Computational Complexity of
Real Structured Singular Value in $\ell_p$ Setting *

Min Yue Fu
Department of Electrical and Computer Engineering,
The University of Newcastle, Newcastle, NSW 2308, Australia
Email: eemf@ee.newcastle.edu.au

Soura Dasgupta
Department of Electrical and Computer Engineering,
University of Iowa, Iowa City, IA 52242, USA
Email: dasgupta@hitchcock.eng.uiowa.edu

Abstract
This paper studies a generalized real structured singular value ($\mu$) problem where uncertain parameters are bounded by an $\ell_p$ norm. Two results are presented. The first one shows that this generalized $\mu$ problem is NP-hard for any given rational number $p \in [1, \infty]$. The NP-hardness holds as long as $k$, the size of the largest repeated block, exceeds 1. This result generalizes the known NP-hardness result for the conventional $\mu$ problem (with $p = \infty$). Our second result, which strengthens the first one, considers the approximability problem of the generalized $\mu$. We show that the problem of obtaining an estimate for the generalized $\mu$ with some guaranteed bound on the relative error remains to be NP-hard, regardless how large this bound is.

* This work was supported by Australian Research Council and NSF grants ECS-9211593 and ECS-9350346. The first author was visiting the University of Iowa when this work was initiated.
1 Introduction

The problem of real structured singular value (real \( \mu \)) arises in many robust control problems where the control system is subject to uncertain parameters. See, e.g., [3, 4, 5, 6, 8] for motivations and references.

Given a matrix \( M \in \mathbb{C}^{n \times n} \) and a set \( \Delta \) described by

\[
\Delta = \{ \Delta = \text{diag}\{\delta_1 I_{k_1}, \cdots, \delta_m I_{k_m}\} \mid \delta_i \in \mathbb{R} \}, \quad k_i > 0, \quad \sum_{i=1}^{m} k_i = n \tag{1.1}
\]

the real \( \mu \) problem is to compute the value of \( \mu_{\Delta}(M) \). This value is defined to be 0 if \( I_n - \Delta M \) is nonsingular for all \( \Delta \in \Delta \), or otherwise

\[
\mu_{\Delta}(M) = (\inf\{\alpha > 0 \mid \det(I_n - \Delta M) = 0, \ ||\delta||_\infty \leq \alpha\})^{-1} \tag{1.2}
\]

where \( ||\cdot||_\infty \) denotes the \( \ell_\infty \) norm and

\[
\delta = \text{diag}\{\delta_1, \cdots, \delta_m\}. \tag{1.3}
\]

Henceforth \( m \) will denote the size of the problem. It is known that the problem of determining if \( \mu_{\Delta}(M) < 1 \) is NP hard, see Poljak and Rohn [11], Braatz et. al. [1], Nemirovskii [10], and Coxson and DeMarco [2]. This negative result means that finding an algorithm for computing \( \mu_{\Delta}(M) \) is very unlikely if the algorithm is forced to require a number of computations that increases at most polynomially in \( m \), i.e. the problem can be solved in polynomial time.

In this paper, we study a generalized \( \mu \) problem by allowing the norm on \( \delta \) to be an \( \ell_p \) norm for any \( p \in [1, \infty] \). More precisely, given \( M, \Delta \), and \( p \in [1, \infty] \), we define \( \mu_{\Delta,p}(M) \) to be zero if \( I_n - \Delta M \) is nonsingular for all \( \Delta \in \Delta \), or otherwise

\[
\mu_{\Delta,p}(M) = (\inf\{\alpha > 0 \mid \det(I_n - \Delta M) = 0, \ ||\delta||_p \leq \alpha\})^{-1} \tag{1.4}
\]

For simplicity, we will denote \( \mu_{\Delta,p}(M) \) by \( \mu_p \). Our objective is to analyze the computational complexity of the \( \mu_p \) problem.

One might hope that the techniques used in [1, 2, 10, 11] for showing the NP-hardness of the \( \mu_\infty \) problem is generalizable to the \( \mu_p \) case. Unfortunately, this is not the case. Examining [1, 2, 10, 11], we find that all the NP-hardness proofs for the \( \mu_\infty \) problem rely (directly or indirectly) on a well-known fact that the following quadratic program is NP-hard for \( p = \infty \): Given a positive-definite and symmetric rational matrix \( Q \), determine if

\[
\max_{\|x\|_p \leq 1} x^T Q x < 1 \tag{1.5}
\]
See Vavasis [13, Exercise 4.3, p. 101]. However, when \( p \neq \infty \), it is not clear whether the problem remains NP-hard. Further, it is known that for \( p = 2 \) the quadratic program above is in the class of P, i.e. a polynomial time algorithm can be formulated to provide its solution; see Ye [14] for such an algorithm. Hence, a new technique is needed to investigate the computational complexity of the \( \mu_p \) problem. The difficulty with the \( \mu_p \) problem, \( p \neq \infty \), is that there is a single constraint on \( \delta \) rather than multiple constraints as in the \( \mu_\infty \) case.

Despite the differences between the \( \mu_\infty \) problem and the \( \mu_p \) problem, \( p \neq \infty \), as discussed above, we present two negative results for the generalized \( \mu \) problems. The first result shows that the generalized \( \mu \) problem remains to be NP-hard.

The second result, which strengthens the first one, deals with the approximability of the generalized \( \mu \). That is, We are interested in knowing how good an estimate can be obtained for \( \mu \) using a polynomial algorithm (polynomial in \( n \)). To be more precise, we have the following definition:

**Definition 1.1** An estimate \( \hat{\mu} \) is called an \( \delta \)-approximation of \( \mu \) for some \( \delta \geq 0 \) if \( |\hat{\mu} - \mu| \leq \delta \mu \).

To motivate this problem, we return to the standard \( \mu \) problem and note several known results. First, Coxson and DeMarco [2] shows that \( \epsilon \)-approximation of \( \mu \) with arbitrarily small \( \epsilon > 0 \) is an NP-hard problem, following a well-known result on the inapproximability of the so-called maxcut problem. A more negative result is offered by Toker [12] showing that computing an upper bound \( \bar{\mu} \) with the guarantee that \( \mu \leq \bar{\mu} \leq Cm^{1-\epsilon} \mu(M) \) for some (very large) constant \( C > 0 \) and (very small) \( \epsilon > 0 \) is an NP-hard problem. Recently, it is shown by Fu [5] that computing a \( \delta \)-approximation problem remains to be NP-hard even for any \( \delta = K(m) \), where \( K(m) \) is any prescribed positive function of \( m \), including exponential functions. The second result in this paper shows that this negative result is still valid for the generalized \( \mu \) problem.

2 Main Results

Define \( k = \min\{k_1, \cdots, k_m\} \), which is the largest size of the repeated blocks, and denote \( \mu_p \) by \( \mu_p(k) \), an explicit function of \( k \). Then, our main results are as follows.

**Theorem 2.1** Given any (fixed) \( k \geq 2 \) and \( p \in [1, \infty) \), the problem of determining if \( \mu_p(k) < 1 \) is NP-hard. Further, the problem of determining if \( \mu_\infty(1) < 1 \) is also NP-hard.

**Theorem 2.2** Let any (fixed) \( k \geq 2 \) and \( p \in [1, \infty] \) be given (including \( p = \infty \)). The problem of determining if \( \mu_p(k) = 0 \) is NP-hard. Subsequently, for any (arbitrarily
large) prescribed function $K(m) > 0$, where $n$ is the dimension of the $\mu_p(k)$ problem, the problem of finding an upper bound $\bar{\mu}$ guaranteeing $\mu_p(k) \leq \bar{\mu} \leq K(m)\mu_p(k)$ is NP-hard. Similarly, for any (arbitrarily small) prescribed function $K(m) > 0$, the problem of finding a lower bound $\underline{\mu}$ guaranteeing $K(m)\mu_p(k) \leq \underline{\mu} \leq \mu_p(k)$ is NP-hard.

Proof: The two theorems above are to be proved together. First, we note that the second part of Theorem 1 is known; see, e.g., [10]. Secondly, we show that the second part of Theorem 2 is implied by the first part of the theorem. Let us prove the result for $\mu$ by contradiction. Suppose there exists $K(m) > 0$ and a polynomial time algorithm which produces a $\mu$ with the guarantee that $\mu_p(k) \leq \bar{\mu} \leq K(m)\mu_p(k)$. Then, $\mu_p(k) = 0$ if and only if $\bar{\mu} = 0$. Hence, this algorithm will be able to solve the $\mu_p(k) = 0$ problem in polynomial time, which contradicts the first part of Theorem 2. A similar argument holds for the lower bound.

Now let us turn to the first part of Theorem 1 and the first part of Theorem 2. As in almost all NP-hardness analysis cases, our basic idea is to polynomially transform a known NP-hard problem to the problem of determining if $\mu_p(k) < 1$. Two problems are said to be related by a polynomial transformation if (i) a polynomial number of operations can be used to transform the first problem to the second; and (ii) the size of the first depends polynomially on the size of the second.

The 0-1 Knapsack Problem: Given an integer vector $c = (c_1, \cdots, c_m)^T$, determining if there exists a binary vector $x = (x_1, \cdots, x_m)^T \in \{-1, 1\}^m$ such that $c^Tx = 0$.

It is well-known that the Knapsack problem is NP-hard; see, e.g., Garey and Johnson [7].

Let $c$ be the integer vector in the Knapsack problem. Two cases are considered: (i) $1 \leq p < \infty$ and (ii) $p = \infty$.

For Case (i), we denote $p = \beta/\alpha$, where $\alpha$ and $\beta$ are coprime positive integers. Without loss of generality, we assume that the size $m$ of the Knapsack problem is such that the number $(2m)^{-1/p}$ is rational. If this is not the case, one can augment enough number of zero components to the vector $c$ such that the new size, say $\hat{m}$, is such that $(2\hat{m})^{-1/p}$ is rational. One particular choice of $\hat{m}$ is given by $\hat{m} = (2m)^{\beta/2}$. Note that such augmentation does not alter the solvability of the Knapsack problem and that the new size $\hat{m}$ is polynomial in $m$.

Define $n = 2m$ and

$$f(\delta) = j \sum_{i=1}^{m} c_i(\delta_i - \delta_{m+i}) + \sum_{i=1}^{m} \left[ (\delta_i + \delta_{m+i})^2 + (\delta_i \delta_{m+i} + d^2)^2 \right], \quad (2.6)$$

where $j = \sqrt{-1}$, $\delta = (\delta_1, \cdots, \delta_n)^T$ and $d = (2m)^{-1/p} > 0$.

For Case (ii), i.e., $p = \infty$, we simply take $\hat{m} = m$ and $d = 1$. This can be viewed as the limiting case of $p \to \infty$. The same definition for $f(\delta)$ will be used.
Obviously, necessary and sufficient conditions for \( f(\delta) = 0 \) for some \( \delta \in \mathbb{R}^n \) are that
\[
\delta_i = -\delta_{m+i}, \quad |\delta_i| = d, \quad i = 1, \ldots, m
\] (2.7)
and
\[
\sum_{i=1}^{m} c_i \delta_i = 0
\] (2.8)
Relating \( x_i \) in the Knapsack problem to \( \delta_i, \ i = 1, \ldots, m \), by
\[
x_i = \frac{\delta_i}{d}
\]
we know that the Knapsack problem has a solution \( x \in \{-1, \ 1\}^m \) if and only if \( f(\delta) = 0 \) for some \( ||\delta||_p \leq 1 \). Since the former problem is NP-hard, it follows that the problem of determining if \( f_1(\delta) \neq 0 \) for all \( ||\delta||_p \leq 1 \) is NP-hard.

Now we need to transform \( f(\delta) \) to some \( \det(I - \Delta M) \) with \( k = 2 \). Define for all \( i \in \{1, \ldots, m\} \):
\[
g_i = [-1 \ 1 - d^i + j c_1 \ \delta_i \ 1 - d^i]' \\
h_i = [1 \ \delta_{m+i} \ -1 - 2d^2 - j c_1 \ 1]'
\]
and
\[
D_i = \begin{pmatrix}
1 & -\delta_{m+i} & 0 & 0 \\
0 & 1 & 0 & 0 \\
\delta_i & 1 & 1 & -2d^2 - 1 \\
0 & 1 & 0 & 1
\end{pmatrix}.
\]
Further define
\[
A(\delta) = \begin{pmatrix}
0 & g_1' & g_2' & \cdots & g_m' \\
h_1 & D_1 & & & \\
h_2 & & D_2 & & \\
\vdots & & & \ddots & \\
h_m & & & & D_m
\end{pmatrix}.
\]
Observe, for each \( i \in \{1, \ldots, m\} \)
\[
\det(D_i) = 1.
\]
Further
\[
g_i' D_i^{-1} h_i = g_i' \begin{pmatrix}
1 & \delta_{m+i} & 0 & 0 \\
0 & 1 & 0 & 0 \\
-\delta_i & -\delta_i \delta_{m+i} - 2d^2 - 2 & 1 & 2d^2 + 1 \\
0 & -1 & 0 & 1
\end{pmatrix} h_i
\]
\[
= c_i (\delta_i - \delta_{m+i}) + (\delta_i + \delta_{m+i})^2 + (\delta_i \delta_{m+i} + d^2)^2.
\]
Thus,
\[
\begin{align*}
\det(A(\delta)) &= -\sum_{i=1}^{m} g_i^t D_i^{-1} h_i \\
&= f\left(\sum_{i=1}^{m} c_i (\delta_i - \delta_{m+i}) + \sum_{i=1}^{m} [(\delta_i + \delta_{m+i})^2 + (\delta_i \delta_{m+i} + d^2)^2]\right) \\
&= f(\delta).
\end{align*}
\]
Since each \(\delta_i\) appears in only two rows in \(A(\delta)\) and that too in an affine fashion, for each \(i \in \{1, \ldots, 2m\},\)
\[
\text{rank} \left[ \frac{\partial A(\delta)}{\partial \delta_i} \right] \leq 2.
\]
Thus we can write \(A(\delta)\) as follows:
\[
A(\delta) = A_0 - \sum_{i=1}^{2m} \delta_i B_i C_i^T
\]
(2.9)
where \(B_i\) and \(C_i\) are matrices with two columns only. Further,
\[
\det A(0) = f(0) = \sum_{i=1}^{m} d^4 = md^4 \neq 0
\]
It follows that
\[
f(\delta) = \det A(0) \det \left( I - \sum_{i=1}^{2m} \delta_i (A_0^{-1} B_i) C_i^T \right)
\]
Let
\[
B = \begin{bmatrix} A_0^{-1} B_1 & \cdots & A_0^{-1} B_{2m} \end{bmatrix}; \quad C = [C_1 \cdots C_{2m}];
\]
\[
M = C^T B; \quad \Delta = \text{diag}\{\delta_1 I_2, \cdots, \delta_{2m} I_2\}
\]
Then,
\[
f(\delta) = md^4 \det(I - B \Delta C^T) = md^4 \det(I - \Delta M)
\]
So, \(\det(I - \Delta M) \neq 0\) for all \(||\delta||_p \leq 1\) if and only if \(f(\delta) \neq 0\) for all \(||\delta||_p \leq 1\), which is NP-hard to determine. Hence, the problem of determining \(\mu_p(k) < 1\) is NP-hard for all \(k_i = 2\) and rational \(p \geq 1\). The fact that the same applies when the \(k_i \geq 2\), follows by taking the above constructed \(M\) and suitably augmenting it with zero rows and columns.

Note in the construction of \(\Delta\) and \(M\) above that \(\det(I - \Delta M) \neq 0\) for all \(||\delta||_p \leq 1\) if and only if \(\det(I - \Delta M) \neq 0\) for all \(\delta\). This is because \(f_1(\delta) \neq 0\) if \(||\delta||_p > 1\). It follows that the problem of determining if \(\mu_p(k) = 0\) is also NP-hard. \(\blacksquare\)
3 Some Remarks

The result in Theorem 1 leaves one question unanswered: Is the problem of determining whether $\mu_p(1) < 1$ NP-hard for $p \neq \infty$? In the following, we offer some remarks on this problem when $p = 2$ and every $k_i = 1$.

First, we note that $f(\delta) = \det(I - \Delta M)$ is a multilinear function in $\delta$ when $k = 1$. If either i) $f(\delta)$ is real and bilinear in $\delta_i$ or ii) $f(\delta)$ is complex and linear in $\delta_i$, then checking if $\mu_2(1) < 1$ is a special quadratic problem with $p = 2$. In these cases, the problem has polynomial complexity (provided that $M$ is rational), as pointed out in Section 1. Unfortunately, this observation does not generalize.

Secondly, we define the unit ball

$$B = \{ \delta : \|\delta\|_2 \leq 1 \}$$

and use the symbol $\partial(X)$ to denote the boundary of a set $X$. We ask the following question: Is $\partial F(B) \subset F(\partial B)$? The motivation of this question is simple because an affirmative answer to this question would make it sufficient to consider $\partial B$ alone to solve the $\mu_2(1)$ problem. Unfortunately, the following example shows that the answer is negative.

Example: Take

$$M = \begin{pmatrix} 0.3846 + 1.9231i & 0.0769 + 1.3846i \\ 0.3846 + 1.9231i & -1.9231 + 0.3846i \end{pmatrix}; \quad \Delta = \text{diag}\{\delta_1, \delta_2\}$$

(3.12)

Both $F(B)$ and $F(\partial B)$ are illustrated in Figure 1 by dots and asterisks, respectively. It is clear that $\partial F(B) \not\subset F(\partial B)$ in this case. Also observed in this example is that $0 \in F(B)$ but $0 \not\in F(\partial B)$.

4 Conclusion

Two results have been presented for the generalized real $\mu$ problem where the uncertain parameters are measured by an $\ell_p$ norm. Our first result shows that computing the generalized $\mu$ problem is NP-hard regardless what $\ell_p$ norm is used to measure the uncertainty block, as long as the minimum block size is 2. This result is strengthened by our second result which shows that the generalized $\mu$ problem is also very hard to approximate in general. Thus, the difficulty in computing or approximating $\mu$ is not unique to the $\ell_\infty$ measure of the uncertainty. The case when $k = 1$ remains unsolved