Open Journal of Applied Sciences, 2017, 7, 348-364
http://www.scirp.org/journal/ojapps
ISSN Online: 2165-3925
ISSN Print: 2165-3917

Computing Structured Singular Values for Delay and Polynomial Eigenvalue Problems

Mutti-Ur Rehman[1], Danish Majeed[2], Naila Nasreen[3], Shabana Tabassum[1]

[1]Department of Mathematics and Statistics, The University of Lahore, Gujrat Campus, Gujrat, Pakistan
[2]Department of Applied Physics, Federal Urdu University of Arts, Science and Technology, Islamabad, Pakistan
[3]Department of Mathematics, COMSATS Institute of Information Technology Islamabad Campus, Islamabad, Pakistan
Email: mutti.abbasi@gmail.com, shabana.tabassum@lbs.uol.edu.pk, mthleaf@gmail.com, naila.12364@gmail.com

How to cite this paper: Rehman, M.-U., Majeed, D., Nasreen, N. and Tabassum, S. (2017) Computing Structured Singular Values for Delay and Polynomial Eigenvalue Problems. *Open Journal of Applied Sciences*, **7**, 348-364.
https://doi.org/10.4236/ojapps.2017.77028

Received: April 27, 2017
Accepted: July 18, 2017
Published: July 21, 2017

Abstract

In this article the computation of the Structured Singular Values (SSV) for the delay eigenvalue problems and polynomial eigenvalue problems is presented and investigated. The comparison of bounds of SSV with the well-known MATLAB routine mussv is investigated.

Keywords

μ-Values, Block Diagonal Uncertainties, Spectral Radius, Low-Rank Approximation, Delay Eigenvalue Problems, Polynomial Eigenvalue Problems

1. Introduction

The μ-values [1] is an important mathematical tool in control theory; it allows discussing the problem arising in the stability analysis and synthesis of control systems to quantify the stability of a closed-loop linear time-invariant systems subject to structured perturbations. The structures addressed by the SSV is very general and allows covering all types of parametric uncertainties that can be incorporated into the control system by using real and complex Linear Fractional Transformations LFT's. For more detail please see [1]-[7] and the references therein for the applications of SSV.

The versatility of the SSV comes at the expense of being notoriously hard, in fact Non-deterministic Polynomial time that is NP hard [8] to compute. The numerical algorithms which are being used in practice provide both upper and lower bounds of SSV. An upper bound of the SSV provides sufficient conditions to guarantee robust stability analysis of feedback systems, while a lower bound provides sufficient conditions for instability analysis of the feedback systems.

The widely used function mussv available in the Matlab Control Toolbox computes an upper bound of the SSV using diagonal balancing and Linear Matrix Inequality techniques [9] [10]. The lower bound is computed by using the generalization of power method developed in [11] [12].

In this paper the comparison of numerical results to approximate the lower bounds of the SSV associated with mixed real and pure complex uncertainties is presented.

Overview of the article. Section 2 provides the basic framework. In particular, it explains how the computation of the SSV can be addressed by an inner-outer algorithm, where the outer algorithm determines the perturbation level ϵ and the inner algorithm determines a (local) extremizer of the structured spectral value set. In Section 3 it is explained that how the inner algorithm works for the case of pure complex structured perturbations. An important characterization of extremizers shows that one can restrict himself to a manifold of structured perturbations with normalized and low-rank blocks. A gradient system for finding extremizers on this manifold is established and analyzed. The outer algorithm is addressed in Section 4, where a fast Newton iteration for determining the correct perturbation level ϵ is developed. Finally, Section 5 presents a range of numerical experiments to compare the quality of the lower bounds to those obtained with mussv.

2. Framework

Consider a matrix $A \in \mathbb{K}^{n,n}$ where $\mathbb{K} = \mathbb{C}(\mathbb{R})$ and an underlying perturbation set with prescribed block diagonal structure,

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_{r_1}, \cdots, s_N I_{r_S}, \Delta_1, \cdots, \Delta_F\right), s_i \in \mathbb{C}(\mathbb{R}), \Delta_j \in \mathbb{C}^{m_j,m_j}\left(\mathbb{R}^{m_j,m_j}\right) \right\}, \quad (1)$$

where I_{r_i} denotes the r_i, r_i identity matrix. Each of the scalars s_i and the m_j, m_j matrices Δ_j may be constrained to stay real in the definition of $\mathbb{BLK}$. The integer N denotes the number of repeated scalar blocks (that is, scalar multiples of the identity) and F denotes the number of full blocks. This implies $\sum_{i=1}^{N} r_i + \sum_{j=1}^{F} m_j = n$. In order to distinguish complex and real scalar blocks, assume that the first $N' \le N$ blocks are complex while the (possibly) remaining $N - N'$ blocks are real. Similarly assume that the first $F' \le F$ full blocks are complex and the remaining $F - F'$ blocks are real. The literature (see, e.g., [1]) usually does not consider real full blocks, that is, $F' = F$. In fact, in control theory, full blocks arise from uncertainties associated to the frequency response of a system, which is complex-valued.

For simplicity, assume that all full blocks are square, although this is not necessary and our method extends to the non-square case in a straightforward way. Similarly, the chosen ordering of blocks should not be viewed as a limiting assumption; it merely simplifies notation.

The following definition is given in [1], where $\|\cdot\|_2$ denotes the matrix 2-norm and I the $n \times n$ identity matrix.

Definition 2.1. [13]. Let $A \in \mathbb{K}^{n \times n}$ and consider a set $\mathbb{BLK}$. Then the SSV (or μ-value) $\mu_{\mathbb{BLK}}(A)$ is defined as

$$\mu_{\mathbb{BLK}}(A) := \frac{1}{\min\left\{\|\Delta\|_2 : \Delta \in \mathbb{BLK}, \det(I - A\Delta) = 0\right\}}. \quad (2)$$

In Definition 2.1, $\det(\cdot)$ denotes the determinant of a matrix and in the following we make use of the convention that the minimum over an empty set is $+\infty$. In particular, $\mu_{\mathbb{BLK}}(A) = 0$ if $\det(I - A\Delta) \neq 0$ for all $\Delta \in \mathbb{BLK}$.

Note that $\mu_{\Delta}(A)$ is a positively homogeneous function, *i.e.*,

$$\mu_{\mathbb{BLK}}(\alpha A) = \alpha\mu_{\mathbb{BLK}}(A) \quad \text{for any } \alpha \geq 0.$$

For $\mathbb{BLK} = \mathbb{K}^{n\times n}$, it follows directly from Definition 2.1 that $\mu_{\mathbb{BLK}} = \|A\|_2$ .For general $\mathbb{BLK}$, the SSV can only become smaller and thus gives us the upper bound $\mu_{\mathbb{BLK}}(A) \leq \|A\|_2$. This can be refined further by exploiting the properties of $\mu_{\mathbb{BLK}}$, see [14]. These relations between $\mu_{\mathbb{BLK}}$ and $\|A\|_2$, the largest singular value of *A*, justifies the name structured singular value for $\mu_{\mathbb{BLK}}(A)$.

The important special case when $\mathbb{BLK}$ only allows for complex perturbations, that is, $N = N'$ and $F = F'$, deserves particular attention. In this case one can write $\mathbb{BLK}^*$ instead of $\mathbb{BLK}$. Note that $\Delta \in \mathbb{BLK}^*$ implies $e^{i\varphi}\Delta \in \mathbb{BLK}^*$ for any $\varphi \in \mathbb{R}$. In turn, there is $\Delta \in \mathbb{BLK}^*$ such that $\rho(A\Delta) = 1$ if and only if there is $\Delta' \in \mathbb{BLK}^*$, with the same norm, such that $A\Delta'$ has the eigenvalue 1, which implies $\det(I - A\Delta') = 0$. This gives the following alternative expression:

$$\mathbb{BLK}^* = \frac{1}{\min\left\{\|A\|_2 : \Delta \in \mathbb{BLK}^*, \rho(A\Delta) = 1\right\}}, \quad (3)$$

where $\rho(\cdot)$ denotes the spectral radius of a matrix. For any nonzero eigenvalue λ of *A*, the matrix $\Delta = \lambda^{-1}I$ satisfies the constraints of the minimization problem shown in Equation (3). This establishes the lower bound $\rho(A) \leq \mu_{\mathbb{BLK}^*}(A)$ for the case of purely complex perturbations. Note that $\mu_{\mathbb{BLK}^*} = \rho(A)$ for $\mathbb{BLK}^* = \{sI : s \in \mathbb{C}\}$. Hence, both the spectral radius and the matrix 2-norm are included as special cases of the SSV.

2.1. A Reformulation Based on Structured Spectral Value Sets [13]

The structured spectral value set of $A \in \mathbb{C}^{n\times n}$ with respect to a perturbation level ϵ is defined as

$$\Lambda_{\epsilon}^{\mathbb{BLK}}(A) = \left\{\lambda \in \Lambda(\epsilon A\Delta) : \Delta \in \mathbb{BLK}, \|\Delta\|_2 \leq 1\right\}, \quad (4)$$

where $\Lambda(\cdot)$ denotes the spectrum of a matrix. Note that for purely complex $\mathbb{BLK}^*$, the set $\Lambda_{\epsilon}^{\mathbb{BLK}}(A)$ is simply a disk centered at 0. The set

$$\Sigma_{\varepsilon}^{\mathbb{BLK}}(A) = \left\{\xi = 1 - \lambda : \lambda \in \Lambda_{\epsilon}^{\mathbb{BLK}}(A)\right\} \quad (5)$$

allows us to express the SSV defined in Equation (2) as

$$\mu_{\mathbb{BLK}}(A) = \frac{1}{\arg\min\left(0 \in \Sigma_{\epsilon}^{\mathbb{BLK}}(A)\right)} \quad (6)$$

that is, as a structured distance to singularity problem. This gives us that $0 \notin \Sigma_{\epsilon}^{\mathbb{BLK}}(A)$ if and only if $\mu_{\mathbb{BLK}}(A) < 1/\epsilon$.

For a purely complex perturbation set $\mathbb{BLK}^*$, one can use Equation (3) to alternatively express the SSV as

$$\mu_{\mathbb{BLK}^*}(A) = \frac{1}{\arg\min\left\{\max|\lambda| = 1\right\}} \tag{7}$$

where $\lambda \in \Lambda_{\epsilon}^{\mathbb{BLK}^*}(A)$ and one can have that $\Lambda_{\epsilon}^{\mathbb{BLK}^*}(A) \subset D$, where D denotes the open complex unit disk, if and only if $\mu_{\mathbb{BLK}^*}(A) < 1/\epsilon$.

2.2. Problem under Consideration [13]

Let us consider the minimization problem

$$\xi(\epsilon) = \arg\min|\xi| \tag{8}$$

where $\xi \in \Sigma_{\epsilon}^{\mathbb{BLK}}(A)$ for some fixed $\epsilon > 0$. By the discussion above, the SSV, $\mu_{\mathbb{BLK}}(A)$ is the reciprocal of the smallest value of ϵ for which $\xi(\epsilon) = 0$. This suggests a two-level algorithm: In the inner algorithm, we attempt to solve Equation (8). In the outer algorithm, we vary ϵ by an iterative procedure which exploits the knowledge of the exact derivative of an extremizer say $\Delta(\epsilon)$ with respect to ϵ. We address Equation (8) by solving a system of Ordinary Differential Equations (ODE's). In general, this only yields a local minimum of Equation (8) which, in turn, gives an upper bound for ϵ and hence a lower bound for $\mu_{\mathbb{BLK}}(A)$. Due to the lack of global optimality criteria for Equation (8), the only way to increase the robustness of the method is to compute several local optima.

The case of a purely complex perturbation set $\mathbb{BLK}^*$ can be addressed analogously by letting the inner algorithm to determine a local optima for

$$\lambda(\epsilon) = \arg\max|\lambda| \tag{9}$$

where $\lambda \in \Lambda_{\epsilon}^{\mathbb{BLK}^*}(A)$ which then yields a lower bound for $\mu_{\mathbb{BLK}^*}(A)$.

3. Pure Complex Perturbations [13]

In this section, we consider the solution of the inner problem discussed in Equation (9) in the estimation of $\mu_{\mathbb{BLK}^*}(A)$ for $A \in \mathbb{C}^{n,n}$ and a purely complex perturbation set

$$\mathbb{BLK}^* = \left\{ diag\left(s_1 I_{r_1}, \cdots, s_N I_{r_N}; \Delta_1, \cdots, \Delta_F\right) : s_i \in \mathbb{C}, \Delta_j \in \mathbb{C}^{m_j, m_j} \right\} \tag{10}$$

3.1. Extremizers

Now, make use of the following standard eigenvalue perturbation result, see, e.g., [15]. Here and in the following, denote $\dot{} = d/dt$.

Lemma 3.1. Consider a smooth matrix family $C : \mathbb{R} \to \mathbb{C}^{n,n}$ and let $\lambda(t)$ be an eigenvalue of $C(t)$ converging to a simple eigenvalue λ_0 of $C_0 = C(0)$ as $t \to 0$. Then $\lambda(t)$ is analytic near $t = 0$ with

$$\dot{\lambda}(0)=\frac{w_0^*C_1v_0}{w_0^*v_0},$$

where $C_1=\dot{C}(0)$ and v_0,w_0 are right and left eigenvectors of C_0 associated to λ_0, that is, $(C_0-\lambda_0 I)v_0=0$ and $w_0^*(C_0-\lambda_0 I)=0$.

Our goal is to solve the maximization problem discussed in Equation (9), which requires finding a perturbation Δ_{local} such that $\rho(\epsilon A\Delta_{local})$ is maximal among all $\Delta\in\mathbb{BLK}^*$ with $\|\Delta\|_2\le 1$. In the following we call λ a largest eigenvalue if $|\lambda|$ equals the spectral radius.

Definition 3.2. [13]. A matrix $\Delta\in\mathbb{BLK}^*$ such that $\|\Delta\|_2\le 1$ and $\epsilon A\Delta$ has a largest eigenvalue that locally maximizes the modulus of $\Lambda_\epsilon^{\mathbb{BLK}^*}(A)$ is called a local extremizer.

The following result provides an important characterization of local extremizers.

Theorem 3.3. [13]. Let

$$\Delta_{local}=diag\left(s_1 I_{r_1},\cdots,s_N I_{r_S};\Delta_1,\cdots,\Delta_F\right),\|\Delta_{local}\|_2=1,$$

be a local extremizer of $\Lambda_\epsilon^{\mathbb{BLK}^*}(A)$. Assume that $\epsilon A\Delta_{local}$ has a simple largest eigenvalue $\lambda=|\lambda|\mathrm{e}^{i\theta}$, with the right and left eigenvectors v and w scaled such that $S=\mathrm{e}^{i\theta}w^*v>0$. Partitioning

$$v=\left(v_1^{\mathrm{T}}\cdots v_N^{\mathrm{T}},v_{N+1}^{\mathrm{T}}\cdots v_{N+F}^{\mathrm{T}}\right)^{\mathrm{T}},\quad u=A^*w=\left(u_1^{\mathrm{T}}\cdots u_N^{\mathrm{T}},u_{N+1}^{\mathrm{T}}\cdots u_{N+F}^{\mathrm{T}}\right)^{\mathrm{T}},\tag{11}$$

such that the size of the components v_k,u_k equals the size of the kth block in Δ_{local}, additionally assume that

$$u_k^*v_k\ne 0,\quad \forall\, k=1,\cdots,N\tag{12}$$

$$\|u_{N+h}\|_2\cdot\|v_{N+h}\|_2\ne 0,\quad \forall\, h=1,\cdots,F.\tag{13}$$

then

$$|s_k|=1,\forall k=1,\cdots,N\quad\text{and}\quad \|\Delta_h\|_2=1,\forall h=1,\cdots,F,$$

that is, all blocks of Δ_{local} have unit 2-norm.

The following theorem allows us to replace full blocks in a local extremizer by rank-1 matrices.

Theorem 3.4. [13]. Let $\Delta_{local}=diag\left(s_1 I_{r_1},\cdots,s_N I_{r_N},\Delta_1,\cdots,\Delta_F\right)$ be a local extremizer and let λ,v,u be defined and partitioned as in Theorem 3.3. Assuming that Equation (13) holds, every block Δ_h has a singular value 1 with associated singular vectors $q_h=\gamma_h w_{N+h}/\|w_{N+h}\|_2$ and $r_h=\gamma_h u_{N+h}/\|u_{N+h}\|_2$ for some $|\gamma_h|=1$. Moreover, the matrix

$$\hat{\Delta}=diag\left(s_1 I_{r_1},\cdots,s_N I_{r_N},a_1b_1^*,\cdots,a_Fb_F^*\right)$$

is also a local extremizer, *i.e.*, $\rho(\epsilon A\Delta_{local})=\rho\left(\epsilon A\hat{\Delta}\right)$.

Remark 3.1. [13]. Theorem 3.3 allows us to restrict the perturbations in the structured spectral value set shown in Equation (4) to those with rank-1 blocks, which was also shown in [1]. Since the Frobenius and the matrix 2-norms of a rank-1 matrix are equal, one can equivalently search for extremizers within the submanifold

$$\mathbb{BLK}_1^* = diag\left(s_1 I_{r_1} \cdots s_N I_{r_N}, \Delta_1 \cdots \Delta_F\right): s_i \in \mathbb{C}, \left|s_i\right| = 1, \Delta_j \in \mathbb{C}^{m_j, m_j}, \left\|\Delta_j\right\|_F = 1. \quad (14)$$

3.2. A system of ODEs to Compute Extremal Points of $\Lambda_\epsilon^{\mathbb{BLK}^*}(A)$ [13]

In order to compute a local maximizer for $|\lambda|$, with $\lambda \in \Lambda_\epsilon^{\mathbb{BLK}^*}(A)$, First construct a matrix valued function $\Delta(t)$ such that a largest eigenvalue $\lambda(t)$ of $\epsilon A \Delta(t)$ has maximal local increase. Then derive a system of ODEs satisfied by this choice of $\Delta(t)$.

Orthogonal projection onto $\mathbb{BLK}^*$. In the following, we make use of the Frobenius inner product $\langle A, B\rangle = \text{trace}\left(A^* B\right)$ for two m,n matrices A, B. Let

$$C_{\mathbb{BLK}^*} = P_{\mathbb{BLK}^*}(C). \quad (15)$$

denote the orthogonal projection, with respect to the Frobenius inner product, of a matrix $C \in \mathbb{C}^{n \times n}$ onto $\mathbb{BLK}^*$. To derive a compact formula for this projection, use the pattern matrix

$$I_{\mathbb{BLK}^*} = diag\left(I_{r_1}, \cdots, I_{r_N}, I_{m_1}, \cdots, I_{m_F}\right), \quad (16)$$

where I_d denotes the d, d -matrix of all ones.

Lemma 3.5. [13]. For $C \in \mathbb{C}^{n,n}$, let

$$C \odot I_{\mathbb{BLK}^*} = diag\left(C_1, \cdots, C_N, C_{N+1}, \cdots, C_{N+F}\right)$$

denote the block diagonal matrix obtained by entrywise multiplication of C with the matrix $I_{\mathbb{BLK}^*}$ defined in Equation (20). Then the orthogonal projection of C onto $\mathbb{BLK}^*$ is given by

$$C_{\mathbb{BLK}^*} = P_{\mathbb{BLK}^*}(C) = diag\left(\gamma_1 I_{r_1}, \cdots, \gamma_N I_{r_N}, \Gamma_1, \cdots, \Gamma_F\right) \quad (17)$$

where $\gamma_i = tr(C_i)/r_i$, $i = 1, \cdots, N$, and $\Gamma_1 = C_{S+1}, \cdots, \Gamma_F = C_{N+F}$.

The local optimization problem [13]**.** Let us recall the setting from Section (3.1): assume that $\lambda = |\lambda| \mathrm{e}^{i\theta}$ is a simple eigenvalue with eigenvectors v, w normalized such that

$$\|w\| = \|v\| = 1, \quad w^* v = \left|w^* v\right| \mathrm{e}^{-i\theta}. \quad (18)$$

As a consequence of Lemma 3.1, see also Equation (15), to have

$$\frac{\mathrm{d}}{\mathrm{d}t}|\lambda|^2 = 2|\lambda| Re\left(\frac{u^* \dot{\Delta} v}{\mathrm{e}^{i\theta} w^* v}\right) = \frac{2|\lambda|}{\left|w^* v\right|} Re\left(u^* \dot{\Delta} v\right), \quad (19)$$

where $u = A^* w$ and the dependence on t is intentionally omitted.

Letting $\Delta \in \mathbb{BLK}_1^*$, with $\mathbb{BLK}_1^*$ as in Equation (18), now we aim at determining a direction $\dot{\Delta} = U$ that locally maximizes the increase of the modulus of λ. This amounts to determining

$$U = diag\left(\omega_1 I_{r_1}, \cdots, \omega_s I_{r_N}, \Omega_1, \cdots, \Omega_F\right) \quad (20)$$

as a solution of the optimization problem

$$U_* = \arg\max\left\{Re\left(u^* U x\right)\right\}$$
$$\text{subject to } Re\left(\overline{\delta}_i \omega_i\right) = 0,\ i = 1:N,$$
$$\text{and } Re\left\langle \Delta_j, \Omega_j \right\rangle = 0,\ j = 1:F. \tag{21}$$

The target function in Equation (20) follows from Equation (19), while the constraints in Equation (21) ensure that U is in the tangent space of $\mathbb{BLK}_1^*$ at Δ. In particular Equation (20) implies that the the norms of the blocks of Δ are conserved. Note that Equation (20) only becomes well-posed after imposing an additional normalization on the norm of *U*. The scaling chosen in the following lemma aims at $U \in \mathbb{BLK}_1^*$.

Lemma 3.6. [13]. With the notation introduced above and v,u partitioned as in Equation (11), a solution of the optimization problem discussed in Equation (20) is given by

$$U_* = diag\left(\omega_1 I_{r_1}, \cdots, \omega_N I_{r_N}, \Omega_1, \cdots, \Omega_F\right), \tag{22}$$

with

$$\omega_i = \nu_i\left(v_i^* u_i - Re\left(v_i^* u_i \overline{s}_i\right) s_i\right), \quad i = 1, \cdots, N \tag{23}$$

$$\Omega_j = \zeta_j\left(u_{N+j} v_{N+j}^* - Re\left\langle \Delta_j, u_{N+j} v_{N+j}^* \right\rangle \Delta_j\right), \quad j = 1, \cdots, F. \tag{24}$$

Here, $\nu_i > 0$ is the reciprocal of the absolute value of the right-hand side in Equation (23), if this is different from zero, and $\nu_i = 1$ otherwise. Similarly, $\zeta_j > 0$ is the reciprocal of the Frobenius norm of the matrix on the right hand side in Equation (24), if this is different from zero, and $\zeta_j = 1$ otherwise. If all right-hand sides are different from zero then $U_* \in \mathbb{BLK}_1^*$.

Corollary 3.7. [13]. The result of Lemma 3.6 can be expressed as

$$U_* = D_1 P_{\mathbb{BLK}^*}\left(uv^*\right) - D_2 \Delta \tag{25}$$

where $P_{\mathbb{BLK}^*}(\cdot)$ is the orthogonal projection and $D_1, D_2 \in \mathbb{BLK}^*$ are diagonal matrices with D_1 positive.

Proof. The statement is an immediate consequence of Lemma 3.5.

The system of ODEs. Lemma 3.6 and Corollary 3.7 suggest to consider the following differential equation on the manifold $\mathbb{BLK}_1^*$:

$$\dot{\Delta} = D_1 P_{\mathbb{BLK}^*}\left(uv^*\right) - D_2 \Delta, \tag{26}$$

where $v(t)$ is an eigenvector, of unit norm, associated to a simple eigenvalue $\lambda(t)$ of $\epsilon A \Delta(t)$ for some fixed $\epsilon > 0$. Note that $u(t), D_1(t), D_2(t)$ depend on $\Delta(t)$ as well. The differential Equation (30) is a gradient system because, by definition, the right-hand side is the projected gradient of $U \mapsto Re\left(u^* U v\right)$.

The following result follows directly from Lemmas 3.1 and 3.6.

Theorem 3.8. [13]. Let $\Delta(t) \in \mathbb{BLK}_1^*$ satisfy the differential Equation (26). If $\lambda(t)$ is a simple eigenvalue of $\epsilon A \Delta(t)$, then $\left|\lambda(t)\right|$ increases monotonically.

The following lemma establishes a useful property for the analysis of stationary points of Equation (26).

Lemma 3.9. [13]. Let $\Delta(t)\in\mathbb{BLK}_1^*$ satisfy the differential Equation (26). If $\lambda(t)$ is a nonzero simple eigenvalue of $\epsilon A\Delta(t)$, with right and left eigenvectors $v(t)$ and $w(t)$ scaled, then

$$P_{\mathbb{BLK}^*}\left(u(t)v(t)^*\right)\neq 0, \tag{27}$$

where $u(t)=A^*w(t)$.

Remark 3.3. The choice of ν_i, η_j originating from Lemma 3.6., to achieve unit norm of all blocks in Equation (25), is completely arbitrary. Other choices would be also acceptable and investigating an optimal one in terms of speed of convergence to stationary points would be an interesting issue.

The following result characterizes stationary points of Equation (26).

Theorem 3.10. [13]. Assume that $\Delta(t)$ is a solution of Equation (26) and $\lambda(t)$ is a largest simple nonzero eigenvalue of $\epsilon A\Delta(t)$ with right/left eigenvectors $v(t)$, $w(t)$. Moreover, suppose that Assumptions (12) and (13) hold for $v(t)$ and $u(t)=A^*w(t)$. Then

$$\frac{\mathrm{d}}{\mathrm{d}t}\left|\lambda(t)\right|^2=0\Leftrightarrow\dot{\Delta}(t)=0\Leftrightarrow\Delta(t)=DP_{\mathbb{BLK}^*}\left(u(t)v(t)^*\right), \tag{28}$$

for a specific real diagonal matrix $D\in\mathbb{BLK}^*$. Moreover if $\lambda(t)$ has (locally) maximal modulus over the set $\Lambda_\epsilon^{\mathbb{BLK}^*}(A)$ then D is positive.

3.3. Projection of Full Blocks on Rank-1 Manifolds [13]

In order to exploit the rank-1 property of extremizers established in Theorem 3.4, one can proceed in complete analogy to [16] in order to obtain for each full block an ODE on the manifold M of (complex) rank-1 matrices. The derivation of this system of ODEs is straightforward; the interested reader can see [17] for full details.

The monotonicity and the characterization of stationary points follows analogously to those obtained for Equation (26); and also refer to [16] for the proofs. As a consequence one can use the ODE in Equation (28) instead of Equation (26) and gain in terms of computational complexity.

3.4. Choice of Initial Value Matrix and ϵ_0 [13]

In our two-level algorithm for determining ϵ use the perturbation Δ obtained for the previous value ϵ as the initial value matrix for the system of ODEs. However, it remains to discuss a suitable choice of the initial values $\Delta(0)=\Delta_0$ and ϵ_0 in the very beginning of the algorithm.

For the moment, let us assume that A is invertible and write

$$I-\epsilon_0 A\Delta_0=A\left(A^{-1}-\epsilon_0\Delta_0\right),$$

which aim to have as close as possible to singularity. To determine Δ_0, one can perform an asymptotic analysis around $\epsilon_0\approx 0$. For this purpose, let us consider the matrix valued function

$$G(\tau)=A^{-1}-\tau\Delta_0,$$

and let denote $\chi(\tau)$ denote an eigenvalue of $G(\tau)$ with smallest modulus. Letting v and w denote the right and left eigenvectors corresponding to $\chi(0)=\chi_0=|\chi_0|e^{i\theta}$, scaled such that $e^{i\theta}w^*v>0$, Lemma 3.1 implies

$$\begin{aligned}\frac{d}{d\tau}|\chi(\tau)|^2\Big|_{\tau=0} &= 2Re(\bar{\chi}\dot{\chi}) = -2Re\left(\bar{\chi}\frac{w^*\Delta_0 v}{w^* v}\right)\\ &= -2|\chi_0|Re\left(\frac{w^*\Delta_0 v}{e^{i\theta}w^* v}\right) = -\frac{2|\chi_0|}{|w^* v|}Re\langle wv^*,\Delta_0\rangle.\end{aligned}$$

In order to have the locally maximal decrease of $|\chi(\tau)|^2$ at $\tau=0$ choose

$$\Delta_0 = DP_{\mathbb{BLK}}\left(wv^*\right), \tag{29}$$

where the positive diagonal matrix D is chosen such that $\Delta_0\in\mathbb{BLK}_1$. This is always possible under the genericity assumptions (12) and (13). The orthogonal projector $P_{\mathbb{BLK}}$ onto $\mathbb{BLK}^*$ can be expressed in analogy to Equation (21) for $P_{\mathbb{BLK}^*}$, with the notable difference that $\gamma_\ell = Re\left(tr\left(C_\ell\right)/r_\ell\right)$ for $\ell=N'+1,\cdots,N$. Note that there is no need to form A^{-1}; v and w can be obtained as the eigenvectors associated to a largest eigenvalue of A. However, attention needs to be paid to the scaling. Since the largest eigenvalue of A is $\frac{1}{|\chi_0|}e^{-i\theta}$, w and v have to be scaled accordingly.

A possible choice for ϵ_0 is obtained by solving the following simple linear equation, resulting from the first order expansion of the eigenvalue at $\tau=0$:

$$|\chi(\epsilon_0)|^2 + \frac{d}{d\tau}|\chi(\tau)|^2\Big|_{\tau=0}\epsilon_0 = 0.$$

This gives

$$\epsilon_0 = \frac{|\chi_0||y^*x|}{2Re\langle wv^*,\Delta_0\rangle} = \frac{|\chi_0||w^*v|}{2\left\|P_{\mathbb{BLK}}\left(wv^*\right)\right\|}. \tag{30}$$

This can be improved in a simple way by computing this expression for ϵ_0 for several eigenvalues of A (say, the m largest ones) and taking the smallest computed ϵ_0. For a sparse matrix A, the matlab function **eigs** (an interface for ARPACK, which implements the implicitly restarted Arnoldi Method [18] [19] allows to efficiently compute a predefined number m of Ritz values.

Another possible, very natural choice for ϵ_0 is given by

$$\epsilon_0 = \frac{1}{\bar{\mu}_{\mathbb{BLK}}(A)} \tag{31}$$

where $\bar{\mu}_{\mathbb{BLK}}(A)$ is the upper bound for the SSV computed by the matlab function mussv.

4. Fast Approximation of $\mu_{\mathbb{BLK}}(A)$ [13]

This section discuss the outer algorithm for computing a lower bound of $\mu_{\mathbb{BLK}}(A)$. Since the principles are the same, one can treat the case of purely complex perturbations in detail and provide a briefer discussion on the extension to the case of mixed complex/real perturbations.

Purely Complex Perturbations

In the following let $\lambda(\epsilon)$ denote a continuous branch of (local) maximizers for

$$\max_{\lambda\in\Lambda_{\epsilon}^{\mathbb{BLK}^*}(A)} |\lambda|,$$

computed by determining the stationary points $\Delta(\epsilon)$ of the system of ODEs in Equation (30). The computation of the SSV is equivalent to the smallest solution ϵ of the equation $|\lambda(\epsilon)| = 1$. In order to approximate this solution, aim at computing $\epsilon^\star$ such that the boundary of the $\epsilon^\star$-spectral value set is locally contained in the unit disk and its boundary $\partial\Lambda_{\epsilon^\star}^{\mathbb{BLK}^*}(A)$ is tangential to the unit circle. This provides a lower bound $1/\epsilon^\star$ for $\mu_{\mathbb{BLK}}(A)$.

Now make the following generic assumption.

Assumption 4.1. [13]. For a local extremizer $\Delta(\epsilon)$ of $\Lambda_{\epsilon}^{\mathbb{BLK}^*}(A)$, with corresponding largest eigenvalue $\lambda(\epsilon)$, assume that $\lambda(\epsilon)$ is simple and that $\Delta(\cdot)$ and $\lambda(\cdot)$ are smooth in a neighborhood of ϵ.

The following theorem gives an explicit and easily computable expression for the derivative of $|\lambda(\epsilon)|$.

Theorem 4.1. [13]. Suppose that Assumption 4.1 holds for $\Delta(\epsilon)\in\mathbb{BLK}_1^*$ and $\lambda(\epsilon)$. Let $v(\epsilon)$ and $w(\epsilon)$ be the corresponding right and left eigenvectors of $\epsilon A\Delta(\epsilon)$, scaled according to Equation (22). Consider the partitioning of $v(\epsilon)$, $u(\epsilon) = A^* w(\epsilon)$, and suppose that Assumptions (12) and (13) hold. Then

$$\frac{\mathrm{d}|\lambda(\epsilon)|}{\mathrm{d}\epsilon} = \frac{1}{\left|w(\epsilon)^* v(\epsilon)\right|}\left(\sum_{i=1}^{N}\left|u_i(\epsilon)^* v_i(\epsilon)\right| + \sum_{j=1}^{F}\left\|u_{N+j}(\epsilon)\right\|\left\|w_{N+j}(\epsilon)\right\|\right) > 0. \quad (32)$$

5. Numerical Experimentation

This section provides the comparison of the lower bounds of SSV computed by well-known Matlab function mussv and the algorithm [13].

We consider the following delay eigenvalue problem of the form:

$$\left(\lambda I - (A_1 + \delta A_1) - (A_2 + \delta A_2)\mathrm{e}^{-\lambda\tau}\right)\upsilon = 0$$

where,

$$A_1 = \begin{bmatrix} -5 & 1 \\ 2 & -6 \end{bmatrix};\ A_2 = \begin{bmatrix} -2 & 1 \\ 4 & -1 \end{bmatrix}$$

and $\tau = 1$.

Example 1. Consider the following two dimensional matrix A_1 taken from above mentioned delay eigenvalue problem.

$$A_1 = \begin{bmatrix} -5 & 1 \\ 2 & -6 \end{bmatrix},$$

along with the perturbation set

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_1, s_2 I_1\right) : s_1, s_2 \in \mathbb{C} \right\}.$$

Apply the Matlab routine mussv, one can obtain the perturbation $\hat{\Delta}$ with

$$\hat{\Delta} = \begin{bmatrix} -0.1429 & 0 \\ 0 & -0.1429 \end{bmatrix},$$

and $\left\|\hat{\Delta}\right\|_2 = 0.1429$. For this example, one can obtain the upper bound $\mu_{PD}^{upper} = 7.0000$ while the same lower bound as $\mu_{PD}^{lower} = 7.0000$.

By using the algorithm [13], one can obtain the perturbation $\epsilon^* \Delta^*$ with

$$\Delta^* = \begin{bmatrix} -1.0000 - 0.0000i & 0 \\ 0 & -1.0000 + 0.0000i \end{bmatrix},$$

and $\epsilon^* = 0.1429$ while $\left\|\Delta^*\right\|_2 = 1$. The same lower bound can be obtained $\mu_{New}^{lower} = 7.0000$ as the one obtained by mussv.

In the following Table 1, it is presented the comparison of the bounds of SSV computed by MUSSV and the algorithm [13] for the matrix A_1 given bellow. In the very first column, it is presented the dimension of the matrix A_1. In the second column, it is presented the set of block diagonal matrices denoted by BLK. In the third, fourth and fifth columns, it is presented the upper and lower bounds μ_u^{mussv}, μ_l^{mussv} computed by MUSSV and the lower bound μ_l^{New} computed by algorithm [13] respectively.

$$A_1 = \begin{bmatrix} -5 & 1 \\ 2 & -6 \end{bmatrix},$$

and

Example 2. Consider the following two dimensional matrix A_2 taken from above mentioned delay eigenvalue problem.

$$A_1 = \begin{bmatrix} -2 & 1 \\ 4 & -1 \end{bmatrix},$$

along with the perturbation set

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_1, s_2 I_1\right) : s_1, s_2 \in \mathbb{C} \right\}.$$

Apply the Matlab routine mussv, one can obtain the perturbation $\hat{\Delta}$ with

$$\hat{\Delta} = \begin{bmatrix} -0.2808 & 0 \\ 0 & -0.2808 \end{bmatrix},$$

Table 1. Comparison of lower bounds of SSV with MATLAB function mussv.

n	BLK	μ_u^{mussv}	μ_l^{mussv}	μ_l^{New}
02	$\left\{diag\left(\Delta_1\right) : \Delta_1 \in \mathbb{C}^{2,2}\right\}$	7.1038	7.1038	7.1038
02	$\left\{diag\left(s_1 I_2\right) : s_1 \in \mathbb{C}\right\}$	7.0000	7.0000	7.0000
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1, s_2 \in \mathbb{R}\right\}$	7.0000	7.0000	7.0000
02	$\left\{diag\left(s_1 I_2\right) : s_1 \in \mathbb{R}\right\}$	7.0000	7.0000	7.0000
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1 \in \mathbb{R}, s_2 \in \mathbb{C}\right\}$	7.0000	7.0000	7.0000
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1 \in \mathbb{C}, s_2 \in \mathbb{R}\right\}$	7.0000	7.0000	7.0000

and $\left\|\hat{\Delta}\right\|_2 = 0.2808$. For this example, one can obtain the upper bound $\mu_{PD}^{upper} = 3.5616$ while the same lower bound as $\mu_{PD}^{lower} = 3.5616$.

By using the algorithm [13], one can obtain the perturbation $\epsilon^* \Delta^*$ with

$$\Delta^* = \begin{bmatrix} -1.0000 - 0.0000i & 0 \\ 0 & -1.0000 + 0.0000i \end{bmatrix},$$

and $\epsilon^* = 0.2808$ while $\left\|\Delta^*\right\|_2 = 1$. The same lower bound can be obtained $\mu_{New}^{lower} = 3.5616$ as the one obtained by mussv.

In Table 2, it is presented the comparison of the bounds of SSV computed by MUSSV and the algorithm [13] for the matrix A_2 given bellow. In the very first column, it is presented the dimension of the matrix A_2. In the second column, it is presented the set of block diagonal matrices denoted by BLK. In the third, fourth and fifth columns, it is presented the upper and lower bounds μ_u^{mussv}, μ_l^{mussv} computed by MUSSV and the lower bound μ_l^{New} computed by algorithm [13] respectively.

$$A_2 = \begin{bmatrix} -2 & 1 \\ 4 & -1 \end{bmatrix},$$

and

We consider the polynomial eigenvalue problems.

Consider the quadratic eigenvalue problem of the form:

$$\sum_{i=1}^{m} \left(A_i + \delta A_i\right) \lambda^{i-1} \upsilon = 0.$$

Example 3. Consider the following three dimensional matrix A_3 taken from above mentioned delay eigenvalue problem.

$$A_3 = \begin{bmatrix} 121 & 18.9 & 15.9 \\ 0 & 2.7 & 0.145 \\ 11.9 & 3.64 & 15.5 \end{bmatrix},$$

along with the perturbation set

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_1, s_2 I_1, s_3 I_1\right) : s_1, s_2, s_3 \in \mathbb{C} \right\}.$$

Apply the Matlab routine mussv, one can obtain the perturbation $\hat{\Delta}$ with

Table 2. Comparison of lower bounds of SSV with MATLAB function mussv.

n	BLK	μ_u^{mussv}	μ_l^{mussv}	μ_l^{New}
02	$\left\{diag\left(\Delta_1\right) : \Delta_1 \in \mathbb{C}^{2,2}\right\}$	4.6708	4.6708	4.6708
02	$\left\{diag\left(s_1 I_2\right) : s_1 \in \mathbb{C}\right\}$	3.5616	3.5616	3.5616
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1, s_2 \in \mathbb{R}\right\}$	3.5616	3.5616	3.5616
02	$\left\{diag\left(s_1 I_2\right) : s_1 \in \mathbb{R}\right\}$	3.5616	3.5616	3.5616
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1 \in \mathbb{R}, s_2 \in \mathbb{C}\right\}$	3.5616	3.5616	3.5616
02	$\left\{diag\left(s_1 I_1, s_2 I_1\right) : s_1 \in \mathbb{C}, s_2 \in \mathbb{R}\right\}$	3.5616	3.5616	3.5616

$$\hat{\Delta}=\begin{bmatrix}0.0081 & 0 & 0\\ 0 & 0.0081 & 0\\ 0 & 0 & 0.0081\end{bmatrix},$$

and $\left\|\hat{\Delta}\right\|_2 = 0.0081$. For this example, one can obtain the upper bound $\mu_{PD}^{upper} = 122.7747$ while the same lower bound as $\mu_{PD}^{lower} = 122.7665$.

By using the algorithm [13], one can obtain the perturbation $\epsilon^*\Delta^*$ with

$$\Delta^*=\begin{bmatrix}-1.0000-0.0000i & 0 & 0\\ 0 & -1.0000+0.0000i & 0\\ 0 & 0 & 1.0000-0.0000i\end{bmatrix},$$

and $\epsilon^* = 0.0081$ while $\left\|\Delta^*\right\|_2 = 1$. The same lower bound can be obtained $\mu_{New}^{lower} = 122.7665$ as the one obtained by mussv.

In Table 3, it is presented the comparison of the bounds of SSV computed by MUSSV and the algorithm [13] for the matrix A_3 given bellow. In the very first column, it is presented the dimension of the matrix A_3. In the second column, it is presented the set of block diagonal matrices denoted by BLK. In the third, fourth and fifth columns, it is presented the upper and lower bounds μ_u^{mussv}, μ_l^{mussv} computed by MUSSV and the lower bound μ_l^{New} computed by algorithm [13] respectively.

$$A_3=\begin{bmatrix}121 & 18.9 & 15.9\\ 0 & 2.7 & 0.145\\ 11.9 & 3.64 & 15.5\end{bmatrix},$$

and

Example 4. Consider the following three dimensional matrix A_4 taken from above mentioned delay eigenvalue problem.

$$A_4=\begin{bmatrix}7.66 & 2.45 & 2.1\\ 0.23 & 1.04 & 0.223\\ 0.6 & 0.756 & 0.658\end{bmatrix},$$

Table 3. Comparison of lower bounds of SSV with MATLAB function mussv.

n	BLK	μ_u^{mussv}	μ_l^{mussv}	μ_l^{New}
03	$\{diag(s_1I_1,\Delta_1): s_1\in\mathbb{C},\Delta_1\in\mathbb{C}^{2,2}\}$	123.6596	123.6596	123.6575
03	$\{diag(\Delta_1,s_1I_1): \Delta_1\in\mathbb{C}^{2,2},s_1\in\mathbb{C}\}$	124.2691	124.2692	124.2691
03	$\{diag(s_1I_3): s_1\in\mathbb{C}\}$	122.7665	122.7665	122.7665
03	$\{diag(\Delta_1): \Delta_1\in\mathbb{C}^{3,3}\}$	124.3212	124.3212	124.3212
03	$\{diag(s_1I_1,s_2I_1,s_3I_1): s_1,s_2,s_3\in\mathbb{R}\}$	122.7665	122.7744	122.7665
03	$\{diag(s_1I_1,\Delta_2): s_1\in\mathbb{R},\Delta_2\in\mathbb{C}^{2,2}\}$	123.6596	123.6596	123.6596
03	$\{diag(\Delta_1,s_1I_1): \Delta_1\in\mathbb{C}^{2,2},s_1\in\mathbb{R}\}$	124.2691	124.2692	124.2691
03	$\{diag(s_1I_1,s_2I_2): s_1\in\mathbb{R},s_2\in\mathbb{C}\}$	122.7665	122.7690	122.7665
03	$\{diag(s_1I_3): s_1\in\mathbb{R}\}$	122.7665	122.7666	122.7665

along with the perturbation set

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_1, s_2 I_1, s_3 I_1\right) : s_1, s_2, s_3 \in \mathbb{C} \right\}.$$

Apply the Matlab routine mussv, one can obtain the perturbation $\hat{\Delta}$ with

$$\hat{\Delta} = \begin{bmatrix} 0.1261 & 0 & 0 \\ 0 & 0.1261 & 0 \\ 0 & 0 & 0.1261 \end{bmatrix},$$

and $\left\|\hat{\Delta}\right\|_2 = 0.1261$. For this example, one can obtain the upper bound $\mu_{PD}^{upper} = 7.9298$ while the same lower bound as $\mu_{PD}^{lower} = 7.9298$.

By using the algorithm [13], one can obtain the perturbation $\epsilon^* \Delta^*$ with

$$\Delta^* = \begin{bmatrix} -1.0000 - 0.0000i & 0 & 0 \\ 0 & -1.0000 + 0.0000i & 0 \\ 0 & 0 & 1.0000 - 0.0000i \end{bmatrix},$$

and $\epsilon^* = 0.1261$ while $\left\|\Delta^*\right\|_2 = 1$. The same lower bound can be obtained $\mu_{New}^{lower} = 7.9298$ as the one obtained by mussv.

In Table 4, it is presented the comparison of the bounds of SSV computed by MUSSV and the algorithm [13] for the matrix A_4 given bellow. In the very first column, it is presented the dimension of the matrix A_4. In the second column, it is presented the set of block diagonal matrices denoted by BLK. In the third, fourth and fifth columns, it is presented the upper and lower bounds μ_u^{mussv}, μ_l^{mussv} computed by MUSSV and the lower bound μ_l^{New} computed by algorithm [13] respectively.

$$A_4 = \begin{bmatrix} 7.66 & 2.45 & 2.1 \\ 0.23 & 1.04 & 0.223 \\ 0.6 & 0.756 & 0.658 \end{bmatrix},$$

and

Example 5. Consider the following three dimensional matrix A_4 taken from above mentioned delay eigenvalue problem.

Table 4. Comparison of lower bounds of SSV with MATLAB function mussv.

n	BLK	μ_u^{mussv}	μ_l^{mussv}	μ_l^{New}
03	$\{diag(s_1 I_1, \Delta_1) : s_1 \in \mathbb{C}, \Delta_1 \in \mathbb{C}^{2,2}\}$	7.9704	7.9704	7.9700
03	$\{diag(\Delta_1, s_1 I_1) : \Delta_1 \in \mathbb{C}^{2,2}, s_1 \in \mathbb{C}\}$	8.2876	8.2847	8.2847
03	$\{diag(s_1 I_3) : s_1 \in \mathbb{C}\}$	7.9298	7.9298	7.9298
03	$\{diag(\Delta_1) : \Delta_1 \in \mathbb{C}^{3,3}\}$	8.3860	8.3860	8.3860
03	$\{diag(s_1 I_1, s_2 I_1, s_3 I_1) : s_1, s_2, s_3 \in \mathbb{R}\}$	7.9298	7.9298	7.9298
03	$\{diag(s_1 I_1, \Delta_2) : s_1 \in \mathbb{R}, \Delta_2 \in \mathbb{C}^{2,2}\}$	7.9704	7.9704	7.9700
03	$\{diag(\Delta_1, s_1 I_1) : \Delta_1 \in \mathbb{C}^{2,2}, s_1 \in \mathbb{R}\}$	8.2876	8.2847	8.2847
03	$\{diag(s_1 I_1, s_2 I_2) : s_1 \in \mathbb{R}, s_2 \in \mathbb{C}\}$	7.9301	7.9298	7.9298
03	$\{diag(s_1 I_3) : s_1 \in \mathbb{R}\}$	7.9298	7.9298	7.9298

$$A_5 = \begin{bmatrix} 17.6 & 1.28 & 2.89 \\ 1.28 & 0.824 & 0.412 \\ 2.89 & 0.413 & 0.725 \end{bmatrix},$$

along with the perturbation set

$$\mathbb{BLK} = \left\{ diag\left(s_1 I_1, s_2 I_1, s_3 I_1\right) : s_1, s_2, s_3 \in \mathbb{C} \right\}.$$

Apply the Matlab routine mussv, one can obtain the perturbation $\hat{\Delta}$ with

$$\hat{\Delta} = \begin{bmatrix} 0.0550 & 0 & 0 \\ 0 & 0.0550 & 0 \\ 0 & 0 & 0.0550 \end{bmatrix},$$

and $\left\| \hat{\Delta} \right\|_2 = 0.0550.$ For this example, one can obtain the upper bound $\mu_{PD}^{upper} = 18.1832$ while the same lower bound as $\mu_{PD}^{lower} = 18.1832$.

By using the algorithm [13], one can obtain the perturbation $\epsilon^* \Delta^*$ with

$$\Delta^* = \begin{bmatrix} -1.0000 - 0.0000i & 0 & 0 \\ 0 & -1.0000 + 0.0000i & 0 \\ 0 & 0 & 1.0000 - 0.0000i \end{bmatrix},$$

and $\varepsilon^* = 0.0550$ while $\left\| \Delta^* \right\|_2 = 1.$ The same lower bound can be obtained $\mu_{New}^{lower} = 18.1832$ as the one obtained by mussv.

In Table 5, it is presented the comparison of the bounds of SSV computed by MUSSV and the algorithm [13] for the matrix A_5 given bellow. In the very first column, it is presented the dimension of the matrix A_5. In the second column, it is presented the set of block diagonal matrices denoted by BLK. In the third, fourth and fifth columns, it is presented the upper and lower bounds μ_u^{mussv}, μ_l^{mussv} computed by MUSSV and the lower bound μ_l^{New} computed by algorithm [13] respectively.

$$A_5 = \begin{bmatrix} 17.6 & 1.28 & 2.89 \\ 1.28 & 0.824 & 0.412 \\ 2.89 & 0.413 & 0.725 \end{bmatrix},$$

and

Table 5. Comparison of lower bounds of SSV with MATLAB function mussv.

n	BLK	μ_u^{mussv}	μ_l^{mussv}	μ_l^{New}
03	$\left\{ diag\left(s_1 I_1, \Delta_1\right) : s_1 \in \mathbb{C}, \Delta_1 \in \mathbb{C}^{2,2} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(\Delta_1, s_1 I_1\right) : \Delta_1 \in \mathbb{C}^{2,2}, s_1 \in \mathbb{C} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(s_1 I_3\right) : s_1 \in \mathbb{C} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(\Delta_1\right) : \Delta_1 \in \mathbb{C}^{3,3} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(s_1 I_1, s_2 I_1, s_3 I_1\right) : s_1, s_2, s_3 \in \mathbb{R} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(s_1 I_1, \Delta_2\right) : s_1 \in \mathbb{R}, \Delta_2 \in \mathbb{C}^{2,2} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(\Delta_1, s_1 I_1\right) : \Delta_1 \in \mathbb{C}^{2,2}, s_1 \in \mathbb{R} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(s_1 I_1, s_2 I_2\right) : s_1 \in \mathbb{R}, s_2 \in \mathbb{C} \right\}$	18.1832	18.1832	18.1832
03	$\left\{ diag\left(s_1 I_3\right) : s_1 \in \mathbb{R} \right\}$	18.1832	18.1832	18.1832

6. Conclusion

In this article the problem of approximating structured singular values for the delay eigenvalue and polynomial eigenvalue problems is considered. The obtained results provide a characterization of extremizers and gradient systems, which can be integrated numerically in order to provide approximations from below to the structured singular value of a matrix subject to general pure complex and mixed real and complex block perturbations. The experimental results show the comparison of the lower bounds of structured singular values with once computed by MUSSV and alogorithm [13].

References

[1] Packard, A. and Doyle, J. (1993) The Complex Structured Singular Value. *Automatica*, **29**, 71-109. https://doi.org/10.1016/0005-1098(93)90175-S

[2] Bernhardsson, B. and Rantzer, A. and Qiu, L. (1994) Real Perturbation Values and Real Quadratic Forms in a Complex Vector Space. *Linear Algebra and Its Applications*, **1**, 131-154.

[3] Chen, J., Fan, M.K.H. and Nett, C.N. (1996) Structured Singular Values with Non-diagonal Structures. I. Characterizations. *IEEE Transactions on Automatic Control*, **41**, 1507-1511.

[4] Hinrichsen, D. and Pritchard, A.J. (2005) Mathematical Systems Theory I, Vol. 48 of Texts in Applied Mathematics. Springer-Verlag, Berlin.

[5] Karow, M., Kokiopoulou, E. and Kressner, D. (2010) On the Computation of Structured Singular Values and Pseudospectra. *Systems & Control Letters*, **59**, 122-129. https://doi.org/10.1016/j.sysconle.2009.12.007

[6] Karow, M., Kressner, D. and Tisseur, F. (2006) Structured Eigenvalue Condition Numbers. *SIAM Journal on Matrix Analysis and Applications*, **28**, 1052-1068. https://doi.org/10.1137/050628519

[7] Qiu, L., Bernhardsson, B., Rantzer, A., Davison, E.J., Young, P.M. and Doyle, J.C. (1995) A Formula for Computation of the Real Stability Radius. *Automatica*, **31**, 879-890.

[8] Braatz, R.P., Young, P.M., Doyle, J.C. and Morari, M. (1994) Computational Complexity of μ Calculation. *IEEE Transactions on Automatic Control*, **39**, 1000-1002.

[9] Fan, M.K.H., Tits, A.L. and Doyle, J.C. (1991) Robustness in the Presence of Mixed Parametric Uncertainty and Unmodeled Dynamics. *IEEE Transactions on Automatic Control*, **36**, 25-38.

[10] Young, P.M., Newlin, M.P. and Doyle, J.C. (1992) Practical Computation of the Mixed μ Problem. *American Control Conference*, 24-26 June 1992, 2190- 2194.

[11] Packard, A., Fan, M.K.H. and Doyle, J. (1998) A Power Method for the Structured Singular Value. *Proceedings of the 27th IEEE Conference on Decision and Control*, 7-9 December 1988, 2132-2137.

[12] Young, P.M., Doyle, J.C., Packard, A., *et al.* (1994) Theoretical and Computational Aspects of the Structured Singular Value. *Systems*, *Control and Information*, **38**, 129-138.

[13] Guglielmi, N., Rehman, M.-U. and Kressner, D. (2016) A Novel Iterative Method to Approximate Structured Singular Values. arXiv preprint arXiv:1605.04103.

[14] Zhou, K., Doyle, J. and Glover, K. (1996) Robust and Optimal Control. Prentice Hall, New Jersey, Volume 40.

[15] Kato, T. (2013) Perturbation Theory for Linear Operators. Springer Science & Business Media, Volume 132.

[16] Guglielmi, N. and Lubich, C. (2011) Differential Equations for Roaming Pseudospectra: Paths to Extremal Points and Boundary Tracking. *SIAM Journal on Numerical Analysis*, **49**, 1194-1209. https://doi.org/10.1137/100817851

[17] Guglielmi, N, and Lubich, C. (2013) Low-Rank Dynamics for Computing Extremal Points of Real Pseudospectra. *SIAM Journal on Matrix Analysis and Applications*, **34**, 40-66. https://doi.org/10.1137/120862399

[18] Lehoucq, R.B. and Sorensen, D.C. (1996) Deflation Techniques for an Implicitly Restarted Arnoldi Iteration. *SIAM Journal on Matrix Analysis and Applications*, **17**, 789-821. https://doi.org/10.1137/S0895479895281484

[19] Lehoucq, R.B., Sorensen, D.C. and Yang, C. (1998) ARPACK Users' Guide: Solution of Large-Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods. SIAM, Volume 6.