Proceedings of the 27th Conference
on Decision and Control
Austin, Texas • December 1988

# COMPUTATION OF THE STABILITY ROBUSTNESS OF LARGE STATE SPACE MODELS WITH REAL PERTURBATIONS*

L. Qiu and E.J. Davison

Department of Electrical Engineering
University of Toronto
Toronto, Ontario, Canada M5S 1A4

**Abstract**

This paper addresses the computational problem encounted by a new method of stability robustness analysis outlined in [1], when applied to large systems. An iterative procedure is developed to compute the singular values and singular vectors of certain classes of large composite matrices; such a procedure can be used to solve problems which, because of dimensionality problems, can not be solved by applying the QR transformation method. The procedure is then applied to the determination of stability robustness bounds of large state space systems with real perturbations using the approach of [1]. Various numerical examples, including a 46th order spacecraft system, are given to illustrate the results obtained.

## 1. Introduction

A considerable amount of attention has been paid in recent years to the area of the robust stability of state space models. One of the problems considered in this area is to find the distance of a real stable matrix to the set of real unstable matrices. This problem can be made precise as follows:

Let $\mathbb{C}_c$ and $\mathbb{C}_d$ be the stable regions in the complex plane for continuous and discrete time systems, respectively, i.e. $\mathbb{C}_c=\{s\in\mathbb{C},\ \mathrm{Re}(s)<0\}$, $\mathbb{C}_d=\{s\in\mathbb{C},\ |s|<1\}$. A matrix $A\in\mathbb{R}^{n\times n}$ is said to be stable in the continuous (or discrete) time case if $\mathrm{sp}(A)\subset\mathbb{C}_c$ (or $\mathbb{C}_d$), where $\mathrm{sp}(A)$ means the spectrum of $A$, otherwise $A$ is said to be unstable. It is desired to find the distance of a given stable matrix $A\in\mathbb{R}^{n\times n}$ from the set of all unstable matrices in $\mathbb{R}^{n\times n}$, which is defined for the continuous and discrete time case by

$$\mu(A)=\inf\{\|\Delta A\|_s:\Delta A\in\mathbb{R}^{n\times n}\text{ and }\mathrm{sp}(A+\Delta A)\not\subset\mathbb{C}_c\} \tag{1}$$

and

$$\nu(A)=\inf\{\|\Delta A\|_s:\Delta A\in\mathbb{R}^{n\times n}\text{ and }\mathrm{sp}(A+\Delta A)\not\subset\mathbb{C}_d\} \tag{2}$$

respectively, where $\|\Delta A\|_s$ means the spectral norm of $\Delta A$.

The problem of computing the exact $\mu(A)$ or $\nu(A)$ for a given $A\in\mathbb{R}^{n\times n}$ has not yet been solved, but various bounds of $\mu(A)$ and $\nu(A)$ have been obtained, e.g. see [1] and the references therein. A new method based on composite matrices has been recently developed to address this problem [1]. Examples have shown that the lower bounds of $\mu(A)$ and $\nu(A)$ derived from this method are the least conservative of all bounds published so far. For systems of moderate size, the computation required is simple and numerically well-defined. However, for large systems, the computation becomes nontrivial, since the proposed method requires the calculation of singular values of certain matrices with dimensions which grow at the rate of $n^2$. The purpose of this paper is to develop an iterative algorithm to compute the singular values of such matrices without actually constructing the matrices. The algorithm borrows from the idea of the block power method of eigenanalysis, see e.g.[4], and is especially suitable for the case when only a few extreme singular values are required. Convergence of the algorithm is guaranteed, and the dimensions of matrices involved in the algorithm are no larger than $n\times n$; the algorithm can therefore be used to extend the results of [1] to the case when $n$ is large.

The following notation will be used throughout this paper. For a matrix $A\in\mathbb{R}^{n\times n}$, $A'$ is the transpose of $A$. $\sigma_i(A)$, $i=1,2,\ldots,n$, denotes the $i$-th singular value of $A$ with order $\sigma_1(A)\geq\sigma_2(A)\geq\cdots\geq\sigma_n(A)$; in particular, $\sigma_1(A)$ and $\sigma_n(A)$ are denoted by $\bar{\sigma}(A)$ and $\underline{\sigma}(A)$ respectively. $\|A\|_s$ denotes the spectral norm of $A$ and $\|A\|_F$ denotes the Frobenius norm of $A$, which have the property that $\|A\|_s=\bar{\sigma}(A)$, $\|A\|_F=\left[\sum_{i=1}^{n}\sigma_i^2(A)\right]^{1/2}$. If $A$ is square, the trace and spectrum of $A$ are denoted by $\mathrm{tr}(A)$ and $\mathrm{sp}(A)$ respectively, and the $i$-th eigenvalue of $A$ is denoted by $\lambda_i(A)$ with no specific order imposed.

## 2. Development

The method of [1] uses the properties of some matrix compositions. Two well-known matrix compositions are the Kronecker product and Kronecker sum [2]. Let $A=[a_{ij}]\in\mathbb{R}^{n\times n}$, $B=[b_{ij}]\in\mathbb{R}^{n\times n}$, and denote the Kronecker product and Kronecker sum of $A$ and $B$ by $A\otimes B$ and $A\oplus B$ respectively.

*This work has been supported by the Natural Sciences and Engineering Research Council of Canada under Grant no. A4396.

Two additional matrix compositions are defined below. Let $(i_1,i_2)$ be the $i$-th pair of integers in the sequence

$$(1,1),(1,2),\ldots,(1,n),(2,2),\ldots,(2,n),(3,3),\ldots,(n-1,n),(n,n). \tag{3}$$

**Definition 1**

$$A\bar{\otimes}B=[c_{ij}]\in\mathbb{R}^{\frac{1}{2}n(n+1)\times\frac{1}{2}n(n+1)},$$

where

$$c_{ij}=\begin{cases}a_{i_1,j_1}b_{i_1,j_1} & \text{if } i_1=i_2 \text{ and } j_1=j_2,\\ \frac{1}{2}(a_{i_1,j_1}b_{i_2,j_2}+a_{i_2,j_2}b_{i_1,j_1}+a_{i_1,j_1}b_{i_2,j_2}+a_{i_2,j_2}b_{i_1,j_1}) & \text{if } i_1\neq i_2 \text{ and } j_1\neq j_2,\\ \frac{\sqrt{2}}{2}(a_{i_1,j_1}b_{i_2,j_2}+a_{i_2,j_2}b_{i_1,j_1}) & \text{otherwise.}\end{cases} \tag{4}$$

Let $(r_1,r_2)$ be the $r$-th pair of integers in the sequence

$$(1,2),(1,3),\ldots,(1,n),(2,3),\ldots,(2,n),(3,4),\ldots,(n-1,n). \tag{5}$$

**Definition 2**

$$A\bar{\bar{\otimes}}B=[d_{rs}]\in\mathbb{R}^{\frac{1}{2}n(n-1)\times\frac{1}{2}n(n-1)},$$

where

$$d_{rs}=\frac{1}{2}(a_{r_1s_1}b_{r_2s_2}-a_{r_1s_2}b_{r_2s_1}-a_{r_2s_1}b_{r_1s_2}+a_{r_2s_2}b_{r_1s_1}). \tag{6}$$

We will call $\bar{\otimes}$ and $\bar{\bar{\otimes}}$ the $\bar{\otimes}$-product and $\bar{\bar{\otimes}}$-product respectively. The corresponding sum operations of $\bar{\otimes}$ and $\bar{\bar{\otimes}}$ can be defined as follows:

**Definition 3**

$$A\bar{\oplus}B=A\bar{\otimes}I_n+I_n\bar{\otimes}B\in\mathbb{R}^{\frac{1}{2}n(n+1)\times\frac{1}{2}n(n+1)}. \tag{7}$$

$$A\bar{\bar{\oplus}}B=A\bar{\bar{\otimes}}I_n+I_n\bar{\bar{\otimes}}B\in\mathbb{R}^{\frac{1}{2}n(n-1)\times\frac{1}{2}n(n-1)}. \tag{8}$$

$A\bar{\oplus}B$ and $A\bar{\bar{\oplus}}B$ will be called the $\bar{\oplus}$-sum and $\bar{\bar{\oplus}}$-sum of $A$, $B$ respectively.

From Definition 1-3, it is easy to see that operations $\bar{\otimes}$, $\bar{\bar{\otimes}}$, $\bar{\oplus}$ and $\bar{\bar{\oplus}}$ are commutative, while operations $\otimes$ and $\oplus$ are not.

The lower bounds of $\mu(A)$ and $\nu(A)$ can then be obtained in terms of these matrix compositions as follows:

**Theorem 1 [1]:** Given $A\in\mathbb{R}^{n\times n}$ such that $\mathrm{sp}(A)\subset\mathbb{C}_c$, then

$$\mu(A)\geq\min\{\underline{\sigma}(A),\ \frac{1}{2}\sigma_{n^2-1}(A\oplus A)\}, \tag{9}$$

$$\mu(A)\geq\frac{1}{2}\underline{\sigma}(A\bar{\oplus}A), \tag{10}$$

$$\mu(A)\geq\min\{\underline{\sigma}(A),\ \frac{1}{2}\underline{\sigma}(A\bar{\bar{\oplus}}A)\}. \tag{11}$$

**Theorem 2 [1]:** Given $A\in\mathbb{R}^{n\times n}$ such that $\mathrm{sp}(A)\subset\mathbb{C}_d$, then

$$\nu(A)\geq\min\{\underline{\sigma}(A-I),\underline{\sigma}(A+I),[\sigma_{n^2-1}(A\otimes A-I)+\bar{\sigma}^2(A)]^{1/2}-\bar{\sigma}(A)\}, \tag{12}$$

$$\nu(A)\geq[\underline{\sigma}(A\bar{\otimes}A-I)+\bar{\sigma}^2(A)]^{1/2}-\bar{\sigma}(A), \tag{13}$$

$$\nu(A)\geq\min\{\underline{\sigma}(A-I),\underline{\sigma}(A+I),[\underline{\sigma}(A\bar{\bar{\otimes}}A-I)+\bar{\sigma}^2(A)]^{1/2}-\bar{\sigma}(A)\}. \tag{14}$$

**Remarks [1]**

(a) If $\underline{\sigma}(A)\leq\frac{1}{2}\sigma_{n^2-1}(A\oplus A)$ or $\underline{\sigma}(A)\leq\frac{1}{2}\underline{\sigma}(A\bar{\bar{\oplus}}A)$, then

88CH2531-2/88/0000-1380$1.00 © 1988 IEEE

$$\mu(A)=\underline{\sigma}(A).$$

If $\min\{\underline{\sigma}(A+I),\underline{\sigma}(A-I)\}\le[\sigma_{n^2-1}(A\otimes A-I)+\bar{\sigma}^2(A)]^{1/2}-\bar{\sigma}(A)$ or $\min\{\underline{\sigma}(A+I),\underline{\sigma}(A-I)\}\le[\underline{\sigma}(A\otimes A-I)+\bar{\sigma}^2(A)]^{1/2}-\bar{\sigma}(A)$, then

$$\nu(A)=\min\{\underline{\sigma}(A+I),\underline{\sigma}(A-I)\}.$$

(b) If $A$ is a normal matrix, then the three expressions on the right hand sides of (9)-(11) are the exact value of $\mu(A)$ which is equal to $\min\{-\mathrm{Re}\lambda_i(A), i=1,2,\ldots,n\}$, and the three expressions on the right hand sides of (12)-(14) are the exact value of $\nu(A)$ which is equal to $\min\{1-|\lambda_i(A)|, i=1,2,\ldots,n\}$.

(c) If $A$ is a 2×2 matrix, then

$$\sigma_3(A\oplus A)=\underline{\sigma}(A\,\bar{\oplus}\,A)=-\mathrm{tr}(A)$$

and

$$\mu(A)=\min\{\underline{\sigma}(A),-\frac{1}{2}\mathrm{tr}(A)\}.$$

(d) A large number of examples (e.g. Example 3 of this paper) show that among bounds (9)-(11), bound (11) is tightest and bound (10) is most conservative, and among bounds (12)-(14), bound (14) is the tightest and bound (12) is most conservative. A proof of this result has not yet been obtained.

If matrix $A$ is moderate in size, the computation required in (9)-(14) is simple and numerically well-defined. However, computational difficulties will arise if the $n\times n$ matrix $A$ is large in size, because the composite matrices required have dimensions $n^2$, $\frac{1}{2}n(n+1)$, $\frac{1}{2}n(n-1)$ respectively. Therefore it is desired to have an alternative way to determine the required singular values without constructing the composite matrices explicitly. In the following, we will develop such a procedure. Before presenting the algorithm, some preliminary results must be established.

Let $A\in\mathbb{R}^{n\times n}$. Consider $A\oplus A$ and $A\otimes A-I$ as linear operators on the inner product space $\mathbb{R}^{n^2}$, mapping $x\in\mathbb{R}^{n^2}$ to $(A\oplus A)x$ and $(A\otimes A-I)x\in\mathbb{R}^{n^2}$ respectively. The inner product on $\mathbb{R}^{n^2}$ is defined in the usual way, i.e. $\langle x,y\rangle=x'y$, $\forall x,y\in\mathbb{R}^{n^2}$. The norm induced by this inner product is the Holder 2-norm $\|\cdot\|_2$.

The $n\times n$ matrix space $\mathbb{R}^{n\times n}$ is also an $n^2$-dimensional inner product space over $\mathbb{R}$, if it is equipped with the inner product $\langle X,Y\rangle=\mathrm{tr}(X'Y)$, $\forall X,Y\in\mathbb{R}^{n\times n}$. The norm in $\mathbb{R}^{n\times n}$ induced by this inner product is the Frobenius norm $\|\cdot\|_F$. Now define a linear operator Vec: $\mathbb{R}^{n\times n}\to\mathbb{R}^{n^2}$ by

$$\mathrm{Vec}\begin{bmatrix}x_{11}&x_{12}&\cdots&x_{1n}\\x_{21}&x_{22}&\cdots&x_{2n}\\\vdots&\vdots&&\vdots\\x_{n1}&x_{n2}&\cdots&x_{nn}\end{bmatrix}=[x_{11}\cdots x_{n1}\ x_{12}\cdots x_{n2}\cdots x_{nn}]'. \quad (15)$$

We need two properties of Vec to proceed.

**Lemma 1 [2]:** Let $X,Y,Z\in\mathbb{R}^{n\times n}$, then

(a) $\mathrm{tr}(X'Y)=[\mathrm{Vec}(X)]'\mathrm{Vec}(Y)$,

(b) $\mathrm{Vec}(XYZ)=(Z'\otimes X)\mathrm{Vec}(Y)$.

Lemma 1(a) implies that Vec is an isomorphism. Under this isomorphism, operators $A\oplus A$ and $A\otimes A-I$ on $\mathbb{R}^{n^2}$ become operators $L_c$ and $L_d$ mapping $X\in\mathbb{R}^{n\times n}$ to $L_c(X)=AX+XA'$ and $L_d(X)=AXA'-X$ respectively. $L_c$ is usually called the Lyapunov transformation. $L_d$ is the discrete time version of the Lyapunov transformation.

The singular values (which are called s-numbers in [3]) of an operator $M$ on an $l$-dimensional inner product space S are usually defined as the square roots of the eigenvalues of $M^*M$, where $M^*$ is the adjoint of $M$ which is defined by $\langle Mu,v\rangle=\langle u,M^*v\rangle$, $\forall u,v\in\mathrm{S}$. The following lemma is a restatement of the matrix singular value decomposition result in the general operator case.

**Lemma 2:** Let $M$ be any linear operator on an $l$-dimensional inner-product space S, and let $\sigma_1,\sigma_2,\ldots,\sigma_n$ be its singular values; then there exist two orthonormal bases of S, $\{u_1,u_2,\ldots,u_l\}$ and $\{v_1,v_2,\ldots,v_l\}$, such that

$$Mu_i=\sigma_i v_i \ \text{ and } \ M^*v_i=\sigma_i u_i, \qquad \forall i=1,2,\ldots,l. \quad (16)$$

Vectors $u_i$ and $v_i$ satisfying (16) are called the right and left singular vectors of $M$ corresponding to singular value $\sigma_i$ respectively. The singular vectors of $M$ are not unique in general. The discussion re the nonuniqueness of the singular vectors of matrices is given in [5]. The general operator case is analogous.

Since the operator $A\oplus A$ and $L_c$ have the relation $A\oplus A=\mathrm{Vec}\cdot L_c\cdot\mathrm{Vec}^{-1}$, and the operator Vec is an isomorphism between inner-product spaces $\mathbb{R}^{n^2}$ and $\mathbb{R}^{n\times n}$, $A\oplus A$ and $L_c$ must have the same singular values. Similarly, $A\otimes A-I$ must have the same singular values as $L_d$.

Let $\mathrm{S}_1\subset\mathbb{R}^{n\times n}$ be the subspace of all symmetric matrices, and let $\mathrm{S}_2\subset\mathbb{R}^{n\times n}$ be the subspace of all skew-symmetric matrices. Formally, $\mathrm{S}_1=\{X\in\mathbb{R}^{n\times n}: X'=X\}$ and $\mathrm{S}_2=\{X\in\mathbb{R}^{n\times n}: X'=-X\}$. The following two lemmas are obtained.

**Lemma 3 [1]**

$$\mathrm{S}_1\perp\mathrm{S}_2 \ \text{ and } \ \mathrm{S}_1\dot{+}\mathrm{S}_2=\mathbb{R}^{n\times n}.$$

**Lemma 4 [1]**

$$L_c(\mathrm{S}_1)\subset\mathrm{S}_1,\ L_c(\mathrm{S}_2)\subset\mathrm{S}_2 \ \text{ and}$$
$$L_d(\mathrm{S}_1)\subset\mathrm{S}_1,\ L_d(\mathrm{S}_2)\subset\mathrm{S}_2.$$

Lemma 3 states that $\mathrm{S}_1$ and $\mathrm{S}_2$ are orthogonal complements to each other. Lemma 4 states that $\mathrm{S}_1$ and $\mathrm{S}_2$ are reducing subspaces of $\mathbb{R}^{n\times n}$ for operators $L_c$ and $L_d$; this implies that the restrictions $L_c|\mathrm{S}_1$, $L_c|\mathrm{S}_2$, $L_d|\mathrm{S}_1$ and $L_d|\mathrm{S}_2$ are well-defined operators. Therefore the singular values of $L_c$ are the singular values of $L_c|\mathrm{S}_1$ together with those of $L_c|\mathrm{S}_2$, and the singular values of $L_d$ are the singular values of $L_d|\mathrm{S}_1$ together with those of $L_d|\mathrm{S}_2$.

Now let us define two linear operators $\Phi$: $\mathrm{S}_1\to\mathbb{R}^{\frac{1}{2}n(n+1)}$ and $\Psi$: $\mathrm{S}_2\to\mathbb{R}^{\frac{1}{2}n(n-1)}$ by

$$\Phi\begin{bmatrix}x_{11}&x_{12}&\cdots&x_{1n}\\x_{12}&x_{22}&\cdots&x_{2n}\\\vdots&\vdots&&\vdots\\x_{1n}&x_{2n}&\cdots&x_{nn}\end{bmatrix}=[x_{11}\ \sqrt{2}x_{12}\cdots\sqrt{2}x_{1n}\,x_{22}\,\sqrt{2}x_{23}\cdots\sqrt{2}x_{2n}\cdots x_{nn}]' \quad (17)$$

and

$$\Psi\begin{bmatrix}0&x_{12}&\cdots&x_{1n}\\-x_{12}&0&\cdots&x_{2n}\\\vdots&\vdots&&\vdots\\-x_{1n}&-x_{2n}&\cdots&0\end{bmatrix}=[\sqrt{2}x_{12}\,\sqrt{2}x_{13}\cdots\sqrt{2}x_{1n}\,\sqrt{2}x_{23}\cdots\sqrt{2}x_{2n}\cdots\sqrt{2}x_{(n-1)n}]'. \quad (18)$$

The following lemmas can now be obtained; their proofs are given in Appendix 1.

**Lemma 5:** Let $X,Y\in\mathrm{S}_1$ and $A\in\mathbb{R}^{n\times n}$, then

(a) $\mathrm{tr}(X'Y)=[\Phi(X)]'\Phi(Y)$,

(b) $\Phi(AX+XA')=(A\,\bar{\oplus}\,A)\Phi(X)$,

(c) $\Phi(AXA')=(A\,\bar{\otimes}\,A)\Phi(X)$.

**Lemma 6:** Let $X,Y\in\mathrm{S}_2$ and $A\in\mathbb{R}^{n\times n}$, then

(a) $\mathrm{tr}(X'Y)=[\Psi(X)]'\Psi(Y)$,

(b) $\Psi(AX+XA')=(A\,\bar{\bar{\oplus}}\,A)\Psi(X)$,

(c) $\Psi(AXA')=(A\,\bar{\bar{\otimes}}\,A)\Psi(X)$.

Lemma 5(a) implies that $\Phi$ is an isomorphism. Lemma 5(b), (c) implies that, under isomorphism $\Phi$, operators $A\,\bar{\oplus}\,A$ and $A\,\bar{\otimes}\,A-I$ on $\mathbb{R}^{\frac{1}{2}n(n+1)}$ are equivalent to operators $L_c|\mathrm{S}_1$ and $L_d|\mathrm{S}_2$ respectively. Similarly, Lemma 6 implies that $\Psi$ is an isomorphism and that operators $A\,\bar{\bar{\oplus}}\,A$ and $A\,\bar{\bar{\otimes}}\,A-I$ on $\mathbb{R}^{\frac{1}{2}n(n-1)}$ are equivalent to operators $L_c|\mathrm{S}_2$ and $L_d|\mathrm{S}_2$ respectively under isomorphism $\Psi$. Each equivalent operator pairs must have the same singular values.

As a summary, we list those equivalent operator pairs which have the same singular values:

$$\begin{aligned}A\oplus A &\ \Leftrightarrow\ L_c\\ A\otimes A-I &\ \Leftrightarrow\ L_d\\ A\,\bar{\oplus}\,A &\ \Leftrightarrow\ L_c|\mathrm{S}_1\\ A\,\bar{\otimes}\,A-I &\ \Leftrightarrow\ L_d|\mathrm{S}_1\\ A\,\bar{\bar{\oplus}}\,A &\ \Leftrightarrow\ L_c|\mathrm{S}_2\\ A\,\bar{\bar{\otimes}}\,A-I &\ \Leftrightarrow\ L_d|\mathrm{S}_2.\end{aligned}$$

Another interpretation of these equivalent operator pairs is that the matrices on the left are the matrix representations of the operators on the right under certain orthonormal bases.

If we can compute the smallest (or second smallest for $L_c$, $L_d$) singular values of those operators on the right of the list, the bounds (9)-(14) are then obtained. In the next section, iterative procedures will be developed to accomplish this computation.

## 3. Algorithm

Given a linear operator $M$ on an $l$-dimensional inner product space S over $\mathbb{R}$, the singular values of $M$ can be computed by constructing the matrix representation of $M$ with respect to an orthonormal basis of S and finding the singular value decomposition of the matrix representation. Usually, the singular value decomposition of a matrix is carried out by using the QR transformation method [5]. However, this is not always the best way to find the singular values of an operator. If $l$ is large, a simple operator on S may result in a large matrix representation. $L_c$ and $L_d$ are examples of such operators. The QR method tends to find all the singular values of a matrix; if only a few singular values are needed, the QR method is unnecessarily expensive. In this section, an iterative method is presented which can efficiently find a few largest (or smallest) singular values of an operator. Our prime interest here is to apply the method to the singular value problem required in the stability robustness analysis of systems, where the operators are transformations in matrix spaces. Another potential application of the method is the singular value problem of large sparse matrices.

Iterative methods for computing the singular values of a matrix have long been a topic in numerical linear algebra. Two such methods based on the Lanczos procedures are reported recently [6]-[8]. One of them uses a Lanczos procedure to transform the matrix into bidiagonal form and then compute the singular values of the bidiagonal matrix using a QR-like algorithm [6]. The other one uses the Lanczos algorithms for the symmetric eigenvalue problem to solve the singular value problem with the recognition that the singular values of matrix $A$ are just the positive eigenvalues of the matrix $\begin{bmatrix} 0 & A \\ A' & 0 \end{bmatrix}$ [7]-[8]. These methods certainly can be adopted to solve the singular value problem required in the stability robustness analysis. However, these methods lead to programs which seem unnecessarily complicated for the case when only a few extreme singular values are needed. Another disadvantage of using the Lanczos procedure is that it is difficult to determine the multiplicity of repeated singular values, which is crucial in the stability robustness analysis. The iterative procedure presented below are similar to the block power method for eigenvalue problems [4]-[5]. They are easy to program with the help of the available matrix computation packages(e.g. PC-MATLAB), and they are especially suitable if the number of singular values needed is much less than the total number of singular values of the operator.

We will first introduce the proposed iterative procedure for the singular value problem of matrices and then translate it into the version for the singular value problem of operators. The reason why we need an operator version of the procedure is that in some situations, such as $L_c$, $L_d$, *etc*, it is more convenient to represent vectors in S in a form other than column vectors.

Consider a matrix $M\in\mathbb{R}^{l\times l}$ which can be regarded as a linear operator in vector space $\mathbb{R}^l$. The adjoint of $M$ is then $M'$. Let $\sigma_1\ge\sigma_2\ge\cdots\ge\sigma_l$ be the singular values of $M$ and let $\{u_1,u_2,\ldots,u_l\}$, $\{v_1,v_2,\ldots,v_l\}$ be corresponding right and left singular vectors respectively. Suppose that it is desired to compute the $m$ dominant singular values of $M$ and their corresponding singular vectors. The major purpose of the iterative procedure is to find the subspace spanned by $\{u_1,u_2,\ldots,u_m\}$. If this is done, $\sigma_i, u_i, v_i, i=1,2,\ldots,m$, can be obtained easily.

First let us assume $\sigma_m>\sigma_{m+1}$. This assumption is only for the sake of the technical simplicity of the presentation. It is not crucial to the resulted algorithm. Note that this assumption also implies $\sigma_m>0$.

Denote

$$U_a=[u_1\ u_2\ \cdots\ u_m] \qquad U_b=[u_{m+1}\ u_{m+2}\ \cdots\ u_l]$$
$$V_a=[v_1\ v_2\ \cdots\ v_m] \qquad V_b=[v_{m+1}\ v_{m+2}\ \cdots\ v_l]$$
$$S_a=\mathrm{diag}[\sigma_1\ \sigma_2\ \cdots\ \sigma_m] \qquad S_b=\mathrm{diag}[\sigma_{m+1}\ \sigma_{m+2}\ \cdots\ \sigma_l].$$

Choose an initial matrix $P^{(0)}\in\mathbb{R}^{l\times m}$ which has orthonormal columns. The subspace spanned by the columns of $P^{(0)}$ is the initial trial subspace. It is clear that there exist $C_a\in\mathbb{R}^{m\times m}$ and $C_b\in\mathbb{R}^{(l-m)\times m}$ such that

$$P^{(0)}=U_aC_a+U_bC_b. \tag{19}$$

It is reasonable to assume that $C_a$ is nonsingular since this is true for "almost all" $P^{(0)}$. Let $\bar{Q}^{(0)}=MP^{(0)}$. Then

$$\bar{Q}^{(0)}=V_aS_aC_a+V_bS_bC_b.$$

Since $\bar{Q}^{(0)}$ has independent columns, there exists a nonsingular matrix $R^{(0)}\in\mathbb{R}^{m\times m}$ such that $\bar{Q}^{(0)}R^{(0)}$ has orthonormal columns. Let $Q^{(0)}=\bar{Q}^{(0)}R^{(0)}$. Then

$$Q^{(0)}=(V_aS_aC_a+V_bS_bC_b)R^{(0)}. \tag{20}$$

Similarly, let $\bar{P}^{(1)}=M'Q^{(0)}$. Then

$$\bar{P}^{(1)}=(U_aS_a^2C_a+U_bS_b^2C_b)R^{(0)}.$$

Since $\bar{P}^{(1)}$ has independent columns, there exists a nonsingular matrix $T^{(0)}\in\mathbb{R}^{m\times m}$ such that $\bar{P}^{(1)}T^{(0)}$ has orthonormal columns. Let $P^{(1)}=\bar{P}^{(1)}T^{(0)}$. Then

$$P^{(1)}=(U_aS_a^2C_a+U_bS_b^2C_b)R^{(0)}T^{(0)}. \tag{21}$$

This process can be repeated; namely, after matrix $P^{(k)}$ is obtained, the updating matrix $P^{(k+1)}$ is determined by

$$Q^{(k)}=MP^{(k)}R^{(k)} \tag{22}$$

$$P^{(k+1)}=M'Q^{(k)}T^{(k)}, \tag{23}$$

where $R^{(k)}$, $T^{(k)}$ are chosen to make $Q^{(k)}$ and $P^{(k+1)}$ have orthonormal columns. $R^{(k)}$ and $T^{(k)}$ are not uniquely determined by above requirement, but they become unique if they are restricted to upper-triangular matrices with positive diagonal matrices. In the iterative procedure defined above, it is actually not necessary to obtain $R^{(k)}$ and $T^{(k)}$ explicitly. $Q^{(k)}$ and $P^{(k+1)}$ can be obtained by orthonormalizing $MP^{(k)}$ and $M'Q^{(k)}$ using the (modified) Gram-Schmidt procedure or QR factorization [5].

Equations (22), (23) give

$$P^{(k)}=(U_aS_a^{2k}C_a+U_aS_b^{2k}C_b)\prod_{i=0}^{k-1}R^{(i)}T^{(i)}$$
$$=U_aS_a^{2k}C_aR+U_bS_b^{2k}C_bR,$$

where $R$ is used to denote $\prod_{i=0}^{k-1}R^{(i)}T^{(i)}$.

Let $x\in\mathbb{R}^l$ be a normalized vector such that $\bar{\sigma}(R)=\|Rx\|_2$ and $y=\dfrac{Rx}{\bar{\sigma}(R)}$. Then

$$\|U_aS_a^{2k}C_aR\|_s\ge\|U_aS_a^{2k}C_aRx\|_2$$
$$=\|U_aS_a^{2k}C_ay\|_2\,\bar{\sigma}(R)$$
$$\ge\underline{\sigma}(U_a)\underline{\sigma}(S_a^{2k})\underline{\sigma}(C_a)\bar{\sigma}(R)$$
$$=\sigma_m^{2k}\underline{\sigma}(C_a)\bar{\sigma}(R),$$

while

$$\|U_bS_b^{2k}C_bR\|_s\le\bar{\sigma}(U^a)\bar{\sigma}(S_b^{2k})\bar{\sigma}(C_b)\bar{\sigma}(R)$$
$$=\sigma_{m+1}^{2k}\bar{\sigma}(C_b)\bar{\sigma}(R).$$

Therefore

$$\frac{\|U_bS_b^{2k}C_bR\|_s}{\|U_aS_a^{2k}C_aR\|_s}\le\frac{\bar{\sigma}(C_b)}{\underline{\sigma}(C_a)}\left[\frac{\sigma_{m+1}}{\sigma_m}\right]^{2k}, \tag{24}$$

which goes to zero as k goes to infinity. Since $\|P^{(k)}\|_s=1$, which implies $\|U_aS_a^{2k}C_aR\|_s\le1$, then $\|U_bS_b^{2k}C_bR\|_s$ must go to zero as $k$ goes to infinity. This means that the trial space spanned by the columns of $P^{(k)}$ approaches the singular vector space spanned by the columns of $U_a$ as $k$ becomes large. For a large enough $k$, denote $P^{(k)}$, $S_a^{2k}C_aR$, $S_b^{2k}C_bR$ by $P, D_a, D_b$ respectively; then

$$P=U_aD_a+U_bD_b\simeq U_aD_a,$$

where $D_a$ is close to a unitary matrix and $D_b$ is close to zero. Let the QR factorization of $MP$ be $\Theta\Omega$ and the singular value decomposition of $\Omega$ be $\Phi\Sigma\Psi'$. Then

$$\Theta\Omega=\Theta\Phi\Sigma\Psi'=MP\simeq V_aS_aD_a.$$

So the approximation of $S_a, U_a, V_a$ can be obtained by $S_a\simeq\Sigma$, $U_a\simeq P\Psi$, $V\simeq\Theta\Phi$.

The iterative algorithm for the $m$ dominant singular values of matrix $M$ can be given as follows.

**Algorithm (matrix version)**

Step 1. Randomly choose an initial matrix $P^{(0)}\in\mathbb{R}^{l\times m}$ with orthonormal columns.

Step 2. (i) Let

$$\bar{Q}^{(k)}=MP^{(k)},$$

orthonormalize the columns of $\bar{Q}^{(k)}$ and let the result be $Q^{(k)}$.

(ii) Let

$$\bar{P}^{(k+1)}=M'Q^{(k)},$$

orthonormalize the columns of $\bar{P}^{(k+1)}$ and let the result be $P^{(k+1)}$.

Repeat (i)-(ii) for $k=0,1,2,\ldots,$ until $\|P^{(k+1)}-P^{(k)}[P^{(k)}]^*P^{(k+1)}\|_F$ is within the error tolerance.

Step 3. Let the QR factorization of $MP^{(k+1)}$ be $\Theta\Omega$ and let the singular value decomposition of $\Omega$ be $\Phi\Sigma\Psi^*$. Then $S_a \approx \Sigma$, $U_a \approx P^{(k+1)}\Psi$, and $V_a \approx \Theta\Phi$.

This algorithm can be easily translated to solve the singular value problem of operators in other forms such as $L_c$, $L_d$ and so on. Let $M$ be an linear operator on an $l$-dimensional inner product space S. Let the inner product in S be $\langle\cdot,\cdot\rangle$ and the induced norm be $\|\cdot\|$. Denote by $\{\sigma_i\}$, $\{u_i\}$, $\{v_i\}$ the singular values and corresponding singular vectors of $M$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_l$. In this case, the trial spaces will be represented by sequences of vectors $\{p_i^{(k)}\}$, $\{q_i^{(k)}\}$, *etc*, instead of matrices $P^{(k)}$, $Q^{(k)}$, *etc*. The orthonormalization of $\{\bar{p}_i^{(k)}\}$ and $\{\bar{q}_i^{(k)}\}$ can be performed using the (modified) Gram-Schmidt procedure. The matrix $\Omega$ can be constructed also using Gram-Schmidt procedure as stated in the following algorithm.

The iterative algorithm for the $m$ dominant singular values of operator $M$ can be stated as follows.

**Algorithm (operator version)**

Step 1. Randomly choose $m$ initial orthonormal vectors $p_i^{(0)} \in S$, $i=1,2,\ldots,m$.

Step 2. (i) Let

$$\bar{q}_i^{(k)} = Mp_i^{(k)} \quad \text{for } i=1,2,\ldots,m,$$

orthonormalize $\{\bar{q}_i^{(k)}\}$ and let the result be $\{q_i^{(k)}\}$.

(ii) Let

$$\bar{p}_i^{(k+1)} = M^* q_i^{(k)} \quad \text{for } i=1,2,\ldots,m,$$

orthonormalize $\{\bar{p}_i^{(k+1)}\}$ and let the result be $\{p_i^{(k+1)}\}$.

Repeat (i)-(ii) for $k=0,1,2,\ldots,$ until

$$\left[\sum_{i=1}^{m}\|p_i^{(k+1)}-\sum_{j=1}^{m}\langle p_i^{(k+1)},p_j^{(k)}\rangle p_j^{(k)}\|^2\right]^{\frac{1}{2}}$$

is within the error tolerance.

Step 3. Let $\bar{q}_i = Mp_i^{(k+1)}$, for $i=1,2,\ldots,m$, and do the following.

(i) For i=1,2 ,..., m, let

$$\omega_{ji}=\begin{cases}\langle q_i,q_j\rangle & 0<j<i\\ 0 & i<j\leq m\end{cases} \qquad q_i=\bar{q}_i-\sum_{j=1}^{i-1}\omega_{ji}q_i$$

$$\omega_{ii}=\|q_i\| \qquad q_i=\frac{q_i}{\omega_{ii}}.$$

(ii) Let $\Omega=[\omega_{ij}]\in R^{n\times n}$ and let the singular value decomposition of $\Omega$ be $\Phi\Sigma\Psi^*$.

Then for $i=1,2,\ldots,m$, $\sigma_i \approx \sigma_{ii}$, $u_i \approx \sum_{j=1}^{m}\psi_{ji}p_j^{(k+1)}$, $v_i \approx \sum_{j=1}^{m}\phi_{ji}q_j$.

It is seen from (24) that the convergence rate of the algorithm depends on the ratio $\frac{\sigma_{m+1}}{\sigma_m}$. If $\sigma_{m+1}$ is close to $\sigma_m$, the convergence will be slow. However, the algorithm provides a flexibility if this is the case, since we can compute $m+r$ singular values such that the property $\sigma_{m+r} \gg \sigma_{m+r+1}$ is satisfied, although only $m$ of the computed singular values are needed. (The $r$ extra trial vectors added to the algorithm in this case are called guard vectors [4]). The convergence can be accelerated considerably by using guard vectors.

Although the algorithm is derived under the assumption that $\sigma_m > \sigma_{m+1}$, it still applies when $\sigma_m = \sigma_{m+1}$. However, the speed of convergence depends not only on $\frac{\sigma_{m+r}}{\sigma_m}$, where $\sigma_{m+r}$ is the largest singular value of $M$ below $\sigma_m$, but also on the distribution of the $m$ dominant singular values. The details of the verification of this case is not be pursued here.

The algorithm is stated in such a way that it computes the largest singular values of operator $M$. Suppose that $M$ has singular values $\sigma_1, \sigma_2, \ldots, \sigma_l$ in descending order and $M$ is nonsingular. It is known that $M^{-1}$ has singular values $\sigma_l^{-1}, \sigma_{l-1}^{-1}, \ldots, \sigma_1^{-1}$ in descending order, and if $u_i$, $v_i$ are right and left singular vectors of $M^{-1}$ corresponding to $\sigma_i$, then they are left and right singular vectors of $M^{-1}$ corresponding to $\sigma_i^{-1}$. Consequently if $M$ and $M^*$ are replaced by $M^{-1}$ and $(M^{-1})^*$ respectively in the algorithm, the algorithm will converge to singular values $\sigma_l^{-1}, \sigma_{l-1}^{-1}, \ldots, \sigma_{l-m+1}^{-1}$ and corresponding singular vectors of $M^{-1}$, and hence the smallest singular values of $M$ and corresponding singular vectors can be obtained. To apply $M^{-1}$ and $(M^{-1})^*$ to vectors $p$ and $q$, two alternative ways can be used. One is to obtain $M^{-1}$ and $(M^{-1})^*$ explicitly and then compute $M^{-1}p$ and $(M^{-1})^*q$. The other way is to take $M^{-1}p$ and $(M^{-1})^*q$ as the solutions $x$ and $y$ to equations

$$Mx=p \quad \text{and} \quad M^*y=q \tag{25}$$

respectively. In many situations, the second way is more advantageous.

The major purpose of this paper is to apply the above algorithms to compute the smallest or the second smallest singular values of operators $L_c$, $L_d$, $L_c|S_1$, $L_d|S_1$, $L_c|S_2$ and $L_d|S_2$. This application is quite straightforward. For operators $L_c$, $L_c|S_1$ and $L_c|S_2$, the equations in (25) are actually Lyapunov equations

$$Ax+xA'=p \quad \text{and} \quad A'y+yA=q$$

respectively. For operators $L_d$, $L_d|S_1$ and $L_d|S_2$, the equations in (25) are actually discrete Lyapunov equations

$$AxA'-x=p \quad \text{and} \quad A'yA-y=q$$

respectively. Since the Lyapunov equations can be solved more easily if $A$ is a triangular matrix, the Schur form of $A$ instead of $A$ itself can be used in the computation. The robustness bounds will not be affected since they are invariant under unitary similarity transformations. When the algorithm is applied to compute the singular values of $L_c|S_1$, $L_c|S_2$, $L_d|S_1$ and $L_d|S_2$, in principle, one only needs to choose the initial trial vectors in the appropriate space $S_1$ or $S_2$, and then the successive trial vectors will stay in space $S_1$ or $S_2$ and converge to the desired results. It should be noted however that due to roundoff error, the successive trial vectors may in fact deviate out of the space $S_1$ or $S_2$ after a certain number of iterations. In this case, the algorithms will now converge to the singular values of $L_c$ or $L_d$ instead. To prevent this from happening, one should re-orient the trial vectors every few iterations to keep them in $S_1$ or $S_2$.

## 4. Numerical Examples

The following examples are computed using PC-MATLAB.

**Example 1**

This example uses the proposed algorithm to calculate the singular values of a 6×6 matrix with clustered singular values, repeated singular values, and very small singular values. The purpose is to show various properties of the algorithm. The matrix considered is

$$M=U\,\mathrm{diag}(1,0.9,0.9,0.2,10^{-10},10^{-12})V,$$

where $U$, $V$ are two randomly chosen 6×6 unitary matrices.

The machine epsilon of the computer used is about $10^{-16}$. The error tolerance for the convergence is set to be 0.01. Table 1 gives a list of the results obtained and the numbers of iterations required when a different number of singular values are requested.

**Table 1**: The results of Example 1

| No. of singular values requested | Singular values obtained | | | No. of iterations |
|---|---|---|---|---|
| $m=1$ | $9.9984\times10^{-1}$ | | | 24 |
| $m=2$ | $9.9980\times10^{-1}$ | $9.0000\times10^{-1}$ | | 19 |
| $m=3$ | 1.0000 | $9.0000\times10^{-1}$ | $9.0000\times10^{-1}$ | 3 |
| $m=4$ | 1.0000<br>$2.0000\times10^{-1}$ | $9.0000\times10^{-1}$ | $9.0000\times10^{-1}$ | 2 |
| $m=5$ | 1.0000<br>$2.0000\times10^{-1}$ | $9.0000\times10^{-1}$<br>$1.0000\times10^{-10}$ | $9.0000\times10^{-1}$ | 2 |

The following remarks are made regarding this example:

(i) The algorithm converges slowly in the case when $m=1$ and $m=2$. This is to be expected since $\frac{\sigma_2}{\sigma_1}$ is close to 1 and $\sigma_2=\sigma_3$. However, the convergence is greatly accelerated when more than two singular values are requested. This shows that the use of guard vectors may be helpful.

(ii) Repeated singular values and their multiplicities can be obtained with no difficulty. This is a reason why the proposed algorithm is more suitable in the application of stability robustness analysis than the Lanczos algorithms.

(iii) Small and large singular values can be obtained simultaneously as long as the ratio of the smallest singular value and the largest one is greater than the machine epsilon.

In the following, the proposed algorithm will be used to calculate the singular values required in the stability robustness analysis of some representative examples.

**Example 2**

The following is a stable matrix obtained from a linear quadratic optimal control design [9]:

$$A=\begin{bmatrix} -0.201 & 0.755 & 0.351 & -0.075 & 0.033 \\ -0.149 & -0.696 & -0.160 & 0.110 & -0.048 \\ 0.081 & 0.004 & -0.189 & -0.003 & 0.001 \\ -0.173 & 0.802 & 0.251 & -0.804 & 0.056 \\ 0.092 & -0.467 & -0.127 & 0.075 & -1.162 \end{bmatrix}.$$

To obtain bounds (9)-(11), one has to compute $\sigma_{24}(A\oplus A)$, $\underline{\sigma}(A\overline{\oplus}A)$ and $\underline{\sigma}(A\overline{\overline{\oplus}}A)$. It has been shown that $\sigma_{24}(A\oplus A)$ is the second smallest singular value of operator $L_c$, and $\underline{\sigma}(A\overline{\oplus}A)$ and $\underline{\sigma}(A\overline{\overline{\oplus}}A)$ are the smallest singular values of $L_c|S_1$ and $L_c|S_2$ respectively. When applying the proposed algorithm to $L_c$ for the two smallest singular values, we obtain 0.1716 and 0.3480 after 13 iterations. When applying the algorithm to $L_c$ for the three smallest singular values, we obtain 0.1716, 0.3480 and 0.3604 after 5 iterations. In both cases, $\sigma_{24}(A\oplus A)=0.3480$ is obtained. The reason for the reduction in the number of iterations when requesting three singular values is because the singular values 0.3480 and 0.3604 are close. By applying the algorithm to $L_c|S_1$ and $L_c|S_2$, $\underline{\sigma}(A\overline{\oplus}A)=0.1716$ and $\underline{\sigma}(A\overline{\overline{\oplus}}A)=0.3604$ are obtained. The number of iterations are 4 and 9 respectively.

The above results are the same as those obtained in [1], where the matrices $A\oplus A$, $A\overline{\oplus}A$ and $A\overline{\overline{\oplus}}A$ are constructed explicitly and their singular values are calculated by the QR transformation method.

$\underline{\sigma}(A)$ can be calculated in the usual way; the result is 0.1116.

Therefore the bounds (9)-(11) are given by:

$$\mu(A)\geq\min\{0.1116,\frac{1}{2}\times0.3480\}=0.1116,$$

$$\mu(A)\geq\frac{1}{2}\times0.1716=0.0858,$$

$$\mu(A)\geq\min\{0.1116,\frac{1}{2}\times0.3604\}=0.1116,$$

respectively. Since $\mu(A)\leq\underline{\sigma}(A)=0.1116$, it is concluded in this case that $\mu(A)=0.1116$.

**Example 3**

This example considers an industrial boiler system which is described by a 9-th order state space model:

$$\dot{x}=Ax+Bu$$

$$y=Cx,$$

where

$$A=\begin{bmatrix} -3.93 & -3.15\times10^{-3} & 0 & 0 & 0 \\ 3.68\times10^{2} & -3.05 & 3.03 & 0 & 0 \\ 2.74\times10^{1} & 7.87\times10^{-2} & -5.96\times10^{-2} & 0 & 0 \\ -6.47\times10^{-2} & -5.20\times10^{-5} & 0 & -2.55\times10^{-1} & -3.35\times10^{-6} \\ 3.85\times10^{3} & 1.73\times10^{1} & -1.28\times10^{1} & -1.26\times10^{4} & -2.91 \\ 2.24\times10^{4} & 1.80\times10^{1} & 0 & -3.56\times10^{1} & -1.04\times10^{-4} \\ 0 & 0 & 2.34\times10^{-3} & 0 & 0 \\ 0 & 0 & 0 & -1.27 & 1.00\times10^{-3} \\ -2.20 & -1.77\times10^{-3} & 0 & -8.44 & -1.11\times10^{-4} \end{bmatrix}$$

$$\begin{bmatrix} 4.03\times10^{-5} & 0 & 0 & 0 \\ -3.77\times10^{-3} & 0 & 0 & 0 \\ -2.81\times10^{-4} & 0 & 0 & 0 \\ 3.60\times10^{-7} & 6.33\times10^{-5} & 1.94\times10^{-4} & 0 \\ -1.05\times10^{-1} & 1.27\times10^{1} & 4.31\times10^{1} & 0 \\ -4.14\times10^{-1} & 9.00\times10^{1} & 5.69\times10^{1} & 0 \\ 2.22\times10^{-4} & -2.03\times10^{-1} & 0 & 0 \\ 7.86\times10^{-5} & 0 & -7.17\times10^{-2} & 0 \\ 1.38\times10^{-5} & 1.49\times10^{-3} & 6.02\times10^{-3} & -1.00\times10^{-10} \end{bmatrix},$$

$$B=\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1.56 & 0 \\ 0 & -5.13\times10^{-6} \\ 8.28 & -1.50 \\ 0 & 1.78 \\ 2.33 & 0 \\ 0 & -2.45\times10^{-2} \\ 0 & 2.94\times10^{-5} \end{bmatrix},\qquad C=\begin{bmatrix} 0&0&0&0&0&1&0&0&0 \\ 0&0&0&0&0&0&0&0&1 \end{bmatrix}.$$

The system is controlled by a static feedback

$$u=Ky$$

such that the closed loop state matrix $A_c=A+BKC$ is stable.

Table 2 gives the stability robustness bounds of $A_c$ for two different controller gain matrices, where the values of $\sigma_{n^2-1}(A_c\oplus A_c)$, $\underline{\sigma}(A_c\overline{\oplus}A_c)$ and $\underline{\sigma}(A_c\overline{\overline{\oplus}}A_c)$ are computed by using the algorithm developed in Section 3. The integers shown in the parentheses are the numbers of iterations required to obtain the result.

**Table 2**: The results of Example 3

| Control gains | | $K_1=\begin{bmatrix}0&0\\0&-0.01\end{bmatrix}$ | $K_2=\begin{bmatrix}0&0\\0&-1\end{bmatrix}$ |
|---|---|---|---|
| $\underline{\sigma}(A_c)$ | | $6.6276\times10^{-9}$ | $6.6688\times10^{-9}$ |
| $\sigma_{n^2-1}(A_c\oplus A_c)$ | | $4.5849\times10^{-9}$(4) | $9.6550\times10^{-9}$(5) |
| $\underline{\sigma}(A_c\overline{\oplus}A_c)$ | | $3.8907\times10^{-11}$(2) | $4.6203\times10^{-13}$(2) |
| $\underline{\sigma}(A_c\overline{\overline{\oplus}}A_c)$ | | $1.3979\times10^{-8}$(2) | $9.6550\times10^{-9}$(2) |
| Lower bound of $\mu(A_c)$ | Bound (9) | $2.2925\times10^{-9}$ | $4.8275\times10^{-9}$ |
| | Bound (10) | $1.9454\times10^{-11}$ | $2.3102\times10^{-13}$ |
| | Bound (11) | $6.6276\times10^{-9}$ | $4.8275\times10^{-9}$ |

It is observed that among bounds (9)-(11), bound (11) is the tightest and bound (10) is the most conservative. When $K_1$ is used, we have that $\mu(A_c)=6.6276\times10^{-9}$, since $\min\{\underline{\sigma}(A_c),\frac{1}{2}\underline{\sigma}(A_c\overline{\overline{\oplus}}A_c)\}=\underline{\sigma}(A_c)$; when $K_2$ is used we have that $4.8275\times10^{-9}\leq\mu(A_c)\leq6.6688\times10^{-9}$.

**Example 4:**

This example considers a 46th order state space model obtained from the design of a 3rd generation spacecraft which has 3 rigid body modes and 20 elastic body modes [10]. The system has 5 inputs and 5 outputs. It is stabilized by a static output feedback controller so that the closed loop poles closest to the imaginary axis are $-1.0026\times10^{-3}\pm j5.3242\times10^{-1}$. Denote the closed loop state matrix by $A_c$. It is desired to have an estimate of $\mu(A_c)$, the distance of $A_c$ to the set of unstable 46×46 real matrices. It is clear that $\mu(A_c)\leq1.0026\times10^{-3}$. On using the algorithm, it is determined after 7 iterations that $\underline{\sigma}(A_c\overline{\overline{\oplus}}A_c)=1.9903\times10^{-3}$. By using the usual QR method, it is determined that $\underline{\sigma}(A_c)=2.0252\times10^{-1}$. Hence, on using bound (11), an estimate of $\mu(A_c)$ is given by:

$$0.9952\times10^{-3}\leq\mu(A_c)\leq1.0026\times10^{-3}.$$

This estimate is very tight.

Note that the dimensions of $A\oplus A$, $A\overline{\oplus}A$ and $A\overline{\overline{\oplus}}A$ are 2116×2116, 1081×1081 and 1035×1035 respectively.

**Example 5**

This example is for the discrete time case. The following 7×7 stable matrix is considered in [1]:

$$A=\begin{bmatrix} -0.1373 & 0.2139 & -0.2831 & 0.2792 & -0.2177 & -0.1298 & 0.0666 \\ 0.0002 & -0.0163 & -0.0438 & -0.0657 & -0.0669 & -0.0473 & -0.0275 \\ 0.0469 & 0.0718 & 0.0896 & 0.0782 & 0.0493 & 0.0224 & 0.0074 \\ 0.0373 & 0.0712 & 0.1124 & 0.1292 & 0.1124 & 0.0712 & 0.0373 \\ 0.0074 & 0.0224 & 0.0498 & 0.0782 & 0.0896 & 0.0718 & 0.0469 \\ -0.0275 & -0.0473 & -0.0669 & 0.0657 & -0.0438 & -0.0163 & 0.0002 \\ -0.0666 & -0.1298 & -0.2177 & -0.2792 & -0.2831 & -0.2139 & -0.1373 \end{bmatrix}.$$

The following data is computed by the usual QR method:

$$\underline{\sigma}(A-I)=0.6705,\quad \underline{\sigma}(A+I)=0.6642,\quad \bar{\sigma}(A)=0.8306.$$

The following data is computed by using the proposed algorithm:

$$\sigma_{n^2-1}(A\bar{\otimes}A-I)=0.9375,\quad \underline{\sigma}(A\bar{\otimes}A-I)=0.7129,\quad \underline{\sigma}(A\bar{\bar{\otimes}}A-I)=0.9375.$$

These results are the same as those obtained in [1]. The stability robustness bounds are as follows:

bound (12): $\nu(A)\geq 0.4451$
bound (13): $\nu(A)\geq 0.3538$
bound (14): $\nu(A)\geq 0.4451$,

which implies that $0.4451\leq\nu(A)\leq 0.6705$.

## 5. Conclusions

A new method is developed in [1] for the robust stability analysis of linear time-invariant state space models with real perturbations. The new method is based on the properties of certain composite matrices and has several advantages over the existing methods in the literature: (i) it gives tighter bounds to the tolerable perturbations of a stable system, (ii) the computation involved is easy and numerically well-defined, provided the size of the system under analysis is moderate. However, the direct application of this method to large systems is limited due to dimensionality problems. To overcome this difficulty, it is desired to have an alternative way to compute the singular values of the composite matrices without explicitly constructing the actual matrices. This problem is addressed in this paper. It is shown that the composite matrices have the same singular values as certain operators in matrix spaces. An iterative procedure is then developed to compute the extreme singular values and corresponding singular vectors of an operator on a finite dimensional inner product space. An immediate application of the procedure enables us to analyze the stability robustness of large systems using the method of [1]. The stability robustness bounds of several non-trivial practical large systems are determined using the proposed algorithm. The results of these examples show not only that the algorithms are effective, but also that the new bounds obtained in [1] are tight.

## References


[1] L. Qiu, E.J. Davison, "A new method for the stability robustness determination of state space models with real perturbation", *Proc. IEEE CDC*, 1988.

[2] A. Graham, *Kronecker Products and Matrix Calculus with Applications*, Ellis Horwood Ltd., England, 1981.

[3] I.C. Gohberg, M.G. Krein, *Introduction to the Theory of Linear nonselfadjoint Operators*, American Mathematical Society, Providence, Rhode Island, 1969.

[4] A. Jennings, *Matrix Computation for Engineers and Scientists*, John Wiley & Sons, London, 1977.

[5] G. Stewart, *Introduction to Matrix Computations*, Academic Press, Inc., Orlando, Florida, 1973.

[6] C.C. Paige, "Bidiagonalization of matrices and solution of linear equations", *SIAM J. Numer. Anal.*, Vol. 11, No. 1, pp 197-209, 1974.

[7] G. Golub, F.T. Luk, M.L. Overton, "A block Lanczos method for computing the singular values and corresponding singular vectors of a matrix", *ACM Trans. Math. Software*, Vol. 7, No. 2, pp 149-169, 1981.

[8] J.K. Cullum, R.A. Willoughby, *Lanczos algorithms for large symmetric eigenvalue computations vol. 1 theory*, Birkhauser, Germany, 1985.

[9] R.V. Patel, M. Toda, "Quantitative measures of robustness for multivariable system", *Proc. Joint Automatic Control Conference*, Paper TD8-A, 1980.

[10] G.B. Sincarsin, "Laboratory Development of Control Techniques for 3rd Generation Spacecraft: Detailed Design", Dynacon Report, Daisy 9, DOC-CR-SP-84-010, March 1984.


## Appendix 1

**Proof of Lemma 5 and Lemma 6:**

Define $E_{ij}\in\mathbb{R}^{n\times n}$ to be a matrix with 1 in the $i,j$-th entry and 0 elsewhere.

Let $(i_1,i_2)$ be the $i$-th pair of integers in the sequence (3) and let

$$U_i=\begin{cases}E_{i_1,i_1} & \text{if } i_1=i_2,\\ \frac{\sqrt{2}}{2}(E_{i_1,i_2}+E_{i_2,i_1}) & \text{otherwise.}\end{cases}$$

Let $(r_1,r_2)$ be the $r$-th pair of integers in the sequence (5) and let

$$V_r=\frac{\sqrt{2}}{2}(E_{r_1,r_2}-E_{r_2,r_1}).$$

Let $u_i=\mathrm{Vec}(U_i)$, $i=1,2,\ldots,\frac{1}{2}n(n+1)$, and $v_i=\mathrm{Vec}(V_i)$, $i=1,2,\ldots,\frac{1}{2}n(n-1)$. Define

$$T_1=[u_1,u_2,\ldots,u_{\frac{1}{2}n(n+1)}]\in\mathbb{R}^{n^2\times\frac{1}{2}n(n+1)}$$

$$T_2=[v_1,v_2,\ldots,v_{\frac{1}{2}n(n-1)}]\in\mathbb{R}^{n^2\times\frac{1}{2}n(n-1)}.$$

It can be easily verified that $[T_1\ T_2]$ is an orthogonal matrix and

$$\mathrm{Vec}(X)=T_1\Phi(X),\quad \forall X\in S_1 \tag{A1}$$

$$\mathrm{Vec}(X)=T_2\Psi(X),\quad \forall X\in S_2. \tag{A2}$$

A lemma is needed to proceed.

**Lemma A1 [1]:** Let $A,B\in\mathbb{R}^{n\times n}$. Then

$$A\bar{\otimes}B=T_1'(A\otimes B)T_1 \tag{A3}$$

$$A\bar{\bar{\otimes}}B=T_2'(A\otimes B)T_2. \tag{A4}$$

Now we are ready to prove Lemma 5 and Lemma 6.

**Proof of Lemma 5:**

(a) $\mathrm{tr}(X'Y)=[\mathrm{Vec}(X)]'\mathrm{Vec}(Y)$ (by Lemma 1)
$=[\Phi(X)]'T_1'T_1\Phi(Y)$
$=[\Phi(X)]'\Phi(Y)$.

(b) $\Phi(AX+XA')=T_1'\mathrm{Vec}(AX+XA')$ (by (A1))
$=T_1'(A\otimes I+I\otimes A)\mathrm{Vec}(X)$ (by Lemma 1)
$=T_1'(A\otimes I+I\otimes A)T_1\Phi(X)$
$=(A\bar{\otimes}I+I\bar{\otimes}A)\Phi(X)$
$=(A\bar{\oplus}A)\Phi(X)$.

(c) $\Phi(AXA')=T_1'\mathrm{Vec}(AXA')$
$=T_1'(A\otimes A)\mathrm{Vec}(X)$
$=T_1'(A\otimes A)T_1\Phi(X)$
$=(A\bar{\otimes}A)\Phi(X)$.

**Proof of Lemma 6:**

Substitute $\Phi$ and $T_1$ in the proof of Lemma 5 by $\Psi$ and $T_2$ respectively, and use the same argument.