

μ -analysis for flexible systems

Jean-François Magni Carsten Döll

ONERA-CERT, Département d'Automatique
B.P. 4025, F31055 Toulouse Cedex 04

Email: magni@cert.fr and doll@cert.fr

μ -analysis for flexible systems

Jean-François Magni* Carsten Döll*,†

* ONERA-CERT, Département d'Automatique
B.P. 4025, F31055 Toulouse Cedex 04

† SUPAERO
B.P. 4025, F31055 Toulouse Cedex 04

Email: magni@cert.fr and doll@cert.fr

Réunion du Groupe Robustesse au LAAS le 4 mars 1998. Note réalisée en assemblant plusieurs articles.

Abstract. This paper presents new tools for computing upper and lower bounds of μ without frequency gridding. The proposed techniques for computing lower bounds of the peaks of the μ -curve, are divided into two steps. The first one consists of finding the perturbation with minimum Frobenius norm that leads to the limit of stability. Using this result as an initialization, the second algorithm finds the perturbation with minimum sigma-max norm such that the system remains at the limit of stability. The limit of stability is considered both from a state space and from a transfer matrix point of view, which leads to two classes of techniques. The lower bounds are validated by using an upper-bound analysis technique that considers standard scalings over a range of frequencies instead of scalings at an isolated frequency. Especially with flexible structures, peaks can be so thin that there is a risk to miss them by gridding methods. Hence, from the fact that frequency gridding is not used, the proposed techniques are more specifically efficient for analyzing flexible structures. Uncertainties can be pure real or mixed.

Keywords. Frequency sweep, μ -lower bound, μ -upper bound.

1 Introduction

Stability analysis of highly flexible structures is a challenging problem. Standard μ -analysis is not efficient for this class of problems for two reasons. First, if frequency gridding is used, peak values of μ are generally missed. Second, uncertainties are usually real, in this case the lower bound of μ cannot be computed with standard numerical tools.

The computation of the structured singular value μ has received much attention. Exact computation was proposed for real uncertainties but this problem is known as being NP-hard [2, 13]. A tractable technique is proposed in [12], the computing time is exponential but up to about twelve real uncertain parameters it is the most efficient technique that we have tested. Unfortunately, this technique is limited to real non repeated uncertainties. In addition, an approximation of the value set by its convex hull induces some conservatism when the limit of stability is found on an edge of the convex hull which is not an edge of the value set (unusual for practical systems). Briefly, exact computation is not realistic, therefore upper and lower bounds must be considered.

All available lower bound computation techniques consist of finding a perturbation which corresponds to the limit of stability. All techniques are more or less heuristic. A fixed point algorithm is proposed in [23, 26]. This algorithm is very efficient for mixed uncertainties in which the complex part is large enough. Unfortunately, when uncertainties are modeled as real parameter variations this algorithm does not converge well enough. It is possible to improve convergence by adding a small amount of artificial complex uncertainty, but this amount needs to be fixed by trial and error and induces approximation in the results that is difficult to evaluate. In [10] it is shown how the perturbation can be tuned so that the small amount of artificial complex uncertainty can be removed. These techniques use frequency gridding. An alternative technique (real uncertainties) is proposed in [4]. The search of instability is limited to two dimensional faces of the uncertainty box. Nevertheless, the involved computing time is exponential.

The upper bounds are defined as convex optimization problems after frequency gridding, see for example [8, 1]. These techniques are reliable when the μ -curve does not present sharp peaks. In order to treat the problem of sharp peaks, frequency sweeping is used (see [24, 16]), the frequency is considered as an additional real uncertainty which should be repeated as many times as the order of the system. Therefore, computationally, this approach rapidly becomes too demanding for real-world applications.

Our main contribution is detailed now. Concerning the lower bound computation, new algorithms which compute a lower bound of the peaks of the μ curve are proposed. These algorithms are efficient for mixed or pure real uncertainties. From the fact that frequency gridding is not used, the proposed algorithms are very fast. This is quite useful in a design cycle in which it is necessary to detect worst cases see [19]. The idea beyond these techniques consists of shifting the eigenvalues toward the imaginary axis with a minimum perturbation. The proposed algorithms are divided into two steps. The first step is used to reach the limit of stability with a perturbation of minimum Frobenius norm. For that purpose the algorithm proposed in [6, 5] was tested but the convergence was very erratic. An adaptation of the ‘‘pole migration’’ of [20] had shown to be much more reliable. The second step consists of minimizing the ‘‘sigma-max norm’’ of the perturbation obtained after the first step while remaining at the limit of stability. The fixed point algorithm of [26] was used, the convergence

properties are improved on account of the good initialization of step 1. But the alternative technique detailed in this paper is more reliable. The efficiency of the proposed lower bound computation is also illustrated in [7].

Concerning the upper bound, the technique proposed here takes advantage, over a frequency interval, of scalings that are computed at a given frequency. This idea can be found for example in [3], however the proposed algorithm is very different. An alternative approach with a different purpose can be found in [11]. The algorithm proposed here checks whether a given test value, for example the maximum value of the lower bound augmented of a given percentage, is larger than the upper bound of μ ([1, 8, 23, 27]) over all frequencies. At each step of the proposed algorithm, scalings are computed at a given frequency. The frequency intervals for which these scalings permit us to conclude that the test value is larger than μ are eliminated. The frequency intervals that can be eliminated are characterized analytically (see Lemma 5.2). If all frequencies can be eliminated, the test value is an upper bound of the peak values of μ , therefore the lower bound is validated.

First, is considered the state-space approach. In §3.1 is presented the proposed algorithm for minimizing the Frobenius norm. In §3.2 is described the second algorithm relative to the “sigma-max norm”. The adaptations of the above algorithms for treating the transfer matrix approach are detailed in §4. In §5 is presented the validation of the lower bound by upper bound analysis. Finally a satellite example controlled by an \mathcal{H}_∞ based design is considered for illustration. The proposed lower bound is also illustrated by reporting the results of [7] in the last section.

2 Notation and preliminary results

Let us consider a quadruple in state space form (A, B, C, D) and in transfer matrix form

$$M(s) = C(sI - A)^{-1}B + D$$

A class of uncertainties Δ (which act as a feedback on the system, see Figure 1) has the following structure:

$$\Delta = \text{diag}(\Delta_1, \dots, \Delta_r) \quad (1)$$

in which Δ_i might be a diagonal matrix of the form $\Delta_i = \delta_i I_{n_i}$ with $\delta_i \in \mathbf{R}$ (real repeated scalar block) or $\delta_i \in \mathbf{C}$ (complex repeated scalar block) or a matrix in $\mathbf{C}^{n_i \times n_i}$ (full complex block). In the part relative to the lower bound Δ_i might also be a matrix in $\mathbf{R}^{n_i \times n_i}$ (full real block). Such a matrix Δ will be called an “admissible perturbation”.

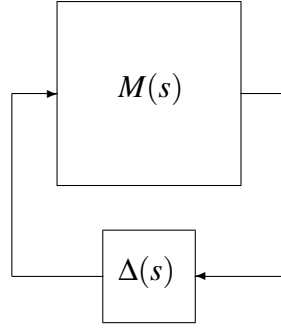


Figure 1: $M - \Delta$ structure for μ -analysis

The real and imaginary part of a complex number are denoted $\Re(\cdot)$ and $\Im(\cdot)$, the maximum singular value $\bar{\sigma}(\cdot)$ the eigenvalue with maximum modulus $\bar{\lambda}(\cdot)$.

First let us recall a well known result:

Lemma 2.1 *If Δ is such that the interconnection (M, Δ) remains well-posed (i.e. for all admissible Δ , $\det(I - D\Delta) \neq 0$) and if s_0 does not belong to the spectrum of A :*

$$\det(I - M(s_0)\Delta) = 0 \Leftrightarrow s_0 \in \text{spectrum}(A + B\Delta(I - D\Delta)^{-1}C) \quad (2)$$

In order to compute the s.s.v. at a complex point λ_t , we address the problem of assigning λ_t in the closed loop spectrum with a “feedback” Δ (belonging to the set of admissible perturbations) of minimum “sigma-max norm”. In view of Lemma 2.1, two almost equivalent definitions will be considered.

Definition 2.2 *The s.s.v. at point λ_t is defined as $1/\bar{\sigma}(\Delta)$ where Δ is an admissible perturbation such that¹*

¹ $\bar{\sigma}(\Delta)$ denotes the maximum singular value of Δ .

1. λ_t belongs to the closed-loop spectrum of (A, B, C, D) with feedback Δ
2. $\bar{\sigma}(\Delta)$ is minimum.

Definition 2.3 The s.s.v. at point λ_t is defined as $1/\bar{\sigma}(\Delta)$ where Δ is an admissible perturbation such that

1. $\det(I - M(\lambda_t)\Delta) = 0$
2. $\bar{\sigma}(\Delta)$ is minimum.

In view of the above definitions, computing μ at point λ_t is the problem of assigning the pole λ_t with minimum norm of the “feedback”. So, finding in one shoot the maximum value of μ over all points on the imaginary axis, is the problem of assigning the value of the imaginary axis for which the required “feedback” is minimum. The following result (see [20]) will be used for shifting the open-loop poles towards the imaginary axis.

Lemma 2.4 The first order approximation $d\lambda$ of the motion of an eigenvalue of the matrix A_0 induced by a gain variation $d\Delta$ of Δ_0 is

$$d\lambda = (uB + tD)d\Delta(Cv + Dw) \quad (3)$$

where v is the right eigenvector of A_0 , u is the left eigenvector of A_0 corresponding to the eigenvalue λ and $w = \Delta_0(I - D\Delta_0)^{-1}Cv$, $t = uB\Delta_0(I - D\Delta_0)^{-1}$.

Using this result iteratively, first, an admissible matrix Δ with minimum Frobenius norm and which move an open-loop pole on the imaginary axis will be computed. Then a second algorithm will be used to find the best candidate with minimum “sigma-max norm”.

3 State-space approach to the computation of a lower bound

Finding a lower bound of the peaks of the μ -curve consists of applying Algorithm 1 (see §3.1) and then Algorithm 2 (see §3.2). The transfer matrix approach which is somewhat similar will be presented in §4, in this case Algorithm 1 will be replaced by an alternative algorithm.

3.1 Computation of the destabilizing perturbation with minimum Frobenius norm

The proposed algorithm is sketched first. Several comments are given afterwards in order to discuss some implementation adaptations that improve the efficiency.

Principle of the algorithm. Let λ denote one of the eigenvalues of $A + B\Delta_0(I - D\Delta_0)^{-1}C$. It is intended to find $d\Delta$ (a variation of Δ_0) that shifts (first order approximation) λ to a vertical line which is distant of a small amount denoted \Re_i . A motion from λ to the vertical line defined by \Re_i is performed as follows. Equation (3) is a linear constraint on $d\Delta$

$$\Re((uB + tD)d\Delta(Cv + Dw)) = \Re_i \quad (4)$$

$d\Delta$ satisfying (4) will be computed such that the Frobenius norm

$$J_1 = \|\Delta_0 + d\Delta\|_F^2 \quad (5)$$

is minimum. This is a problem of quadratic optimization under linear constraints. *Such a problem will be solved at each iteration of the proposed algorithm.* The details on the way the structure of Δ is taken into account in order to solve (4) minimizing J_1 in (5) are given in Appendix A. In order to avoid initialization problems due to the fact that, when the criterion J_1 is minimum for large values of $d\Delta$, the first order approximation of Lemma 2.4 is no longer valid, we shall minimize a combination of criteria J_1 and J_0 , where J_0 is defined by

$$J_0 = \|d\Delta\|_F^2 \quad (6)$$

At the beginning of the algorithm, J_0 is considered, then a combination of J_0 and J_1 that becomes equal to J_1 (see (8)).

Algorithm 1.

Step 1 - Initialization. Choose the initial open-loop eigenvalues that are to be moved towards the imaginary axis. Choose also the expected number of steps (say N) that will be used in order to shift each initial eigenvalue to the target. For each initial eigenvalue λ , perform the following steps. Set $i = -1$ and $\Delta_0 = 0$.

Step 2 - Compute u, v, w, t and solve (4), for $d\Delta$ having the admissible structure, for a variation of $\Re(\lambda)$ given by $\Re_i = -\Re(\lambda)/(N - i)$ i.e.

$$\Re((uB + tD)d\Delta(Cv + Dw)) = \Re_i \quad (7)$$

such that

$$\frac{(N-i-1)\|d\Delta\|_F^2 + (i+1)\|\Delta_0 + d\Delta\|_F^2}{N} \quad (8)$$

is minimum.

Step 3 - Set $i = \min(i+1, N-1)$, $\Delta_0 = \Delta_0 + d\Delta$. After Δ_0 is updated, select the new closed loop eigenvalue that is the closest to $\lambda + \Re_i$ that will become the new λ . If λ is close enough to the imaginary axis, stop, otherwise go to step 2.

Comments relative to Algorithm 1.

Measure of the controllability. In order to reduce the computing time it is useful to apply the algorithm only to a subset of the poles of $M(s)$. The use of the bandwidth knowledge of the system behavior can help. It is also possible to apply a controllability measure. From Lemma 2.4,

$$\text{Contr}(\lambda) = \|uB + tD\| \|Cv + Dw\| \quad (9)$$

can be considered as such a measure. Only the most controllable eigenvalue might be chosen, however only one peak of the μ -curve would be detected (and it might not be the highest one). The number of poles to be considered varies depending on the system, so it must be fixed by trial and errors. From an experimental point of view, about five initial poles is far enough for most of the systems that we have analyzed.

Details on computation involved at Step 2. Equation (7) is a linear constraint relative to the entries of $d\Delta$ and (8) is a quadratic criterion. An intermediate step which consists of writing the entries of Δ_0 and $d\Delta$ as real vectors ξ_0 and $d\xi$ (in which uncertainties are not repeated) is necessary, see Appendix A. Equation (7) can be written

$$Ad\xi = b \quad (10)$$

(A being a row vector) and J_0 and J_1 in (8) become respectively

$$J_0 = d\xi^T H_0 d\xi \quad \text{and} \quad J_1 = d\xi^T H_1 d\xi + 2c_1 d\xi$$

The computation of the matrices A , $H_0 (= H_1)$ and c_1 is treated in Appendix A. Finally the problem to be solved at each step is a least square problem.

Number of steps, precision. The criterion in (8) is a combination of J_0 and J_1 . At the first step, $i = -1$ so J_0 is optimized. When i increases, the part of J_1 increases. When i is saturated at $N-1$ the criterion is exactly equal to J_1 . Usually, $N = 20$ is enough. After these initial iterations it is worth adding some iterations (about 10) for improving the assignment on the imaginary axis.

Continuity of pole motion. A limitation comes from the fact that Equation (3) is valid only for non repeated eigenvalues. It happens sometimes that two branches (defined by the motion of the pole in the complex plane) cross each other or that two complex conjugate branches lead to a separation point to continue on the real axis. Numerical problems arise only if during iterations these pathologies are almost exactly encountered, which is not so likely to occur. Otherwise, saddle points on a branch can occur where first order approximations are not valid anymore (see §6).

3.2 Computation of the destabilizing perturbation with minimum sigma-max norm

After having used the algorithm of the previous section, we have at our disposal a matrix Δ_0 which assigns a pole λ_r on the imaginary axis. But the norm that was minimized for obtaining this result is not the right one. So, a second algorithm is proposed: the assignment of $\Re(\lambda_r)$ is preserved while the convergence towards a matrix “ Δ ” with minimum sigma-max norm is performed.

Several possibilities are offered. Ferreres and Fromion [10] propose a very efficient technique that can be used only if the uncertainty blocs are all real repeated uncertainties. The main idea consists of using linear programming in which the criterion is a bound on the maximum value of the uncertain parameters, the linear constraint being the first order condition corresponding to the assignment on the imaginary axis (*i.e.* Equation (13)). This technique is not evaluated here as it cannot be used with mixed uncertainties.

The power algorithm of [26] was also considered. Its convergence is very good for mixed or complex uncertainties. In the pure real case, convergence is erratic and fails in most cases. It was expected that a good initialization, as for example the results of Algorithm 1 (§3.1), would improve convergence in the pure real case. The description of the power algorithm, its adaptation to Algorithm 1, its frequency sweep version and discussion are given in Appendix B. Briefly, when the power algorithm is initialized by the result of Algorithm 1 it is faster than Algorithm 2 presented below. But Algorithm 2 is much more reliable as convergence is almost insured, in addition computing time is still reasonable.

Principle of the proposed algorithm. Considering the singular value decomposition (s.v.d.) of $\Delta_0 = USV^*$ and denoting V_1 the first column vector of V , the maximum singular value of Δ_0 is

$$\bar{\sigma}(\Delta_0) = \sqrt{V_1^* \Delta_0^* \Delta_0 V_1} \quad (11)$$

so we have to minimize (11). For that purpose a new criterion

$$J_2 = V_1^* (\Delta_0 + d\Delta)^* (\Delta_0 + d\Delta) V_1 \quad (12)$$

is defined in which V_1 is relative to Δ_0 . As it is expected to consider small variations of Δ_0 we shall have

$$J_2 \approx \bar{\sigma}(\Delta_0 + d\Delta)$$

It is this approximation J_2 of the maximum singular value that will be considered.

Algorithm 2.

Step 1 - Initialization. Perform Algorithm 1. Let Δ_0 denote the resulting admissible perturbation and λ the eigenvalue which is approximatively on the imaginary axis. Choose the number of iterations N . Set $i = 0$.

Step 2 - Compute u, v, w, t and solve, for $d\Delta$ having the admissible structure,

$$\Re((uB + tD)d\Delta(Cv + Dw)) = -\Re(\lambda) \quad (13)$$

such that

$$\frac{(N-i)\|\Delta_0 + d\Delta\|_F^2 + iV_1^*(\Delta_0 + d\Delta)^*(\Delta_0 + d\Delta)V_1}{N} \quad (14)$$

is minimum, in which V_1 is relative to the s.v.d. of Δ_0 .

Step 3 - Set $i = \min(i+1, N-1)$, $\Delta_0 = \Delta_0 + d\Delta$. After Δ_0 is updated, select the new closed loop eigenvalue that is the closest to $\Im(\lambda)$ that will become the new λ . If the value of J_2 becomes stationary, stop, otherwise go to step 2.

Comments relative to Algorithm 2.

Details on computation involved at Step 2. Equation (13) is a linear constraint relative to the entries of $d\Delta$ and (14) is a quadratic criterion. An intermediate step which consists of writing the entries of Δ_0 and $d\Delta$ as real vectors ξ_0 and $d\xi$ (in which uncertainties are not repeated) is necessary, see Appendix A. Equation (13) can be written as Equation (10) and J_1 and J_2 in (14) become respectively

$$J_1 = d\xi^T H_1 d\xi + 2c_1 d\xi \quad \text{and} \quad J_2 = d\xi^T H_2 d\xi + 2c_2 d\xi$$

See Appendix A for details. Finally the problem to be solved at each step is a least square problem.

Coalescence of singular values at the optimum. In order to alleviate the presentation of the algorithm we ignored a very important fact. If the algorithm is run as above, the optimization will end when the two leading singular values of Δ become equal. It is worth noting that Algorithm 2 tends to tune the Frobenius optimal perturbation by reducing the leading singular values. Coalescence of two or more leading singular values is therefore usual. In order to optimize further after the two leading singular values are equal, a simple improvement is required. The details are given in the Appendix A. This improvement reduces to testing that some singular values become equal and then to adding some rows to the linear equation that corresponds to Equation (13).

Evolution of the criterion, number of iterations. The criterion in (14) is a combination of J_1 and J_2 . At the first step, $i = 0$, J_1 is dominant in the composite criterion. When i increases, the part of J_2 increases. When i is saturated at $N-1$ the criterion is equal to $(1/N)J_1 + ((N-1)/N)J_2$. Note that J_2 is not minimized without considering a small amount of J_1 because, as shown in the appendix, J_2 , unlike J_1 , is not a definite quadratic form. Here again, it is better to transform ‘‘smoothly’’ the criterion from J_1 to J_2 , because if $(1/N)J_1 + ((N-1)/N)J_2$ were used abruptly, for some systems the variation of Δ would be too large taking into account the first order approximation behind Equation (13). Usually, $N = 20$ is enough. After these initial iterations it is worth adding some iterations in order to treat singular values coalescence. The number of additional iterations depends on the number of singular values which become equal. Globally, 80 iterations are enough for most systems.

4 Transfer matrix approach to the computation of a lower bound

This section describes a technique similar to the one presented in §3.1, but it is the transfer matrix that is applied instead of the state space representation. This alternative technique is interesting because it is an additional tool that may converge better than the previous one in some cases. In fact, all lower bounds computation techniques are based on iterative techniques without guarantee of global convergence. So, it is very useful to have as many complementary tools as possible.

The proposed technique being very similar to the previous one, only the differences are pointed out. The main difference is that, instead of assigning a pole on the imaginary axis (see Definition 2.2) we want $\det(I - M(s)\Delta)$ to become equal to zero for some value of s on the imaginary axis (see Definition 2.3).

It is well known that the loop in Figure 1 is equivalent to a similar loop where $M(s)$ and Δ are replaced respectively by \bar{M} and $\bar{\Delta}$ where:

$$\bar{M} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} ; \bar{\Delta} = \begin{bmatrix} \frac{1}{s}I & 0 \\ 0 & \Delta \end{bmatrix}$$

As we are interested in poles on the imaginary axis s takes the form $1/j\omega$. We shall denote

$$\delta = 1/\omega$$

moving the complex number j inside \bar{M} , we have to consider \bar{M}' and the interconnection $\bar{\Delta}'$:

$$\bar{M}' = \begin{bmatrix} -jA & B \\ -jC & D \end{bmatrix} ; \bar{\Delta}' = \begin{bmatrix} \delta I & 0 \\ 0 & \Delta \end{bmatrix}$$

At this step, the following property is clearly satisfied:

$$\det(I - \bar{M}'\bar{\Delta}') = 0 \Leftrightarrow \det(I - M(\frac{j}{\delta})\Delta) = 0$$

Therefore, the problem to be treated consists of shifting a pole of the system $(I, -\bar{M}', I, 0)$ from 1 to the origin. But the difference with the algorithm of §3.1 is that it is not the norm of $\bar{\Delta}' + d\bar{\Delta}'$ that must be considered, because this artificial perturbation contains the frequency (δ). The norm to be considered is the norm of the matrix

$$\begin{bmatrix} 0 & I \end{bmatrix} (\bar{\Delta}' + d\bar{\Delta}') \begin{bmatrix} 0 \\ I \end{bmatrix}$$

The Frobenius norm of the above matrix is not a positive *definite* criterion because the weighting relative to δ is zero. In order to obtain a regular problem, an additional step is required. This step consists of eliminating δ from the constraints and criteria.

Elimination of δ . Applying Lemma 2.4 to the system $(I, -\bar{M}', I, 0)$ (instead of (A, B, C, D)) for a variation $d\lambda$, an equation linear with respect to δ and the entries of Δ (denoted in vector form ξ) is obtained (see (10):

$$A \begin{bmatrix} d\delta \\ d\xi \end{bmatrix} = d\lambda \quad (15)$$

Let

$$A = \begin{bmatrix} a_r + ja_i & A_r + jA_i \end{bmatrix}$$

and

$$d\lambda = b_r + jb_i$$

considering the imaginary and real parts

$$\begin{aligned} a_r d\delta + A_r d\xi &= b_r \\ a_i d\delta + A_i d\xi &= b_i \end{aligned} \tag{16}$$

after elimination of $d\delta$, only one equation remains

$$(a_i A_r - a_r A_i) d\xi = a_i b_r - a_r b_i \tag{17}$$

Finally, the frequency (or δ) is eliminated from the considered variable, therefore the criterion relative to the Frobenius norm is again positive definite (see Appendix A.1).

This discussion is summarized now. The differences between the algorithm of §3.1 and the one of this section (**Algorithm 1'**) reduce to

- The motion of the considered poles is not towards the imaginary axis but towards the origin, therefore replace the right hand side $\Re_i = -\Re(\lambda)/(N-i)$ in (7) by $-\lambda/(N-i)$.
- Replace (A, B, C, D) by $(I, -\bar{M}', I, 0)$ for the computation of the left hand side of (7) (and remove “ \Re ”). Let denote u and v the left and right eigenvectors of $I - \bar{M}'\bar{\Delta}_0'$, Equation (7) becomes:

$$-u\bar{M}'d\bar{\Delta}'v = -\lambda/(N-i) \tag{18}$$

- This constraint being written in the explicit linear form of (15) eliminate δ in order to obtain (17) instead of (10).
- After the quadratic programming problem is solved and $d\xi$ is known, it remains to compute $d\delta$ by considering Equation (16) ($\rightarrow \bar{\Delta}'$ can be evaluated).

Other adaptations are straightforward. It is possible to adapt Algorithm 2 to the transfer matrix approach. This possibility was not tested, because Algorithm 2 as described in §3.2 can be used indifferently after Algorithm 1 or Algorithm 1'. Note also that the first step is preponderant for convergence, so it is not worth developing an alternative second algorithm.

5 Upper bound for validation of the lower bound

As for all lower bound of μ , the lower bound of the peaks presented in the previous sections has an heuristic base. There is no guarantee that the proposed algorithms will lead to all the peaks of μ with much accuracy. All computation of a lower bound must be validated by an upper bound. As already mentioned in the introduction, standard tools are not reliable on account of frequency gridding. If frequency sweeping is used a too high level of computing time is required.

Here we take advantage of the knowledge of the lower bounds of peak values of μ . Let μ_{\max} be the highest lower bound available. In order to validate this result say with 10 % accuracy, it suffices to consider a test value $\mu_T = 1.1 \mu_{\max}$ and to check that μ is lower than this value for all frequencies.

First is treated the problem of finding the interval of frequencies for which a given pair of scalings D_0 and G_0 enables to conclude that μ is less than μ_T . Afterward, this result will be organized as an algorithm.

5.1 Elimination of frequency intervals

Two sets of scalings having the same structure than Δ (see (1)) are considered:

- $\mathcal{D} : D_i = D_i^* > 0$ and $D_i \in \mathbb{C}^{n_i \times n_i}$ for real and complex repeated blocks. $D_i = d_i I_{n_i}$ and $d_i > 0$ for full complex blocks.
- $\mathcal{G} : G_i = G_i^*$ and $G_i \in \mathbb{C}^{n_i \times n_i}$ for real repeated blocks. $G_i = 0$ for other blocks.

We shall use the following well known result (see [8])

Lemma 5.1 *The Laplace variable s being fixed, let $D \in \mathcal{D}$, $G \in \mathcal{G}$ and β be such that*

$$M(s)^* D M(s) + j(GM(s) - M^*(s)G) \leq \beta^2 D \quad (19)$$

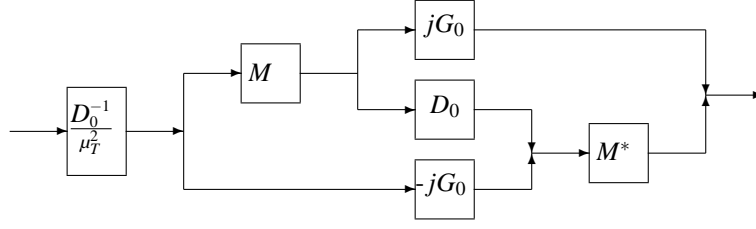
then

$$\mu(M(s)) \leq \beta \quad (20)$$

The problem of finding the best upper bound β of $\mu(M(s))$ is equivalent to the problem of minimizing β under the LMI condition of Equation (19). It is known as a generalized eigenvalue problem. If β is fixed *a priori*, it is also possible to find D and G satisfying (19) by solving an LMI.

The complete algorithm is described in the next section. Here we just consider a single step of this algorithm. A test value denoted μ_T is assumed to be chosen. The problem that is considered here consists of finding an interval of frequencies for which the upper bound of μ of Equations (19) - (20) remains less than μ_T .

Let us consider the scalings D_0 and G_0 that satisfy (19) for $s = j\omega_0$ and $\beta = \mu_T$.

Figure 2: Decomposition of $\bar{M}(j\omega)$

Lemma 5.2 Let (A, B, C, D) be a state space realization of $M(s)$. For a given value μ_T and two scalings G_0 and D_0 such that

$$M(j\omega_0)^* D_0 M(j\omega_0) + j(G_0 M(j\omega_0) - M^*(j\omega_0) G_0) < \mu_T^2 D_0 \quad (21)$$

and

$$1 \notin \text{spectrum}\left((D^T D_0 D + j(G_0 D - D^T G_0)) \frac{D_0^{-1}}{\mu_T^2}\right) \quad (22)$$

we have

$$\mu(M(j\omega)) \leq \mu_T$$

for $\omega \in [\omega^- \ \omega^+]$ where $j\omega^-$ and $j\omega^+$ are the eigenvalues of the matrix H the closest to $j\omega_0$ ($\omega^- < \omega_0 < \omega^+$) where:

$$H = \bar{A} + \bar{B}(I - \bar{D})^{-1} \bar{C}$$

and

$$\begin{aligned} \bar{A} &= \begin{bmatrix} A & 0 \\ C^T D_0 C & -A^T \end{bmatrix}; \quad \bar{B} = \begin{bmatrix} B \\ C^T D_0 D - jC^T G_0 \end{bmatrix} \frac{D_0^{-1}}{\mu_T^2} \\ \bar{C} &= \begin{bmatrix} D^T D_0 C + jG_0 C & -B^T \end{bmatrix} \\ \bar{D} &= (j(G_0 D - D^T G_0) + D^T D_0 D) \frac{D_0^{-1}}{\mu_T^2} \end{aligned}$$

Proof. Note that the condition of Equation (22) ensures that $(I - \bar{D})$ is a non singular matrix. It can be assumed that A has no eigenvalues on the imaginary axis because otherwise $\mu(M(j\omega))$ would have infinite values. In turn, the matrix \bar{A} can also be assumed to have no eigenvalues on the imaginary axis, therefore the condition of Lemma 2.1 concerning the spectrum of the matrix “ A ” will be ignored (see below). From (19-20), we have $\mu(M(j\omega)) \leq \mu_T$ if

$$\bar{\lambda}\left((M(j\omega)^* D_0 M(j\omega) + j(G_0 M(j\omega) - M^*(j\omega) G_0)) \frac{D_0^{-1}}{\mu_T^2}\right) < 1 \quad (23)$$

Let us denote

$$\bar{M}(j\omega) = (M(j\omega)^* D_0 M(j\omega) + j(G_0 M(j\omega) - M^*(j\omega) G_0)) \frac{D_0^{-1}}{\mu_T^2}$$

The limit of the validity of the scalings D_0 and G_0 around ω_0 is tested by computing the values of ω such that

$$\det(I - \bar{M}(j\omega)) = 0 \quad (24)$$

The matrix $\bar{M}(j\omega)$ can be viewed as being the matrix “ M ” in equation (2) (the matrix “ Δ ” in (2) being the identity matrix).

The matrix $\bar{M}(j\omega)$ can be decomposed as shown in Figure 2, where $M(j\omega) = C(j\omega I - A)^{-1}B + D$ and $M^*(j\omega) = (-B^T)(j\omega I - (-A^T))^{-1}C^T + D^T$. The system $(\bar{A}, \bar{B}, \bar{C}, \bar{D})$ is clearly a state space realization of \bar{M} valid on the imaginary axis.

So, in view of Lemma 2.1, we have shown that the roots of Equation (24) are the eigenvalues of the matrix H . In order to terminate the proof of the lemma, it remains to show that the relevant eigenvalues of H are $j\omega^-$ and $j\omega^+$. From (21), Equation (23) is satisfied for $\omega = \omega_0$, therefore the conclusion proceeds from a continuity argument. ■

A similar result can also be derived using the scalings proposed in [27] with minor amendments.

5.2 Proposed algorithm

Assume that a test value μ_T is known, see the discussion in the introduction of §5.

Algorithm 3.

Step 1 - Perform Algorithms 1 - 2 (or Algorithm 1' - 2) and choose the test value μ_T and the frequency range $[\omega_{min} \ \omega_{max}]$ for which $\mu_T > \mu(M(s))$ will be checked. Set $\omega_i = \omega_{min}$. Go to Step 2.

Step 2 - Compute $M = M(j\omega_i)$ and solve the generalized eigenvalue problem that gives D and G as in (19) with β minimized (standard upper bound of μ at $s = j\omega_i$). If the value of β is larger than μ_T we can conclude that μ_T is not a good candidate for an upper bound over the interval $[\omega_{min} \ \omega_{max}]$ (end of the algorithm). Otherwise go to Step 3.

Step 3 - If the condition number of D is too large solve an alternative LMI problem that gives new scalings D and G . This step is detailed and justified in the comment given below. Go to Step 4.

Step 4 - At this step $\mu_T > \beta$ therefore Equation (19) is still valid at $\omega = \omega_i$, with the current scalings D and G , if β is replaced by μ_T . So, (21) is satisfied and ω^+ can be computed as stated in Lemma 5.2. Compute ω^+ and set $\omega_i = \omega^+$. Go to Step 5.

Step 5 - If $\omega_i > \omega_{max}$ we can conclude that μ is less than μ_T over the interval $[\omega_{min} \ \omega_{max}]$ (end of the algorithm). If the series $\{\omega_i\}$ converges toward a finite value, we can conclude that μ_T is not a good candidate for an upper bound over the interval $[\omega_{min} \ \omega_{max}]$ (end of the algorithm). Otherwise go to Step 2.

Comments relative to Algorithm 3. First let us define the mapping $v_{D,G}(\omega)$ as being the minimum value of β that satisfies (19) for given D , G and $M(s) \rightarrow M(j\omega)$.

Illustration of the algorithm. The above algorithm takes advantage of scalings computed at a given “nominal” frequency over a neighborhood. This fact can be illustrated by drawing $v_{D,G}(\omega)$. As long

as $v_{D,G}(\omega)$ is less than μ_T we know that $\mu(M(j\omega)) (< v_{D,G}(\omega))$ is less than μ_T (see Lemma 5.1). Figure 3 gives an illustration of the way the frequencies for which the upper bound of μ is less than μ_T are eliminated. The sign “*” denotes the standard (point wise) upper bound of μ . The curves represent the corresponding mappings $v_{D,G}(\omega)$. Here all frequencies are eliminated for a test value $\mu_T = 0.3$ so the maximum value of μ is less than 0.3.

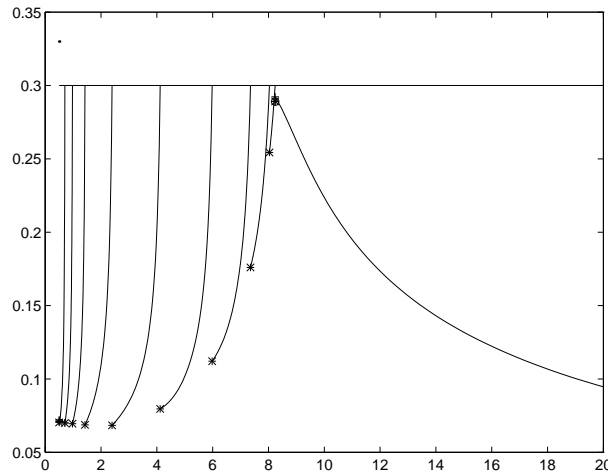


Figure 3: Illustration of Algorithm 3

Possible numerical problems. The shape of $v_{D,G}(\omega)$ is depicted in Figures 3 and 6. The curve becomes almost vertical when the frequency deviates from the “nominal” one. This problem becomes more and more stringent when μ presents sharp peaks or when μ_T is too small and that the series ω_i converges toward a finite value. Numerical problems arise: finding eigenvalues of the matrix H on the imaginary axis becomes more and more troublesome. In order to deal with this problem the scalings D and G are computed such that (19) hold true for additional frequencies (in addition to the “nominal” one). This problem is still an LMI problem, see [11, 9]. In this way the curve $v_{D,G}(\omega)$ is less vertical and the numerical problems disappear. Other strategies are possible for dealing with this problem. For example it is possible to solve the LMI problem of (19) with a value of β fixed between its minimal possible value and the test value μ_T .

Invertibility of $I - \bar{D}$. When $D = 0$, the problem of the invertibility of $I - \bar{D}$ cannot be encountered. Otherwise if it occurs it can be dealt with as above by changing the scalings D and G .

6 Illustrative example: a flexible satellite

The example of [14] (flexible satellite) is considered. It is characterized by six uncertain real parameters repeated twice and by 28 states (including the controller states).

6.1 Standard μ -analysis

A standard computation of an upper bound of μ by frequency sweeping (see [24, 16]) requires a huge amount of computing time. This is due to the fact that the frequency is considered as a real “uncertain” parameter repeated 28 times. It was not possible to perform a single step of this analysis tool. Therefore only frequency gridding based techniques are discussed.

In Figures 4 and 5 are given the results. The solid line is the upper bound of μ obtained using the μ -toolbox with option 'wC' (100 points between 0.5 and 20 Rd/s, 3 hours².) The lower bound obtained using the same toolbox with default option is plotted in a dotted line. As already stated the interest of drawing this lower bound is useless (in the real uncertainty case). Finally the additional signs denote the lower bounds of the peaks of μ obtained by our algorithms (§3.1-§3.2)

- \diamond : lower bound of peak A of μ after performing Algorithm 1
- ∇ : lower bound of peak B of μ after performing Algorithm 1
- \circ : lower bound of peak A of μ after performing Algorithm 2
- \triangle : lower bound of peak B of μ after performing Algorithm 2

There are still peaks at higher frequencies obviously not detected by our lower bound, but in fact, it is the upper bound which is erratic. Figures 8 and 9 prove that these peaks do actually not exist.

Note that peak A at about 2 Rd/s is almost invisible considering the upper bound curve. In Figure 5 a “zoom” around this frequency (20 points between 1.8 and 2.2 Rd/s) shows that this almost invisible peak really exists. This adds additional computation time of about 30min with option 'wC'. This analysis shows that frequency gridding must be avoided for flexible structures. The combination of the high amount of computing time and of the huge number of points to be considered lead to non-feasibility.

6.2 Proposed μ -analysis

Using the lower bound of §3.1 and §3.2, about one minute of computing time suffices to obtain the two peaks $\mu_A = 0.64$ at 2.03 Rd/s and $\mu_B = 0.72$ at 0.76 Rd/s, see also small circles in Figure 6.

²Using a very fast computer. With other workstations, 21 hours were needed for the same computation. Other options are faster but the results are more conservative, options 'wu' and 'wc' lead to large fancy peaks at high frequencies.

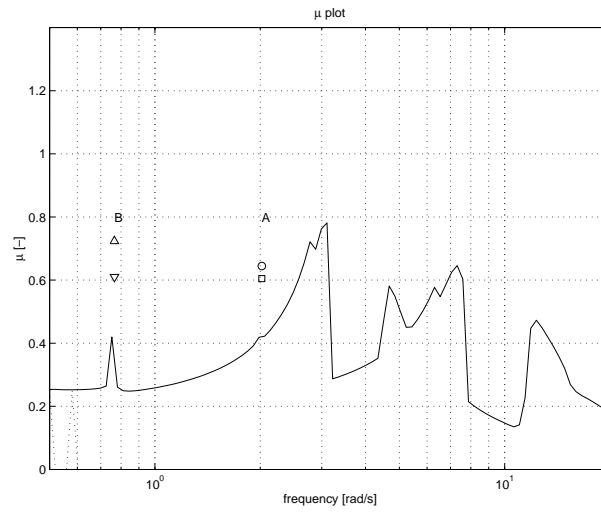


Figure 4: Upper bound computed by frequency gridding and lower bounds resulting from Algorithm 1 and 2

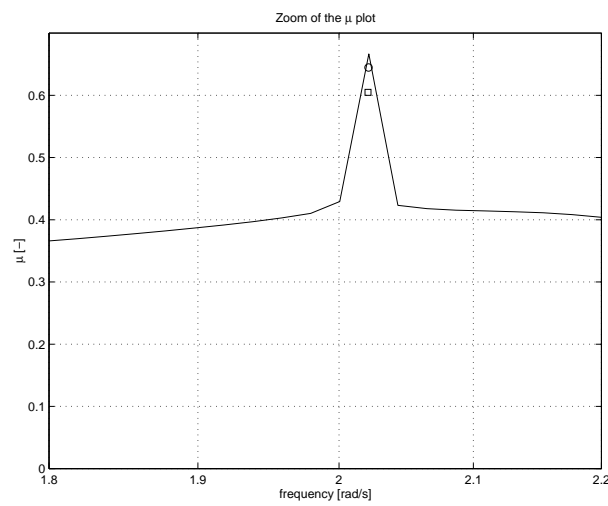


Figure 5: “Zoom” between 1.8 and 2.2 Rd/s of Figure 4

Using the proposed validation by upper bound analysis a few minutes suffice to obtain the result depicted in Figure 6 (see below for a more precise comparison of computing time). The proposed algorithm permits us to conclude that the peak values of μ are less than 0.75. Therefore the highest one belongs to $[0.72 \ 0.75]$.

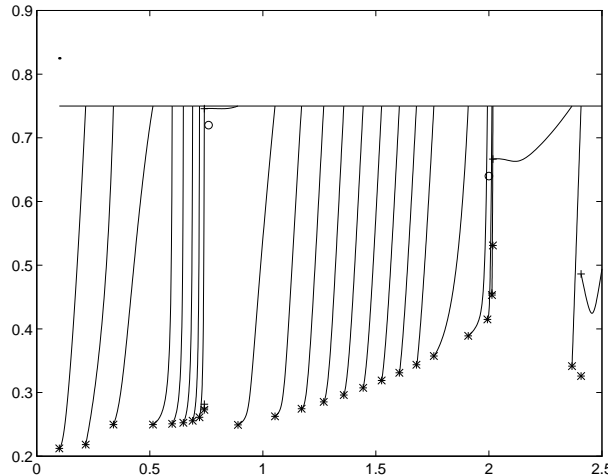


Figure 6: Illustration of the proposed algorithm (flexible satellite).

As computing time depends significantly on the computer and on the considered toolbox, the best way for estimating the advantage of using the proposed approach consists of comparing the number of pairs of scalings that are required. Using frequency gridding, about 1000 points are necessary for identifying all the peaks. With Algorithm 3, 25 points suffice. So we have divided by 40 the amount of computing time. Note that a simple adaptation of Algorithm 3 such that frequencies are eliminated on *both* sides of the peaks will lead to a much faster validation.

6.3 Some further validations

In Figure 7 the cross-section ξ_1, ξ_4 of the stability domain of the satellite is depicted, *i.e.* the set of parameters ξ_1 and ξ_4 for which the system remains stable is plotted by '+', the set of parameters which destabilizes the system is plotted by '.'. We recognize immediately that the perturbation matrices obtained by Algorithms 1 and 2 lead the system in fact to the limit of stability. Concerning the Frobenius norm approach, the result is as it has to be on the smallest hyperball (here a circle). On the other hand, the results with minimal σ_{max} norm are of course on the smallest hypercube (here a square). Especially point *B* illustrates the fact that the second algorithm is important as it has migrated by a not negligible manner from B_{Frob} to $B_{\sigma_{max}}$. In terms of point *A*, it is a special case, it does not change a lot.

The evolution of the eigenvalues during the performance of Algorithm 1 is demonstrated in Figures 8 and 9.

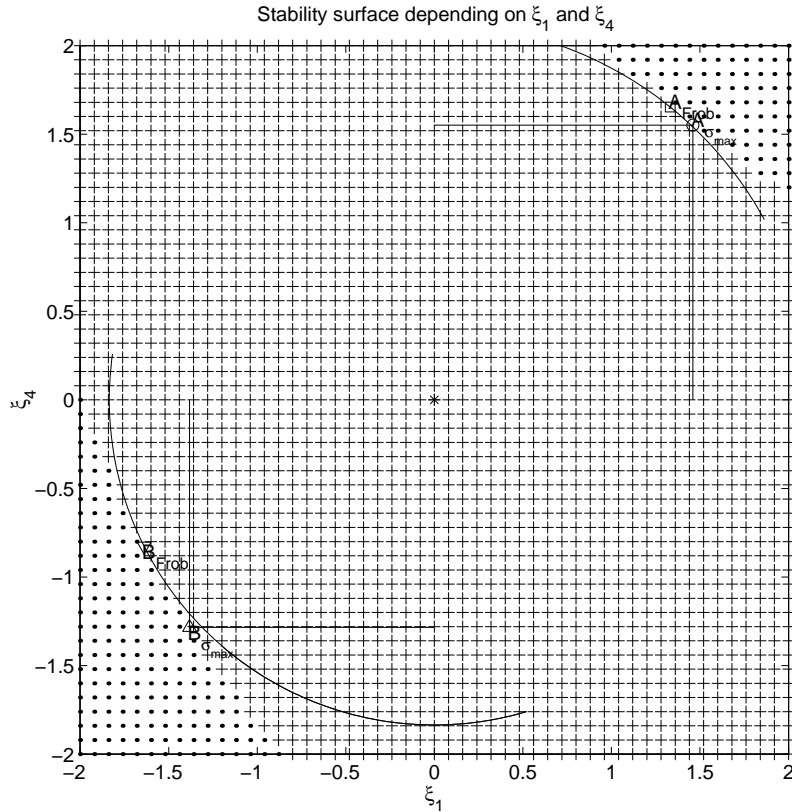


Figure 7: ξ_1, ξ_4 cross-section of the stability domain of the satellite

Conclusions:

- Figure 8 depicts the migration of pole A towards the imaginary axis when applying Algorithm 1 to pole A . In the meantime, pole B migrates to the real axis.
- Figure 9 depicts the migration of pole B towards the imaginary axis when applying Algorithm 1 to pole B . In the meantime, pole A migrates to the real axis.
- Note that the poles corresponding to the flexible modes do not move at all. Someone could remark that this is just the case for Δ in the directions of ξ_1 and ξ_4 , but investigations in other directions have confirmed that result. Retain as well that the standard upper bound algorithm produces the fancy peaks exactly at the frequencies where these poles arise (obviously scalings D and G exists to destabilize the system) even if these poles have no intention to move, *i.e.* the proof for the non-existence of the fancy peaks mentioned earlier.
- In Figure 8 one problem of Algorithm 1 is shown. Pole B moves to the real axis and reaches a separation point. Imagine that we would have tried to move this pole to the imaginary axis by a perturbation matrix Δ which effects that movement. Once the real axis is reached the first order approximation does not give enough information. Physically speaking, we do not know how to shift the eigenvalue: on its left or on its right side.

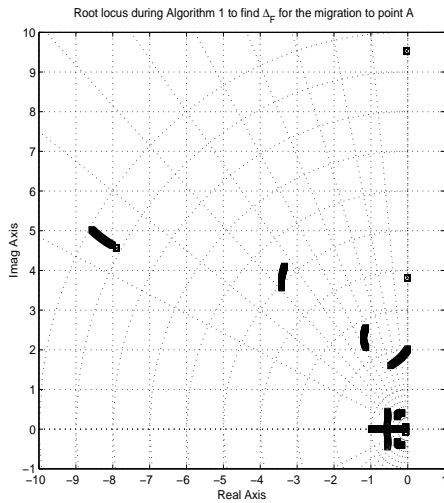


Figure 8: Evolution of eigenvalues during the performance of Algorithm 1 to shift pole A

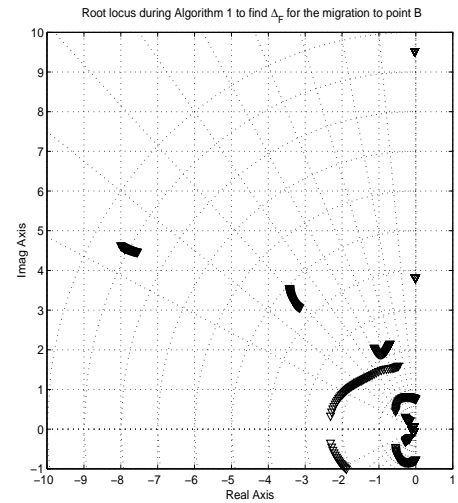


Figure 9: Evolution of eigenvalues during the performance of Algorithm 1 to shift pole B

- A further problem is not depicted in these Figures. A shifted pole can cross a pole which moves freely. A discontinuity arises. After the discontinuity, the algorithm might choose the wrong branch. Resolution: Use an additional continuity constraint related to the corresponding eigenvectors.
- Here, we have seen that a perturbation Δ_A pushes pole A to the imaginary axis and perturbation Δ_B pole B . However there exists as well cases where perturbations Δ_A and $\Delta_{A'}$ push one and the same pole A to the imaginary axis. As our algorithm chooses only once a pole, it is not able to detect both perturbations in one run. This occurs if the stability surface of Figure 7 is symmetric and is too plane for a first order approximation to determine $d\lambda$. Such a case is not fictitious. In another cross-section of the stability domain of this satellite such a problem was detected. To overcome use slight initial perturbations Δ_0 to lead the search already from the beginning or add second order terms in the linear constraint on $d\lambda$.

Similar plots as those in Figures 8 and 9 illustrates the evolution of the eigenvalues on the imaginary axis when performing Algorithm 2 as well as the evolution of eigenvalues to the origin when performing Algorithm 1'.

7 Analysis of the robustness of aircraft autopilots

In this section are reported the results of [7] and [15] relative to robustness analysis of a set of controllers. These controllers were designed using a great variety of techniques, see Table 1. In this table “MM” means multimodel eigenstructure assignment, “MO” multi-objective optimization, “EA” eigenstructure assignment, “FL” fuzzy logic, “MS” μ -synthesis, “CC” classical control, “LY” (Lya-punov) quadratic stability, “MF” model following, “HI” \mathcal{H}_∞ . The considered model is an aircraft defined in [21]. Details on the way the considered controllers were derived can be found in this book. The standard form is described in [18, 22], where four uncertain parameters (mass, horizontal and vertical position of the center of gravity, time delay) are repeated respectively 17, 15, 3 and 5 times.

Controller	frequency	lower bound	upper bound
MM-12	7.59	0.31	0.35
MO-16	6.37	0.33	0.34
EA-22	0.42	0.34	0.38
FL-15	5.82	0.41	0.43
MS-11	0.60	0.45	0.48
CC-13	0.87	0.48	0.49
LY-14	0.75	0.54	0.56
MF-25	0.65	0.63	0.64
EA-18	6.96	0.77	0.81
HI-09	0.69	0.88	0.91
MS-19	15.0	1.27	1.25
HI-21	1.28	1.49	1.50

Table 1: Comparison of the maximum value of μ using the proposed lower bound and the standard upper bound

In Figures 10-21 the solid line is the upper bound of μ with frequency gridding (MATLAB μ -Toolbox). Here the system is not flexible, so, gridding is reliable (except in Figure 16). The dashed line is the lower bound obtained (using the same toolbox). The signs “*” are the lower bounds of the peaks found using Algorithm 1 (§3.1) and Algorithm 2 (§3.2). In Figure 16 the lower bound of the leading peak is larger than the upper bound on account of gridding.

Comments. First, finding only lower bounds of the peaks was much faster than finding the upper bound. The purpose of showing all these results is to demonstrate that even for very robust controllers (taken from robust control design challenge), even if there are not sharp peaks, even if there are up to 70 states in some cases, the proposed technique works quite well. Some peaks are missed for example in Figures 11, 14, 18. But in all cases the maximum value of μ is well estimated (see Table 1). These results were obtained without tuning the algorithm parameters (number of iterations, proximity tests, options...). So, even if the proposed lower bound computation technique was tailored for flexible structures it can be used for most systems.

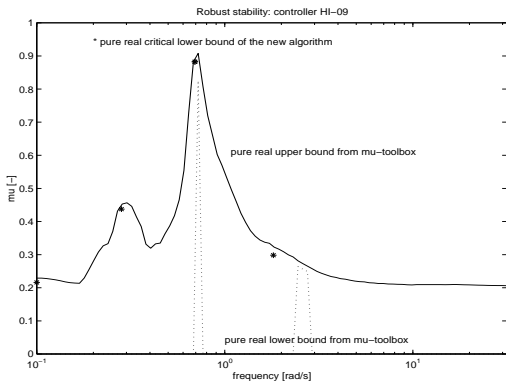


Figure 10: \mathcal{H}_∞ -synthesis (HI-09)

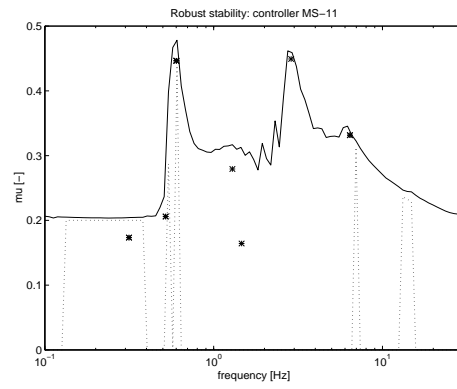


Figure 11: μ -synthesis (MS-11)

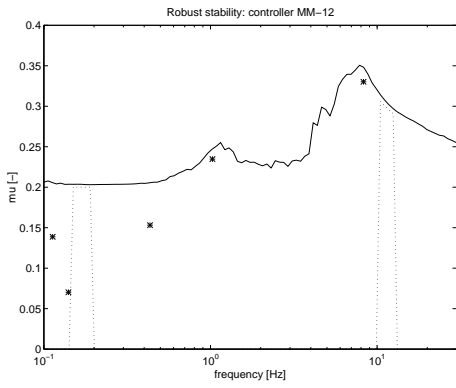


Figure 12: Modal-multimodel approach (MM-12)

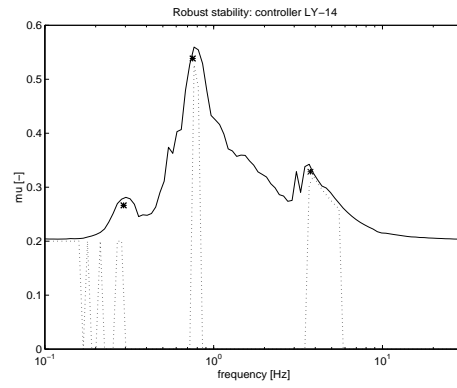


Figure 13: Quadratic stability (LY-14)

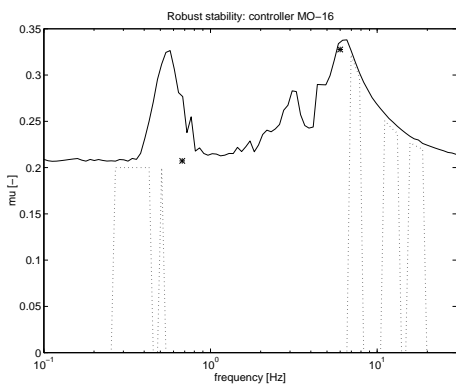


Figure 14: Multi-objective optimization (MO-16)

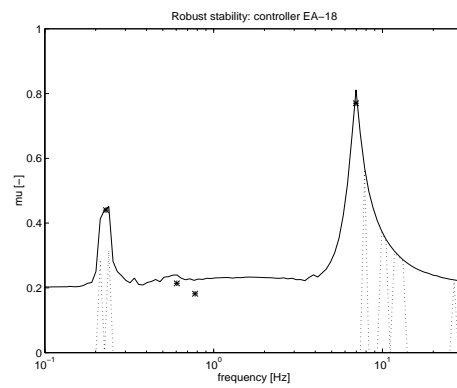


Figure 15: Eigenstructure assignment (EA-18)

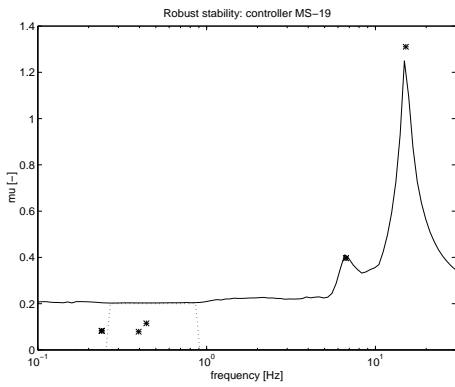


Figure 16: μ -synthesis (MS-19)

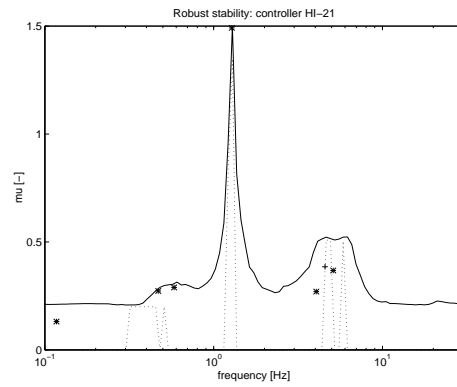


Figure 17: \mathcal{H}_∞ -synthesis (HI-21)

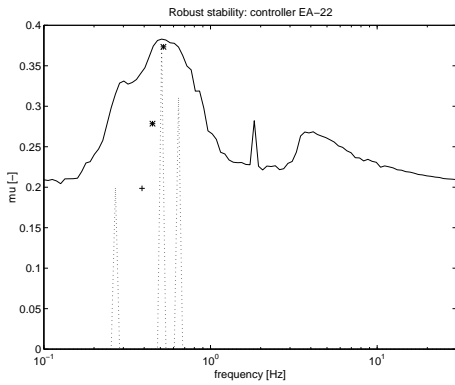


Figure 18: Eigenstructure assignment (EA-22)

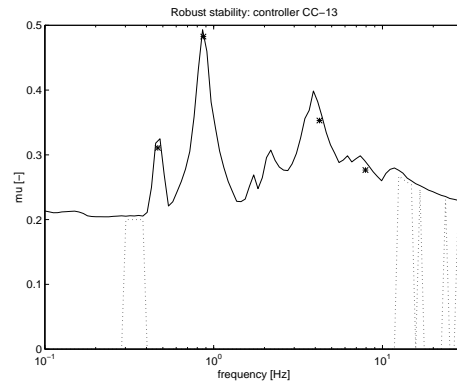


Figure 19: Classical control (CC-13)

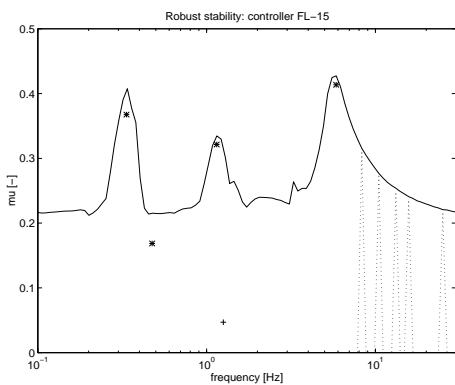


Figure 20: Fuzzy control (“linearized”) (FL-15)

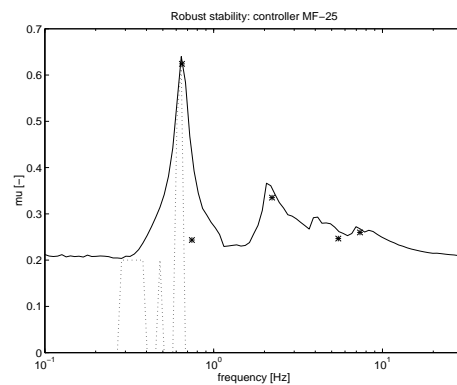


Figure 21: Model of reference (MF-25)

A Appendix: quadratic criteria and linear constraints

Two optimization problems are considered. Minimizing a combination of criteria (5) and (6) or a combination of criteria (5) and (12) under the linear equality constraint (4) (or equivalent, see §4).

To treat the problems with standard tools, these equations must be written in the following form:

$$J_0 = d\xi^T H_0 d\xi \quad (25)$$

$$J_1 = d\xi^T H_1 d\xi + 2 c_1 d\xi \quad (26)$$

$$J_2 = d\xi^T H_2 d\xi + 2 c_2 d\xi \quad (27)$$

$$A d\xi = b \quad (28)$$

with $d\xi$ a real column vector in which all the free entries of the structured matrix $d\Delta$ will appear as follows $d\xi = [\delta_1 \ \delta_2 \ \dots \ \delta_n]^T$.

A.1 Transformation of J_0 and J_1 to quadratic forms

Matrix H_0 of Equation (25) has to be defined taking into account the structure of ξ . First it is checked that J_0 and J_1 are almost similar, therefore, it will be sufficient to consider J_0 . Assume that H_0 as in (25) is known, then,

$$J_1 = (\xi_0 + d\xi)^T H_0 (\xi_0 + d\xi)$$

so

$$J_1 = d\xi^T H_0 \xi_0 + \xi_0^T H_0 d\xi + \xi_0^T H_0^T d\xi$$

plus a constant term that can be ignored. Therefore (H_0 is a real symmetric matrix) we have

$$H_1 = H_0 \quad \text{and} \quad c_1 = \xi_0^T H_0$$

The criterion J_0 can be decomposed:

$$J_0 = \text{trace}(d\Delta^* d\Delta) = \sum_i \text{trace}(d\Delta_i^* d\Delta_i)$$

the terms $\text{trace}(d\Delta_i^* d\Delta_i)$ will be considered separately and the subscript i will be omitted.

Case of repeated variables. Considering the case of a twice repeated complex variables

$$d\Delta = \begin{bmatrix} \delta_1 + j\delta_2 & 0 \\ 0 & \delta_1 + j\delta_2 \end{bmatrix}$$

The corresponding “sub-criterion” is $\text{tr}(d\Delta^* d\Delta)$ or

$$\begin{aligned} &= 2\delta_1^2 + 2\delta_2^2 \\ &= \begin{bmatrix} \delta_1 & \delta_2 \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} \\ &= \xi^T H_0 \xi \end{aligned}$$

The case of a real variable is easily derived: $\delta_2 = 0$, so, H_0 becomes $H_0 = 2$. More generally, the number “2” above has to be replaced by the number of times the variable is repeated.

Case of full blocks. Let consider a full 2×2 complex block,

$$d\Delta = \begin{bmatrix} \delta_1 + j\delta_5 & \delta_3 + j\delta_7 \\ \delta_2 + j\delta_6 & \delta_4 + j\delta_8 \end{bmatrix}$$

the “sub-criteria” will come out like

$$\text{tr} \begin{bmatrix} \delta_1 - j\delta_5 & \delta_2 - j\delta_6 \\ \delta_3 - j\delta_7 & \delta_4 - j\delta_8 \end{bmatrix} \begin{bmatrix} \delta_1 + j\delta_5 & \delta_3 + j\delta_7 \\ \delta_2 + j\delta_6 & \delta_4 + j\delta_8 \end{bmatrix}$$

being

$$(\delta_1^2 + \delta_5^2) + (\delta_2^2 + \delta_6^2) + (\delta_3^2 + \delta_7^2) + (\delta_4^2 + \delta_8^2)$$

from which the matrix H_0 follows easily: obviously, H_0 is the unit (8×8) matrix. In the real case, setting to zero $\delta_5, \dots, \delta_8$, H_0 will form a unit matrix with a dimension equal to 4. More generally, the matrix H_0 is the unit matrix of a dimension equal to the number (real case) or twice the number of elements in the perturbation matrix Δ (complex case).

Note that from the above results, it is clear that J_0 and J_1 positive definite quadratic forms.

A.2 Transformation of J_2 into a quadratic form

Now, J_2 as defined by Equation (12) is treated in order to derive the equivalent form of equation (27). The criterion to be considered is $J_2 = v^*(\Delta_0^* + d\Delta^*)(\Delta_0 + d\Delta)v$ so

$$J_2 = v^*d\Delta^*d\Delta v + 2v^*\Delta_0^*d\Delta v$$

plus a constant term that is ignored. By using the result of §A.3, the second term is easily written as

$$v^*\Delta_0^*d\Delta v = c_2 d\xi \quad (29)$$

Therefore it remains to treat the first term. So, it can be assumed that $J_2 = v^*d\Delta^*d\Delta v$. This criterion can be decomposed as $J_2 = \sum_i v_i^*d\Delta_i^*d\Delta_i v_i$. Each term of this sum will be considered separately and the subscript i will be omitted. The subscripts that will be used correspond to the components of v_i .

Case of repeated variables. For example a twice repeated complex variable is considered:

$$d\Delta = \begin{bmatrix} \delta_1 + j\delta_2 & 0 \\ 0 & \delta_1 + j\delta_2 \end{bmatrix}$$

The “sub-criterion” to be considered is:

$$\begin{bmatrix} \bar{v}_1 & \bar{v}_2 \end{bmatrix} \begin{bmatrix} \delta_1 - j\delta_2 & 0 \\ 0 & \delta_1 - j\delta_2 \end{bmatrix} \begin{bmatrix} \delta_1 + j\delta_2 & 0 \\ 0 & \delta_1 + j\delta_2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}$$

or in the standard form of Equation (27):

$$\begin{bmatrix} \delta_1 & \delta_2 \end{bmatrix} \begin{bmatrix} \bar{v}_1 v_1 + \bar{v}_2 v_2 & 0 \\ 0 & \bar{v}_1 v_1 + \bar{v}_2 v_2 \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}$$

More generally, the number of terms in the sum is equal to the number of times the uncertainty is repeated. In the real case, it suffices to set $\delta_2 = 0$, the second diagonal term disappears.

Case of full blocks. Let consider the following example:

$$d\Delta = \begin{bmatrix} \delta_1 + j\delta_5 & \delta_3 + j\delta_7 \\ \delta_2 + j\delta_6 & \delta_4 + j\delta_8 \end{bmatrix}$$

The corresponding ‘‘sub-criterion’’ is:

$$\begin{aligned} J_2 &= \begin{bmatrix} v_1 & v_2 \end{bmatrix} \begin{bmatrix} \delta_1 + j\delta_5 & \delta_3 + j\delta_7 \\ \delta_2 + j\delta_6 & \delta_4 + j\delta_8 \end{bmatrix} \begin{bmatrix} \delta_1 - j\delta_5 & \delta_2 - j\delta_6 \\ \delta_3 - j\delta_7 & \delta_4 - j\delta_8 \end{bmatrix} \begin{bmatrix} \bar{v}_1 \\ \bar{v}_2 \end{bmatrix} \\ &= \begin{bmatrix} v_1 & v_2 \end{bmatrix} \left(\begin{bmatrix} \delta_1 + j\delta_5 \\ \delta_2 + j\delta_6 \end{bmatrix} \begin{bmatrix} \delta_1 - j\delta_5 & \delta_2 - j\delta_6 \end{bmatrix} + \begin{bmatrix} \delta_3 + j\delta_7 \\ \delta_4 + j\delta_8 \end{bmatrix} \begin{bmatrix} \delta_3 - j\delta_7 & \delta_4 - j\delta_8 \end{bmatrix} \right) \begin{bmatrix} \bar{v}_1 \\ \bar{v}_2 \end{bmatrix} \end{aligned}$$

after commutation of scalar numbers

$$J_2 = \begin{bmatrix} \delta_1 - j\delta_5 & \delta_2 - j\delta_6 \end{bmatrix} H_2 \begin{bmatrix} \delta_1 + j\delta_5 \\ \delta_2 + j\delta_6 \end{bmatrix} + \begin{bmatrix} \delta_3 - j\delta_7 & \delta_4 - j\delta_8 \end{bmatrix} H_2 \begin{bmatrix} \delta_3 + j\delta_7 \\ \delta_4 + j\delta_8 \end{bmatrix}$$

where

$$H_2 = \begin{bmatrix} \bar{v}_1 \\ \bar{v}_2 \end{bmatrix} \begin{bmatrix} v_1 & v_2 \end{bmatrix}$$

So

$$J_2 = \begin{bmatrix} \delta_1 & \delta_2 \end{bmatrix} H_2 \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} \delta_5 & \delta_6 \end{bmatrix} H_2 \begin{bmatrix} \delta_5 \\ \delta_6 \end{bmatrix} - j \begin{bmatrix} \delta_1 & \delta_2 \end{bmatrix} H_2 \begin{bmatrix} \delta_5 \\ \delta_6 \end{bmatrix} + \dots$$

So, the considered criterion can be written:

$$\xi^T H_2 \xi$$

where

$$H_2 = \begin{bmatrix} H_2 & 0 & -jH_2 & 0 \\ 0 & -H_2 & 0 & -jH_2 \\ -jH_2 & 0 & H_2 & 0 \\ 0 & -jH_2 & 0 & -H_2 \end{bmatrix}$$

The standard form of Equation (27) is obtained. In the case of full real blocs, it suffices to set to zeros $\delta_5, \dots, \delta_8$ and to remove superfluous blocs in the matrix H_2 . The generalization is obvious.

Note that (unlike H_0), the matrix H_2 is singular because H_2 is a matrix of rank one.

A.3 The transformation of the linear equality constraint

Now are considered Equations (7), (13), (18), (29) and (31). These equations can be written under the form $x^* \Delta y = b$ or $\sum_i x_i^* \Delta_i y_i = b$. The terms $x_i^* \Delta_i y_i$ will be treated separately. The subscript i will be omitted. The subscript that will be used correspond to the components of x_i and y_i . We have to detail the matrix A of equation (28).

Case of repeated variables. A twice repeated complex variable can be expressed as follows

$$\begin{aligned} & \begin{bmatrix} \bar{x}_1 & \bar{x}_2 \end{bmatrix} \begin{bmatrix} \delta_1 + j\delta_2 & 0 \\ 0 & \delta_1 + j\delta_2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \\ &= \bar{x}_1\delta_1y_1 + j\bar{x}_1\delta_2y_1 + \bar{x}_2\delta_1y_2 + j\bar{x}_2\delta_2y_2 \\ &= \begin{bmatrix} (\bar{x}_1y_1 + \bar{x}_2y_2) & j(\bar{x}_1y_1 + \bar{x}_2y_2) \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} \end{aligned}$$

More generally $A = [x^*y \ jx^*y]$. In the real case $\delta_2 = 0$ so $A = x^*y$.

Case of full blocks. Considering a complex full block, it emerges

$$\begin{aligned} & \begin{bmatrix} \bar{x}_1 & \bar{x}_2 \end{bmatrix} \begin{bmatrix} \delta_1 + j\delta_5 & \delta_3 + j\delta_7 \\ \delta_2 + j\delta_6 & \delta_4 + j\delta_8 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \\ &= \bar{x}_1\delta_1y_1 + \bar{x}_2\delta_2y_1 + \bar{x}_1\delta_3y_2 + \bar{x}_2\delta_4y_2 \\ &\quad + j(\bar{x}_1\delta_5y_1 + \bar{x}_2\delta_6y_1 + \bar{x}_1\delta_7y_2 + \bar{x}_2\delta_8y_2) \\ &= \begin{bmatrix} y_1[\bar{x}_1 \ \bar{x}_2] & y_2[\bar{x}_1 \ \bar{x}_2] & j y_1[\bar{x}_1 \ \bar{x}_2] & j y_2[\bar{x}_1 \ \bar{x}_2] \end{bmatrix} \begin{bmatrix} \delta_1 \\ \delta_2 \\ \vdots \\ \delta_8 \end{bmatrix} \end{aligned}$$

More generally

$$A = [1 \ j] \otimes (y^T \otimes x^*)$$

(where \otimes is the Kronecker product). The real case is similar:

$$A = y^T \otimes x^*$$

A.4 Coalescence of singular values in Algorithm 2

For simplicity, the algorithm of §3.2 was presented ignoring the fact that, at the optimum, usually more than two singular values become equal. Let ξ denote, as above, the vector of uncertainties appearing in Δ . At each step of Algorithm 2, a criterion is optimized subject to a linear constraint

$$A d\xi = b \tag{30}$$

(this constraint insure that a pole remains on the imaginary axis). When two or more singular values of the blocks Δ_i of Δ are equal to the maximum singular value of Δ some additional constraints must be used in order to ensure that these singular values vary simultaneously of the same amount.

The singular value decomposition of Δ is

$$\Delta = U S V^*$$

where $U^*U = I$ and $V^*V = I$, clearly

$$S = U^* \Delta V$$

As S a real diagonal matrix $S = \text{diag}\{\sigma_1, \sigma_2, \dots\}$:

$$\sigma_i = u_i^* \Delta v_i$$

So, the first order variation is (see [17], p. 419)

$$d\sigma_i = \Re(u_i^* d\Delta v_i) \quad (31)$$

Using the results of Appendix A.3, this equations can be written

$$d\sigma_i = A_i d\xi \quad (32)$$

for some row vector A_i . If for example

$$\sigma_1 = \sigma_2 = \dots = \sigma_i, \quad 2 \leq i \leq n \quad (33)$$

in order to preserve this property, we must have:

$$d\sigma_1 = d\sigma_2 = \dots = d\sigma_i$$

Using Equation (32)

$$\begin{bmatrix} A_1 - A_2 \\ \vdots \\ A_1 - A_i \end{bmatrix} d\xi = \mathbf{0}$$

So, considering also (30),

$$\begin{bmatrix} A \\ A_1 - A_2 \\ \vdots \\ A_1 - A_i \end{bmatrix} d\xi = \begin{pmatrix} b \\ 0 \end{pmatrix} \quad (34)$$

Finally, in order to adapt Algorithm 2 to singular value coalescence, it suffices to replace Equation (30) by Equation (34).

B Appendix: Use of the power algorithm instead of Algorithm 2

In [26] the lower bound problem of $\mu(M(j\omega))$ is translated into an alignment problem of left and right eigenvectors at a global maximum of the real spectral radius problem³, $\bar{\lambda}_R(QM(j\omega))$ where Q is a normalized element of the set of admissible perturbations Δ . It is demonstrated in this reference that the standard power algorithm (SPA) like stated in Table 2 (without entering into details) gives a solution to the lower bound problem of μ .

In the algorithm of Table 2, the vectors b , a , z , w are partitioned according to the bloc structure of Δ as

$$b = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \quad a = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \quad z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix} \quad w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \quad (38)$$

if

$$\Delta = \begin{bmatrix} \delta_r & 0 & 0 \\ 0 & \delta_c & 0 \\ 0 & 0 & \Delta_C \end{bmatrix}$$

The interest of this algorithm is that if it converges, it converges very fast and gives a reliable and accurate result frequency by frequency.

Besides the inherent problem of frequency gridding, its big problem is if the number of complex blocs is zero, *i.e.* a pure real problem, then the SPA converges very badly. Even if the number of repeated scalars δ_r , δ_c is greater than the number of complex full blocs Δ_C the convergence characteristics are not so well. Just in the case where the number of complex full blocs Δ_C is significantly higher than those of the scalar blocs the algorithm converges very well.

One reason seems to be the initialization which is a bit arbitrary. Another problem are limit cycles arising even at equilibrium points for β_i .

Hence, we thought to overcome the problems by first applying a better initialization and second by offering the possibility of frequency sweeping.

Proposed initialization. Only the pure real case is considered, so the vectors with subscripts 2 or 3 in Equation (38) are not initialized. After applying Algorithm 1, a matrix Δ_F with minimum Frobenius norm and such that a pole is at the limit of stability ($j\omega_F$) is known.

- Choose $\beta = \hat{\beta}_1 = \tilde{\beta}_1 = \bar{\sigma}(\Delta_F)$ as the lower bound found by using Algorithm 1.
- Choose $Q = \Delta_F / \bar{\sigma}(\Delta_F)$ as a normalized perturbation matrix, then set $\tilde{q}_1 = Q$ and $\hat{q}_1 = Q$.
- The initialization for b_0 et w_0 is as follows. Choose $M = M(j\omega_F) = C(j\omega_F I - A)^{-1} B + D$ with ω_F the frequency where a pole achieves the limit of stability by Δ_F

By inspection of the algorithm of Table 2, after convergence (omitting the subscripts):

$$\begin{array}{ll} Mb & = \beta a \\ b & = Qa \end{array} \quad \begin{array}{ll} M^* z & = \beta w \\ z & = Q^* w \end{array}$$

³The real spectral radius is the maximum real eigenvalue: $\bar{\lambda}_R(\cdot)$

Mixed SPA.

1. **Initialization:** Choose an initial value for M , b_0 , w_0 , $\tilde{\beta}_1$, $\hat{\beta}_1$, \tilde{q}_1 and \hat{q}_1 .

2. **Iterations:**

(a) **Actualization:**

$$\begin{aligned}
 \tilde{\beta}_{k+1} a_{k+1} &= M b_k \\
 z_{1_{k+1}} &= \tilde{q}_{k+1} w_{1_k} \\
 z_{2_{k+1}} &= \frac{w_{2_k}^* a_{2_{k+1}}}{|w_{2_k}^* a_{2_{k+1}}|} w_{2_k} \quad , \quad z_{3_{k+1}} = \frac{\|w_{3_k}\|}{\|a_{3_{k+1}}\|} a_{3_{k+1}} \\
 \hat{\beta}_{k+1} w_{k+1} &= M^* z_{k+1} \\
 b_{1_{k+1}} &= \hat{q}_{k+1} a_{1_{k+1}} \\
 b_{2_{k+1}} &= \frac{a_{2_{k+1}}^* w_{2_{k+1}}}{|a_{2_{k+1}}^* w_{2_{k+1}}|} a_{2_{k+1}} \quad , \quad b_{3_{k+1}} = \frac{\|a_{3_{k+1}}\|}{\|w_{3_{k+1}}\|} w_{3_{k+1}}
 \end{aligned} \tag{35}$$

where $\tilde{\beta}_{k+1}$ and $\hat{\beta}_{k+1}$ are initialized > 0 such that $\|a_{k+1}\| = \|w_{k+1}\| = 1$ and with \tilde{q}_{k+1} and \hat{q}_{k+1} resulting from

$$\begin{aligned}
 \tilde{\alpha}_{k+1} &= \text{sign}(\hat{q}_k) \frac{|b_{1_k}|}{|a_{1_{k+1}}|} + \Re(a_{1_{k+1}}^* w_{1_k}) \\
 \text{If } |\tilde{\alpha}_{k+1}| &\geq 1 \text{ then } \tilde{q}_{k+1} = \frac{\tilde{\alpha}_{k+1}}{|\tilde{\alpha}_{k+1}|} \\
 &\text{else } \tilde{q}_{k+1} = \tilde{\alpha}_{k+1}
 \end{aligned} \tag{36}$$

$$\begin{aligned}
 \hat{\alpha}_{k+1} &= \text{sign}(\tilde{q}_{k+1}) \frac{|b_{1_k}|}{|a_{1_{k+1}}|} + \Re(a_{1_{k+1}}^* w_{1_{k+1}}) \\
 \text{If } |\hat{\alpha}_{k+1}| &\geq 1 \text{ then } \hat{q}_{k+1} = \frac{\hat{\alpha}_{k+1}}{|\hat{\alpha}_{k+1}|} \\
 &\text{else } \hat{q}_{k+1} = \hat{\alpha}_{k+1}
 \end{aligned}$$

(b) **Verification of convergence**

3. **Calculation of μ and q_r :**

$$\begin{aligned}
 \mu &= \max(\tilde{\beta}, \hat{\beta}) \\
 q_r &= \frac{q(\mu)}{\mu}
 \end{aligned} \tag{37}$$

Table 2: Standard Power Algorithm

and hence

$$(MQ - \beta I) a = 0 \quad \text{and} \quad (M^* Q^* - \beta I) w = 0 \quad (39)$$

Clearly, $\det(I - M\Delta_F) = 0$, so, $\det(\beta I - MQ) = 0$, therefore a and w can be computed as the right and left (normalized) eigenvectors of the matrix MQ corresponding to the eigenvalue β . The initialization of b_0 and w_0 follows:

$$b_0 = Qa, \quad w_0 = w$$

Frequency sweep version. Here, a frequency gridding around the fixed frequency ω_F is necessary. Let us use the equivalence of the loop in Figure 1 with:

$$\widehat{M} = \begin{bmatrix} -jA & B \\ -jC & D \end{bmatrix}; \quad \widehat{\Delta} = \begin{bmatrix} \delta I & 0 \\ 0 & \Delta \end{bmatrix} \quad (40)$$

already proved in §4 with $\delta = 1/\omega$. Since the frequency sweep should start at ω_F and a perturbation matrix Δ_F , suppose

$$\delta = \delta_0 + \delta_\omega \quad (41)$$

with

$$\begin{aligned} \delta_0 &= 1/\omega_F \\ \delta_\omega &= 0 \\ \Delta &= \Delta_F \end{aligned}$$

during the initialization. If we introduce δ_0 in the complex matrix \widehat{M} we obtain new matrices \overline{M} and $\overline{\Delta}$ as follows

$$\overline{M} = \begin{bmatrix} \overline{A} & \overline{B} \\ \overline{C} & \overline{D} \end{bmatrix}; \quad \overline{\Delta} = \begin{bmatrix} \delta_\omega I & 0 \\ 0 & \Delta \end{bmatrix}$$

with

$$\begin{aligned} H &= \delta_0 (I + j\delta_0 A)^{-1} \\ \overline{A} &= -jA - AHA \\ \overline{B} &= B - jAHB \\ \overline{C} &= -jC - CHA \\ \overline{D} &= D - jCHB \end{aligned}$$

The rest of the aforementioned real SPA remains unchanged, *i.e.* $Q = \overline{\Delta}$ and $\beta = \mu_F$. After the iterations, isolate δ_ω and Δ from Q and calculate the critical frequency ω belonging to Δ and μ as follows:

$$\omega = \frac{1}{\delta_0 + \delta_\omega}$$

Concluding remarks. The convergence characteristics are much better:

- It is now possible to find very sharp peaks which have been missed by gridding.

- Also in pure real problems (where no lower bound was given with SPA) an accurate lower bound is determined. If the SPA converged even in the pure real case, the new algorithm converges as well to the same value.

But nevertheless, there remains convergence problems due to the above mentioned limit cycles on b_i , a_i , z_i and w_i while β_i has already converged even if initialized near the global maximum. Probably, this comes from the fact that in certain cases $\bar{\lambda}_R(QM) \ll \bar{\lambda}(QM)$ for the Q finding the maximum of $\bar{\lambda}_R(QM)$ so that during the actualization the SPA magnifies the component of the spectral radius in place of the real spectral radius. A small error on b_i , a_i , z_i and w_i is hence amplified and thus, the SPA risks to diverge, see Tierno, Young [25]. Furthermore, the changes are not yet adapted to mixed perturbations.

Henceforth, if we want a reliable result in all cases, we have to develop a completely new algorithm, for example Algorithm 2.

References

- [1] C. Beck and J.C. Doyle. Mixed μ upper bound computation. In *Proc. 31st IEEE Conference on Decision and Control*, pages 3187–3192, Tucson, Arizona, USA, December 1992.
- [2] R.P. Braatz, P.M. Young, J.C. Doyle, and M. Morari. Computational complexity of μ calculation. *IEEE Transactions on Automatic Control*, AC-39(5):1000–1002, 1994.
- [3] T.L. Craig, A.L. Tits, and P. Van Dooren. A fast algorithm for the computation of an upper bound of the μ -norm. In *Proc. 13th Triennial IFAC World Congress, San Francisco, USA*, D:59–64, July 1996.
- [4] R.L. Dailey. A new algorithm for the real structured singular value. in *Proceedings American Control Conference*, 1990.
- [5] J. David and B. De Moor. On the calculation of the Euclidian parameter margin. In *Proc. European Control Conference*, pages 380–384, 1993.
- [6] B. De Moor. Structured total least square and l2 approximation problems. *Linear Algebra and its Applications*, 188:163–205, 1993.
- [7] C. Döll, J.F. Magni, G. Looye, and S. Bennani. Robustness analysis applied to autopilot design. part 2: Evaluation of a new tool for μ -analysis. *ICAS Congress, Melbourne, Australia*, September 1998.
- [8] M.K.H. Fan, A.L. Tits, and J.C. Doyle. Robustness in the presence of mixed parametric uncertainties and unmodelled dynamics. *IEEE Transactions on Automatic Control*, AC-36(1):25–38, 1991.
- [9] E. Feron. Robustness of linear systemes against parametric uncertainties: towards consistent stability indicators. In *Proc. of the 34th Conf. on Decision and Contr., New Orleans*, pages 1425–1430, 1995.
- [10] G. Ferreres. *A Practical Approach to Robustness Analysis with Aeronautical Applications*. Kluwer Academics/Plenum Press, 1999.
- [11] G. Ferreres and J.M. Biannic. A μ analysis technique without frequency gridding. In *Proc. of the American Control Conference*, 1998.
- [12] G. Ferreres and J.F. Magni. Robustness analysis using the mapping theorem without frequency gridding. In *Proc. IFAC World Congress, San Francisco*, Vol H:7–12, 1996.
- [13] M. Fu. The real structured singular value is hardly approximable. *IEEE Transactions on Automatic Control*, 42(9):1286–1288, September 1997.
- [14] M. Gauvrit and D. Alazard. Parametric worst-case analysis by prabi method. application to flexible space structures. In *Proc. 2th IFAC Symposium on Robust Control Design, ROCOND'97, Budapest, Hungary*, June 1997.

- [15] G. Grübel. An other view on the design challenge achievements. In “*Robust Flight Control*” *Lecture Note in Control and Information Sciences Springer-Verlag*, 224:603–609, 1997.
- [16] A. Helmersson. A finite frequency method for μ -analysis. In *Proc. 3rd European Control Conference, Rome, Italy*, pages 171–176, 1995.
- [17] J.L. Junkins and Y. Kim. *Introduction to dynamics and control of flexible structures*, volume 32 of *Education Series*. AIAA, Washington, DC, USA, 1st edition, 1993.
- [18] G. Looye, S. Bennani, and Grübel G. Robustness analysis applied to autopilot design. part 1: μ -analysis of design entries to a robust flight control benchmark. *ICAS Congress, Melbourne, Australia*, September 1998.
- [19] J.F. Magni, Y. Le Gorrec, and C. Chiappa. A multimodel-based approach to robust and self-scheduled control design. In *Proc. 37th I.E.E.E. Conf. Decision Contr., Tampa, Florida*, pages 3009–3014, 1998.
- [20] J.F. Magni and A. Manouan. Robust flight control design by eigenstructure assignment. In *Proc. of the IFAC Symposium on Robust Control, Rio de Janeiro, Brasil*, pages 388–393, September 1994.
- [21] J.F. Magni, J. Terlouw, and S. Bennani (Eds.). *Robust Flight Control*. Lecture Notes in Control and Information Sciences, No 224. Springer-Verlag, 1997.
- [22] D. Moormann, A. Varga, G. Looye, and Grübel G. Robustness analysis applied to autopilot design. part 3: automated generation of LFT-based parametric uncertainty descriptions applied to the RCAM aircraft model. *ICAS Congress, Melbourne, Australia*, September 1998.
- [23] A. Packard and J.C. Doyle. The complex structured singular value. *Automatica*, 29(1), 1993.
- [24] A. Sideris. Elimination of frequency search from robustness tests. *IEEE Transactions on Automatic Control*, AC-37(10):1635–1640, 1992.
- [25] J.E. Tierno and P.M. Young. An improved μ lower bound via adaptive power iteration. in *Proceedings Conference on Decision and Control*, pages 3181–3186, 1992.
- [26] P.M. Young and J.C. Doyle. Computation of μ with real and complex uncertainties. in *Proceedings Conference on Decision and Control*, pages 1230–1235, 1990.
- [27] P.M. Young, M.P. Newlin, and J.C. Doyle. Computing bounds for the mixed μ problem. *International Journal of Robust and Nonlinear Control*, 5(6):573–590, 1995.