an automobile, at the simplest level it may be appropriate to think of the vehicle as a rigid body. If further detail is necessary, it may be appropriate to model the car as the interconnection of springs, rods, beams, dampers, and so on. Each of these parts could be further described by finite element models.

The second choice made is the inclusion uncertainty in the models. For a real system, there is a gap between reality and a mathematical model. Hence, finding an exact model is impossible and characterizing the inexactness is critical for making guarantees on system performance using the mathematical model. In the case of a resistor, there is uncertainty on the exact value of the resistance and the parasitics. When a model is reduced, the eliminated dynamics must be covered by uncertainty. This is useful when cruder models with more uncertainty are sufficient for answering a particular question which may lead to reduced computation.

The next choice made is a tree structured hierarchy. This hierarchy defines a partial ordering of all the models for a system. The partial ordering is necessary because different phenomena should be modelled at different levels of accuracy depending on the question to be answered. The model needed to predict the emissions for a car would be quite different than one useful for passenger comfort.

This hierarchy also follows from the reductionist point of view of the subdivision of a system. For example, modelling all the facets of a car is quite a task, but the natural approach is to break up the car into more tractable components and model them individually. So rather than modelling the entire car, a car is an interconnection of an engine, transmission, exhaust system, cooling system, suspension, etc., and each of the components is modelled separately.

A benefit of the tree structure is the connection with the object oriented philosophy. The tree defines how the components interact with each other, which simplifies future modifications, because if a system is modified, only the modified component and the models which depend upon this component needs to be remodelled, and the other component models will remain intact.

Another choice is that the model will be constructed, implemented and used on a computer. As a result, the data structures should be convenient and tractable for computer implementation as opposed to writing them out by hand. The reason for this choice is that as more complicated systems have more and more detailed models and are to be analyzed or simulated, it is intractable to do them any other way.

The proposed framework is defined by a hierarchical tree structure of the components, an interconnection structure of the components, and a fundamental data type for a component. Throughout this paper, an inductor is used to demonstrate the features of the framework.

8.2 Component Modelling

In the same way that a system has many different models, so does a component; after all, the component is a system itself. For the component, there are a number of composite models representing the entire system. These models would have their own hierarchy. Imagine that the component consists of a discrete switch which trades of complexity and fidelity. This switch may be multidimensional to be compatible with the partial order of models. If none of the component models are satisfactory, then the component is described by the subcomponents which make it up.

Hence the component model has two modes. The first mode is when the switch points to a particular model. In the second mode the switch points to the reticulation, and the component model defines the interconnection of subcomponents, which may also be modelled by subcomponents.

First Mode

Consider the ideal inductor shown in Figure 8.1 with behavior described by

$$\frac{d}{dt}(Li) = v$$

with inductance L, where v is an input, i is an output, and ϕ is the flux in the inductor. The resulting LFT model (Figure 1.1) of the system is given

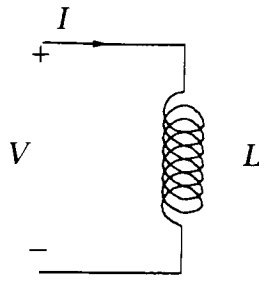

Figure 8.1: Inductor

by $u = v$, $x = \phi$, $y = i$, $\Delta = \int$, and $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{c|c} 0 & 1 \\ \hline 1/L & 0 \end{array}\right]$. As a result of choosing this model of an inductor, two problems arise. First, if the effects of uncertainty in L, nonlinearities and parasitics, are to be investigated, there is no convenient way to do this without starting over. Second, by assuming that v is an input and i is an output, our model may be incompatible with other components with which it is interconnected as shown in Figure 8.2. Figure 8.2 represents an input/output interpretation of two inductors connected in series ($i_1 = i_2$). The connection of the two

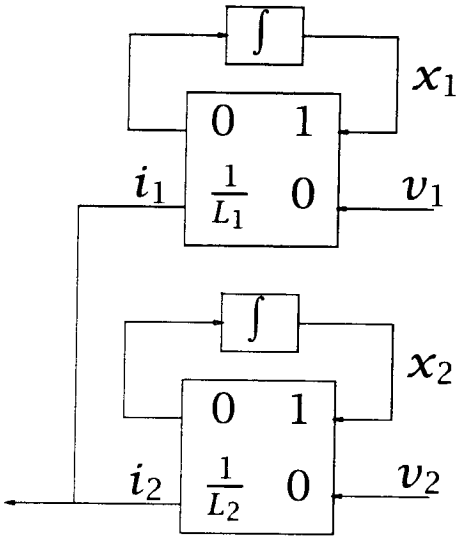

Figure 8.2: Series Interconnection of Inductors

outputs is not properly defined within the input/output framework. These two problems are addressed next.

Problem 1: Uncertainty

The solution to the first problem is easily addressed in the LFT framework. For the inductor example in Figure 8.1, the modelled equations should be replaced by $d\phi/dt = v$ and $\phi = L(I) = L_0(1 + \delta)i$, where L_0 is a constant which represents the nominal value of the inductor and δ is an unknown operator. The model of the uncertain inductor is given by $y = i$, $x = [\phi, x_2]$, $u = v$, $\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|c} 0 & 0 & 1 \\ 1/L_0 & -1/L_0 & 0 \\ \hline 1/L_0 & -1/L_0 & 0 \end{array}\right]$, and $\Delta = \begin{bmatrix} \int & 0 \\ 0 & \delta \end{bmatrix}$. x_2 is not a state in the conventional sense but the output δ [5]. δ could represent time variation in the inductance due to saturation, variations in the core geometry, magnetic links to other components, nonlinearities, and so on.

Problem 2: Interconnection Compatibility

LFTs provide a flexible modelling framework for incorporating uncertainty descriptions, but the input-output assumption is not desirable for the modelling of interconnected systems. It is not a priori known which variables should be treated as inputs and which should be treated as outputs [44].

For tree structured hierarchical models, at the interconnections there isn't a notion of signal flow. The interconnection variables become internal variables to the system rather than inputs or outputs. So it is more natural to not make the distinction between inputs and outputs in modelling.

To address the second problem, systems will be represented in an implicit LFT form where $0 = (\Delta \star M)w$ as shown in Figure 1.2. There is no issue of compatibility between implicit LFT models of components because no input-output partition has been made.

For the inductor example, the implicit model of the inductor is given by $d\phi/dt = v$, $L_0(1 + \delta)(i) = \phi$, $w = [v, i]$,

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{cc|cc} 0 & L_0 & 0 & L_0 \\ 0 & 0 & 1 & 0 \\ \hline -1 & 0 & 1 & 0 \end{array}\right], \text{and } \Delta = \begin{bmatrix} \frac{d}{dt} & 0 \\ 0 & \delta \end{bmatrix}.$$

Generality

For the fundamental components in this modelling system, the model must be general enough to describe any situation which may occur. The model should have some information about the component, so that each component isn't just an arbitrary operator. The information is contained in the nominal value of the component. There should be no irreversible assumptions made. For example, the standard circuit equation for an inductor is $V = L \cdot di/dt$. If this were the fundamental model of an inductor, the assumption has been made that L is a constant. If L were blindly replaced by $L(t)$, the resulting model would be incorrect. The actual equations for an inductor are given in equations 8.1 and 8.2.

$$\phi = Li \tag{8.1}$$

$$kv = d\phi/dt \tag{8.2}$$

Once the model is constructed and analysis, synthesis, or simulation is the next step, then any assumptions can be made about our uncertainty like Δ_L is a real parameter, a bounded operator, etc., but this is after the modelling process and a part of the identification process.

The weakest possible assumptions about the inductor (L,k) are to be made, so that wide variety of uncertainty assumptions can be made at the analysis level. L and k are assumed to be a non-commuting indeterminates (NCIs), i.e., it could be an arbitrary nonlinear, time-varying operator. For a real inductor, its nominal value is of use in describing its operation. For modelling, $L := L_0 + \Delta_L$ and $k := 1 + \Delta_v$, where Δ_L and Δ_v are NCIs and L_0 is a "place holder" for the nominal inductance. L_0 is used to describe the ideal model of an inductor. NCI's act as "place holders" for uncertainty descriptions. It is important to note that by setting $L := L_0 + \Delta_L$ we have not committed to anything. This can be undone by defining $\Delta_L := -L_0 + \Delta_{Lnew}$. The L_0 term is added because the nominal value is presumably useful in describing the operation of the system.

Second Mode

In the second mode the component describes the interconnection of subcomponents. Given that the leaves of the model tree are implicit models and it is the nodes in the second mode which connect the system together and form the composite model. It follows that the second mode models are themselves implicit systems which only describe an algebraic constraint on the signals that are routed through the associated node.

8.3 Component Data Type

There is no fundamental difference between modes 1 and 2. Mode 1 is a special case of mode 2 with only one subcomponent. The distinction is useful for the problem of model choice 8.4. It is natural to have more than one possible model available in mode 1. Often a number of high level models for a component are desirable. Similarly, it is possible to have multiple decompositions available in mode 2, but this is somewhat inconsistent with the reductionist philosophy, and this flexibility will not be exploited here.

Each component is modelled as a switch as shown in Figure 8.3. This

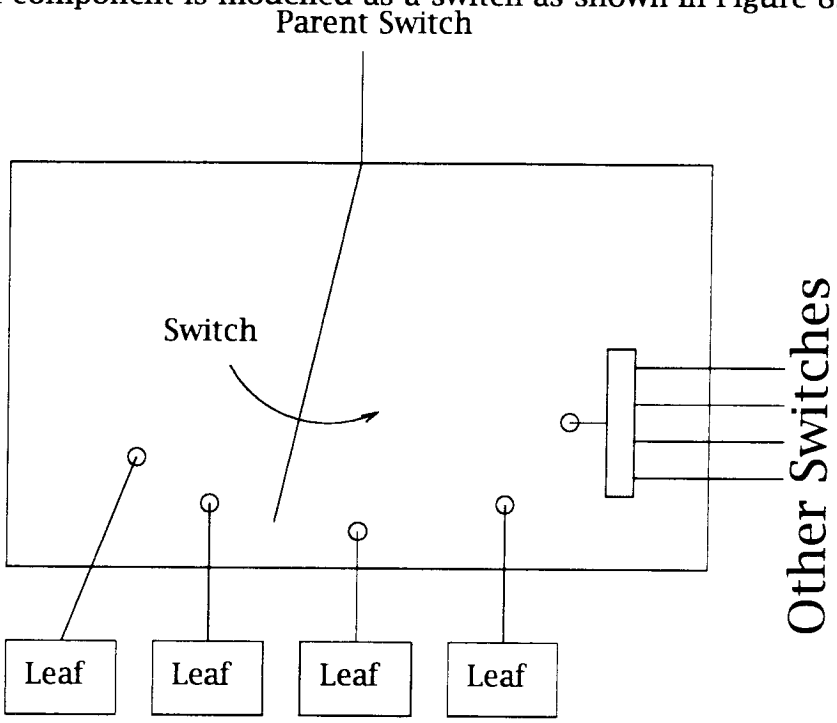

Figure 8.3: Component Model

switch is connected to other switches or up to one other switch (to no other switches if it is the root component) and one leaf model.

The hierarchical structure of a system modelled using this framework is a tree (Figure 8.4). At each component node of the tree there is a component model. Each leaf node is associated with a leaf model.

Both the leaf and component models are represented by implicit LFT systems. For the component models, the generalized state dimension is zero. The system variables w for such models are partitioned as:

$$w = \begin{bmatrix} w_m & l_m \end{bmatrix}.$$

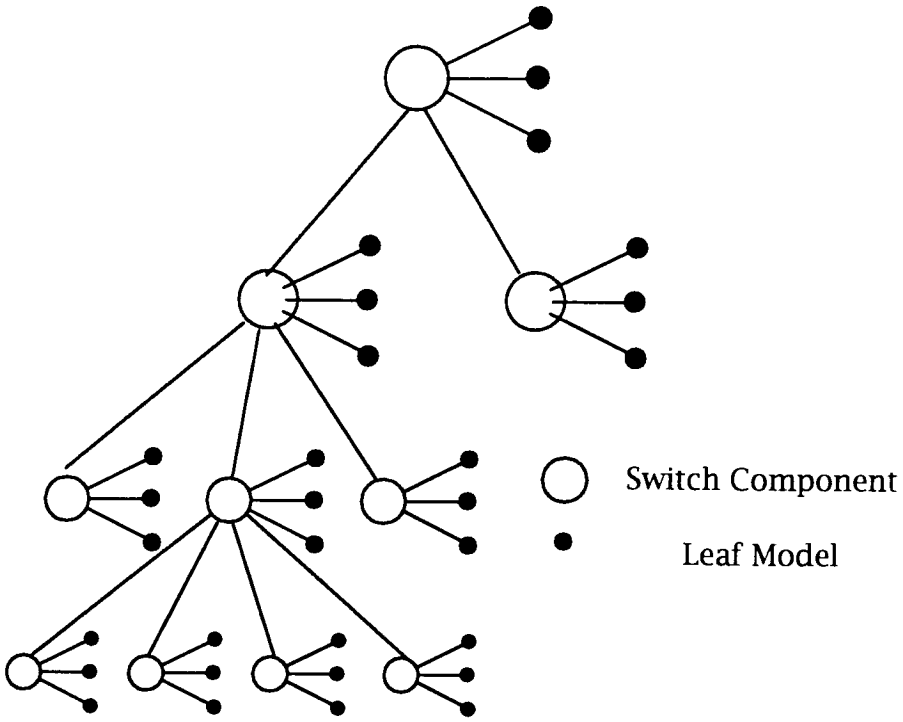

Figure 8.4: Hierarchical Structure

w_m are the manifest variables or the signals of interest for the outside world. l_m are the latent manifest variables or the signals needed to interconnect the subcomponents and connect them to the outside world or to connect the leaf node to the outside world. In mode 1, it follows that the dimension of w_m is equal to the dimension of l_m. In mode 2, it is expected that the dimension of w_m is less than the dimension of l_m because of the extra signals need to describe the interconnection of the subcomponents.

The implicit system used to model the switch or component model is given by

$$0 = \begin{bmatrix} T_1 & T_2 \end{bmatrix} w.$$

The switches only contain interconnection information. The dynamical and uncertainty information is contained within the leaf models.

The system variables for the leaf models, w, are just the manifest variables w_m (interconnection variables from the switch). The dynamics at the leaf node are given by $0 = (M \star \Delta)w$ where

$$\Delta = diag\left(\frac{d}{dt}I, q_1 I, \cdots, q_n I, \Delta_1, \cdots, \Delta_n\right) \tag{8.3}$$

q_i are arbitrary constants. Δ_i are noncommuting indeterminates and can be used for unmodeled dynamics or nonlinearities to be specified later. Δ

is used to describe the dynamics and uncertainty of the nominal component. Within Δ the standard dynamic element is $\frac{d}{dt}$ instead of the standard $\int$. The reason for this choice is that technically $\int$ is not an operator because an initial condition must be specified. Ultimately all that is being done is writing differential equations in a distributed way. Hopefully this effort is distributed in a way that is useful for the engineering endeavor.

8.4 System Refinement/Model Choice

So far an elaborate framework for representing hierarchical uncertain models has been presented. Assuming that such a construction exists and all of the NCIs have been refined to have a specific description, there is still the issue of choosing a model from this hierarchy given the question to be answered. Putting this into a robustness analysis context, what is the simplist model which is good enough to answer a particular question with sufficient accuracy. One solution is to compute all models which are good enough and choose the simplist, but this is computationally intractable as you get combinatoric growth with all the leaf models.

The approach proposed is a scheme for choosing which leaf model to refine. Associated with each leaf is an uncertainty structure. Presumably, as that component is refined, the uncertainty will be reduced. This is assuming that the partial ordering of the models is consistent with the associated behaviors. By finding the uncertainty to which the model's predictive performance is most sensitive, then refinement of that component should lead to improved model performance. If this process is started from the simplist high level model, then the model will be systematically refined until a satisfactory model is found. In general, this approach will not yield the simplest model but may be good enough. If the robustness analysis formulation for the question and associated performance can be cast as a μ problem, then the sensitivity computation can be cast as n-skew μ problems where n is the number of leaf models in the system model. The component for which skew-μ is the greatest would be a heuristic for the component to which our current model is most sensitive and whose refinement would help the predictive power of the model the most.

8.5 Flaw

In this proposed reductionist approach to hierarchical modelling, there is a flaw. The assumption is made that the leaf models can be refined inde-

pendently and they will still be compatible. The big problem is incorporating continuum phenomenon described by PDEs which are interconnected along a boundary. From a computation and representation standpoint, although these phenomenon may be from different domains like fluids and structures in the case of aeroelastic instabilities (flutter), the modelling and the models must be linked so that the discretizations are compatible. At the lowest level, essentially all models are linked in this manner and there is no distribution to the hierarchy. It becomes a strict ordering rather than a partial ordering. In trying to deal with these sorts of linkages between subcomponents, the reductionist approach breaks down.

Chapter 9

Implicit Identification

In the process of modelling the interconnection of systems, the interconnection is described by a constraint on a set of signals or interconnection variables. For example, if two masses are bolted together so that the interconnection can be considered rigid, then the resulting constraint is that the position and orientation of the masses at the interconnection point are the same. Additionally, the forces and torques at the interconnection are equal in magnitude but in opposite direction. For building system models from component models, the component models must include the interconnection signals.

Often the natural choices for inputs and outputs for a subsystem do not contain the interconnection variables. This arises when applying black box identification methodologies. Within these methodologies, the system is excited by sending a known signal to the actuators of a system and a model is constructed from the data from the sensors. If the system does not have actuators and sensors at the interconnection point then a model appropriate for interconnection cannot be constructed using a traditional black box methodology. In this presentation, this problem will be addressed by the use of *passive excitation* to identify the system. By passive excitation, it is meant that the interconnection point is not driven by an actuator but a separate but known system.

For this problem it is assumed that for the system to be identified, the interconnection point cannot be driven, meaning that there are no actuators at the interconnection point. Additionally, the system will be connected to known system. The desire is to construct a sufficiently general model of the original system, so that when the system is interconnected with a known system the new input-output model can be determined without repeating the identification procedure for the aggregate system. When the term input-output model is used, it is referring to the model relating the input of the actuators to the output of the sensors for a particular

attached structure.

If a black box identification of the system is carried out with a known system attached at the interconnection point, the resulting model is not general enough to predict the model for any known system at the attachment. Presumably there is partial information in this input-output model and the associated attached system which would be useful in constructing a general model for predicted input-output models.

The following questions arise naturally: what additional information needs to be known to construct a general model? How many input-output models are needed to construct a general model? What sorts of systems should be interconnected with the system? Can everything about the system be determined?

The motivation for this sort of modelling is that a system can interact with the outside world via more than just the inputs and outputs defined by the sensors and actuators on a system. For example, a car is affected not only by the surface of the road, but also by how it has been loaded (people, luggage, toys). The "manifest" variables, or signals of interest, of our more general model are the inputs, outputs, and the relevant interconnection variables. These would be all the signals that come from or relate to the world outside of our original system.

A natural framework for describing this model is the behavioral framework. Although this problem can be considered in other frameworks, the behavioral framework provides a convenient and systematic method for interconnecting systems, especially when it is not known to what the system might be connected. It also simplifies the issue of partitioning the interconnection variables into inputs and outputs; this partition is not necessary so it isn't done.

9.1 Behavioral Framework

For a detailed description of the framework, the reader is referred to Willems [44]. Further the behavioral framework is a special case of the implicit LFT framework presented in Section 1.3. In the behavioral framework, all of the external variables are on equal footing. There is no distinction between inputs and outputs, much in the same way as models are constructed from first principles. From first principles, only the pertinent equations are written. There is no distinction between inputs and outputs; they are merely system variable or potential boundary conditions. This formulation is natural for defining interconnections. Determining which

signals are inputs and which are outputs depends on what the system is interconnected with. For example, the flexible structure to be discussed later has manifest variables of inputs and outputs, and accelerations and forces for an interconnection point on the structure. If nothing is attached to the structure, then the forces at the interconnection point are exactly 0. This would lead to the interpretation that the forces are inputs and the accelerations are outputs. The system could also be connected to a rigid point in which case the accelerations are exactly 0. The forces at the interconnection are then whatever they must be to satisfy this condition, hence the forces would be interpreted as outputs. Thus, if a framework is chosen such that the interconnection variables aren't on an equal footing, then one would have to alter the model and algorithm, depending on how the variables were initially partitioned.

The behavioral model of a system defines the set of allowable trajectories for the system. A trajectory is a time evolution of all the manifest variables. For example, a point mass (=m Kg) with manifest variables of force (=**F** Newtons) and acceleration (=**a** M/s^2) satisfies:

$$\mathbf{F}(t) = m\mathbf{a}(t)$$

If either $\mathbf{F}(t)$ or $\mathbf{a}(t)$ is defined, then the remaining variables are defined. Obviously not all trajectories are allowable, that is both $\mathbf{F}(t)$ and $\mathbf{a}(t)$ cannot be defined arbitrarily. The representation used here is the implicit LFT representation for $\Delta = \frac{1}{s}I$.

9.2 Problem Statement

For the development presented here, it is assumed that the signals can be partitioned like the ON model, which includes the interconnection variables, shown in Figure 9.1.
ι is the actuated inputs, θ is the measured outputs, ϕ is the first set of interconnection variables which are difficult to actuate, and α is second set of interconnection variables which could be sensed. The purpose of partitioning the interconnection variables is that a particular system that relates ϕ and α will be used. It is assumed that (A, B) is controllable and $[C_2\ D_2]$ is right invertible. If this is not true then the identification problem is hopeless. The first condition is obvious and the second condition says that α can essentially be driven arbitrarily by ι.

The goal is to determine the ON model of a system from the Input-Output models associated algebraic relationships between ϕ and α. There

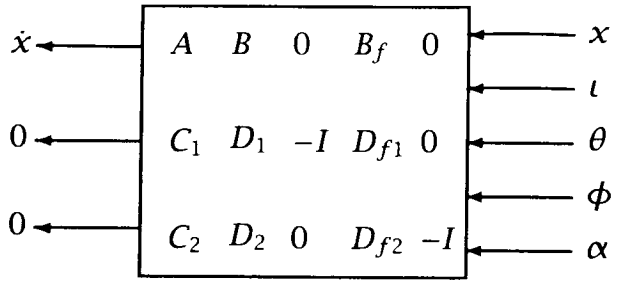

Figure 9.1: ON Representation of an Interconnection System

are a number of reasons for choosing algebraic relationships between ϕ and α. First, it has the same flavor as using differential-algebraic relationship between ϕ and α, in that it leads to the same conclusions. Second, the differential-algebraic formulation is not an interesting extension and only serves to worsen the algebra involved. Third, for some physical problems an algebraic relationship corresponds to simple elements like a masses and moment of inertias to a mechanical structure, where ϕ are the interconnection forces (and torques) and α are the interconnection accelerations (and angular accelerations).

Note that for physical reasons, it isn't always possible to implement any algebraic relationship between ϕ and α. The relationship is restricted to a block diagonal structure with repeated elements similar to the μ uncertainty structures; this constraint is chosen based on the mechanical motivation of attaching masses to a known structure. It isn't possible to have different masses for translation in the x and y directions without resorting to an elaborate physical setup.

The motivation for this sort of identification is to avoid attaching actuators and sensors to obtain the interconnection signals to determine the behavioral model.

9.3 No Information Problem

It is assumed that only the inputs can be driven by an arbitrary signal and only the outputs can be measured, that is to say there is no information about the interconnection. How much information about the general ON model can be extracted from the various I/O models and the associated attached systems?

Claim: A unique solution up to an equivalence transformation for the structured ON model (Figure 9.1) consistent with the I/O data does not exist.

Argument: The basic idea is to assume there exists a unique solution and construct another solution that is consistent with all of the I/O data. Another realization of the interconnected system is shown in Figure 9.2, where M defines the algebraic constraint associated with the interconnection system. There is an algebraic loop in this block diagram. The physical

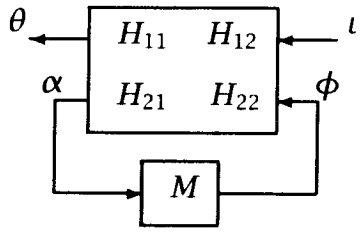

Figure 9.2: System Interconnected with an Algebraic Relation on Interconnection Variable

interpretation of the algebraic loop for the case of mechanical systems is equivalent to a mass being bolted with a mass which is a part of the original system, where the interconnection variables are forces and accelerations.

The reason for this switch in realizations from an ON model to a transfer matrix model form is to motivate the argument graphically rather than algebraically, making the following argument easier to understand. H_{ij} are transfer matrices which are equivalent to the assumed unique solution of the ON model for a particular partitioning of the interconnection variables and M is the algebraic constraint associated with the test interconnection. Physically there may be some restrictions on the possible M. For example, $M \in \Omega$, where Ω is the set of realizable mass matrices. Hence Ω has no negative masses and no masses that accelerate perpendicularly to applied forces. The elements of Ω must be constant and may have structure. Additionally, with particular choices of the accelerations and forces, $M \in \Omega$ is a diagonal matrix which may have relationships between the elements on the diagonal.

Assuming that K commutes with all $M \in \Omega$ (KM=MK) then Figure 9.3 is consistent with all of the I/O data and what is inside the dashed box is indistinguishable for the set of all K. Therefore, at best the general model can be solved up to the set of systems parameterized by K by doing I/O models only with algebraic constraints. Figure 9.4 describes the set of

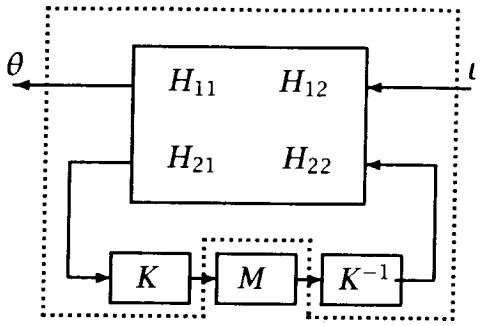

Figure 9.3: Interconnected with Algebraic Constraint and Commuting K

all consistent models. If the attached system commutes with all possible

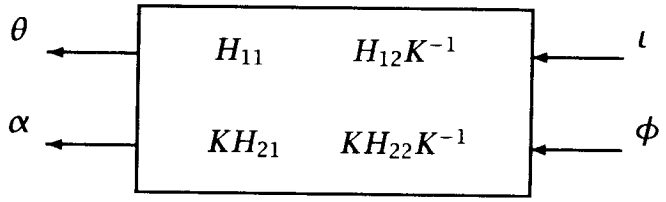

Figure 9.4: Consistent Models Defined by Commuting K's

K's, then any consistent model will predict the correct I/O model, but in general, where M is not a scalar, it is easy to construct more complicated systems which violates the structure assumed for M and doesn't commute with all K's.

9.4 Half Information Problem

In this problem, it is assumed that half of the interconnection signals are available. Specifically, α is measured. Either the system was originally designed so that there are appropriate sensors at this point, or they have been added to the original system. For the case of mechanical systems this would be the addition of accelerometers.

If the system is identified with the interconnection constraint $\phi = 0$,

then the state space I/O model from Figure 9.1

$$\left[\begin{array}{c|c} A & B \\ \hline C_1 & D_1 \\ C_2 & D_2 \end{array}\right]$$

is obtained. Because α can be measured, they can be considered a part of the set of outputs for when the structure is I/O modeled. So the entire ON model of the system (Figure 9.1) is determined except for B_f, D_{f1}, and D_{f2}.

Next the system is identified with a known interconnection constraint,

$$D_{if}\phi = -\alpha. \tag{9.1}$$

The identified model is:

$$\left[\begin{array}{c|c} \overline{A} & \overline{B} \\ \hline \overline{C}_1 & \overline{D}_1 \\ \overline{C}_2 & \overline{D}_2 \end{array}\right] \tag{9.2}$$

The I/O model predicted by the ON model of the system with the known interconnection constraint in (9.1) can be constructed. The ON model (Figure 9.1) is interconnected with the ON model of the algebraic constraint,

$$0 = \left[\begin{array}{cc} D_{if} & I \end{array}\right]\left[\begin{array}{c} \phi_2 \\ \alpha_2 \end{array}\right].$$

The interconnection variables for the mass are α_2 and ϕ_2, where the interconnection is defined by (Using (1.5))

$$T_1 = \left[\begin{array}{cccc} 0 & 0 & I & 0 \\ 0 & 0 & 0 & I \end{array}\right]$$

and

$$T_2 = \left[\begin{array}{cccc} 0 & 0 & I & 0 \\ 0 & 0 & 0 & -I \end{array}\right].$$

The choice of $\alpha = \alpha_2$ and $\phi = -\phi_2$ is motivated from mechanical systems, where the forces are opposite and the accelerations are equal. The ON representation of the interconnected system is: This can be simplified to determine the predicted I/O model of the system by solving for θ and α in terms of x and ι only:

$$\left[\begin{array}{c|c} A + B_f T C_2 & B + B_f T D_2 \\ \hline C_1 + D_{f1} T C_2 & D_1 + D_{f1} T D_2 \\ -D_{if} T C_2 & -D_{if} T D_2 \end{array}\right] \tag{9.3}$$

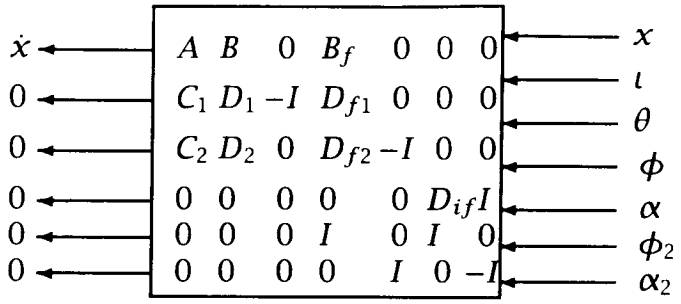

Figure 9.5: ON Rep. of a Structure Connected to D_{if}^{-1}

where

$$T \triangleq -(D_{if} + D_{f2})^{-1} \tag{9.4}$$

Now the predicted I/O model of the interconnected system (9.3) is equated with the identified I/O model (9.2) of the interconnected system. The state space models may differ by a similarity transformation-S and still be equivalent:

Unknown: $S, B_f, T, D_{f1}, D_{f2}$

Given: $A, B, C_1, C_2, D_1, D_2, \overline{A}, \overline{B}, \overline{C}_1, \overline{C}_2, \overline{D}_1, \overline{D}_2, D_{if}$

$$\begin{bmatrix} S^{-1}(A + B_f T C_2) S & S^{-1}(B + B_f T D_2) \\ (C_1 + D_{f1} T C_2) S & D_1 + D_{f1} T D_2 \\ -(D_{if} T C_2) S & -D_{if} T D_2 \end{bmatrix} = \begin{bmatrix} \overline{A} & \overline{B} \\ \overline{C}_1 & \overline{D}_1 \\ \overline{C}_2 & \overline{D}_2 \end{bmatrix} \tag{9.5}$$

Using algebraic manipulations, (9.5) can be converted into

$$\begin{bmatrix} AS - S\overline{A} - B_f D_{if}^{-1} \overline{C}_2 \\ S\overline{B} + B_f D_{if}^{-1} \overline{D}_2 \\ C_1 S - D_{f1} D_{if}^{-1} \overline{C}_2 \\ D_{f1} D_{if}^{-1} \overline{D}_2 \end{bmatrix} = \begin{bmatrix} 0 \\ B \\ \overline{C}_1 \\ D_1 - \overline{D}_1 \end{bmatrix} \tag{9.6}$$

which is linear in the unknowns. There exists a unique least squares solution to this system of linear equations under mild conditions. The only remaining unknowns are: T and D_{f2}

$$T \begin{bmatrix} C_2 S & D_2 \end{bmatrix} = -D_{if}^{-1} \begin{bmatrix} \overline{C}_2 & \overline{D}_2 \end{bmatrix}$$

$$T = -D_{if}^{-1} \left[\begin{array}{cc} \overline{C}_2 & \overline{D}_2 \end{array} \right] R \tag{9.7}$$

where R is defined by

$$I = \left[\begin{array}{cc} C_2 S & D_2 \end{array} \right] R$$

and from the definition of T (9.4),

$$D_{f2} = -T^{-1} - D_{if}. \tag{9.8}$$

Now all of the terms in the ON model have been found. The ON model of the system has been determined from two I/O models of the system with different attached systems, no attachment and a mass.

9.5 Concluding Remarks

This development is a possible methodology for doing the behavioral identification of a flexible structure when some of the signals cannot be actuated actively. The methodology would be the attachment of a known system for passive excitation and the addition of sensors to measure some of the interconnection information. Accelerometers are easy to add to a structure, where trying to drive this interconnections and measuring the driving forces (strain gauges) is much more difficult. This lends to the formulation of the half information problem.

The solution/computation presented in (9.6), (9.7), and (9.8) is a proof that there is sufficient information from the two black box identification procedure to reconstruct the model. The linear system of equations is overconstrained and for a real problem there is no solution. The least squares approach to construct a solution would minimize the error in the state space terms, which is not the desired minimization of the gap [20, 43] between the behaviors. This computation should be reformulated.

Chapter 10

Conclusions and Future Directions

The probabilistic formulation for μ is a natural extension of the μ-framework can be used to describe probabilistic sets of model. This in turn can be used to address probabilistic robustness questions. A simple approach to this problem is to extend the branch and bound methodologies to perform an advanced method of gridding of parameter space to obtain local information. This local information can be used to compute bounds to the probabilistic robustness questions.

This approach has a significant limitation in the growth of the computation cost as a function of problem size. The linear cut approach should improve the cost associated with gridding the boundary of singularity and make the computation for small sized problems tractable. This is a natural direction for future extensions of the work presented. It is unlikely that the exotic gridding techniques will be useful for problems with a large number of parameters unless the problem is sensitive only to a few parameters. In which case, identifying these parameters would be crucial to the success of the methodology.

Although the formulation of spherical μ and its extensions are not useful for the linear cut problem that motivated their development, they do increase the richness of the set of uncertain models in the LFT framework that admit analysis. The connection between ellipsoidal uncertainty and parameter estimation procedures would result in tighter analysis for a broader class of robustness questions than are currently being addressed in the literature.

In order to apply the separating hyperplane argument to construct an upper bound for a μ problem, it is necessary to convert the operator description of Δ to a quadratic signal space description. For many operator descriptions this is not possible. Generalizing the implicit construction for computing an upper bound to μ with linear cuts, tractable LMI upper bounds for very exotic set descriptions for $B\Delta$ can be computed.

The machinery developed for the fast computation for the spherical μ upper bound can be extended to a broader class of LMIs and even further to general linear order inequalities. The contribution of this work is describing special structure that can be exploited for the purpose of improved computation, and describing a class of "easy" LMIs. It may be possible to extend the computational benefits to problems that don't strictly satisfy the cone-preserving structure, but "part" of the problem satisfies this structure.

A promising future direction which incorporates a number of the tools and ideas developed in this presentation is the probabilistic robustness analysis of systems with a large number of probabilistic parameters. Although this problem is computational death using the obvious extensions to the gridding approach presented, but using another approach it may admit computationally tractable "tight" bounds. The motivation for this approach is Paganini's typical set modelling of white noise [35]. The same autocorrelation set description can be applied to the probabilistic parameters. One of the autocorrelation constraints is exactly the spherical constraint. The remainder cannot be converted to signal space descriptions, but by using the implicit construction to apply constraints on the operators the other autocorrelation constraints can be exploited. The result is a *big* implicit μ problem with the intersection of a hypercubic and spherical region.

Bibliography

[1] V. Balakrishnan, Stephen Boyd, and S. Balemi. Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems. *International Journal of Robust and Nonlinear Control*, 1(4):295–317, October–December 1991.

[2] Gary J. Balas. *Robust Control of Flexible Structures, Theory and Experiment.* PhD thesis, California Institute of Technology, 1990.

[3] B. R. Barmish and C. M. Lagoa. The uniform distribution: A rigorous justification for its use in robustness analysis. In *Proceedings of the 35^{th} Conference on Decision and Control*, 1996.

[4] C. L. Beck and J. C. Doyle. Model reduction of behavioural systems. In *Proceedings of the 32^{nd} Conference on Decision and Control*, 1993.

[5] Carolyn Beck, Raffaello D'Andrea, Fernando Paganini, Wei-Min Lu, and John Doyle. A state space theory of uncertain systems. In *Proc. Int. Fed. Auto. Control*, 1996.

[6] Carolyn Beck and John C. Doyle. Mixed μ upper bound computation. In *Proceedings of the 31^{st} Conference on Decision and Control*, pages 3187–3192, 1992.

[7] Abraham Berman and Robert J. Plemmons. *Nonnegative Matrices in the Mathematical Sciences.* Academic Press, New York, 1979.

[8] Stephen P. Boyd, Laurent El Ghaoui, Eric Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM, 1994.

[9] Richard D. Braatz and O. D. Crisalle. Robustness analysis for systems with ellipsoidal uncertainty. *Submitted to International Journal of Robust and Nonlinear Control*, October 1997.

[10] 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.

[11] Jie Chen, Michael K. H. Fan, and Carl N. Nett. The structured singular value and stability of uncertain polynomials: The generalized μ. *Systems and Control Letters*, 23, 1994.

[12] Jie Chen, Michael K. H. Fan, and Carl N. Nett. The structured singular value and stability of uncertain polynomials: A missing link. *Systems and Control Letters*, 23:97-109, 1994.

[13] Raffaello D'Andrea and Fernando Paganini. Interconnection of uncertain behavioral systems for robust control. In *Proceedings of the 32nd Conference on Decision and Control*, pages 3642-3647, 1993.

[14] Raffaello D'Andrea and Fernando Paganini. Why behave? In *Proceedings of the 32nd Conference on Decision and Control*, 1993.

[15] Raffaello D'Andrea, Fernando Paganini, and John C. Doyle. Uncertain behavior. In *Proceedings of the 32nd Conference on Decision and Control*, pages 3891-3896, 1993.

[16] John C. Doyle. Analysis of feedback systems with structured uncertainty. *IEE Proceedings, Part D*, 129(6):242-250, November 1982.

[17] M. Fu. The real structured singular value is hardly approximable. *IEEE Transactions on Automatic Control*, 49, 1997.

[18] Pascal Gahinet, Arkadi Nemirovski, Alan J. Laub, and Mahmoud Chilali. *LMI Control Toolbox.* The MathWorks Inc., Natick, Mass., May 1995.

[19] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP Completeness.* W. H. Freeman, New York, 1979.

[20] T. T. Georgiou and M. C. Smith. Optimal robustness in the gap metric. *IEEE Transactions on Automatic Control*, 35(7):673-686, July 1990.

[21] Gene H. Golub and Charles F. Van Loan. *Matrix Computations.* The Johns Hopkins University Press, Baltimore, 1987.

[22] Roger A. Horn and Charles R. Johnson. *Matrix Analysis.* Cambridge University Press, New York, 1985.

[23] S. Khatri and P. A. Parrilo. Guaranteed bounds for probabilistic μ. In *Proceedings of the 37th Conference on Decision and Control*, 1998.

[24] S. Khatri and P. A. Parrilo. Spherical μ. In *Proceedings of the American Control Conference*, 1998.

[25] Lennart Ljung. *System Identification, Theory for the User.* Information and System Sciences Series. Prentice-Hall, New Jersey, 1987.

[26] A. Megretski. Necessary and sufficient conditions of stability: A multiloop generalization of the circle criterion. *IEEE Transactions on Automatic Control*, 38:753-756, 1993.

[27] A. Megretski and S. Treil. S-procedure and power distribution inequalities: A new method in optimization and robustness of uncertain systems. *Mittag-Leffler Institute*, 1991.

[28] Yurii E. Nesterov and Arkadi Nemirovski. *Interior point polynomial methods in convex programming*, volume 13 of *Studies in Applied Mathematics.* SIAM, Philadelphia, PA, 1994.

[29] M. Newlin. *Model Validation, Control, and Computation.* PhD thesis, California Institute of Technology, 1996.

[30] Matthew P. Newlin and Sonja T. Glavaski. Advances in the computation of the μ lower bound. In *Proceedings of the American Control Conference*, pages 442-446, 1995.

[31] Matthew P. Newlin and Roy S. Smith. A generalization of the structured singular value and its application to model validation. *IEEE Transactions on Automatic Control*, 43(7):901-907, 1998.

[32] Matthew P. Newlin and Peter M. Young. Mixed μ problems and branch and bound techniques. *International Journal of Robust and Nonlinear Control*, pages 145-164, 1996.

[33] E. E. Osborne. On preconditioning of matrices. *Journal of the Assoc. for Comp. Mach.*, 7:338-349, 1960.

[34] Andrew K. Packard and John C. Doyle. The complex structured singular value. *Automatica*, 29:71-109, 1993.

[35] F. Paganini. *Sets and Constraints in the Analysis of Uncertain Systems.* PhD thesis, California Institute of Technology, 1995.

[36] P. A. Parrilo and S. Khatri. Closed form solutions for a class of LMIs. In *Proceedings of the American Control Conference*, 1998.

[37] Kameshwar Poolla and Ashok Tikku. Robust performance against time-varying structured perturbations. *ieeeTAC*, 40:1589–1601, 1995.

[38] L. R. Ray and R. F. Stengel. A monte carlo approach to the analysis of control systems robustness. *Automatica*, 3, 1993.

[39] R. Tyrrell Rockafellar. *Convex Analysis.* Princeton University Press, Princeton, New Jersey, 1970.

[40] Michael G. Safonov. Stability margins for diagonally perturbed multivariable feedback systems. *IEE Proceedings, Part D*, 129(6):251–256, November 1982.

[41] Jeff S. Shamma. Robust stability with time varying structured uncertainty. *IEEE Transactions on Automatic Control*, 39:714–724, 1994.

[42] R. F. Stengel and L. R. Ray. Stochastic robustness of linear time-invariant systems. *IEEE Transactions on Automatic Control*, 36, 1991.

[43] Glenn Vinniecombe. Frequency domain uncertainty and the graph topology. *IEEE Transactions on Automatic Control*, 38:1371–1383, 1993.

[44] Jan C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Transactions on Automatic Control*, 36:259–294, 1991.

[45] P. Young, M. Newlin, and J. Doyle. Structured singular value analysis with real and complex uncertainties. In *Proceedings of the 30^{th} Conference on Decision and Control*, 1991.

[46] P. M. Young, M. P. Newlin, and J. C. Doyle. Let's get real. CDS Technical Memo CIT-CDS 92-001, California Institute of Technology, Pasadena, CA 91125, September 1992.

[47] Peter M. Young and John C. Doyle. A lower bound for the mixed μ problem. Submitted to IEEE Transactions on Automatic Control.

[48] Peter M. Young, Matthew P. Newlin, and John C. Doyle. Let's get real. In *Robust Control Theory*, pages 143–173. Springer-Verlag, 1995. IMA Proceedings Volume 66.

[49] Kemin Zhou, Keith Glover, and John Doyle. *Robust and Optimal Control.* Prentice Hall, 1995.

[50] X. Zhu, Y. Huang, and J. Doyle. Soft vs. hard bounds in probablistic robustness analysis. In *Proceedings of the American Control Conference*, 1996.