

Mixed μ Upper Bound Computation

Carolyn Beck John Doyle
Electrical Engineering, M/S 116-81
California Institute of Technology
Pasadena, CA 91125

Abstract

This paper considers computation of the mixed real and complex μ upper bound expressed in terms of Linear Matrix Inequalities (LMIs). We build on two existing methods, Osborne's for balancing matrices, and the method of centers as proposed by Boyd and El Ghaoui. These methods are compared and a hybrid algorithm that combines the best features of each is proposed. Numerical experiments suggest that this hybrid algorithm provides an efficient method to compute the upper bound for mixed μ .

1 Introduction

The structured singular value, μ , was first introduced in [1] as a method for analyzing the robustness of systems with respect to structured uncertainties. A large variety of robust stability and robust performance problems can easily be reformulated as μ computation problems, with respect to a block structure which we denote by Δ . In general, the exact computation of μ is quite difficult, thus computational efforts have emphasized methods for determining upper and lower bounds, with fairly good results for problems containing complex only uncertainties. One very efficient method for approximating the complex upper bound which is widely used is based on Osborne's method for preconditioning matrices [2]. An alternative method which is often used and is efficient for small size problems has been developed based on results due to Safonov [3].

For problems containing both real and complex uncertainties, i.e., mixed μ problems, computational methods for determining the upper bound are either accurate but too slow, or efficient but not consistently accurate. An efficient but somewhat ad-hoc algorithm for computing the upper bound developed by Young and Newlin [4] first balances the mixed μ upper bound problem statement in such a way that the Osborne method for the complex μ upper bound can be applied yielding a good initial estimate for the solution. The problem is then reformulated in a more standard format and a subgradient method applied, resulting in a more accurate solution. Fan has implemented a cutting plane algorithm which solves the mixed μ upper bound problem formulated in [5]. Although the cutting plane method is fairly accurate, it is relatively slow. An algorithm for solving the mixed μ problem which is both accurate and efficient remains to be developed.

Recently, much attention has been focused on determining efficient and accurate computational schemes for solving a class of problems to which both the mixed and complex μ upper bound problems belong. The problems considered in many of these efforts are written as maximum eigenvalue or maximum generalized eigenvalue optimization problems, which are equivalently formulated as linear matrix inequality (LMI) optimization problems. Nesterov and Nemirovsky [6], Boyd and El Ghaoui [7], and Jarre [8] have utilized interior point methods for solving these LMI problems with promising results. Fan [9] has also implemented an algorithm for solving this class of problems. We have been primarily motivated by two methods in our search for determining a satisfactory computational solution to the mixed μ upper bound problem: interior point methods for solving LMI optimization problems, and the Osborne method for computing the complex μ upper bound.

In Section 2, both the complex and mixed μ problems are defined,

along with associated bounds. A general description of LMI optimization via interior point methods is given in Section 3, along with a brief outline of recent results due to Nesterov and Nemirovsky, Jarre, and Boyd and El Ghaoui. In Section 4, Osborne's method for preconditioning matrices and the application of this method to the complex μ upper bound problem is discussed. An extension of Osborne's method to the mixed μ upper bound problem is described in Section 5. A comparison, including numerical experiments, of the computational methods is presented in Section 6, and in Section 7 a brief summary of our findings is given.

2 Preliminaries

The notation used in this paper is fairly standard, and is essentially a simplified version of that used in [5] and [4]. Given a matrix $M \in \mathbb{C}^{n \times n}$, we denote the complex conjugate transpose by M^* , the spectral radius by $\rho(M)$, the maximum singular value by $\bar{\sigma}(M)$, and the structured singular value by $\mu(M)$. Consider the system shown in Figure 1, where $M \in \mathbb{C}^{n \times n}$ represents what is known about the system, and Δ is a block diagonal matrix representing what is varied or unknown about the system, i.e., perturbations or uncertainties. The definition of μ for this

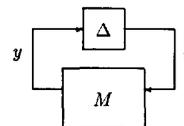


Figure 1: Standard System

system depends explicitly on the Δ block structure. For the purpose of this paper, the Δ block structure is restricted to the following set. Given two non-negative integers m_r and m_c , with $m_r + m_c = n$,

$$\Delta = \{diag[\delta_1^r, \dots, \delta_{m_r}^r, \delta_1^c, \dots, \delta_{m_c}^c] : \delta_i^r \in \mathbb{R}, \delta_i^c \in \mathbb{C}\}.$$

Note that for the purely complex case $m_r = 0$.

Definition 1 The structured singular value, $\mu_\Delta(M)$, of a matrix $M \in \mathbb{C}^{n \times n}$ with respect to a block structure Δ is defined as

$$\mu_\Delta(M) = \left(\min_{\Delta \in \Delta} \{\bar{\sigma}(\Delta) : \det(I - \Delta M) = 0\} \right)^{-1} \quad (1)$$

with $\mu_\Delta = 0$ if no $\Delta \in \Delta$ solves $\det(I - \Delta M) = 0$.

In the purely complex case, we conclude directly from the definition of μ

$$\rho(M) \leq \mu_\Delta(M) \leq \bar{\sigma}(M). \quad (2)$$

A substantial gap may exist between ρ and $\bar{\sigma}$, thus these bounds are refined by considering transformations on M which do not affect $\mu_\Delta(M)$ but do affect ρ and $\bar{\sigma}$. Since we are only interested in the upper bound in this paper, we discuss only transformations on M which affect $\bar{\sigma}(M)$. For a more in depth discussion of the lower bound, see for example [4] and [10]. Consider the following subset of $\mathbb{C}^{n \times n}$

$$D = \{diag[d_1, \dots, d_{m_c}] : 0 < d_i \in \mathbb{R}\}$$

Note that for $\Delta \in \mathbf{\Delta}$ and $D \in \mathcal{D}$,

$$D\Delta = \Delta D \quad (3)$$

which leads to the following theorem.

Theorem 1 For all $D \in \mathcal{D}$

$$\mu_{\Delta}(M) = \mu_{\Delta}(DM D^{-1}). \quad (4)$$

Therefore the upper bound for the complex μ problem can be tightened to

$$\mu_{\Delta}(M) \leq \inf_{D \in \mathcal{D}} \bar{\sigma}(DM D^{-1}). \quad (5)$$

In the mixed μ problem, the upper bound is more complicated. As one simple approach, we could cover the real perturbations with complex perturbations and use the complex μ upper bound, as this would cover the perturbation set $\mathbf{\Delta}$. However, this method does not exploit any of the phase information given by the real perturbations, resulting in an upper bound which is too conservative. The upper bound defined in [5] does use the phase information given by the real perturbations, and is at least as good as the complex μ upper bound and often much better. Consider the following subsets of $\mathbf{C}^{n \times n}$.

$$\mathcal{D} = \{\text{diag}[d_1, \dots, d_{m_r}, d_{m_r+1}, \dots, d_{m_r+m_c}] : 0 < d_i \in \mathbf{R}\}$$

$$\mathcal{G} = \{\text{diag}[g_1, \dots, g_{m_r}, 0, \dots, 0] : g_i \in \mathbf{R}\}$$

Using this notation, an upper bound for the mixed μ problem is given in the following theorem which is taken from [5].

Theorem 2 Given a matrix $M \in \mathbf{C}^{n \times n}$, and a compatible block structure $\mathbf{\Delta}$, suppose α_* is the result of the minimization problem

$$\alpha_* = \inf_{D \in \mathcal{D}, G \in \mathcal{G}} \left[\min_{\alpha \in \mathbf{R}} \{\alpha : M^*DM + j(GM - M^*G) - \alpha D \leq 0\} \right] \quad (6)$$

then $\mu_{\Delta}(M) = 0$ if $\alpha_* \leq 0$, otherwise

$$\mu_{\Delta}(M) \leq \sqrt{\alpha_*} \quad (7)$$

The upper bound given by this theorem is easily written as an LMI optimization problem, as discussed in the following section.

3 Optimization Methods for LMIs

General LMI problems can be described by the following decision problem:

$$\text{Does } \exists X : A^*XA - B^*XB + XC + C^*X + D \leq 0 \quad (8)$$

subject to: $X \in \mathcal{X}$

where \mathcal{X} describes the feasible set for X , i.e., the set of block diagonal structured Hermitian matrices, and A, B, C and D are constant matrices having corresponding block partitions. In the complex μ upper bound case X must also be positive definite [11]. Henceforth, we will denote $A^*XA - B^*XB + XC + C^*X + D$ by $\mathcal{A}(X)$. One way of reformulating the above LMI decision problem as an optimization problem is

$$\min_X \lambda_{\max} \mathcal{A}(X) \quad (9)$$

subject to: $X \in \mathcal{X}$

where $\lambda_{\max} \mathcal{A}(X)$ is referred to as the objective function. When $\mathcal{A}(X)$ is an Hermitian matrix which is affine in X , $\lambda_{\max} \mathcal{A}(X)$ belongs to the class of convex, non-differentiable functions for which numerous optimization methods have been developed. Specifically, the minimization of the maximum eigenvalue of a symmetric matrix which depends linearly on a real set of parameters has received considerable attention [12], [13], [8].

As special a case of the problem represented by (8), the mixed μ upper bound optimization problem can be formulated as an LMI decision problem as follows:

$$\text{Does } \exists D, G : M^*DM + j(GM - M^*G) - \alpha D \leq 0 \text{ and } D \geq 0 \quad (10)$$

subject to: $D \in \mathcal{D}, G \in \mathcal{G}$

where \mathcal{D} and \mathcal{G} are defined as in Section 2, $M \in \mathbf{C}^{n \times n}$, and $\alpha \in \mathbf{R}$. We can easily put (10) into the exact form given in (8) or (9) by defining $X_{\mu} = \begin{bmatrix} D & 0 \\ 0 & G \end{bmatrix}$, $A_{\mu} = \begin{bmatrix} M \\ 0 \end{bmatrix}$, $B_{\mu} = \begin{bmatrix} \beta I \\ 0 \end{bmatrix}$, $C_{\mu} = \begin{bmatrix} 0 \\ jM \end{bmatrix}$, and $D_{\mu} = 0$ where $\beta = \sqrt{\alpha}$. Thus the mixed μ upper bound problem is easily put into LMI form. However, to "solve" the mixed μ upper bound, a series of iterations on α must be performed, that is, a series of LMIs must be optimized, with decreasing α values. Thus the mixed μ upper bound problem in LMI optimization form is

$$\min \left\{ \alpha : \min_{D, G} \lambda_{\max}(M^*DM + j(GM - M^*G) - \alpha D) \leq 0 \right\} \quad (11)$$

subject to: $D \in \mathcal{D}, G \in \mathcal{G}$.

Alternatively, LMI problems in general, and more specifically the mixed μ upper bound problem defined in [5] can be reformulated as generalized eigenvalue optimization problems [7], [15],

$$\mu_{\Delta}(M) \leq \min_{D, G} \lambda_{\max}(M^*DM + j(GM - M^*G), D) \quad (12)$$

subject to: $D \in \mathcal{D}, G \in \mathcal{G}$

where $\lambda_{\max}(X, Y)$ for the pair $X = X^*, Y = Y^*$ is defined as

$$\lambda_{\max}(X, Y) = \inf \{ \lambda \in \mathbf{R} : \lambda Y - X > 0 \}. \quad (13)$$

The minimization of the maximum generalized eigenvalue of a pair of such matrices is not a convex optimization problem, in contrast to (9). However it is quasiconvex, for which a number of reliable optimization methods exist, for example cutting plane methods such as the ellipsoid algorithm developed by Shor, Nemirosky and Yudin, or Kelley's cutting plane algorithm [17], [18]. This problem has been studied more recently (see [7] and the references therein). Applications of interior point methods and the theory of *self-concordant* functions, which are discussed in the following subsection, have yielded promising results [14], [6], [7].

3.1 Interior Point Methods

Interior and exterior point methods are used to convert constrained minimization problems to unconstrained minimization problems, to which optimization algorithms such as Newton's method are applied. Both methods utilize a *penalty* function, usually constructed from the constraint functions, which is added to the objective function. These penalty functions greatly increase the objective function *cost* for violation of constraints. Interior point methods are not applicable to problems with equality constraints, whereas exterior point methods may be used on problems with equality and/or inequality constraints.

The basic idea behind interior point methods is to convert the inequality constraint functions into a *barrier* function; that is, a function whose value approaches infinity near the border of the feasible region. For example, suppose we have a standard optimization problem with the objective function denoted by f and inequality constraint functions denoted by $h_i, 1 \leq i \leq m$, so that our optimization problem is stated as

$$\min_x f(x) \quad (14)$$

subject to: $h_i(x) \geq 0, 1 \leq i \leq m$.

One commonly used barrier function is the logarithmic barrier function,

$$\phi(x, r) = -r \sum_{i=1}^m \ln h_i(x) \quad (15)$$

where the parameter r determines the weight of the function. An alternative barrier function is given by

$$\phi(x, r) = r^2 \sum_{i=1}^m \frac{1}{h_i(x)}. \quad (16)$$

A new unconstrained optimization problem is then given by

$$\min_x P(x, r) = \min_x (f(x) + \phi(x, r)) \quad (17)$$

where r decreases as the iterations progress. Under mild conditions, $\lim_{r \rightarrow 0} (\min_x P(x, r)) = \min_x f(x)$ ([21], [20]). This minimum is achieved in the interior of the feasible region, since near the boundary of the feasible set, $P(x, r)$ approaches infinity.

3.2 Applications of Interior Point Methods and the Method of Centers

Recently, interior point methods have been applied to a variety of LMI problems with favorable results. The interest in interior point methods for solving LMIs is due mainly to the work of Nesterov and Nemirovsky, [14], [6], who propose an algorithm for solving problems such as (17), and show that a large class of logarithmic barrier functions satisfy a desirable property called *strong self-concordance*. Nesterov and Nemirovsky's algorithm utilizes Newton's method in the following manner. Starting from some initial point x^1 , iteratively determine $x^* = \operatorname{argmin} P(x, r)$ such that at the k th iteration

$$x^{k+1} = x^k - s^k H^{-1}(x^k) g(x^k) \quad (18)$$

where $H(x)$ is the Hessian and $g(x)$ the gradient of $P(x, r)$ with respect to x , and s^k is the step size of the k th iteration. A simple rule for determining the step size s^k is given. Nesterov and Nemirovsky prove that if the function ϕ is strongly self-concordant, then computable bounds exist on the number of iterations of (18) required to find x^* to within a given tolerance.

As one application of Nesterov and Nemirovsky's methods, Jarre [8] develops an interior point algorithm which utilizes a logarithmic barrier function for a problem such as that given in (9). He applies this algorithm to (9) and compares the results to those found for other convex programming methods. Jarre's interior point algorithm requires fewer iterations than either the subgradient method or the cutting-plane algorithm employed by Boyd and Yang [13]. Similarly, Boyd and El Ghaoui [7] apply Nesterov and Nemirovsky's methods to the following generalized eigenvalue problem.

$$\begin{aligned} \min \lambda_{\max}(A(x), B(x)) \\ \text{subject to: } B(x) > 0 \end{aligned} \quad (19)$$

where $A(x)$ and $B(x)$ are symmetric matrices which depend affinely on $x \in \mathbb{R}^n$, and satisfy a given set of assumptions. They utilize the following logarithmic barrier function

$$\phi(x) = -\log \det \mathcal{F}(x) \quad (20)$$

where

$$\mathcal{F}(x) = \lambda B(x) - A(x) \quad (21)$$

and it is assumed that λ is chosen so that $\mathcal{F}(x) > 0$. This barrier function is analytic, and strictly convex. Additionally, the Hessian and gradient functions for $\phi(x)$, $H(x)$ and $\operatorname{grad}(x)$, are easily computed as

$$\operatorname{grad}_i(x) = -\operatorname{Trace}(\mathcal{F}(x)^{-1} \mathcal{F}_i) \quad (22)$$

and

$$H_{ij}(x) = \operatorname{Trace}(\mathcal{F}(x)^{-1} \mathcal{F}_i \mathcal{F}(x)^{-1} \mathcal{F}_j) \quad (23)$$

where $\mathcal{F}_i = \frac{\partial \mathcal{F}}{\partial x_i}$. It is easily shown that the resulting quadratic approximation can be determined by a single Frobenius norm minimization, which does not require forming the Hessian. The optimum point $x^* = \operatorname{argmin} \phi(x)$ is often referred to as the *analytic center* of $\mathcal{F}(x)$ [8], [7].

Boyd and El Ghaoui implement an iterative optimization algorithm for (19) which is based on the *method of centers* due to Huard [19], so called because at each iteration the analytic center of $\mathcal{F}(x)$ is determined for the current λ value. The algorithm starts at an initial point (λ^1, x^1) and proceeds as follows:

$$\lambda^{k+1} = (1 - \theta) \lambda_{\max}(A(x^k), B(x^k)) + \theta \lambda^k \quad (24)$$

$$x^{k+1} = x^*(\lambda^{k+1}) \quad (25)$$

where $0 < \theta < 1$. The method of centers proposed by Huard is completed with $\theta = 0$. The step given by (25) represents a separate iterative process. At each stage of the algorithm represented by (24) and (25) the analytic center, $x^*(\lambda^{k+1})$, is determined using an iterative Newton method similar to that proposed by Nesterov and Nemirovsky, however, the step size s^k is determined by an exact line search. Boyd and El Ghaoui provide a convergence analysis for their method of centers, indicating that λ^k converges to a minimum, λ^* , with at least geometric convergence. Additionally, stopping criteria are provided for this algorithm. See [7] for a more thorough description of this algorithm and its properties.

Boyd and El Ghaoui's method of centers immediately extends to affine functions of diagonal matrices, X , i.e., $A(X)$, $B(X)$ and $F(X)$, with $F_i = \frac{\partial \mathcal{F}}{\partial x_{ii}}$. As an example consider the mixed μ upper bound problem given by (12). In this case, the variable set is denoted by (D, G) and the equations corresponding to (20) and (21) are

$$\phi(D, G) = -\log \det \mathcal{F}(D, G) \quad (26)$$

where

$$\mathcal{F}(D, G) = \lambda B(D, G) - A(D, G), \quad (27)$$

$$B(D, G) = D, \text{ and } A(D, G) = M^* D M + j(GM - M^* G).$$

Again it is assumed that λ is chosen so that $\mathcal{F}(D, G) > 0$. The complex case is easily represented by the above with $G = 0$, i.e.,

$$\mathcal{F}(D) = \lambda D - M^* D M. \quad (28)$$

An application of Boyd and El Ghaoui's algorithm to the mixed μ upper bound problem given by (12) is discussed in [15]. We have also implemented a version of this algorithm for comparison to the extended Osborne method described in Section 5. The Osborne method for the complex μ upper bound is described first in the following section.

4 Osborne's Method for Preconditioning Matrices

Osborne presents a method in [2] for determining norm-reducing similarity transformations for square matrices. The norm to which he refers is the Frobenius, or Euclidean, matrix norm defined as follows. Let $M \in \mathbb{C}^{n \times n}$, then

$$\|M\|_F = \left(\sum_{i,j=1}^n |m_{ij}|^2 \right)^{\frac{1}{2}} = (\operatorname{Trace}(M^* M))^{\frac{1}{2}}. \quad (29)$$

The method Osborne develops yields a series of similarity transformations, D_k , such that

$$\lim_{k \rightarrow \infty} \|D_k M D_k^{-1}\|_F = \inf_D \|D M D^{-1}\|_F. \quad (30)$$

The resulting norm-reduced matrix is often referred to as *balanced* since the Euclidean norms of the corresponding rows and columns are equal. This method is often used to precondition matrices before calling other linear algebra routines, such as eigenvalue decompositions and least squares problems. The computation cost is $O(n^2)$ which is negligible compared to many standard linear algebra computations which are $O(n^3)$.

Due to the following well-known relation

$$\frac{1}{\sqrt{n}} \|M\|_F \leq \bar{\sigma}(M) \leq \|M\|_F \quad (31)$$

the norm reduction method given by Osborne for the Frobenius norm of $D M D^{-1}$ is also a norm reduction method for the complex μ upper bound when the similarity transformations, D , are restricted to the set defined by \mathcal{D} . In fact, solving for $D^* = \operatorname{argmin}_{\mathcal{D}} \|D M D^{-1}\|_F$ by Osborne's method results in reasonably good approximations to the problem $D^* = \operatorname{argmin}_{\mathcal{D}} \bar{\sigma}(D M D^{-1})$.

Osborne presents an iterative method which is summarized in the following. For a given matrix M , denote the ij th element by m_{ij} ,

the i th row by $m_{i\bullet}$ and the i th column by $m_{\bullet i}$. Let e_i represent the column vector where the i th element is 1 and all other elements are 0. We consider the Euclidean norm for vectors and define

$$R_i = \|m_{i\bullet} - m_{ii}e_i\|_E, \quad (32)$$

$$S_i = \|m_{\bullet i} - m_{ii}e_i^T\|_E \quad (33)$$

and

$$\bar{d}_i = \left(\frac{S_i}{R_i} \right). \quad (34)$$

Note that the positive, real number \bar{d}_i is chosen to minimize $\bar{d}_i^2 R_i^2 + \frac{1}{\bar{d}_i^2} S_i^2$ and makes the resulting scaled row and column have equal norm. The rows and columns of M are modified in a cyclic order, such that at the k th iteration, the i th row and column are altered where $i - 1$ is the least positive residue of $k - 1$ modulo n . The process is initialized with $D_1 = I$. At each iteration, the following steps are completed:

1. Given D_k , determine

$$\bar{D}_k = \begin{cases} \text{diag}[1, \dots, 1, \bar{d}_i, 1, \dots, 1] & i < n \\ \text{diag}[\bar{d}_i^{-1}, \dots, \bar{d}_i^{-1}, 1] & i = n \end{cases}$$

where \bar{d}_i is computed using equations (32) - (34) for the matrix $M_k = D_k M D_k^{-1}$.

2. Set $D_{k+1} = D_k \bar{D}_k$.

The iterations proceed until \bar{d}_i is sufficiently close to 1 for n consecutive steps. Proof of convergence for the statement given in (30) is provided by Osborne in [2]. Since the resulting D_k 's are diagonal at every step, and the elements \bar{d}_i are real and positive, the resulting transformation is always an element of the set \mathcal{D} . If either R_i or S_i is zero, M is called *degenerate* and a simple modification of the algorithm is required. We will assume, without loss of generality, that such a degeneracy does not arise.

4.1 Osborne's Method for the Complex μ Upper Bound

Osborne's method can be implemented exactly as suggested by Osborne [2] and described above but we actually use an alternative that uses more matrix oriented computations and avoids the looping required by Osborne's algorithm. This is important when using interpretive languages like Matlab, which is used for the numerical experiments in this paper. The modified version of Osborne's method, which we refer to as the element-by-element (EBE) approach, is outlined as follows. Define R_i , S_i and \bar{d}_i as in equations (32) - (34). Initialize the process with $D_1 = I$ and $M_1 = M$. At each iteration the following steps are completed:

1. Determine $M_{k+1} = D_k^{\frac{1}{2}} M_k D_k^{-\frac{1}{2}}$.
2. For M_{k+1} , determine $\bar{d}_i \forall i \in \{1, \dots, n\}$.
3. Set $D_{k+1} = \text{diag}[\bar{d}_1, \dots, \bar{d}_n]$.

The iterations proceed until, as above, D_k is sufficiently close to I . The EBE approach is much faster in Matlab than the original Osborne method. The application of Osborne's method to the computation of the complex μ upper bound, via the EBE method, yields an extremely fast and often accurate solution.

It can be shown that for each M_{k+1} , the step in D_{k+1} is equivalent to a modified Newton step for $\|DM D^{-1}\|_F$, where the Hessian is replaced by its diagonal. As this Hessian is diagonally dominant, the EBE method is a good approximation to a full Newton's method, but is much faster, as the full Hessian is not computed or inverted.

Although Osborne's method nearly always results in a good approximation to $\min \bar{\sigma}(DM D^{-1})$, examples for which this method fails are easily constructed. Consider the inequality in (31). The left inequality is achieved on normal matrices (i.e., matrices for which $MM^* = M^*M$), and the right inequality is achieved on rank one

matrices. For example, consider the $n \times n$ rank one matrix X , with $X_{ij} = 1, \forall i, j$ and the $n \times n$ matrix $Y = \sqrt{(n)}I$, and construct the matrix

$$M = \begin{bmatrix} 0 & X \\ Y & 0 \end{bmatrix}$$

Thus $\bar{\sigma}(M) = \bar{\sigma}(X) = n$. Note that M is already balanced in the Osborne sense, that is, the Euclidean norm of the i th row is equivalent to the Euclidean norm of the i th column, for every $i \in \{1, \dots, n\}$. The minimum, however, of $\bar{\sigma}(DM D^{-1}) = n^{3/4}$ is obtained by letting $d = n^{-1/4}$ and $D = \text{diag}[dI, I]$. Thus the optimal Osborne solution can be off by as much $n^{1/4}$.

Although the gap is usually much lower than $n^{1/4}$, we'd like to determine a method which performs more consistently and which extends Osborne to the mixed μ case. The mixed μ upper bound problem can be formulated as the minimization of a maximum singular value [4],

$$\min_{D, G} \bar{\sigma} \left((I + G^2)^{-\frac{1}{4}} \left(\frac{DM D^{-1}}{\beta} - jG \right) (I + G^2)^{-\frac{1}{4}} \right) \quad (35)$$

however, minimizing the Frobenius norm of (35) results in a good solution for the D matrix, but not for the G matrix. Thus we have chosen to look for a blend of Osborne's method and the method of centers proposed by Boyd and El Ghaoui as a means for computing the mixed μ upper bound.

5 Extension of Osborne's Method to Mixed Upper Bound

First, consider the function

$$\Gamma(\lambda, D) = \frac{\text{Det}(D)}{\text{Det}(\mathcal{F}(D))} = \left(\text{Det}(\lambda I - D^{-1} M^* D M) \right)^{-1}. \quad (36)$$

For large values of λ we can approximate $\Gamma(\lambda, D)$ by

$$\Gamma(\lambda, D) \approx \lambda^n \left(1 + \frac{1}{\lambda} \text{Trace}(D^{-1} M^* D M) + \dots \right). \quad (37)$$

Since $\text{Trace}(D^{-1} M^* D M) = \|D^{-\frac{1}{2}} M D^{\frac{1}{2}}\|_F^2$, minimizing $\Gamma(\lambda, D)$ for large λ is essentially equivalent to Osborne's method. If we used the logarithmic function

$$\Phi(\lambda, D) = \log \Gamma(\lambda, D) = -\log \text{det}(\mathcal{F}(D)) + \log \text{det}(D) \quad (38)$$

or in the mixed μ case

$$\Phi(\lambda, D, G) = \log \frac{\text{Det}(D)}{\text{Det}(\mathcal{F}(D, G))} = -\log \text{det}(\mathcal{F}(D, G)) + \log \text{det}(D). \quad (39)$$

in the method of centers, then Osborne's method would be a limiting case of the method of centers for large λ . Note that in trying to minimize $\Phi(\lambda, D, G)$, the $\text{Det}(D)$ term prevents us from merely increasing the values of the elements of D without limit. Unfortunately, $\Phi(\lambda, D, G)$ is not a barrier function for (D, G) , due to the $\text{Det}(D)$ term, and in fact is not a convex function. Thus the Hessian for $\Phi(\lambda, D, G)$ is indefinite and Newton type methods will not work.

Although we cannot completely connect Osborne's method with the standard method of centers, these considerations do suggest an extension of Osborne's method. The actual function we will consider is

$$\Psi(\lambda, D, G) = \log \frac{\text{Det}(\epsilon I + D)}{\text{Det}(\mathcal{F}(D, G))} \quad (40)$$

If we write $\Psi(\lambda, D, G)$ as a function of a single pair of d_i and g_i for some i , while holding the other variables constant, it is easy to minimize Ψ analytically. (Note that $\text{Det}(\mathcal{F}(D, G))$ is quadratic in each d_i , g_i pair when the remaining variables are held constant.) Thus we can obtain what we will refer to as the EBE method by simply using the d_i and g_i so obtained as a search direction.

Recall that Osborne's method corresponds to choosing a large value for λ , leaving it constant, and then optimizing $\Psi(\lambda, D, G)$. However, as we actually desire to minimize λ , not $\Psi(\lambda, D, G)$, we complete only a few iterations of the EBE algorithm, and then update λ . This approach is more successful for the mixed μ problem than leaving λ fixed or updating it after many EBE iterations. Also, unlike Osborne, we do a line search on Ψ . This modified Osborne method gives a better bound than standard Osborne in the complex case, and extends to the mixed problem, where it also seems to give good bounds. At the same time, it is much cheaper computationally than the method of centers. Unfortunately, it does not always converge to the optimum bound, and so is best suited to computing initial starting values for other methods, like the method of centers.

Our best algorithm to date involves a combination that begins with iterations of the EBE method, and then shifts to the method of centers using the function given by (26) with one d_i variable held constant. The remainder of this paper will explore in more detail the properties of this algorithm. In particular, we will compare more closely our EBE algorithm with the method of centers to better understand how the shift from one to the other should take place.

6 Comparison of EBE versus Method of Centers

In implementing the method of centers for the mixed μ upper bound problem, the gradient and Hessian formulas given by (22) and (23) can be simplified in such a manner that no matrix trace terms need to be computed. For example, if we denote

$$E = \mathcal{F}(D, G)^{-1}, \quad J = \mathcal{F}(D, G)^{-1} M^*, \quad K = M \mathcal{F}(D, G)^{-1} M^* \quad (41)$$

where M is the given system matrix as originally defined, the gradients given by (22) can be simplified to the following form. For gradients taken over the d_i variables

$$\text{grad}_i(D, G) = -\alpha^2 E_{ii} + K_{ii}, \quad (42)$$

and for gradients taken over the g_i variables

$$\text{grad}_i(D, G) = 2\Im(J_{ii}). \quad (43)$$

Similar formulas may be obtained for the Hessian. Additionally, as mentioned in Section 5, one d_i variable corresponding to a complex perturbation is held constant.

While the standard method of centers uses Newton's method, it may be more efficient to avoid computing full Newton steps when possible. The main problem with using Newton steps to find the analytic center is that the initial steps it gives are extremely small, resulting in a very slow initial progression to the analytic center. The Hessian for either $\phi(D, G)$ or $\text{Det}(\mathcal{F}(D, G))^{-1}$ near the boundary of the region where $(\mathcal{F}(D, G))$ is positive definite contains extremely large values, thus the steps are small and may even be in poor directions. As the Hessian for $\phi(D, G)$ has strictly smaller values than that for $\text{Det}(\mathcal{F}(D, G))^{-1}$, it is best to use the former in the method of centers.

An additional problem with the method of centers is that when λ is far from the optimum, minimizing $-\log\det$ between each λ update is wasteful since we really want to minimize λ . It is typically much better to update λ before the analytic center is found, for two reasons. The first is that the Newton steps to compute the analytic center are expensive and their number should be minimized. The second is that in the process of finding the analytic center, it is typical that the optimal λ initially goes down, but then increases as the analytic center is approached. Thus stopping the iteration before the analytic center is found not only saves expensive Newton steps, but also results in even smaller values for λ at each iteration.

6.1 Numerical experiments

A numerical experiment illustrates the relationship between the EBE method and the method of centers. For each algorithm, there are

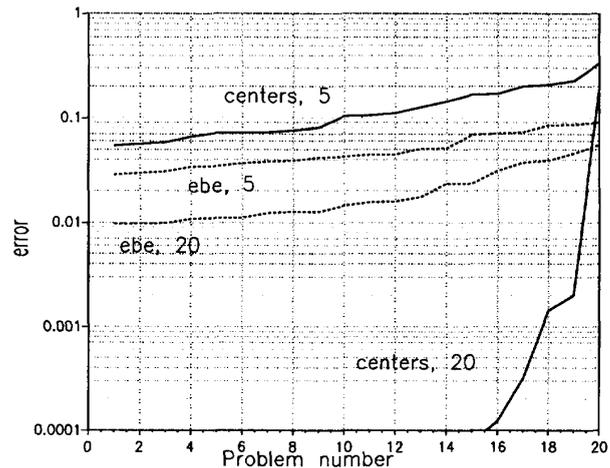


Figure 2: Errors for EBE and method of centers for 20 problems using 5 and 20 iterations

numerous choices that can be made that affect the outcome, and we have not fully explored which choices are best. Recall that the general algorithm for the method of centers consists of the λ iteration in (24) with x^{k+1} taken as the analytic center in (25). The EBE algorithm we tested against this involves an alternative method for computing x^{k+1} . Specifically, x^{k+1} is found by computing a descent direction in Ψ using the EBE method described above, followed by a line search on Ψ . This EBE computation is repeated once before the λ is updated. In contrast, the method of centers uses several Newton steps to find the analytic center before updating λ . Thus each iteration in λ of the method of centers is substantially more expensive than the EBE algorithm.

The two algorithms are compared in Figure 2. The data for this plot was obtained by generating 100 pseudo-random 5×5 complex matrices using a method that we have found gives reasonable test matrices similar to what occurs in frequency responses in typical engineering applications [16]. Our best hybrid algorithm, described later, of EBE and method of centers was then run until it converged to what we believe is a good approximation to the optimal upper bound and the matrix was divided by this value so that its upper bound was approximately 1. Then 5 and 20 λ iterations of each of the EBE and method of centers algorithms was run on the matrices, giving a total of 4 runs on each matrix. These 4 runs all used the same initial conditions of $D = I$, $G = 0$.

The plots in Figure 2 show the errors for the 20 worst problems out of 100 for each run. Note that after 5 iterations, the EBE algorithm has less than half the error of the method of centers, despite being much less expensive. Furthermore, the 80 matrices not shown all had errors below 3% after 5 EBE iterations and the worst matrix had an error below 10%. After 20 iterations, however, the error for the EBE method has not improved substantially and is much worse than the method of centers. This suggests that the EBE method be used to obtain initial values of the variables for a subsequent application of the method of centers. Better yet, a hybrid of the two algorithms that transitions smoothly from one to the other can be constructed.

6.2 A hybrid algorithm

Our experience and analysis strongly suggest that the EBE algorithm is superior initially but can fail to converge to the optimum, while the method of centers is guaranteed to converge but can be much slower. The best algorithm would be a combination of the two, starting with EBE and ending with the method of centers. This combination of

algorithms is easily constructed, yielding what could be viewed as an approximate method of centers.

The final hybrid algorithm consists of the same λ iteration as the method of centers in (24) and (25), restated here for convenience,

$$\lambda^{k+1} = (1 - \theta)\lambda_{\max}(A(x^k), B(x^k)) + \theta\lambda^k \quad (44)$$

$$x^{k+1} = x^*(\lambda^{k+1}). \quad (45)$$

The hybrid algorithm differs from the standard method of centers in the computation of x^* . Initially, x^* is computed by the EBE method described above. As iterations proceed, more steps are taken to obtain x^* before updating λ , and gradually Newton steps are added as well. Eventually, the Newton steps are run until the analytic center is found and thus the final algorithm executed is exactly the method of centers.

At any stage of the hybrid algorithm, the computation of x^* can be characterized by the number of EBE steps and the number of Newton steps taken. We have not resolved how to choose these numbers, and so far have used a fixed schedule plus some simple stopping tests. Our experience so far suggests that even a naive hybrid algorithm has substantially better performance than either the EBE or method of centers alone. Since the upper bound being computed will be compared with a lower bound, the likely strategy will be to achieve some specified bound, or run until convergence. Hopefully, this will usually need only the initial EBE iterations.

7 Summary

We have attempted to directly compare Osborne's method and the method of centers proposed by Boyd and El Ghaoui for computing the mixed μ upper bound. We have proposed a modified Osborne method, which we called the EBE method, for the computation of the mixed μ upper bound which is much faster than the standard method of centers far from the optimum, but which can fail to converge to the optimum. The EBE method should serve roughly the same purpose for mixed μ upper bound as Osborne did for the complex μ upper bound. The EBE method is slower but more accurate than standard Osborne on purely complex problems.

The best algorithm in the class we considered here appears to be a hybrid of the EBE and method of centers algorithms. Although this algorithm is reasonably efficient, there are still opportunities for improvements. For example, the strict splitting of the iteration between updating λ and x in (44) and (45) is somewhat artificial. Furthermore, the convergence rate of the method of centers can be quite slow and it might be advantageous to use a quadratically convergent algorithm near the optimum ([9], [12]).

Acknowledgements

We would like to thank Micheal Fan and Stephen Boyd for preprints of their papers and/or copies of software, as well as valuable discussions about the material in this paper. We would also like to thank Peter Young and Matt Newlin for lending their Matlab and μ expertise to this project. This work was supported by AFOSR, NSF, and Rockwell.

References

- [1] Doyle, J., 1982, "Analysis of Feedback Systems with Structured Uncertainty", *IEE Proceedings, Part D*, Vol. 129, pp. 242-250
- [2] Osborne, E. E., 1960 "On Pre-Conditioning of Matrices", *Journal of the Assoc. for Comp. Mach.*, Vol. 7, pp. 338-349
- [3] Safonov, M., 1982 "Stability Margins for Diagonally Perturbed Multivariable Feedback Systems", *IEE Proceedings*, Vol. 129, Part D, pp. 251-256
- [4] Young, P., Newlin, M., Doyle, J., 1991 "Structured Singular Value Analysis with Real and Complex Uncertainties", *Proceedings, 1991 IEEE CDC*, Brighton, England.
- [5] Fan, M. K. H., Tits, A. L., Doyle, J., 1991, "Robustness in the Presence of Mixed Parametric Uncertainty and Unmodeled Dynamics", *IEEE Transactions on Automatic Control* Vol. 36, No. 1, January 1991
- [6] Nesterov, Yu.E., and Nemirovsky, A.S., 1990, **Optimization over Positive Semidefinite Matrices: mathematical background and users manual**, USSR Academy of Science, Central Econ. and Math. Inst., Moscow.
- [7] Boyd, S., and El Ghaoui, L., 1991, "Method of Centers for Minimizing Generalized Eigenvalues", to appear in *Linear Algebra and its Applications*, special issue on Numerical Linear Algebra Methods in Control, Signals and Systems.
- [8] Jarre, F., 1991, "An Interior-Point Method for Minimizing the Maximum Eigenvalue of a Linear Combination of Matrices", Technical Report SOL 91-8, Stanford, California.
- [9] Fan, M. K. H., 1992 "A Quadratically Convergent Algorithm on Minimizing the Largest Eigenvalue of a Symmetric Matrix", submitted to *Linear Algebra and its Applications*, August 1992
- [10] Tierno, J. E., Young, P. M., 1992 "An Improved Lower Bound for μ ", *Proceedings, 1992 IEEE CDC*, Phoenix, Arizona
- [11] Doyle, J., Packard, A., and Zhou, K., 1991, "Review of LFTs, LMIs and μ ", *Proceedings, 1991 IEEE CDC*, Brighton, England.
- [12] Overton, M.L., 1990, "Large-Scale Optimization of Eigenvalues", New York University, Department of Computer Science Report No. 505.
- [13] Boyd, S., and Yang, Q., 1989, "Structured and Simultaneous Lyapunov Functions for System Stability Problems," *Int. Journal of Control*, Vol. 49, No. 6.
- [14] Nesterov, Yu.E., and Nemirovsky, A.S., 1989, **Self-Concordant Functions and Polynomial Time Methods in Convex Programming**, USSR Academy of Science, Central Econ. and Math. Inst., Moscow.
- [15] Balakrishnan, V., Feron, E., Boyd, S., El Ghaoui, L., 1992 "Computing Bounds for the Structured Singular Value via an Interior Point Algorithm", *Proceedings, 1992 ACC*, Chicago, Illinois.
- [16] Newlin, M.P. and Young, P.M., 1992 "Mixed μ Problems and Branch and Bound Techniques", *Proceedings of 31st Conference on Decision and Control*
- [17] Shor, N.Z., 1985 **Minimization Methods for Non-differentiable Functions**, Springer Series in Computational Mathematics, Springer-Verlag, Berlin.
- [18] Boyd, S. and Barratt, C., 1991, **Linear Controller Design**, Prentice Hall, New Jersey.
- [19] Huard, P., 1967, "Resolution of Mathematical Programming with Nonlinear Constraints by the Method of Centers", North Holland, Amsterdam, 1967
- [20] Fiacco, A.V., and McCormick, G.P., 1990, **Nonlinear Programming: Sequential Unconstrained Minimization Techniques**, SIAM Classics in Applied Mathematics, Pennsylvania.
- [21] Luenberger, D.G., 1989, **Linear and Nonlinear Programming**, Addison-Wesley, Massachusetts.