mathematics

MDPI

Article

Computing μ-Values for Real and Mixed μ Problems

Mutti-Ur Rehman [1], Muhammad Tayyab [2,*] and Muhammad Fazeel Anwar [1]

[1] Department of Mathematics, Sukkur IBA University, Sukkur 65200, Pakistan; mutti.rehman@iba-suk.edu.pk (M.R.); fazeel.anwar@iba-suk.edu.pk (M.F.A.)
[2] Faculty of Engineering Sciences, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, District Swabi, Khyber Pakhtunkhwa, Topi 23640, Pakistan
* Correspondence: muhammad.tayyab@giki.edu.pk

Received: 25 June 2019; Accepted: 24 August 2019; Published: 5 September 2019



Abstract: In various modern linear control systems, a common practice is to make use of control in the feedback loops which act as an important tool for linear feedback systems. Stability and instability analysis of a linear feedback system give the measure of perturbed system to be singular and non-singular. The main objective of this article is to discuss numerical computation of the μ-values bounds by using low ranked ordinary differential equations based technique. Numerical computations illustrate the behavior of the method and the spectrum of operators are then numerically analyzed.

Keywords: eigenvalues; singular values; low-rank ODEs

1. Introduction

In system theory, μ-value known as structured singular value (SSV) is a well-defined tool, first introduced by Doyle [1]. It is used to analyze the robustness of linear feedback systems when they are subject to uncertainties, particularly structured uncertainties. The μ-value tool has vast applications in control engineering, for instance, for the analysis of controller of an unstable system, the MIMO margins by using μ and for the space shuttle robustness analysis [2]. For more applications, we refer [3,4] and the references therein.

The numerical approximation of SSV appears as NP-hard problem [5,6]. The computation of the upper bounds of μ-values is a hard problem; in fact, the NP-hard is in nature when only real uncertainties appear [7].

The power method [3,8] is used to approximate the lower bounds of μ-values when only pure complex uncertainties are under consideration. This is a seemingly robust numerical method due to the fact that it depends solely on matrix-vector products. The power method is easy to apply only when mixed real perturbations and complex perturbations are taken into account [9–11]. This method resembles a mixture of power techniques used for the computation of eigenvalues and singular values of the constant matrices. This method also prove that μ-values act as an equilibrium point of the power technique. The power method fails to numerically compute the lower bounds of μ-values only when pure real uncertainties are under discussion. We refer to [12–20] and references therein for a comprehensive study of the numerical algorithms to compute numerically the bounds of μ-values from below.

An approximation of bounds of μ-values involve fundamental difficulties when one considers pure real uncertainties because the real μ-values might be discontinuous function of the given problem data [21,22]. Thus the available algorithms [13–15,19] to deal such problems have a computational cost which has exponential growth in nature. The idea of regularization to such a problem helps to reduce computational cost by making use of the mixed real/complex power iteration.

In this article, we use an iterative method [23] in order to numerically compute the μ-values bounds from below for a set of constant matrices generated randomly. The idea is based on a two level algorithm described in a great detail in [23]. The inner algorithm provides basis to develop a gradient system of ordinary differential equations (ODEs) and eventually solves this system by making use of extremizers of structured spectral-value sets. On the other hand, The outer algorithm helps to adjust the small positive parameter which acts as valid perturbation $\epsilon > 0$, when making use of fast Newton's iteration.

This paper is organized as follows: In Section 2 we provide the basic definitions of pseudo-spectra and structured singular values. In Section 3 we give the reformulation of the definition of structured singular values. Section 4 of this article contains the computation of pseudo-spectra, where in Section 5 we explain how the computation of structured singular values can be addressed by making use of an inner-outer algorithm. In the outer algorithm, we determine a small perturbation level ϵ while in inner algorithm we determines a local extremizer of the structured spectral value set. Section 6 refers to numerical experiments to compare the lower bounds of SSV obtained with algorithm [23] to those obtained with MATLAB function mussv. Section 7 summarizes our conclusions.

2. Preliminaries

Definition 1. *The uncertainties in terms of mixed real and complex scalar blocks are denoted by $\mathbb{B}_{rc}$ and defined by*

$$\mathbb{B}_{rc} := \{Diag(\delta_1 I_1, \delta_2 I_2, ..., \delta_s I_s; \delta_{s+1} I_{s+1}, \delta_{s+2} I_{s+2}, ..., \delta_{s+n} I_{s+n}) : \delta_i \in \mathbb{R}\ \forall i = 1:s, \delta_{s+j} \in \mathbb{C}\ \forall j = 1:n\}. \quad (1)$$

Definition 2. *The set of block diagonal matrices in terms of pure real scalar blocks is denoted by $\mathbb{B}_r$ and defined by*

$$\mathbb{B}_r := \{Diag(\delta_1 I_1, \delta_2 I_2, ..., \delta_s I_s) : \delta_i \in \mathbb{R}\ \forall i = 1:s\}. \quad (2)$$

Definition 3. *The set of uncertainties in terms of pure full complex blocks are denoted by $\mathbb{B}_f$, defined as*

$$\mathbb{B}_f := \{Diag(\Delta_1, \Delta_2, ..., \Delta_F) : \Delta_j \in \mathbb{C}^{m_j, m_j}\ \forall j = 1:F\}. \quad (3)$$

Definition 4. *The uncertainties in terms of blocks consist of r repeated real scalar blocks and F square full complex blocks are denoted by $\mathbb{B}_R$ and defined as follows*

$$\mathbb{B}_R := \{Diag(\delta_1 I_{r_1}, \delta_2 I_{r_2}, ..., \delta_s I_{r_S}, \Delta_1, \Delta_2, ..., \Delta_F) : \delta_i \in \mathbb{R}, \Delta_j \in \mathbb{C}^{m_j, m_j}\}. \quad (4)$$

Definition 5. *The block diagonal operators entirely consist of S repeated complex scalar blocks and F square full complex blocks are denoted by $\widehat{\mathbb{B}}_R$ and defined as*

$$\widehat{\mathbb{B}}_R := \{Diag(\delta_1 I_1, \delta_2 I_2, ..., \delta_S I_S, \Delta_1, \Delta_2, ..., \Delta_F) : \delta_i \in \mathbb{C}, \Delta_j \in \mathbb{C}^{m_j, m_j}\}. \quad (5)$$

Definition 6. *The spectrum of a square complex valued matrix $M \in \mathbb{C}^{n,n}$ is represented by $\Lambda(M)$ and mathematically defined as*

$$\Lambda(M) = \{\lambda \in \mathbb{C} : det\,(\lambda I - M) = 0\}. \quad (6)$$

Definition 7. *The pseudo-spectrum of a n-dimensional complex valued operator is represented by $\Lambda_\epsilon(M)$ and defined as:*

$$\Lambda_\epsilon(M) = \left\{\lambda \in \mathbb{C} : \| (\lambda I - M)^{-1} \| \geq \frac{1}{\epsilon}\right\}. \quad (7)$$

Definition 8. *The μ-value of $M \in \mathbb{C}^{n\times n}$ underlying block diagonal operator $\mathbb{B}$ is mathematically denoted by $\mu_{\mathbb{B}_R}(M)$ and defined as follows*

$$\mu_{\mathbb{B}_R}(M) := \begin{cases} 0, & \text{if} \quad |I - M\Delta| \neq 0\,, \forall\ \Delta \in \mathbb{B}_R\,, \\ \dfrac{1}{\min\{||\Delta||_2 : |I - M\Delta| = 0,\ \Delta \in \mathbb{B}_R\}}\,, & \text{otherwise}\,. \end{cases} \tag{8}$$

3. Reformulation of μ-Values

In this section we give the reformulation of μ-values based on structured spectral value sets [24]. The reformulation of μ-values allows us to construct an optimization problem in order to maximize the spectrum of the perturbed system.

Definition 9. *The ϵ-spectral valued set of an n-dimensional complex operator underlying a small positive real parameter $\epsilon > 0$ is mathematically denoted by $\Delta_\epsilon^{\mathbb{B}_R}(M)$ and defined as:*

$$\Lambda_\epsilon^{\mathbb{B}_R}(M) := \{\lambda \in \Lambda(\epsilon M\Delta), \Delta \in \mathbb{B}_R,\ ||\Delta||_2 \leq 1\}, \tag{9}$$

where the quantity $\Lambda(\epsilon M\Delta)$ represent the spectrum of an operator $(\epsilon M\Delta)$ while $||\Delta||_2$ represents the greatest singular value of an operator $\Delta \in \mathbb{B}_R$.

Definition 10. *For a given n-dimensional complex operator M and an underlying perturbation in the form of $\mathbb{B}$, the spectral value set is represented by $\Sigma_\epsilon^{\mathbb{B}_R}(M)$ and is defined as the following*

$$\Sigma_\epsilon^{\mathbb{B}_R}(M) := \{\eta = 1 - \lambda :\ \lambda \in \Lambda_\epsilon^{\mathbb{B}_R}(M)\}. \tag{10}$$

Definitions 9 and 10 make it possible to formulate again μ-value as a maximization problem.

Definition 11. *For $M \in \mathbb{C}^{n\times n}$ and a set of pure complex uncertainties $\widehat{\mathbb{B}}_R$, μ-value is defined as the following*

$$\mu_{\widehat{\mathbb{B}}_R}(M) = \frac{1}{arg\{\max|\lambda| = 1,\ \lambda \in \Lambda_\epsilon^{\mathbb{B}_R}(M)\}}, \tag{11}$$

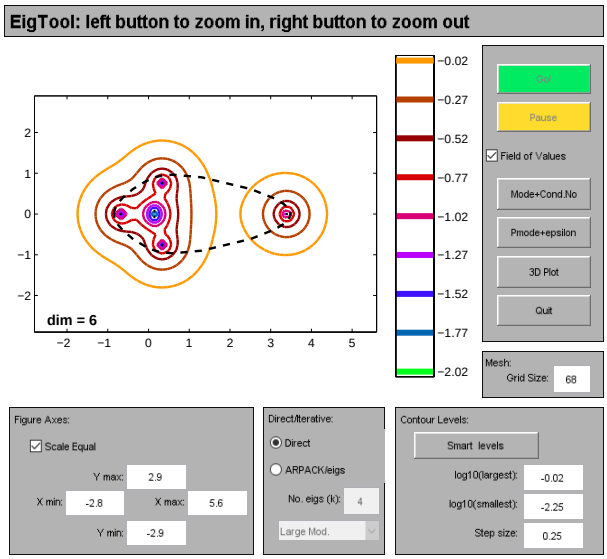
with $\widehat{\mathbb{B}}_R = \{Diag(\delta_1 I_{c_1}, \delta_2 I_{c_2}, ..., \delta_S I_{c_S}, \Delta_1, \Delta_2, ..., \Delta_F) :\ \delta_i \in \mathbb{C}, \Delta_j \in \mathbb{C}^{m_j, m_j}\}$.

4. Pseudospectra of Square Matrices

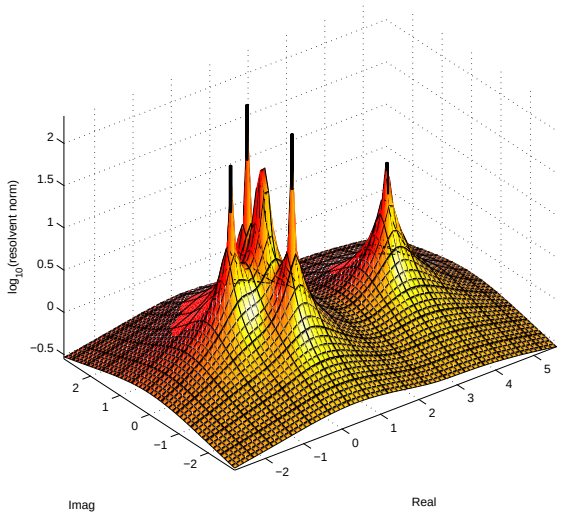
In this section, the pseudospectra for matrices are under consideration. For this purpose we make use of the software package EigTool [25]. EigTool is routinely used to plot the unstructured pseudospectra of the matrices under consideration. Figures 1 and 2 show the graphical representation of the pseudospectrum of 6-dimensional and 4-dimensional real valued matrices.

We consider a six dimensional real valued matrix M as

$$M = \begin{bmatrix} 0.8147 & 0.2785 & 0.9572 & 0.7922 & 0.6787 & 0.7060 \\ 0.9058 & 0.5469 & 0.4854 & 0.9595 & 0.7577 & 0.0318 \\ 0.1270 & 0.9575 & 0.8003 & 0.6557 & 0.7431 & 0.2769 \\ 0.9134 & 0.9649 & 0.1419 & 0.0357 & 0.3922 & 0.0462 \\ 0.6324 & 0.1576 & 0.4218 & 0.8491 & 0.6555 & 0.0971 \\ 0.0975 & 0.9706 & 0.9157 & 0.9340 & 0.1712 & 0.8235 \end{bmatrix}.$$



(**a**) Pseudospectrum of 6-dimensional real matrix



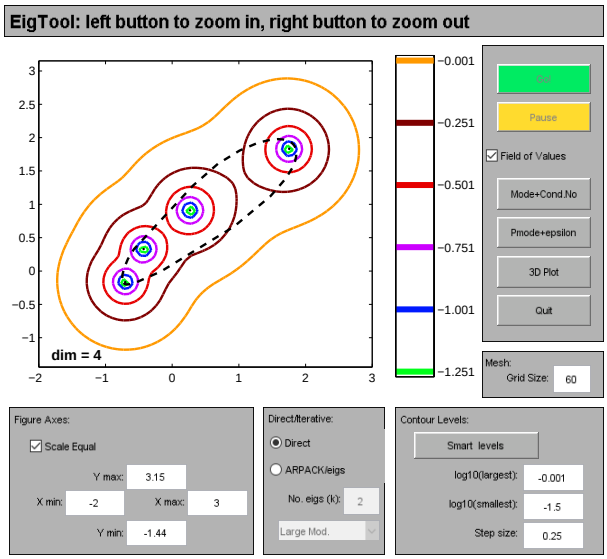
(**b**) 3-dimensional representation of the pseudospectrum

Figure 1. MATLAB interface for computing pseudospectrum.

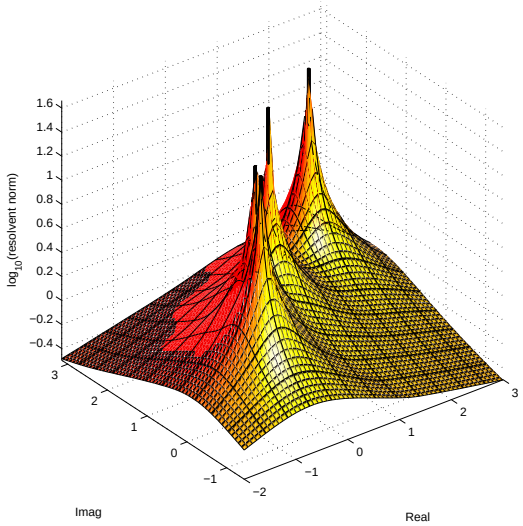
Next, we consider a four dimensional complex valued matrix M as

$$M = \begin{bmatrix} 0.2374 + 0.9338i & 0.1048 + 0.4876i & 0.6035 + 0.0372i & 0.7581 + 0.6099i \\ 0.5309 + 0.1875i & 0.1123 + 0.7690i & 0.9644 + 0.6733i & 0.4326 + 0.0594i \\ 0.0915 + 0.2662i & 0.7844 + 0.3960i & 0.4325 + 0.4296i & 0.6555 + 0.3158i \\ 0.4053 + 0.7978i & 0.2916 + 0.2729i & 0.6948 + 0.4517i & 0.1098 + 0.7727i \end{bmatrix}.$$

and compute pseudo-spectra with EigTool.



(**a**) Pseudospectrum of 6-dimensional real matrix



(**b**) 3-dimensional representation of the pseudospectrum

Figure 2. MATLAB interface for computing pseudospectrum.

5. Proposed Methodology

In this section, the main objective is to discuss the optimization problem as mentioned in Definition 11, we make use of a numerical approach which is based on low-ranked ordinary differential equations. The numerical method is composed of a two-level algorithm, the inner algorithm and an outer algorithm. In this Section, we first discuss the inner-algorithm which shows how to give a theoretical construction to the system of ordinary differential equations. Secondly, we discuss how to deal with such systems mathematically. Next, we discuss the outer algorithm, which helps to modify the small positive real parameter $\epsilon > 0$. The outer algorithm is based on Newton's iteration for approximating the roots of non-linear equations. The outer-algorithm computes an exact derivative of an extremizer say, $\Delta(\epsilon)$ for $\Delta \in \mathbb{B}$ and $\epsilon > 0$. Complete details of the numerical method under consideration are given in [23,24]. Finally, we discuss the computation of an extremizer. For this purpose, we approximate the derivative of an eigenvalue matrix $\Lambda(p)$ of a smooth matrix family say $A(p)$ for some fixed parameter p.

5.1. Inner-Algorithm

5.1.1. The Basic Theory

Consider some smooth matrix family $A(p) \in \mathbb{C}^{n,n}$ for a small parameter p. Let $\Lambda(p) \in \mathbb{C}^{n,n}$ and $X(p) \in \mathbb{C}^{n,n}$ be the eigenvalue and eigenvector matrices corresponding to $A(p)$. In matrix form

$$A(p)X(p) = X(p)\Lambda(p)\,. \tag{12}$$

In Equation (12), computation of eigenvector matrix $X(p)$ is not unique unless all of the eigenvectors remain fixed. For distinct eigenvalues all of the eigenvectors are computed up to a constant multiplier.

The theory of computing $X(p)$ is extended if eigenvalue derivatives are repeated while the second order derivatives of eigenvalues remain distinct. This theory is further generalized to complex-valued and Hermitian matrix.

Here and hereafter the symbol $(\prime)$ denotes the derivative while we write $A = A(p)$, that is, we omit the dependency on p. By differentiating Equation (12) we get,

$$A^{'}X + AX^{'} = X^{'}\Lambda + X\Lambda^{'},$$

or

$$A^{'}X - X\Lambda^{'} = -AX^{'} + X^{'}\Lambda. \tag{13}$$

Our goal is to determine $\Lambda^{'}$, the derivative of eigenvalues matrix. For this purpose, we multiply Equation (13) with X^{-1} throughout to have,

$$X^{-1}A^{'}X - X^{-1}X\Lambda^{'} = -X^{-1}AX^{'} + X^{-1}X^{'}\Lambda.$$

Since $X^{-1}X = I$, this implies:

$$X^{-1}A^{'}X - \Lambda^{'} = -X^{-1}AX^{'} + X^{-1}X^{'}\Lambda, \tag{14}$$

let $L = X^{-1}X^{'}$, we find

$$X^{'} = XL\,,$$

thus Equation (14) gives:

$$X^{-1}A^{'}X - \Lambda^{'} = -X^{-1}AXL + L\Lambda,$$

or

$$X^{-1}A^{'}X - \Lambda^{'} = -X^{-1}X\Lambda L + L\Lambda.$$

Therefore,

$$X^{-1}A^{'}X - \Lambda^{'} = -\Lambda L + L\Lambda. \tag{15}$$

Finally, this gives the required result for the computation of the derivative of spectrum of matrix valued function $A(p)$ as follows

$$X^{-1}A^{'}X - \Lambda^{'} = 0, \qquad \text{or} \qquad \Lambda^{'} = X^{-1}A^{'}X\,.$$

5.1.2. Approximation of Extremizers

A matrix valued function $\Delta \in \mathbb{B}$ having the largest eigenvalue bounded above by 1 and the matrix valued function $(I - \epsilon M\Delta)$ having a smallest eigenvalue which minimizes the modulus of structured spectral value set, $\sum_{\epsilon}^{\mathbb{B}}(M)$ is known as an extremizer. The following theorem computes an extremizer for a chosen smallest complex number belonging to the set $\sum_{\epsilon}^{\mathbb{B}}(M)$.

Theorem 1 ([23]). *For a perturbation $\Delta \in \mathbb{B}$ having the block diagonal structure*

$$\Delta = \{Diag(\delta_1 I_1, \ldots \delta_{s'} I_{s'}, \delta_{s'+1} I_{s'+1}, \ldots \delta_S I_S; \Delta_1, \ldots, \Delta_F)\},$$

with $\|\Delta\|_2 = 1$, acts as a local extremizer of structured spectral value set. For a simple smallest eigenvalue $\lambda = |\lambda| e^{\iota\theta}, \theta \in \mathbb{R}$ of matrix valued function $(I - \epsilon M\Delta)$ having right and left eigenvetors x and y scaled as $S = e^{\iota\theta} y^* x$ and let $z = M^* y$. The non-degeneracy conditions

$$\begin{gathered} z_k^* x_k \neq 0, \ \forall = 1 : S^{'} \\ Re(z_k^* x_k) \neq 0, \ \forall = 1 : S^{'} + 1 : S \\ and \quad ||z_{s+h}||.||x_{s+h}|| \neq 0, \ \forall h = 1 : F, \end{gathered}$$

hold. Then magnitude of each complex scalar $\delta_i \, \forall i = 1 : s$ appears to be exactly equal to 1 while each full block possesses a unit 2-norm.

Proof. For proof we refer to [23]. □

5.1.3. Gradient System of ODEs

The gradient system of ODEs for an admissible perturbation $\Delta \in \mathbb{B}$ to approximate a local extremizer of smallest eigenvalue $\lambda = |\lambda| \epsilon^{i\theta}$, is obtained as,

$$\begin{gathered} \dot{\delta}_i = \nu_i (x_i^* z_i - Re(x_i^* z_i \bar{\delta}_i)\delta_i); \quad i = 1 : s^{'} \\ \dot{\delta}_l = sign(Re(z_l^* x_l)) \Psi_{(-1,1)}(\delta_l); \quad l = s^{'} + 1 : s \\ \dot{\Delta}_j = \nu_j (z_{s+j} x_{s+j}^* - Re\langle \Delta_j; z_{s+j} x_{s+j}^* \rangle); \quad j = 1 : F, \end{gathered}$$

where $\delta_i \in \mathbb{C}$, $\forall i = 1 : s^{'}$, $\delta_l \in \mathbb{R}$ for $l = s^{'} + 1$ and $\Psi_{(-1,1)}$ is a characteristic function. For a further discussion on the construction of gradient system of ODEs in above equations, we refer to [23].

5.2. Outer-Algorithm

In an outer-algorithm the main aim is to vary $\epsilon > 0$, the perturbation level, by means of a fast Newton's iteration. In turn, $1/\epsilon$ will provide the approximation of lower bound of μ-value.

We make use of fast Newton's iteration in order to solve a problem

$$|\lambda(\epsilon)| = 1, \qquad \epsilon > 0. \tag{16}$$

In order to solve Equation (16), we need to compute

$$\frac{d}{d\epsilon} \left(|\lambda(\epsilon)| \right),$$

the derivative.

The following Theorem 2 helps us to compute $\frac{d}{d\epsilon}(|\lambda(\epsilon)|)$, when $|\lambda(\epsilon)|$ is simple and $\Delta(0)$, $\lambda(0)$ are assumed to remains smooth in the neighboring region of perturbation level $\epsilon > 0$.

Theorem 2 ([23]). *Consider matrix valued function $\Delta \in \mathbb{B}$. Let x and y as a function of perturbation level $\epsilon > 0$ acts as right and left eigenvectors of matrix valued function $(\epsilon M\Delta)$. Consider the scaling of these vector accordingly to Theorem 1. Let $z = M^* y$ and assume that non-degenracy conclusions as discussed in Theorem 1 yield then*

$$\frac{d}{d\epsilon}(|\lambda(\epsilon)|) = \frac{1}{|y(\epsilon^*)x(\epsilon)|} \left(\sum_{i=1}^{s} |z_i(\epsilon)^* x_i(\epsilon)| + \sum_{j=1}^{F} ||z_{s+j}(\epsilon)||.||y_{s+j}(\epsilon)|| \right) > 0. \tag{17}$$

Proof. For proof we refer to [23]. □

Remark 1. *Choice of suitable initial value matrix and initial perturbation level: For a suitable choice of the initial value matrix Δ_0 and an initial perturbation level ϵ_0, we refer to [23].*

6. Numerical Computation

In this section, we present numerical results for both lower and upper bounds of μ-values for randomly generated matrices in MATLAB (R2018a, MathWorks, Natick, MA, USA). The numerical computation of the lower bounds of structured singular values with the help of Nalgo, show the tighter results for the μ-values lower bounds computed by means of well-known MATLAB routine mussv. In each experiment the admissible uncertainties ∇ and E are computed with the help of mussv and Nalgo [23] respectively.

Example 1. *First we consider a six dimensional real valued matrix M*

$$M = \begin{bmatrix} 0.8147 & 0.2785 & 0.9572 & 0.7922 & 0.6787 & 0.7060 \\ 0.9058 & 0.5469 & 0.4854 & 0.9595 & 0.7577 & 0.0318 \\ 0.1270 & 0.9575 & 0.8003 & 0.6557 & 0.7431 & 0.2769 \\ 0.9134 & 0.9649 & 0.1419 & 0.0357 & 0.3922 & 0.0462 \\ 0.6324 & 0.1576 & 0.4218 & 0.8491 & 0.6555 & 0.0971 \\ 0.0975 & 0.9706 & 0.9157 & 0.9340 & 0.1712 & 0.8235 \end{bmatrix}.$$

We choose block uncertainties $\mathcal{B} = Diag\{\Delta_1 : \Delta_1 \in \mathbb{C}^{6,6}\}$. The admissible uncertainty ∇ is given as below

$$\nabla = \begin{bmatrix} 0.0615 & 0.0421 & 0.0607 & 0.0427 & 0.0479 & 0.0453 \\ 0.0692 & 0.0473 & 0.0683 & 0.0480 & 0.0539 & 0.0509 \\ 0.0545 & 0.0373 & 0.0538 & 0.0378 & 0.0425 & 0.0401 \\ 0.0628 & 0.0430 & 0.0619 & 0.0436 & 0.0489 & 0.0462 \\ 0.0609 & 0.0417 & 0.0601 & 0.0423 & 0.0475 & 0.0448 \\ 0.0727 & 0.0497 & 0.0717 & 0.0504 & 0.0566 & 0.0535 \end{bmatrix},$$

with $\|\nabla\|_2 = 0.3157$. The main drawback with mussv is that it does not compute the unit 2-norm of each block of admissible perturbation Δ and by this way it loose the optimality condition. The optimality condition requires that: not only the admissible perturbation Δ should possess unit 2-norm but each of its block also possesses a unit 2-norm.

The approximated lower bound of μ-value is 3.1677. Moreover, by using the algorithm [23], the admissible perturbation E is obtained as

$$E = \begin{bmatrix} 0.1813 & 0.1691 & 0.1611 & 0.1079 & 0.1286 & 0.1718 \\ 0.2059 & 0.1921 & 0.1830 & 0.1225 & 0.1460 & 0.1952 \\ 0.2140 & 0.1997 & 0.1902 & 0.1273 & 0.1518 & 0.2028 \\ 0.2397 & 0.2237 & 0.2131 & 0.1427 & 0.1700 & 0.2273 \\ 0.1844 & 0.1721 & 0.1639 & 0.1097 & 0.1308 & 0.1748 \\ 0.1179 & 0.1100 & 0.1048 & 0.0702 & 0.0836 & 0.1118 \end{bmatrix},$$

with $\|E\|_2 = 1$. The algorithm [23] computes the unit 2-norm of admissible perturbation E. It does not violate the optimality condition and hence computes the unit 2-norm against each block of E. The lower bound approximation is 3.5693 which is a much better approximation than the one computed numerically.

If instead of full complex blocks, one chose the uncertainties to be repeated as complex blocks, then the mussv routine will not compute the desired admissible perturbations such that each repeated complex block possesses a unit 2-norm. On the hand, Nalgo computes each repeated complex block

in such a way that not only the admissible perturbation possesses a unit 2-norm but each individual block also has a largest singular value to be exactly equal to 1.

In Table 1, we present the comparison for the lower and upper bounds of μ-value approximated numerically with the help of mussv and an efficient numerical algorithm [23] for matrix M in Example 1. The first column presents the size of an operator M, where the second, third and fourth columns represent the approximations towards both upper and lower bounds which are computed numerically and the lower bounds μ_l^{Nalgo} computed with the help of [23] respectively.

Table 1. Computation of bounds of μ-values.

n	μ_u^{mussv}	μ_l^{mussv}	μ_l^{Nalgo}
6	2.6579	2.6504	2.8158
6	2.9758	2.9630	2.4962
6	2.9778	2.9654	2.9835
6	2.7771	2.7696	2.7611
6	2.8483	2.8464	2.5008

Example 2. *We consider a four dimensional complex valued matrix M*

$$M = \begin{bmatrix} 0.2374 + 0.9338i & 0.1048 + 0.4876i & 0.6035 + 0.0372i & 0.7581 + 0.6099i \\ 0.5309 + 0.1875i & 0.1123 + 0.7690i & 0.9644 + 0.6733i & 0.4326 + 0.0594i \\ 0.0915 + 0.2662i & 0.7844 + 0.3960i & 0.4325 + 0.4296i & 0.6555 + 0.3158i \\ 0.4053 + 0.7978i & 0.2916 + 0.2729i & 0.6948 + 0.4517i & 0.1098 + 0.7727i \end{bmatrix}.$$

We choose the perturbation set $\mathcal{B} = Diag\{\Delta_1 : \Delta_1 \in \mathbb{C}^{4,4}\}$. The perturbation set contains only pure complex full blocks. The admissible perturbation ∇ is obtained as

$$\nabla = \begin{bmatrix} 0.0520 - 0.0934i & 0.0694 - 0.1567i & 0.1073 - 0.0797i & 0.0794 - 0.1041i \\ 0.0371 - 0.0539i & 0.0513 - 0.0914i & 0.0699 - 0.0424i & 0.0543 - 0.0588i \\ 0.0144 - 0.0590i & 0.0144 - 0.0963i & 0.0468 - 0.0598i & 0.0278 - 0.0690i \\ 0.0352 - 0.0752i & 0.0452 - 0.1252i & 0.0788 - 0.0676i & 0.0558 - \ 0.0850i \end{bmatrix},$$

with $\|\nabla\|_2 = 0.4176$.

The numerical result for the lower bound of μ-value is 2.3948*. Moreover by using the algorithm [23], the admissible perturbation E is obtained as*

$$E = \begin{bmatrix} 0.1332 - 0.2201i & 0.1503 - 0.1958i & 0.1388 - 0.1551i & 0.0912 - 0.2321i \\ 0.1095 - 0.1970i & 0.1253 - 0.1763i & 0.1165 - 0.1403i & 0.0723 - 0.2062i \\ 0.2603 - 0.1604i & 0.2652 - 0.1255i & 0.2309 - 0.0887i & 0.2224 - 0.1960i \\ 0.1974 - 0.1872i & 0.2081 - 0.1577i & 0.1849 - 0.1194i & 0.1579 - \ 0.2113i \end{bmatrix},$$

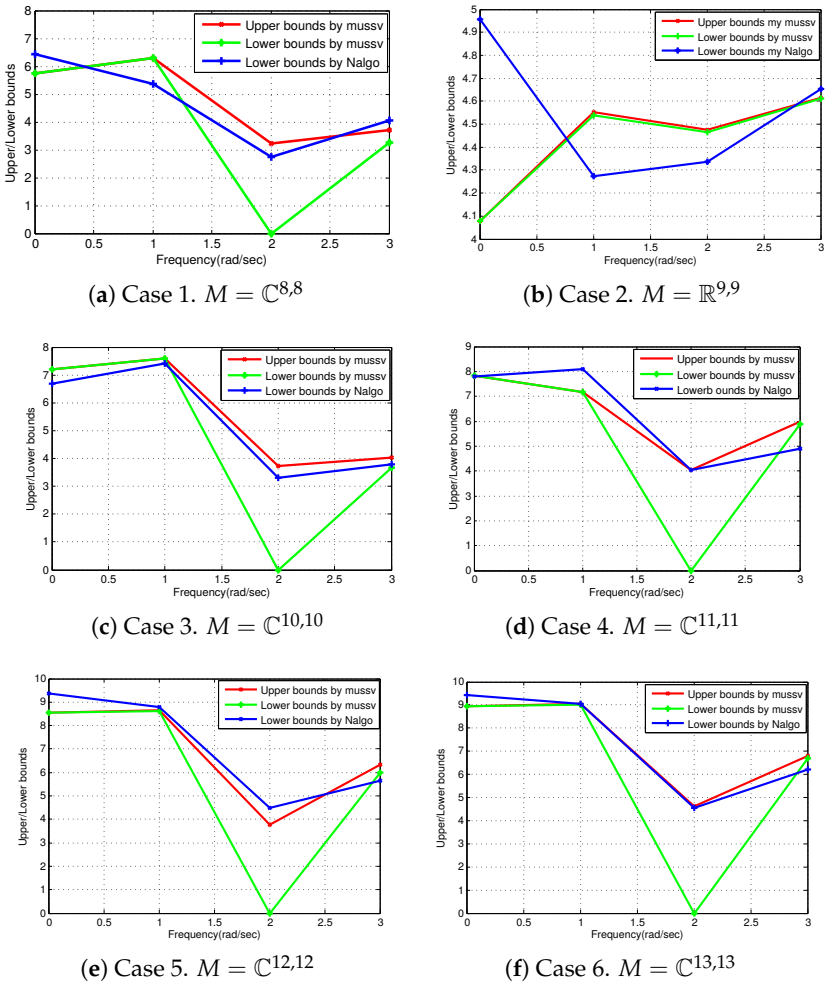
with $\|E\|_2 = 1$. The lower bound approximation is 2.6983*. If instead of full complex blocks, one choose the uncertainties to be repeated complex blocks then the mussv routine will not compute the desired admissible perturbations such that each repeated complex block possesses a unit 2-norm. On the other hand, Nalgo computes each repeated complex block in a such a way that not only the admissible perturbation possesses a unit 2-norm but each individual block also have a largest singular value to be exactly equal to 1.*

Table 2 represents the comparison of the μ-values approximated numerically with the mussv routine and an iterative method [23] for matrix M taken in Example 2. In the very first column, we present the size of the matrix M. In the second column, the upper bounds of μ-values are presented while lower bounds are presented in the third and fourth columns, respectively.

Table 2. Computation of bounds of μ-values.

n	μ_u^{mussv}	μ_l^{mussv}	μ_l^{Nalgo}
6	2.7392	2.7379	2.7885
6	1.5106	0.0000	1.5027
6	3.1980	3.1945	3.0202
6	0.7588	0.0000	1.3416
6	3.0524	3.0515	2.9231

In Figure 3, we show numerical approximation of bounds of μ-values for matrix valued function $M(\omega_i)$ at various frequencies ω_i, $\forall i = 1:5$. In Cases 1–6, the matrix valued functions $M(\omega_i)$ are obtained in MATLAB by using the "rand" command. The geometrical interpretation of μ-values bounds show the comparison of μ-values bounds computed numerically from below with the help of Nalgo and the one computed with the help of mussv. The frequency response ω_i is the quantitative measure of output of $M - \Delta$ system.



(**a**) Case 1. $M = \mathbb{C}^{8,8}$ (**b**) Case 2. $M = \mathbb{R}^{9,9}$

(**c**) Case 3. $M = \mathbb{C}^{10,10}$ (**d**) Case 4. $M = \mathbb{C}^{11,11}$

(**e**) Case 5. $M = \mathbb{C}^{12,12}$ (**f**) Case 6. $M = \mathbb{C}^{13,13}$

Figure 3. Numerical approximation of bounds of μ-values at different frequency response.

7. Conclusions

In this article, we presented numerical computation of lower and upper bounds of μ-values a well-known mathematical quantity in control. The μ-values discuss robustness and performance of linear feedback systems in control and their bounds examine stability analysis of such systems. The numerical computations show that the attained results for μ-values bounds from below are more

appropriate than those obtained by MATLAB function mussv. The EigTool discussed the geometrical interpretation of the pseudo-spectrum of the family of matrices.

Author Contributions: M.R. and M.T. both contributed in conceptualization, methodology, writing manuscript and validation of results, while M.F.A. contributed in writing-review and editing.

Funding: This research received no external funding.

Acknowledgments: The authors acknowledge the anonymous reviewers for their potential inputs to increase the quality of the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Doyle, J. Analysis of feedback systems with structured uncertainties. *IEE Proc. D Control Theory Appl.* **1982**, *129*, 242–250. [CrossRef]
2. Balas, G.J.; Doyle, J.C.; Glover, K.; Packard, A.; Smith, R. *µ-Analysis and Synthesis Toolbox User'S Guide*; The Mathworks Inc.: Natick, MA, USA, 1995.
3. Packard, A.; Doyle, J. The complex structured singular value. *Automatica* **1993**, *29*, 71–109. [CrossRef]
4. Ferreres, G. *A Practical Approach to Robustness Analysis with Aeronautical Applications*; Springer Science & Business Media: Berlin, Germany, 1999.
5. Braatz, R.P.; Young, P.M.; Doyle, J.C.; Morari, M. Computational complexity of mu-calculation. *IEEE Trans. Autom. Control* **1994**, *39*, 1000–1002. [CrossRef]
6. Demmel, J. The componentwise distance to the nearest singular matrix. *SIAM J. Matrix Anal. Appl.* **1992**, *13*, 10–19. [CrossRef]
7. Fu, M. The real structured singular value is hardly approximable. *IEEE Trans. Autom. Control* **1997**, *42*, 1286–1288.
8. Packard, A.; Fan, M.K.H.; Doyle, J. A power method for the structured singular value. In Proceedings of the 27th IEEE Conference on Decision and Control, Austin, TX, USA, 7–9 December 1988; pp. 2132–2137.
9. Young, P.M.; Doyle, J.C. Computation of mu with real and complex uncertainties. In Proceedings of the 29th IEEE Conference on Decision and Control, Honolulu, HI, USA, 5–7 December 1990; pp. 1230–1235.
10. Young, P.M.; Doyle, J.C. A lower bound for the mixed mu-problem. *IEEE Trans. Autom. Control* **1997**, *42*, 123–128. [CrossRef]
11. Young, P.M.; Newlin, M.P.; Doyle, J.C. Practical computation of the mixed μ problem. In Proceedings of the 1992 IEEE American Control Conference, Chicago, IL, USA, 24–26 June 1992; pp. 2190–2194.
12. Bates, D.G.; Mannchen, T. Improved computation of mixed μ bounds for flight control law robustness analysis. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2004**, *218*, 609–619. [CrossRef]
13. Dailey, R.L. A new algorithm for the real structured singular value. In Proceedings of the IEEE 1990 American Control Conference, San Diego, CA, USA, 23–25 May 1990; pp. 3036–3040.
14. De Gaston, R.R.E.; Safonov, M.G. Exact calculation of the multiloop stability margin. *IEEE Trans. Autom. Control* **1988**, *33*, 156–171. [CrossRef]
15. Elgersma, M.; Freudenberg, J.; Morton, B. Polynomial methods for the structured singular value with real parameters. *Int. J. Robust Nonlinear Control* **1996**, *6*, 147–170. [CrossRef]
16. Hayes, M.J.; Bates, D.G. Postlethwaite, I. New tools for computing tight bounds on the real structured singular value. *J. Guid. Control Dyn.* **2001**, *24*, 1204–1213. [CrossRef]
17. Newlin, M.P.; Glavaski, S.T. Advances in the computation of the mu-lower bound. In Proceedings of the 1995 IEEE American Control Conference-ACC'95, Seattle, WA, USA, 21–23 June 1995; pp. 442–446.
18. Newlin, M.P.;Young, P.M. Mixed μ problems and branch and bound techniques. *Int. J. Robust Nonlinear Control* **1997**, *7*, 145–164. [CrossRef]
19. Sideris, A.; Pena, R.S.S. Mixed μ Fast computation of the multivariable stability margin for real interrelated uncertain parameters. *IEEE Trans. Autom. Control* **1989**, *34*, 1272–1276. [CrossRef]
20. Sideris, A.; Pena, R.S.S. Mixed μ Robustness margin calculation with dynamic and real parametric uncertainty. In Proceedings of the 1988 IEEE American Control Conference, Atlanta, GA, USA, 15–17 June 1988; pp. 1201–1206.

21. Barmish, B.R.; Khargonekar, P.P.; Shi, Z.C.; Tempo, R. *Robustness Margin Need Not Be a Continuous Function of the Problem Data*; Elsevier: Amsterdam, The Netherlands, 1990.
22. Packard, A.; Pandey, P. Robustness Continuity properties of the real/complex structured singular value. *IEEE Trans. Autom. Control* **1993**, *38*, 415–428. [CrossRef]
23. Guglielmi, N.; Rehman, M.-U.; Kressner, D. A novel iterative method to approximate structured singular values. *SIAM J. Matrix Anal. Appl.* **2017**, *38*, 361–386. [CrossRef]
24. Rehman, M.-U.; Anwar, M.F. Robustness μ-Values for Matrices Corresponding to Symmetries in Control Systems. *Int. J. Anal. Appl.* **2018**, *16*, 673–688.
25. Wright, T.G.; Trefethen, L.N. Robustness Eigtool. 2002. Available online: http://www.comlab.ox.acuk/pseudospectra/eigtool (accessed on 15 March 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).