Computation of the Real Structured Singular Value via Polytopic Polynomials

B. C. Chang* and O. Ekdal†
Drexel University, Philadelphia, Pennsylvania 19104
and
H. H. Yeh‡ and S. S. Banda§
Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio 45433

For a large class of linear time-invariant systems with real parametric perturbations, the coefficient vector of the characteristic polynomial is a multilinear function of the real parameter vector. Based on this multilinear mapping relationship together with the recent developments for polytopic polynomials and parameter domain partition technique, an iterative algorithm for computing the real structured singular value is proposed. The algorithm requires neither frequency search nor Routh's array symbolic manipulations and allows the dependency among the elements of the parameter vector. Moreover, the number of the independent parameters in the parameter vector is not limited to three as is required by many existing structured singular-value computation algorithms.

I. Introduction

STABILITY robustness is an important issue in the analysis and design of control systems. Currently, there are two major approaches to stability-robustness analysis. One is the structured-singular-value (SSV)[1-7] or the multivariable-stability-margin (MSM)[8-12] approach and the other is the perturbed-characteristic-polynomials approach.[13-18] Several significant progresses have been made in both approaches. In this paper, an iterative algorithm of computing the real SSV and the real MSM is developed based on the existing results in both approaches.

For a large class of linear time-invariant systems with real parametric perturbations, the coefficient vector of the characteristic polynomial is a multilinear function of the real parameter vector. Based on this multilinear mapping together with the recent results by De Gaston and Safonov,[10] Sideris and Pena,[12] Bartlett et al.,[14] and Bouguerra et al.,[19] an algorithm for computing the real structured singular value is proposed. The algorithm requires neither frequency search nor Routh's array symbolic manipulations and allows the dependency among the elements of the parameter vector. Moreover, the number of the independent parameters in the parameter vector is not limited to three, as is required by many existing structured-singular-value computation algorithms.

According to Doyle and Safonov,[1-2,8-9] all of the plant uncertainties, structured or unstructured, unmodeled dynamics or parametric perturbations, can be described by the blocked diagram shown in Fig. 1. In Fig. 1, $\Delta(s)$ = block diag $\{\Delta_1(s),\Delta_2(s),\ldots,\Delta_m(s)\}$ and $M(s)$ is the nominal system that includes the nominal plant and the stabilizing controller. Doyle[1-2] and Safonov[8-10] defined the structured singular value (SSV) and the multivariable stability margin (MSM), respectively, based on the preceding perturbation structure and used them as analysis tools for robust stability. The SSV and the MSM are nonconservative scalar stability-margin measures for multivariable systems.

Algorithms[1-7] to compute the SSV are available only for those cases where the number of perturbation blocks are less than or equal to three. The computational problem for the cases with more than three perturbation blocks is still an unsolved problem.

One important special case of plant uncertainties is the real parametric perturbation. In this case the perturbation matrix $\Delta(s)$ is a real diagonal matrix. The SSV defined for this case is called the real SSV. Algorithms for computing the real SSV[6-7] are also only available for those cases where the number of independent perturbation parameters are less than or equal to three. The MSM defined for this case is called the real MSM. An iterative algorithm for computing the real MSM for real diagonal Δ was developed by De Gaston and Safonov[10] and generalized by Pena and Sideris.[11] There is no limitation on the number of perturbation parameters. However, this iterative algorithm is complicated since an extensive frequency search is required. Sideris and Pena[12] eliminated the need of frequency search by requiring the first column of the Routh's array to be positive. This approach requires symbolic manipulations. Besides, the first-column elements of the Routh's array usually are not multilinear functions of the parameter vector even if the original characteristic coefficients are. For those elements to be multilinear functions of the parameters, more dependent parameters need to be created which will cause unnecessary complexity.

In this paper, we assume that the perturbation matrix Δ in Fig. 1 is real diagonal, i.e., $\Delta = \text{diag}\{\delta_1,\delta_2,\ldots,\delta_m\}$ and the nominal system $M(s)$ is a rational matrix with real coefficients. If the parameters vary independently and $-1 \le \delta_i \le 1$, $i = 1,2,\ldots,m$, the parameter perturbation domain $\mathcal{D}$ can be described as a hypercube $\bar{\mathcal{D}}$ with 2^m vertices $(\pm 1,\ldots,\pm 1)$ in the m-dimensional real space. In general, the perturbation matrix Δ can be written as $\Delta = \text{diag}\{\delta_1 I_{m1},\delta_2 I_{m2},\ldots,\delta_r I_{mr}\}$ where I_{mi} is the identity matrix of order mi and $m1 + m2 + \cdots + mr = m$. That is, $\delta_1 = \delta_2 = \cdots = \delta_{m1} = \delta_1$, $\delta_{m1+1} = \delta_{m1+2} = \cdots = \delta_{m1+m2} = \delta_2,\ldots$,etc. In this case, the parameter perturbation domain $\mathcal{D}$ is an r-dimensional hyperrectangle inside the m-dimensional hypercube $\bar{\mathcal{D}}$. The system is said to be *robustly stable in* $\mathcal{D}$ if and only if it is stable for every parameter vector $\delta = [\delta_1\ \delta_2\ \cdots\ \delta_m]^T$ in $\mathcal{D}$. Throughout the paper, we may use "$\mathcal{D}$ is stable" to replace "the system is robustly stable in $\mathcal{D}$" whenever there is no confusion.

Received June 20, 1989; revision received Dec. 18, 1989.

*Associate Professor, Department of Mechanical Engineering and Mechanics. Member of AIAA.
†Graduate Student, Department of Mechanical Engineering and Mechanics.
‡Electronic Engineer.
§Aerospace Engineer. Associate Fellow AIAA.

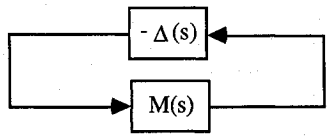

Fig. 1 Standard structure for a perturbed closed-loop system.

The *real multivariable stability margin* (real MSM) k_M is defined as the largest real constant k such that the closed-loop system remains robustly stable in $k\mathcal{D}$, where $k\mathcal{D}$ is the enlarged (or shrunk) parameter perturbation domain of $\mathcal{D}$, i.e.,

$$k\mathcal{D} := \{\delta: \delta = [\delta_1,..,\delta_1,\delta_2,..,\delta_2,...,\delta_r,..,\delta_r] \in \boldsymbol{R}^m \text{ and } |\delta_i| \le k, i = 1,2,...,r\} \quad (1)$$

The enlarged (or shrunk) hypercube of $\bar{\mathcal{D}}$, $k\bar{\mathcal{D}}$ is

$$k\bar{\mathcal{D}} := \{\delta: \delta \in \boldsymbol{R}^m \text{ and } |\delta_i| \le k, i = 1,2,...,m\}. \quad (2)$$

Recall that the real structured singular value (real SSV) μ is defined as

$$\mu := [\min\{k \mid \det[I + M(j\omega)\Delta] = 0 \text{ for some } \omega \text{ and } \Delta \in \boldsymbol{X}(k)\}]^{-1} \quad (3)$$

where

$$\boldsymbol{X}(k) = \{\Delta \mid \operatorname{diag}\{\delta_1 I_{m1},\delta_2 I_{m2},...,\delta_r I_{mr}\} \text{ with } |\delta_i| \le k, \text{ for all } i\} \quad (4)$$

That is, the real SSV μ is the reciprocal of the smallest k such that the system is unstable in $k\mathcal{D}$. It is easy to see that the relation between the real SSV μ and the real MSM k_M is

$$\mu = 1/k_M \quad (5)$$

As mentioned in the first paragraph, several significant results[13-18] have been obtained in the perturbed-characteristic-polynomials approach. Probably the most famous are the Kharitonov's theorems[13] that apply to the special case with a hyperrectangular perturbed region in the coefficient space. In this special case, the coefficients of the characteristic polynomial vary independently and the robust stability of the system can be easily determined by four bounding characteristic polynomials. Unfortunately, the Kharitonov's theorems cannot be applied to our problem since the coefficient variations of the characteristic polynomial are not independent.

Bartlett et al.[14] developed an important theorem that is applicable to the case when the coefficients of characteristic polynomial are linearly dependent. The theorem is now well-known as the Edge Theorem: For the set of characteristic polynomials inside a polytope $\mathcal{P}$ in the coefficient space, every polynomial $\mathcal{P}$ is stable if and only if all the exposed edges of $\mathcal{P}$ are stable. This simplifies the stability checking tremendously since checking the stability of exposed edges is much simpler than checking that of the full $\mathcal{P}$. The exposed-edge stability checking is done by sweeping t from 0 to 1 such that

$$t\alpha^i + (1-t)\alpha^j \quad (6)$$

are all stable for all vertices α^i and α^j of $\mathcal{P}$.

Bialas[16] and Fu and Barmish[15] reduced the checking of the exposed-edge sweep stability to a one-shot test. They showed that $t\alpha^i + (1-t)\alpha^j$ is stable for all $t \in [0,1]$ if and only if the real eigenvalues of $-H_i H_j^{-1}$ are all negative where α^i and α^j are assumed to be stable and H_i and H_j are the Hurwitz matrices for α^i and α^j respectively. Recently, a fast algorithm based on Chapellat and Bhattacharyya's Segment Lemma[20] was proposed by Bouguerra et al.[19] for checking the stability of the exposed edges. The computation in the algorithm mainly depends on the number of vertices instead of the edges and therefore reduces the computation burden due to the "combinatoric explosion."

There are no such celebrated properties in the parameter space as those in the coefficient space discovered by Kharitonov[13] and Bartlett et al.[14] The closed-loop system may be unstable inside $\bar{\mathcal{D}}$ although it is stable at all of the edges of the hypercube $\bar{\mathcal{D}}$.[21] So far, there is no easy way of checking robust stability in the parameter space.

For each parameter vector δ in the parameter perturbation domain $\mathcal{D}$, there is a corresponding characteristic polynomial, i.e., a coefficient vector α in the coefficient space. Let $\mathcal{I}(\mathcal{D})$ be the image of $\mathcal{D}$ in the coefficient space. The closed-loop system is robustly stable in $\mathcal{D}$ if and only if every characteristic polynomial in $\mathcal{I}(\mathcal{D})$ is stable. Although several significant results for robust stability have been obtained in the coefficient space, there is no efficient way to check robust stability for $\mathcal{I}(\mathcal{D})$ since $\mathcal{I}(\mathcal{D})$ usually is neither a Kharitonov's hyperrectangle[13] nor a polytope considered by Bartlett et al.[14]

Define the polytope $\mathcal{P}(\bar{\mathcal{D}})$ as the convex hull of the 2^m image points in the $(n+1)$-dimensional coefficient space mapped from the vertices of $\bar{\mathcal{D}}$, where n is the degree of the characteristic polynomial. If the mapping is multilinear, then the image of the edges of the hypercube $\bar{\mathcal{D}}$ will be the straight-line segments connecting the corresponding mapped vertices. The image of $\mathcal{D}$, $\mathcal{I}(\mathcal{D})$, is a subset of $\mathcal{I}(\bar{\mathcal{D}})$ and therefore is a subset of the polytope $\mathcal{P}(\bar{\mathcal{D}})$. Under the condition of the multilinear mapping, the stability of the edges of $\mathcal{P}(\bar{\mathcal{D}})$ will guarantee the robust stability in $\bar{\mathcal{D}}$. The multilinear mapping can be easily achieved by assuming that the nominal system $M(s)$ in Fig. 1 is a rational matrix with real coefficients.

Now, we have an easy way to check the sufficient condition for the robust stability in $\mathcal{D}$ by using its corresponding polytope $\mathcal{P}(\bar{\mathcal{D}})$. The sufficient condition is still not enough to determine the real MSM k_M. Any k such that the polytope $\mathcal{P}(k\bar{\mathcal{D}})$ remains stable, say k_L, can be served as a lower bound for k_M. However, there may exist some $k > k_L$ such that $k\mathcal{D}$ is stable although the corresponding polytope $\mathcal{P}(k\bar{\mathcal{D}})$ is unstable.

Any k that causes instability of $k\mathfrak{D}$ or $\mathcal{J}(k\bar{\mathfrak{D}})$ qualifies as an upper bound for k_M. To facilitate the description of the relations between the parameter space and the coefficient space, let the edges (vertices, resp.) of $\mathcal{P}(k\bar{\mathcal{D}})$ that are mapped from the edges (vertices, resp.) that are parallel to an axis of coordinates of $k\mathcal{D}$ be called the crucial edges (vertices, resp.) and those that are not crucial be called noncrucial edges (vertices, resp.). The noncrucial edges include two kinds of edges: supplemental edges and fictitious edges. The supplemental edges are the image of the edges of $k\bar{\mathcal{D}}$ that are not in $k\mathcal{D}$. The fictitious edges are the edges of $\mathcal{P}(k\bar{\mathcal{D}})$ that are not mapped from the edges of $k\bar{\mathcal{D}}$. The crucial edges are all in $\mathcal{I}(k\mathcal{D})$. Thus, some k that causes instability at the crucial edges of $\mathcal{P}(k\bar{\mathcal{D}})$, say k_U, may be used as an upper bound for k_M. If the lower and the upper bounds coincide or are close enough, we have the real MSM k_M and the real SSV μ. The objective of the iterative algorithm to be presented in the paper is to reduce the gap of the lower and the upper bounds. When the gap is smaller than the desired accuracy ε, i.e., $|k_U - k_L| \le \varepsilon$, we have the real MSM $k_M = k_L$ and the real SSV $\mu = 1/k_M$.

The paper is organized as follows. In Sec. II, the mapping properties from the parameter space to the coefficient space and the theories on which the iterative algorithm is developed will be demonstrated. The detailed iterative algorithm for computing the real multivariable stability margin is presented in Sec. III. In Sec. IV, two examples are used to illustrate our algorithm. Section V is the conclusion.

II. Multilinear Mapping and the Polytopic Polynomials

The robust stability of the perturbed system in Fig. 1 is determined by $\det[I + M(s)\Delta]$. There are two lemmas available for checking the robust stability. They are listed as follows.

Lemma 1

The closed-loop system is robustly stable in $\mathcal{D}$ if and only if $M(s)$ is asymptotically stable and

$$\det[I + M(j\omega)\Delta] \neq 0 \qquad \text{for all } \omega \text{ and all } \delta \text{ in } \mathcal{D} \tag{7}$$

Lemma 2

The closed-loop system is robustly stable in $\mathcal{D}$ if and only if all of the zeros of

$$\det[I + M(s)\Delta] \qquad \text{for all } \delta \text{ in } \mathcal{D} \tag{8}$$

are in the open left half of the s plane.

The existing computational algorithms for the SSV and the MSM are all developed based on Lemma 1 while the approach to be presented in the paper is based on Lemma 2.

Recall that $\Delta = \text{diag}\{\delta_1 I_{m1}, \delta_2 I_{m2}, \ldots, \delta_r I_{mr}\} = \text{diag}\{\delta_1, \delta_2, \ldots, \delta_m\}$, where I_{mi} is the identity matrix of order mi and $m1 + m2 + \cdots + mr = m$. That is, $\delta_1 = \delta_2 = \cdots = \delta_{m1} = \delta_1$, $\delta_{m1+1} = \delta_{m1+2} = \cdots = \delta_{m1+m2} = \delta_2, \ldots$, etc. The parameters $\delta_1, \delta_2, \ldots, \delta_r$ are assumed independent and $-1 \le \delta_i \le 1, i = 1,2,\ldots,r$. The parameter perturbation domain $\mathcal{D}$ is an r-dimensional hyperrectangle inside the m-dimensional hypercube $\bar{\mathcal{D}}$ with 2^m vertices $(\pm 1, \ldots, \pm 1)$ in the m-dimensional real space.

First of all, we will establish the relationship between the coefficients $\{\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_n\}$ of the characteristic polynomial

$$\alpha_0 s^n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} + \cdots + \alpha_n \tag{9}$$

and the perturbation parameters. It is interesting to note that if $M(s)$ is a rational matrix with real coefficients then the coefficient vector $\alpha = [\alpha_0 \ \alpha_1 \ \alpha_2 \ \cdots \ \alpha_n]^T$ is a multilinear function of the parameter vector $\delta = [\delta_1 \ \delta_2 \ \cdots \ \delta_m]^T$. That is, if we only allow one of the δ_i, say δ_j, to vary and keep the rest $(m-1)$ of the δ_i constant then α is a linear function of δ_j.

Theorem 1

Consider the perturbed system in Fig. 1 where $\Delta = \text{diag}\{\delta_1, \delta_2, \ldots, \delta_m\}$ and $M(s)$ is rational with real coefficients. The coefficients α_i, $i = 0,1,2,\ldots,n$, of the characteristic polynomial of the perturbed system are multilinear functions of δ_i.

Proof. See Appendix.

Remark

The coefficients α_i, $i = 0,1,2,\ldots,n$, of the characteristic polynomial of the perturbed system in general are not multilinear functions of δ_i unless $\mathcal{D} = \bar{\mathcal{D}}$.

The following lemma is a direct consequence of the multilinear mapping between δ and α.

Lemma 3

In the parameter space, draw a line segment with ending points E_1 and E_2 and the line is parallel to an axis of coordinates. The images of these two ending points are denoted by $\mathcal{I}(E_1)$ and $\mathcal{I}(E_2)$, respectively. The image of the line segment in the coefficient space is a straight-line segment that connects $\mathcal{I}(E_1)$ and $\mathcal{I}(E_2)$ if the mapping from the parameter space to the coefficient space is multilinear.

The parameter perturbation domain $\mathcal{D}$ is an r-dimensional hyperrectangle inside the m-dimensional hypercube $\bar{\mathcal{D}}$ with 2^m vertices whose edges are parallel to the axes of coordinates. $\mathcal{I}(\mathcal{D})$ and $\mathcal{I}(\bar{\mathcal{D}})$ are the images of $\mathcal{D}$ and $\bar{\mathcal{D}}$, respectively, in the coefficient space. The polytope $\mathcal{P}(\bar{\mathcal{D}})$ is the convex hull of the 2^m image points in the $(n+1)$-dimensional coefficient space mapped from the vertices of $\bar{\mathcal{D}}$. In the following, we will show that if the mapping is multilinear, the image of $\bar{\mathcal{D}}$, i.e., $\mathcal{I}(\bar{\mathcal{D}})$, is a subset of the polytope $\mathcal{P}(\bar{\mathcal{D}})$.

Theorem 2

$$\mathcal{I}(\bar{\mathcal{D}}) \subset \mathcal{P}(\bar{\mathcal{D}}) \tag{10}$$

Proof. See Appendix.

Theorem 3

If $\bar{\mathcal{D}}_1 \subset \bar{\mathcal{D}}_2$, then

$$\mathcal{I}(\bar{\mathcal{D}}_1) \subset \mathcal{I}(\bar{\mathcal{D}}_2) \tag{11}$$

and

$$\mathcal{P}(\bar{\mathcal{D}}_1) \subset \mathcal{P}(\bar{\mathcal{D}}_2) \tag{12}$$

Proof. See Appendix.

Theorem 4

The hypercube or hyperrectangle $\bar{\mathcal{D}}$ is cut into two equal subdomains $\bar{\mathcal{D}}_1$ and $\bar{\mathcal{D}}_2$ by a hyperplane that is orthogonal to an axis of coordinates. Then

$$\mathcal{I}(\bar{\mathcal{D}}) \subset \mathcal{P}(\bar{\mathcal{D}}_1) \cup \mathcal{P}(\bar{\mathcal{D}}_2) \subset \mathcal{P}(\bar{\mathcal{D}}) \tag{13}$$

Proof. See Appendix.

In the following, a simple case with two-dimensional parameter space and three-dimensional coefficient space will be used to illustrate the basic concept on which our algorithm is developed.

In Fig. 2a, the hypercube $\bar{\mathcal{D}}$ is a square with edges V_1V_2, V_2V_3, V_3V_4, and V_4V_1 parallel to either δ_1 axis or δ_2 axis and the perturbed parameter domain $\mathcal{D}$ is the straight-line segment connecting V_2 and V_4. In Fig. 2b, X_1, X_2, X_3, and X_4 are the images of the vertices V_1, V_2, V_3, and V_4 of $\bar{\mathcal{D}}$, respectively, in the three-dimensional coefficient space. The polytope $\mathcal{P}(\bar{\mathcal{D}})$, i.e., the convex hull of the four image points X_1, X_2, X_3, and X_4, is a pyramid with six edges. X_2 and X_4 are crucial vertices since they are in $\mathcal{I}(\mathcal{D})$. From Theorem 1, the four edges X_1X_2, X_2X_3, X_3X_4, and X_4X_1 are the images of V_1V_2, V_2V_3, V_3V_4, and V_4V_1, respectively, and therefore are supplemental edges. The other two edges of the polytope, X_1X_3 and X_4X_2, are fictitious edges that may not even be in the image of $\bar{\mathcal{D}}$.

The image of $\bar{\mathcal{D}}$ ($\mathcal{D}$ resp.), $\mathcal{I}(\bar{\mathcal{D}})$ [$\mathcal{I}(\mathcal{D})$ resp.], can be constructed as follows. Let V_a and V_b be the center points of the line segments V_1V_4 and V_2V_3, respectively. It is easy to see that the line segment drawn between V_a and V_b, i.e.,

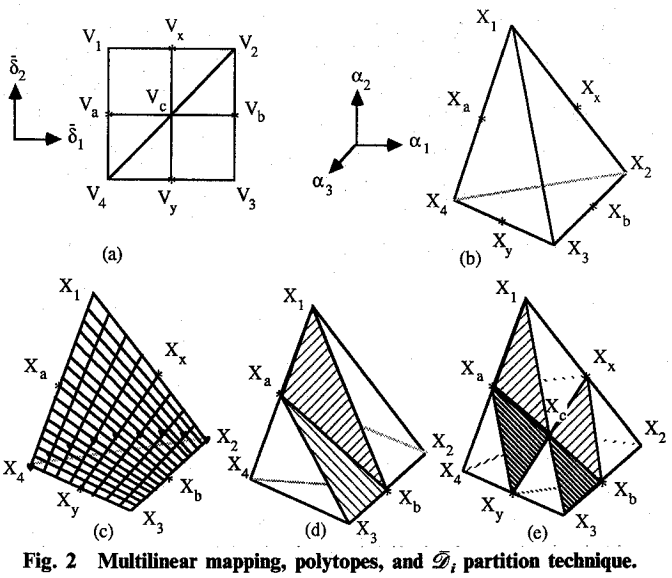

Fig. 2 Multilinear mapping, polytopes, and $\bar{\mathcal{D}}_i$ partition technique.

V_aV_b, is parallel to V_1V_2 and V_4V_3, and therefore parallel to the $\bar{\delta}_1$ axis. Since the mapping is multilinear, $\mathcal{I}(V_aV_b)$ will be the straight-line segment X_aX_b, where $X_a = \mathcal{I}(V_a)$ and $X_b = \mathcal{I}(V_b)$ are the center points of the line segments X_1X_4 and X_2X_3, respectively. Similarly, $\mathcal{I}(V_xV_y)$ will be the straight-line segment X_xX_y, where $X_x = \mathcal{I}(V_x)$ and $X_y = \mathcal{I}(V_y)$ are the center points of the line segments X_1X_2 and X_4X_3, respectively. It is easy to see that X_aX_b and X_xX_y are both in $\mathcal{I}(\bar{\mathcal{D}})$ and the intersection point of X_aX_b and X_xX_y, i.e., X_c, is in $\mathcal{J}(\mathbb{D})$. Repeating the above mapping, we can see that $\mathcal{J}(\bar{\mathbb{D}})$ and $\mathcal{J}(\mathbb{D})$ can be constructed as that shown in Fig. 2c. In Fig. 2c, $\mathcal{J}(\bar{\mathbb{D}})$ is a saddle-shaped surface and is inside the pyramid $X_1X_2X_3X_4$, i.e., $\mathcal{P}(\bar{\mathcal{D}})$. The ♥ curve in $\mathcal{I}(\bar{\mathcal{D}})$ is the image of $\mathcal{D}$.

Note that $\mathcal{I}(\mathcal{D}) \subset \mathcal{I}(\bar{\mathcal{D}}) \subset \mathcal{P}(\bar{\mathcal{D}})$. There is an easy way to check the stability of $\mathcal{P}(\bar{\mathcal{D}})$ and therefore a sufficient condition for the stability of $\mathcal{I}(\mathcal{D})$ can be determined without too much effort. That is, $\mathcal{I}(\mathcal{D})$ is stable if all of the six edges of $\mathcal{P}(\bar{\mathcal{D}})$: X_1X_2, X_2X_3, X_3X_4, X_4X_1, X_1X_3, and X_4X_2, are stable.[14] The stability of the line segment X_iX_j is determined by using the segment stability-checking algorithm proposed by Bouguerra et al.[19]

In the proposed algorithm for computing the stability margin k_M, we need to check whether a subdomain $\mathcal{D}_i$ is stable. Consider $\mathcal{D}_i$ as the line segment V_2V_4 in Fig. 2a and its corresponding polytope $\mathcal{P}(\bar{\mathcal{D}}_i)$ as shown in Fig. 2b. $\mathcal{D}_i$ is stable if $\mathcal{P}(\bar{\mathcal{D}}_i)$ is stable. If $\mathcal{P}(\bar{\mathcal{D}}_i)$ is unstable and the instability is caused by the crucial edges or vertices, then $\mathcal{D}_i$ is unstable. If $\mathcal{P}(\bar{\mathcal{D}}_i)$ is unstable but the instability is not caused by the crucial edges or vertices, then the information is not enough to determine the stability of $\mathcal{D}_i$. In this case, a partition technique should be used and repeated until the stability of $\mathcal{D}_i$ is determined. The partition technique is illustrated briefly as follows.

The hypercube $\bar{\mathcal{D}}_1$ (square $V_1V_2V_3V_4$) is partitioned equally into four hyperrectangles $\bar{\mathcal{D}}_{i1}$ (square $V_4V_aV_cV_y$), $\bar{\mathcal{D}}_{i2}$ (square $V_cV_xV_2V_b$), $\bar{\mathcal{D}}_{i3}$ (square $V_aV_1V_xV_c$), and $\bar{\mathcal{D}}_{i4}$ (square $V_yV_cV_bV_3$), by two hyperplanes (line V_aV_b and line V_xV_y) that are orthogonal to the axes δ_1 and δ_2. From Fig. 2, it is easy to see that $\mathcal{I}(\bar{\mathcal{D}}_1) \subset \mathcal{I}(\bar{\mathcal{D}}_{i1}) \cup \mathcal{I}(\bar{\mathcal{D}}_{i2}) \subset \mathcal{P}(\bar{\mathcal{D}}_{i1}) \cup \mathcal{P}(\bar{\mathcal{D}}_{i2})$ where the polytopes $\mathcal{P}(\bar{\mathcal{D}}_{i1})$ and $\mathcal{P}(\bar{\mathcal{D}}_{i2})$ are the pyramids $X_4X_aX_cX_y$ and $X_cX_xX_2X_b$, respectively. If both $\mathcal{P}(\bar{\mathcal{D}}_{i1})$ and $\mathcal{P}(\bar{\mathcal{D}}_{i2})$ are stable, then $\mathcal{D}_i$ is stable. If either $\mathcal{P}(\bar{\mathcal{D}}_{i1})$ or $\mathcal{P}(\bar{\mathcal{D}}_{i2})$ is unstable and the instability is caused by the crucial edges or vertices (note that X_4, X_2, and X_c are crucial vertices), then $\mathbb{D}_i$ is unstable. If $\mathcal{P}(\bar{\mathbb{D}}_{i1})$ or $\mathcal{P}(\bar{\mathbb{D}}_{i2})$ is unstable but the instability is not caused by the crucial vertices or edges, then the $\bar{\mathbb{D}}_{ij}$ corresponding to the unstable $\mathcal{P}(\bar{\mathbb{D}}_{ij})$ should be partitioned further and the partition process is continued until the stability of $\mathbb{D}_i$ is determined.

From Fig. 3, we can see that the perturbed parameter domain $\mathbb{D}_i$ (line V_4V_2) is inside the shaded hyperrectangles (squares), and as the number of the shaded hyperrectangles becomes very large the image of $\mathbb{D}_i$ will be close to the union of the polytopes corresponding to the shaded hyperrectangles. In our algorithm, the number of partitioning usually is small since the partition process is terminated whenever all $\mathcal{P}(\bar{\mathcal{D}}_{ij})$s are stable or some crucial vertex or edge is unstable. The partition process needs to continue only if some $\mathcal{P}(\bar{\mathcal{D}}_{ij})$s are unstable and the instability is not caused by the crucial vertices or edges. Besides, only the $\bar{\mathcal{D}}_{ij}$s related to the unstable $\mathcal{P}(\bar{\mathcal{D}}_{ij})$s need to be partitioned further.

In the next section, a detailed algorithm based on the preceding theorems for computing the real MSM k_M and the real SSV μ will be proposed.

III. Algorithm for the Real Structured Singular Value

The algorithm is developed based on the multilinear mapping between the parameter and the coefficient spaces. Some recent results by De Gaston and Safonov,[10] Sideris and Pena,[12] Bartlett et al.,[14] and Bouguerra et al.[19] are used in the algorithm. The iterative algorithm is designed to find the largest positive real number k, i.e., the real MSM k_M, such that the system in Fig. 1 remains robustly stable for every possible perturbation in $k_M\mathcal{D}$. Then the real SSV μ can be obtained from $\mu = 1/k_M$.

Recall that $\bar{\mathcal{D}}$ is the hypercube with 2^m vertices $(\pm 1,...,\pm 1)$ in the m-dimensional real space and the parameter perturbation domain $\mathcal{D}$ is an r-dimensional hyperrectangle inside the m-dimensional hypercube $\bar{\mathcal{D}}$. In Fig. 1, the nominal system $M(s)$ is fixed and the shape of the perturbation domain $\mathcal{D}$ is also fixed. In the computation of stability margin, we will just enlarge or shrink the perturbation domain $\mathcal{D}$ by a factor k until k is the largest positive real number such that the system in Fig. 1 remains robustly stable for every possible perturbation in $k\mathcal{D}$. Recall that "$k\mathcal{D}$ is stable" means "the system in Fig. 1 remains robustly stable for every possible perturbation in $k\mathcal{D}$" and "$\mathcal{I}(k\mathcal{D})$ [resp. $\mathcal{P}(\bar{\mathcal{D}})$] is stable" means "every characteristic polynomial in $\mathcal{I}(k\mathcal{D})$ [resp. $\mathcal{P}(\bar{\mathcal{D}})$] is stable."

Since $\mathcal{I}(k\mathcal{D})$ is a coefficient-space image of $k\mathcal{D}$, $k\mathcal{D}$ is stable if and only if $\mathcal{I}(k\mathcal{D})$ is stable. From Theorems 2 and 3, we have $\mathcal{I}(k\mathcal{D}) \subset \mathcal{I}(k\bar{\mathcal{D}}) \subset \mathcal{P}(k\bar{\mathcal{D}})$ which implies that if $\mathcal{P}(k\bar{\mathcal{D}})$ is stable then $\mathcal{I}(k\mathcal{D})$ is stable and therefore so is $k\mathcal{D}$. Hence, any k such that the polytope $\mathcal{P}(k\mathcal{D})$ remains stable, say k_L, can be served as a lower bound for k_M. The stability of the polytope $\mathcal{P}(k\mathcal{D})$ can be easily checked by using the recent results developed by Bartlett et al.[14] and Bouguerra et al.[19]

Any k that causes instability of $k\mathcal{D}$ or $\mathcal{I}(k\mathcal{D})$ qualifies as an upper bound for k_M. In the proposed algorithm, some k that causes instability at the crucial vertices or edges of the polytopes corresponding to $k\bar{\mathcal{D}}$ or its subdomains, say k_{Ui}, will be used as an upper bound for k_M. If the lower and the upper bounds coincide or are close enough, we have the stability margin k_M. Otherwise, the iterative algorithm needs to continue until the gap of the lower and the upper bounds is small enough.

In an iteration loop of the algorithm, for a given k we construct a polytope $\mathcal{P}(k\bar{\mathcal{D}})$. If $\mathcal{P}(k\bar{\mathcal{D}})$ is stable, k_L is updated by k. In the case of unstable $\mathcal{P}(k\bar{\mathcal{D}})$, the instability may be caused by the crucial or the noncrucial edges. If it is caused by the crucial vertices or edges, the upper bound k_U shall be updated by k. In case that the instability is caused by the noncrucial edges, no information can be used to update k_L or k_U. In this situation, the perturbation domain-partition technique[10,12] is employed.

The domain partition procedure is described as follows. Referring to Fig. 4, the perturbation domain $k\mathcal{D}$ is the line segment V_aV_d and $k\bar{\mathcal{D}}$ is the hypercube (square) that encloses $k\mathcal{D}$. Assume $k_L\mathcal{D}$ (line segment V_bV_c) is stable and $k\bar{\mathcal{D}}$ is unstable and the instability is caused by noncrucial vertices or edges. To determine the stability of $k\mathcal{D}$, the following partition technique is used. The domain $k\mathcal{D}$ is partitioned into three parts: $k_L\mathcal{D}$ (line segment V_bV_c), $\mathcal{D}_1$ (line segment V_aV_b), and $\mathcal{D}_2$ (line segment V_cV_d). Enclose $\mathcal{D}_1$ and $\mathcal{D}_2$ by

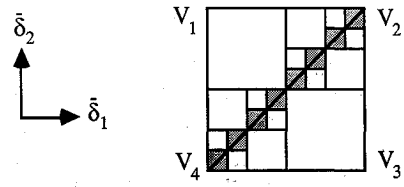

Fig. 3 Further partitioning in the parameter space.

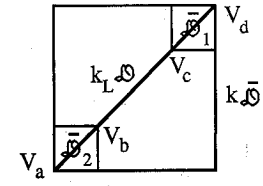

Fig. 4 Partitioning in the parameter space.

hyperrectangles $\bar{\mathscr{D}}_1$ and $\bar{\mathscr{D}}_2$, respectively. Now, there are three possibilities: 1) Both $\mathscr{P}(\bar{\mathscr{D}}_1)$ and $\mathscr{P}(\bar{\mathscr{D}}_2)$ are stable, then $k\mathscr{D}$ is stable and the lower bound k_L shall be updated by k. 2) Some crucial vertex or edge of $\mathscr{P}(\bar{\mathscr{D}}_1)$ or $\mathscr{P}(\bar{\mathscr{D}}_2)$ is unstable, then $k\mathscr{D}$ is unstable and the upper bound k_U shall be updated by k. 3) Either or both of $\mathscr{P}(\bar{\mathscr{D}}_1)$ and $\mathscr{P}(\bar{\mathscr{D}}_2)$ is unstable but the instability is not caused by crucial vertices or edges, then no information can be used to update k_L or k_U and therefore the $\bar{\mathscr{D}}_1$ with unstable $\mathscr{P}(\bar{\mathscr{D}}_i)$ shall be partitioned further until all polytopes that contain parts of $\mathscr{I}(k\mathscr{D})$ are stable (update the lower bound k_L by k) or some crucial vertex or edge is unstable (update the upper bound k_U by k). The iterative procedure is repeated with a new $k=(k_U+k_L)/2$ or $k=2k_L$ (if k_U is not available) until (k_U-k_L) is negligible and then the stability margin is $k_M=k_L$.

The algorithm for computing the real MSM k_M and the real SSV μ is listed as follows.

Algorithm

1) Initialize both the lower and the upper bounds $k_L=k_U=0$. Set the partition indicator $p=0$. Set ε for the accuracy of k_M. Set initial trial value at $k=1$.

2) Check the stability of the corresponding polytope $\mathscr{P}(k\bar{\mathscr{D}})$. If it is stable, go to step 3. Otherwise, go to step 5.

3) Update k_L by k and check k_U. If $k_U=0$, update k by $2k_L$ and go to step 2. If $k_U\neq 0$ (it must have been updated and must be $k_U>k_L$), go to step 4.

4) Set $k=(k_U+k_L)/2$. If $p=0$, then go to step 2. Otherwise go to step 5.

5) If the instability of the polytope $\mathscr{P}(k\bar{\mathscr{D}})$ (or the polytopes of the relevant subhyperrectangles as resulted from the partitioning in step 7) is due to crucial vertices or crucial edges ($k\mathscr{D}$ is unstable), go to step 8. Otherwise, go to step 6.

6) Partition the hyperrectangles with unstable polytopes by the hyperplanes orthogonal to the axes of coordinates. Only the subhyperrectangles that enclose $(k\mathscr{D}-k_L\mathscr{D})$ (we call these the relevant subhyperrectangles) need to be considered.

7) If the polytopes of the relevant subhyperrectangles under consideration are all stable, set $p=1$ and go to step 3. Otherwise, go to step 5.

8) $k\mathscr{D}$ is unstable. Update k_U by k and check (k_U-k_L). If $(k_U-k_L)>\varepsilon$, go to step 4. Otherwise, set $k_M=k_L$ and stop.

In the algorithm, ε, the tolerable gap between the lower bound k_L and the upper bound k_U, is a small number preset to determine the computation accuracy. The algorithm is employed to determine the stability margin k_M by narrowing down the gap between the lower bound k_L and the upper bound k_U. The perturbation domain-partitioning loop is described in step 6. Only the hyperrectangles enclosing $(k\mathscr{D}-k_L\mathscr{D})$ and corresponding to unstable polytopes whose crucial vertices and edges are stable need to be partitioned further and the domain-partitioning loop is terminated whenever all of the polytopes under consideration are stable or any crucial vertex or edge is unstable. The set $(k\mathscr{D}-k_L\mathscr{D})$ consists of all of the points in $k\mathscr{D}$, which is not in $k_L\mathscr{D}$. Theoretically, the domain-partitioning loop should be terminated in finite steps unless the unstable region is of measure zero (e.g., a singular point, the area of a singular unstable region is zero) inside $k\mathscr{D}$. However, in practical computation a preset counter can be placed in the loop to avoid the loop from repeating too many times for a given k. The partition indicator p initially is set at zero and is set to be one in step 7 whenever $k\mathscr{D}$ is determined to be stable after partitioning. If a $k_L\mathscr{D}$ is determined to be stable after partitioning, then the polytope $\mathscr{P}(k\bar{\mathscr{D}})$ is unstable for $k>k_L$. Therefore, in step 4 after setting a new k, we go to step 5 instead of 2 if $p\neq 0$.

IV. Examples

In this section, two examples are used to illustrate our algorithm. The first example is from De Gaston and Safonov[10] in which $\mathscr{D}=\bar{\mathscr{D}}$, i.e., the elements in the parameter vector δ are independent and the other example has dependency among the entries of δ. In the following, the vertices of the hypercube and their image points in the coefficient space are denoted by V_i and X_i, respectively. The line segment between X_i and X_j is represented by X_iX_j. Note that both V_i and X_i are functions of k.

Example 1

De Gaston and Safonov[10] developed an algorithm for computing the stability margin based on the multilinear mapping between the parameter space and the complex plane.[22] They used an example to illustrate their algorithm. We will use exactly the same example to illustrate our algorithm and then the solutions of both approaches can be compared.

The system to be considered is shown in Fig. 5 where the plant $P(s)$ and the controller $C(s)$ are described by

$$P(s)=\frac{\delta_{1a}}{s(s+\delta_{2a})(s+\delta_{3a})}, \qquad C(s)=\frac{s+z_1}{s+p_1}$$

respectively. The parameters in the preceding expressions are given by

$$z_1=2 \text{ rad/s} \qquad p_1=10 \text{ rad/s}$$

$$\delta_{1a}=\delta_{1o}(1+\delta_1) \qquad \delta_{1o}=800 \qquad |\delta_1|\le 0.1$$

$$\delta_{2a}=\delta_{2o}+\delta_2 \qquad \delta_{2o}=4 \text{ rad/s} \qquad |\delta_2|\le 0.2 \text{ rad/s}$$

$$\delta_{3a}=\delta_{3o}+\delta_3 \qquad \delta_{3o}=6 \text{ rad/s} \qquad |\delta_3|\le 0.3 \text{ rad/s}$$

The perturbation part of the perturbed closed-loop system can be pulled out and represented by a diagonal matrix and therefore the perturbed closed-loop system in Fig. 5 can be restructured as that in Fig. 1 where $\Delta=\text{diag}\{\delta_1,\delta_2,\delta_3\}$ and the nominal closed-loop system $M(s)$ has a realization (A,B,C) as follows:

$$A=\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -10 & -800 & 3200 \\ 1 & 0 & -4 & 0 \\ 0 & 0 & 1 & -6 \end{bmatrix}, \quad B=\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -800 \\ -1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$C=\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The vertices of the perturbation domain $k\mathscr{D}$ in the three-dimensional parameter space are numbered as shown in Fig. 6.

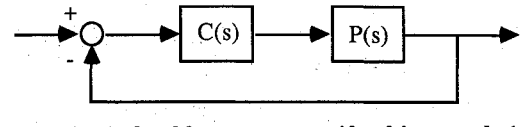

Fig. 5 A closed-loop system considered in example 1.

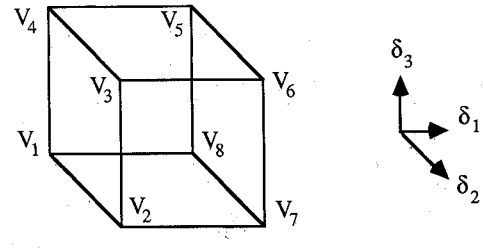

Fig. 6 Perturbation domain of example 1.

Recall that the vertices of $k\mathscr{D}$ are denoted by V_i. The vertices of $k\mathscr{D}$ with $k=1$ are

$$V_1=(-0.1,-0.2,-0.3), \qquad V_2=(-0.1,0.2,0.3)$$

$$V_3=(-0.1,0.2,0.3)$$

$$V_4=(-0.1,-0.2,0.3), \qquad V_5=(0.1,-0.2,0.3)$$

$$V_6=(0.1,0.2,0.3)$$

$$V_7=(0.1,0.2,-0.3), \qquad V_8=(0.1,-0.2,-0.3)$$

Since $M(s)$ is a rational matrix with real coefficients, the coefficients of the characteristic polynomial are multilinear functions of the parameters δ_1, δ_2, and δ_3. The characteristic polynomial

$$\alpha_0 s^4+\alpha_1 s^3+\alpha_2 s^2+\alpha_3 s+\alpha_4$$

can be obtained from $\det[sI-(A-B\Delta C)]$ and then

$$\alpha_0=1$$

$$\alpha_1=20+\delta_2+\delta_3$$

$$\alpha_2=124+16\delta_2+14\delta_3+\delta_2\delta_3$$

$$\alpha_3=1040+800\delta_1+60\delta_2+40\delta_3+10\delta_2\delta_3$$

$$\alpha_4=1600+1600\delta_1$$

As we expect, the coefficient vector $\alpha=[\alpha_0\ \alpha_1\ \alpha_2\ \alpha_3\ \alpha_4]^T$ is a multilinear function of the parameter vector $\delta=[\delta_1\ \delta_2\ \delta_3]^T$.

The objective is to compute the stability margin k_M that is the largest k such that $k\mathscr{D}$ is stable, i.e., the closed-loop system remains robustly stable in $k\mathscr{D}$. For each k, we map the eight vertices of $k\mathscr{D}$ into the coefficient space as X_i, $i=1,2,\ldots,8$, from which the polytope $\mathscr{P}(k\mathscr{D})$ can be constructed. It is easy to see that the polytope $\mathscr{P}(k\mathscr{D})$ has 12 crucial edges and 16 fictitious edges. If these 28 edges are all stable, i.e., $\mathscr{P}(k\mathscr{D})$ is stable, then $k\mathscr{D}$ is stable. If any of the crucial edges is unstable, then so is $k\mathscr{D}$. Note that unstable $\mathscr{P}(k\mathscr{D})$ does not imply unstable $k\mathscr{D}$, since the instability may be only caused by some fictitious edges that are not in $\mathscr{I}(k\mathscr{D})$.

Let's go through the algorithm listed in the previous section. First of all, we initialize both the lower and the upper bounds $k_L=k_U=0$ and set the accuracy $\varepsilon=10^{-6}$. For $k=1$, $\mathscr{P}(k\mathscr{D})$ is stable and we have $k_L=1$ and update k to 2. For $k=2$, again $\mathscr{P}(k\mathscr{D})$ is stable and therefore we update k_L to 2 and k to 4, respectively. For $k=4$, $\mathscr{P}(k\mathscr{D})$ is unstable and the instability occurs at X_8, which is crucial, and k_U is updated to 4.

Now, we are in the loop of bisection to narrow down the gap between k_L and k_U to be less than ε. After 21 iterations, we have $k_L=3.417395$ and $k_U=3.417396$ such that $\mathscr{P}(k_L\mathscr{D})$ is stable while $\mathscr{P}(k_U\mathscr{D})$ is unstable and the instability occurs at X_8. Therefore, we have the real MSM $k_M=3.417395$.

From the preceding computation, we find that the no. 8 vertex of the perturbation domain is the critical point. Thus, we can check our solution at V_8 of the parameter space. When $k=k_M=3.417395$, the closed-loop system at V_8 is stable since its characteristic values are

$$\begin{aligned}&-16.3521921088081\\&-0.00000016598452 \pm 8.2282006559442I\\&-1.93911005922287\end{aligned}$$

If we increase k a little bit to $k=k_U=3.417396$, the closed-loop system at V_8 becomes unstable since its characteristic values are

$$\begin{aligned}&-16.352192213492\\&0.00000014646442 \pm 8.228200893344I\\&-1.9391100794366\end{aligned}$$

De Gaston and Safonov[10] obtain a margin $k=3.44$ that is close to our result. However, when $k=3.44$, the closed-loop system at V_8 is unstable since its characteristic values are

$$\begin{aligned}&-16.354555659044\\&0.00706068635988 \pm 8.2335635562319I\\&-1.9395657136757\end{aligned}$$

That means 3.44 could not be a correct margin. One of the major reasons that De Gaston and Safonov's algorithm[10] is complicated is that it requires frequency ω sweep. For the same example, they found that at $\omega=8.22$ rad/s, $\det[I+kM(j\omega)\Delta]$ is approximately equal to zero at V_8 when $k=3.44$. To have an accuracy of 10^{-6} for the stability margin, they need a more accurate ω, say $\omega=8.2282$ rad/s, which needs extensive frequency search. Our approach does not require frequency ω search.

The real SSVμ for the system is $\mu=1/k_M=1/3.417395=0.29262055$.

Example 2

Assume that the nominal system $M(s)$ in Fig. 1 has a state-space representation (A,B,C) as

$$A=\begin{bmatrix}-2.7 & -2 & -1.5 & -0.5\\ -1.5 & -4 & -1.5 & -1.5\\ -0.2 & 0 & -3 & 0\\ 1.5 & 2 & 3.5 & -0.7\end{bmatrix}, \qquad B=\begin{bmatrix}1 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 1\\ 0 & 1 & 0\end{bmatrix}$$

$$C=\begin{bmatrix}-0.3 & 0 & 0 & 0\\ 0 & 0 & 0 & -0.3\\ -0.3 & 0 & 0 & 0\end{bmatrix}$$

and the perturbation matrix Δ in Fig. 1 is given by

$$\Delta=\operatorname{diag}\{\bar{\delta}_1,\bar{\delta}_2,\bar{\delta}_3\}$$

where

$$\bar{\delta}_1=\delta_1, \qquad \bar{\delta}_2=\bar{\delta}_3=\delta_2$$

$$-1\le\delta_1\le 1, \qquad -1\le\delta_2\le 1$$

The set of the characteristic polynomials of the perturbed system can be described by

$$\alpha_0 s^4+\alpha_1 s^3+\alpha_2 s^2+\alpha_3 s+\alpha_4$$

where

$$\alpha_0=1$$

$$\alpha_1=10.4-0.3\delta_1-0.3\delta_2$$

$$\alpha_2=38.14-2.31\delta_1-2.91\delta_2+0.45\delta_3+0.09\delta_1\delta_2$$

$$\alpha_3=58.12-5.97\delta_1-8.28\delta_2+1.74\delta_3+0.63\delta_1\delta_2-0.135\delta_2\delta_3$$

$$\alpha_4=31.36-5.22\delta_1-6.84\delta_2+0.48\delta_3+1.08\delta_1\delta_2-0.27\delta_2\delta_3$$

The parameter perturbation domain is shown in Fig. 7. The shaded area $V_1V_8V_6V_3$ is the perturbation domain $\mathscr{D}$ in which $\bar{\delta}_1=\delta_1$, $\bar{\delta}_2=\bar{\delta}_3=\delta_2$ and the hypercube $V_1V_2V_3V_4V_5V_6V_7V_8$ is $\bar{\mathscr{D}}$. In the following, we will use the algorithm in Sec. III to compute k_M. Initially, $k_L=k_U=0$ and ε is set as 10^{-4}. $\mathscr{P}(k\bar{\mathscr{D}})$ is stable for $k=1$ and therefore k_L is updated to

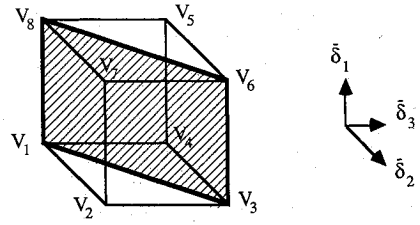

Fig. 7 Perturbation domain of example 2.

1. For $k = 5$, the polytope $\mathcal{P}(k\bar{\mathcal{D}})$ is unstable and the instability is caused by the crucial vertices X_3, X_6 and by the noncrucial vertex X_7.

Therefore k_U is updated to 5. As a next step, k is chosen as $(1+5)/2 = 3$. $\mathcal{P}(k\bar{\mathcal{D}})$ is stable for $k = 3$ and therefore k_L is updated to 3. Following the iteration steps, $k = (3+5)/2 = 4$, we find that the polytope $\mathcal{P}(k\bar{\mathcal{D}})$ is unstable for the value of $k = 4$ and the instability is caused by the crucial vertex X_6. Therefore k_U is updated to 4. Iterating k between 3 and 4, as it is suggested in steps 2–8, we find that the polytope $\mathcal{P}(k\bar{\mathcal{D}})$ is unstable for the value of $k = 3.6297$ and the instability is caused by the crucial vertex X_6. Hence, $k_U = 3.6297$. $\mathcal{P}(k\bar{\mathcal{D}})$ is stable at $k = 3.6296$ and we have $k_L = 3.6296$. Now, $|k_U - k_L| \leq \varepsilon$ and therefore we have the real MSM $k_M = 3.6296$. The real SSV is $\mu = 1/k_M = 1/3.6296 = 0.2755$.

V. Conclusions

The proposed real structured singular-value computation is developed based on the multilinear mapping, the edge theorem, the segment lemma and its extension, and the perturbation domain partition technique. The computational burden due to the combinatoric explosion of the number of edges has been reduced a lot by using the segment lemma and its extension. The convergence usually is not a problem unless a singular unstable region is inside or very close to the perturbation domain, which hardly happens in the real world. If that happens, a preset counter can be used to prevent the loop from repeating too many times for the same k. The drawback of leaving the loop prematurely is that the upper bound may not be tight. However, the largest computable lower bound k_L can still be found. Although the algorithm requires no frequency search, the frequency at which the system becomes unstable can be computed from the real structured singular value and the critical point at which the stable perturbation domain touches the unstable region. This critical point can be evaluated by using the segment lemma and its extension.

Appendix

Proof of Theorem 1

Without loss of generality, we will show that α_i, $i = 0,1,2,...,n$, are linear (affine) functions of δ_1, while $\delta_2,...,\delta_m$ remain unchanged. Let $m_{ij}(s)$ be the i-j entry of the matrix $M(s)$. Then

$$\det[I + M(s)\Delta] = \begin{vmatrix} 1+m_{11}\delta_1 & m_{12}\delta_2 & \cdots & m_{1m}\delta_m \\ m_{21}\delta_1 & 1+m_{22}\delta_2 & \cdots & m_{2m}\delta_m \\ \vdots & \vdots & & \vdots \\ m_{m1}\delta_1 & m_{m2}\delta_2 & \cdots & 1+m_{mm}\delta_m \end{vmatrix}$$

$$= \begin{vmatrix} 1 & m_{12}\delta_2 & \cdots & m_{1m}\delta_m \\ 0 & 1+m_{22}\delta_2 & \cdots & m_{2m}\delta_m \\ \vdots & \vdots & & \vdots \\ 0 & m_{m2}\delta_2 & \cdots & 1+m_{mm}\delta_m \end{vmatrix}$$

$$+\delta_1 \begin{vmatrix} m_{11} & m_{12}\delta_2 & \cdots & m_{1m}\delta_m \\ m_{21} & 1+m_{22}\delta_2 & \cdots & m_{2m}\delta_m \\ \vdots & \vdots & & \vdots \\ m_{m1} & m_{m2}\delta_2 & \cdots & 1+m_{mm}\delta_m \end{vmatrix} := \hat{m}_1(s) + \delta_1 \hat{m}_2(s)$$

Here $\hat{m}_1(s)$ and $\hat{m}_2(s)$ are rational functions with real coefficients. Now, we can write

$$\hat{m}_1(s) + \delta_1 \hat{m}_2(s) = \frac{n_1(s)}{d_c(s)d_1(s)} + \delta_1 \frac{n_2(s)}{d_c(s)d_2(s)}$$

$$= \frac{n_1(s)d_2(s) + \delta_1 n_2(s)d_1(s)}{d_c(s)d_1(s)d_2(s)}$$

where the $\{n_1(s), d_c(s)d_1(s)\}$ and $\{n_2(s), d_c(s)d_2(s)\}$ are coprime polynomial pairs, and $d_c(s)$ is the greatest common factor of the denominator polynomials of $\hat{m}_1(s)$ and $\hat{m}_2(s)$. The polynomials $n_1(s)d_2(s)$ and $n_2(s)d_1(s)$ can be represented as

$$a_0 s^n + a_1 s^{n-1} + a_2 s^{n-2} + \cdots + a_n$$

and

$$b_0 s^n + b_1 s^{n-1} + b_2 s^{n-2} + \cdots + b_n$$

respectively. Therefore, the characteristic polynomial is

$$\alpha_0 s^n + \alpha_1 s^{n-1} + \alpha_2 s^{n-2} + \cdots + \alpha_n$$

$$= (a_0 + \delta_1 b_0)s^n + (a_1 + \delta_1 b_1)s^{n-1} + (a_2 + \delta_1 b_2)s^{n-2}$$

$$+ \cdots + (a_n + \delta_1 b_n)$$

i.e.,

$$\alpha_i = a_i + \delta_1 b_i, \qquad i = 0,1,2,...,n$$

Proof of Theorem 2

Recall that $\bar{\mathcal{D}} = \{\delta: -1 \leq \delta_i \leq 1,\ i = 1,2,...,m\}$. To complete the proof, we need to show that for any $\delta^* \in \bar{\mathcal{D}}$, the image of δ^* is inside the polytope $\mathcal{P}(\bar{\mathcal{D}})$ if the mapping is multilinear. Let

$$\delta^* = [\delta_1^*, \delta_2^*, \delta_3^*, ..., \delta_m^*]^T \in \bar{\mathcal{D}}$$

and define

$$\delta^*(k) = [\delta_1^*, \delta_2^*, ..., \delta_k^*, \sigma_{k+1}, \sigma_{k+2}, ..., \sigma_m]^T$$

where $0 \leq k \leq m$, and σ_i is either $+1$ or -1, for $k+1 \leq i \leq m$. Note that $\delta^*(k)$ represents one of the 2^{m-k} points which associate with δ^* on the boundary hyperplanes of $\bar{\mathcal{D}}$. $\delta^*(0)$ is one of the vertices of $\bar{\mathcal{D}}$ and $\delta^*(m) = \delta^*$. We shall prove by mathematical induction that $\mathcal{I}(\delta^*(m)) \in \mathcal{P}(\bar{\mathcal{D}})$. Assume that the image of the 2^{m-k} $\delta^*(k)$ are all in $\mathcal{P}(\bar{\mathcal{D}})$ and define

$$\delta^*(k^+) = [\delta_1^*, \delta_2^*, ..., \delta_k^*, +1, \sigma_{k+2}, ..., \sigma_m]^T$$

and

$$\delta^*(k^-) = [\delta_1^*, \delta_2^*, ..., \delta_k^*, -1, \sigma_{k+2}, ..., \sigma_m]^T$$

Let the line segment connecting $\delta^*(k^+)$ and $\delta^*(k^-)$ be denoted by l_k. Obviously, l_k is parallel to the δ_{k+1} axis, since the only coordinate change from $\delta^*(k^+)$ to $\delta^*(k^-)$ is along the δ_{k+1} axis. Therefore, the image of l_k is also in $\mathcal{P}(\bar{\mathcal{D}})$ by virtue of Lemma 3. Since $\delta^*(k+1)$ is a point on l_k, its image must also be in $\mathcal{P}(\bar{\mathcal{D}})$. The fact that the image of $\delta^*(0)$ is in $\mathcal{P}(\bar{\mathcal{D}})$ is established by the definition of $\mathcal{P}(\bar{\mathcal{D}})$. This completes the proof.

Proof of Theorem 3

The proof is similar to that of De Gaston and Safonov.[10] The only difference is that their codomain is the complex plane while ours is the coefficient space. The roots of the characteristic equation, i.e., the eigenvalues of A-$B\Delta C$, are

uniquely determined by the parameter vector δ and therefore so is the coefficient vector α. For any $\delta^x \in \bar{\mathcal{D}}_1$, there exists a $\delta^y \in \bar{\mathcal{D}}_2$ such that $\delta^y = \delta^x$ and thus $\mathcal{I}(\delta^y) = \mathcal{I}(\delta^x)$. Hence, $\mathcal{I}(\bar{\mathcal{D}}_1) \subset \mathcal{I}(\bar{\mathcal{D}}_2)$. Since $\mathcal{P}(\bar{\mathcal{D}}_1)$ is the smallest convex set that contains all of the points of $\mathcal{I}(\bar{\mathcal{D}}_1)$, Eq. (12) is a direct consequence of Eq. (11).

Proof of Theorem 4

Again, the proof is similar to that of De Gaston and Safonov.[10] The vertices of $\mathbb{D}_1$ and $\mathbb{D}_2$ are mapped into $\mathcal{P}(\mathbb{D})$ and therefore $\mathcal{P}(\bar{\mathcal{D}}_1)$, $\mathcal{P}(\bar{\mathcal{D}}_2)$, and $\mathcal{P}(\bar{\mathcal{D}}_1) \cup \mathcal{P}(\bar{\mathcal{D}}_2)$ are contained in $\mathcal{P}(\bar{\mathcal{D}})$. Since $\mathcal{I}(\bar{\mathcal{D}}_1) \subset \mathcal{P}(\bar{\mathcal{D}}_1)$, $\mathcal{I}(\bar{\mathcal{D}}_2) \subset \mathcal{P}(\bar{\mathcal{D}}_2)$, and $\mathcal{I}(\bar{\mathcal{D}}_1) \cup \mathcal{I}(\bar{\mathcal{D}}_2) = \mathcal{I}(\mathcal{D})$, we have $\mathcal{I}(\bar{\mathcal{D}}) \subset \mathcal{P}(\bar{\mathcal{D}}_1) \cup \mathcal{P}(\bar{\mathcal{D}}_2)$.

Acknowledgments

This research was supported in part by the Air Force Office of Scientific Research under Contract F33615-88-C-3600 and in part by the National Science Foundation under Grant ECS-8796269.

References

[1]Doyle, J., "Analysis of Feedback Systems with Structured Uncertainties," *IEE Proceedings*, Vol. 129, Pt. D, No. 6, 1982, pp. 242–250.

[2]Doyle, J., "Structured Uncertainty in Control System Design," *Proceedings of the 24th IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, Dec. 1985, pp. 260–265.

[3]Fan, M. K. H., and Tits, A. L., "Characterization and Efficient Computation of the Structured Singular Value," *IEEE Transactions on Automatic Control*, Vol. AC-31, No. 8, 1986, pp. 734–743.

[4]Fan, M. K. H., and Tits, A. L., "Geometric Aspects in the Computation of the Structured Singular Value," *Proceedings of the 1986 American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1986, pp. 437–441.

[5]Safonov, M. G., and Doyle, J., "Minimizing Conservativeness of Robustness Singular Values," edited by S. G. Tzefestas, *Multivariable Control*, Reidel, Amsterdam, the Netherlands, 1984.

[6]Morton, B. G., and McAfoos, R. M., "A Mu-Test for Real Parameter Variations," *Proceedings of the 1985 American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1985, pp. 135–138.

[7]Jones, R. D., "Structured-Singular-Value Analysis for Real Parameter Variations," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, AIAA, New York, 1987, pp. 1424–1432.

[8]Safonov, M. G., and Athans, M., "Gain and Phase Margin for Multiloop LQG Regulators," *IEEE Transactions on Automatic Control*, Vol. AC-22, No. 2, 1977, pp. 173–179.

[9]Safonov, M. G., "Stability Margins of Diagonally Perturbed Multivariable Feedback Systems," *Proceedings of the 20th IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, Dec. 1981, pp. 1472–1478.

[10]De Gaston, R. E., and Safonov, M. G., "Exact Calculation of the Multiloop Stability Margin," *IEEE Transactions on Automatic Control*, Vol. AC-33, No. 2, 1988, pp. 156–171.

[11]Pena, R. S. S., and Sideris, A., "A General Program to Compute the Multivariable Stability Margin for Systems with Parametric Uncertainty," *Proceedings of the 1988 American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1988, pp. 317–322.

[12]Sideris, A., and Pena, R. S. S., "Fast Computation of the Multivariable Stability Margin for Real Interrelated Uncertain Parameters," *Proceedings of the 1988 American Control Conference*, Inst. of Electrical and Electronics Engineers, Piscataway, NJ, June 1988, pp. 1483–1488.

[13]Kharitonov, V. L., "Asymptotic Stability of an Equilibrium Position of a Family of Systems of Linear Differential Equations," *Differential'nye Uraveniya*, Vol. 14, No. 11, 1978, pp. 1483–1485.

[14]Bartlett, A. C., Hollot, C. V., and Lin, H., "Root Locations on an Entire Polytope of Polynomials: It Suffices to Check the Edges," *Math. Control Signal Systems*, Vol. 1, No. 1, 1988, pp. 61–71.

[15]Fu, M., and Barmish, B. R., "Stability of Convex and Linear Combinations of Polynomials and Matrices Arising in Robustness Problems," *Proceedings of the Conference on Information Sciences and Systems*, Johns Hopkins University, 1987.

[16]Bialas, S., "A Necessary and Sufficient Condition for the Stability of Convex Combinations of Stable Polynomials or Matrices," *Bulletin of the Polish Academy of Sciences, Technical Sciences*, Vol. 33, No. 9–10, 1985, pp. 473–480.

[17]Wei, K. H., and Yedavalli, "Invariance of Strict Hurwitz Property for Uncertain Polynomials with Dependent Coefficients," *IEEE Transactions on Automatic Control*, Vol. AC-32, No. 10, 1987, pp. 907–909.

[18]Keel, L. H., Bhattacharyya, S. P., and Howze, J. W., "Robust Control with Structured Perturbations," *IEEE Transactions on Automatic Control*, Vol. AC-33, 1988, pp. 68–78.

[19]Bouguerra, H., Chang, B.-C., Yeh, H. H., and Banda, S. S., "Fast Stability Checking for the Convex Combination of Stable Polynomials," *IEEE Transactions on Automatic Control*, Vol. 35, No. 5, 1990, pp. 586–588.

[20]Chapellat, H., and Bhattacharyya, S. P., "A Generalization of Kharitonov's Theorem for Robust Stability on Interval Plants," Electrical Engineering Dept., Texas A&M Univ., TCSP Research Rept. 88-004, Jan. 1988.

[21]Barmish, B. R., Fu, M., and Salem, S., "Stability of a Polytope of Matrices: Counterexamples," *IEEE Transactions on Automatic Control*, Vol. AC-33, No. 6, 1988, pp. 569–572.

[22]Zadeh, L. A., and Desoer, C. A., *Linear System Theory*, McGraw-Hill, New York, 1963, Chap. 9, pp. 474–475.