Skew Mu Toolbox (SMT): improvements and additional tools

G. Ferreres and J-M. Biannic
ONERA-CERT / DCSD
BP 4025, F-31055 Toulouse Cedex 4
ferreres@cert.fr, biannic@cert.fr

October 2004

Abstract

This second version contains several improvements of the toolbox. It moreover contains additional tools for the treatement of largely repeated parametric uncertainties. The more repeated the parametric uncertainties are, the higher the computational requirement is when computing the mixed μ upper bound with an LMI solver. Nevertheless the routine *mu.m* of the μ Analysis and Synthesis Toolbox provides a (suboptimal) value of the mixed μ upper bound with a reasonable computational requirement at a single frequency point. On the basis of *mu.m* we propose different efficient methods, either to eliminate frequency intervals inside which μ is guaranteed to be below a given threshold, or to compute a guaranteed μ upper bound over a union of frequency intervals. *Note that this document only contains additional notes w.r.t. the first one, which should be read before.*

Available on the web pages :

http://www.cert.fr/dcsd/idco/perso/Biannic/mypage.html
http://www.cert.fr/dcsd/idco/perso/Ferreres/index.html

1 Structure of the document

Acronyms: LHP (Left Half Plane), LFR (Linear Fractional Representation), LFT (Linear Fractional Transformation), LTI (Linear Time Invariant), RHP (Right Half Plane), s.s.v. (structured singular value).

The document is structured as follows. The second section describes miscellaneous improvements with respect to the first version of the Skew Mu Toolbox. The third section, which is the main one, proposes two computational methods for the case of largely repeated parametric uncertainties. The first technique eliminates frequency intervals inside which the (skew) s.s.v. is guaranteed to be less than a given value, while the second one computes a guaranteed μ upper bound over a union of frequency intervals [4, 3, 2], noting that this second technique is not applicable to skew μ problems. In both cases the computational burden is minimised, and the routine *mu.m* of the μ Analysis and Synthesis Toolbox is used as a basis. The 4th section summarises the methods to compute bounds of the (skew) s.s.v. either at a single frequency point, or on a frequency interval. The last section proposes examples of calls to the main routines of the toolbox. Remember all these routines can also be interactively called with the routine *robust.m*, whose interface is very simple: if $M(s)-\Delta(s)$ is the standard interconnection structure, it's enough to enter the structure of the model perturbation $\Delta(s)$ and a state-space representation of $M(s)$.

2 Improvements

Note as a preliminary that at least Matlab 6.1 must be used. The LMI Control Toolbox is required as an LMI solver, as well as the basic Control System Toolbox (and Simulink to some extent). The μ Analysis and Synthesis Toolbox is not necessary, but a few computational tools can not be used without it. Here is a summary of the modifications w.r.t. the first version of the toolbox:

- *first_thing_to_do.m*, which defined the Matlab path in the first version, is replaced by *path2SMT.m*, whose content is to be adjusted by the user as a preliminary, before using the toolbox. No question is asked concerning the availability of the μ Analysis and Synthesis Toolbox, its existence is automatically detected inside the routines.
- The directory *Mu_max* is deeply modified :
  - *Mu_max/mu_max_1.m*, which computes a guaranteed μ upper bound over a union of frequency intervals, is extended to the case of a truncated sector (the preliminary version only dealt with the Left Half Plane). As in the whole toolbox the degree of stability alfa is *negative* for a stable plant.

  - $Mu\_max/mu\_max\_2.m$ still only deals with the LHP, but routines $Mu\_max/mu\_max\_1f.m$ and $Mu\_max/mu\_max\_3.m$, which did not exist in the first version (see the next section), deal with the LHP and with a truncated sector.
  - Sub-routines (*calc_interval.m* and *testdg.m*) related to the scalings validation problem have been extended to handle the case of a truncated sector.
  - Specific Sub-routines of *mu_max_3.m* have been added to perform specific operations (reduction, length computation or bounding) on a list of intervals. These routines are : *reduc_tab.m*, *length_tab.m* and *bnd_interval.m*.

- A directory *Mu_elim* is added, corresponding to the elimination of frequency intervals inside which the (skew) s.s.v. is guaranteed to be less than a given value. The main routine is *mu_elim.m*, which only deals with the LHP.

- *Routines/skew_mu_ub.m* and *Routines/skew_mu_ub_bis.m* are modified for the special case of some robust performance problems. In the case of skew μ problems, if the part of the model perturbation Δ whose size is free is just a full complex block or a non-repeated complex scalar (while the remaining part of Δ is maintained inside its unit ball), it's useless to solve a generalized eigenvalue problem with the LMI Control Toolbox. Minimizing a linear objective is enough.

- The structure of the subdirectories is deeply modified. The directory *Delays* no more exists. Its content is now in the main directory $SMTv2$, except $worst\_case\_margin.m$ in the directory *Routines* which is slightly modified. Moreover the directories *Routines_1*, *Routines_2_no_musyn* and *Routines_2_with_musyn* are merged into a single one *Routines*. Inside this directory, routines $calc\_freq\_resp.m$, *mixed_mu_lb.m*, *mixed_mu_ub.m* and $mu\_frequency\_gridding.m$ are (slightly) modified, while *gen_grid.m* and $mu\_ub2.m$ are added.

- In the Applications directory, a new *.mat* file has been added : *Applications/generic_rep_unc.mat*. It contains a high-order system ($n = 60$) with poorly damped flexible modes. Furthermore, the structure Δ describes highly repeated parametric uncertainties. This challenging example was built by following a specific procedure which is similar to what can be found in $demo\_ACC98.m$ (see the main directory). It is devoted to illustrate "LMI free" techniques which are available in $Mu\_max/mu\_max\_1f.m$ and $Mu\_max/mu\_max\_3.m$.

3 The case of largely repeated uncertainties

Let $M(s) - \Delta(s)$ the standard interconnection structure. If $\Delta(s)$ contains repeated real or complex uncertainties $\delta_i I_{q_i}$, the more repeated the scalar uncertainties are (i.e. the larger the q_i are), the larger the computational time of the LMI solver is. A solution is to use the routine $mu.m$ of the μ Analysis and Synthesis Toolbox, which provides a (suboptimal) value of the mixed μ upper bound with a reasonable computational requirement at a single frequency point: an option of $mu.m$ enables to tune the degree of (sub)optimality of the result (see the help of this routine), and the less suboptimal the result is, the larger the computational time is. On the basis of $mu.m$ we propose either to eliminate frequency intervals, inside which the s.s.v. μ is guaranteed to be less than a given value, or to compute a guaranteed μ upper bound over a union of frequency intervals. In both cases the computational burden is minimised.

3.1 Elimination of frequency intervals satisfying a μ test

We just give the principle of the method, the reader is referred to the help of the routine $mu\_elim.m$ for further details. The issue is to eliminate frequency intervals, inside which $\mu(M(j\omega))$ is guaranteed to be less than a given value γ. To this aim a suboptimal value μ_0 of the mixed μ upper bound is computed at a frequency point ω_0, as well as associated D_0, G_0 scaling matrices. Assume that $\mu_0 < \gamma$, and let $\mu_{D,G}(M(j\omega))$ the value of the mixed μ upper bound at frequency ω, for a given value (D, G) of scaling matrices. The maximal size frequency interval around ω_0 is computed, inside which $\mu_{D_0,G_0}(M(j\omega))$ is less than γ: thus $\mu(M(j\omega))$ is less than γ on this frequency interval. A suboptimal value μ_1 of the mixed μ upper bound is then computed at a new frequency point $\omega_1 \ldots$

Remarks:
(i) It appears that suboptimal values of the mixed μ upper bound provide better results than an optimal one, i.e. the frequency interval around ω_0, inside which $\mu_{D_0,G_0}(M(j\omega))$ is less than γ, is larger. But if the value is too suboptimal, it may happen that $\mu_0 \geq \gamma$. As a consequence, in the routine $mu\_elim.m$, the choice of the option of $mu.m$ is crucial: our experience is that good results are obtained with opt = 'f', 'u' or 'c', while opt = 'C' which corresponds to a higher accuracy is not advised. it is moreover possible to add complex uncertainties to the original one, i.e. the scaling matrices, which are computed for the new regularised problem, are suboptimal for the original one. The tuning parameter *epsilon* of $mu\_elim.m$ corresponds to the quantity of complex uncertainties. $epsilon = 0.01$ (i.e. 1 %) is advised, but in some cases very bad results are obtained when introducing complex uncertainties, so that $epsilon = 0$ is to be chosen.
(ii) The case of skew μ is very simple to handle, since the skew s.s.v. $\nu(M)$ is less than γ if the classical s.s.v. $\mu(\begin{bmatrix} I & 0 \\ 0 & \gamma \end{bmatrix} M)$ is less than 1 (the partition $\begin{bmatrix} I & 0 \\ 0 & \gamma \end{bmatrix}$ corresponds to the parts of the structured model perturbation Δ, which are maintained inside the unit ball or whose size is free). It just becomes a classical μ test.

3.2 Computation of the maximal value of μ

In this second version of the Toolbox, 4 different routines are now available to compute a reliable estimate of the maximal value of μ over a frequency range (from which the robustness margin is immediately deduced). As already mentioned, routines $mu\_max\_1.m$ and $mu\_max\_2.m$ already existed in the previous version of the Toolbox. They are both based on the LMI solver.

We focus here on $mu\_max\_1f.m$ and more specifically on $mu\_max\_3.m$ which have been developed essentially to cope with largely repeated uncertainties in high-order (flexible) systems.

- Routine $mu\_max\_1f.m$ and $mu\_max\_1.m$ are based on the same algorithm. In both cases, a μ upper-bound is computed so as to be valid simultaneously at two distinct (but rather close) frequency points. Consequently the associated scaling matrices (D and G) are suboptimal which makes them more "robust" versus frequency variations and greatly improves the performances of the frequency elimination technique. However, in the new implementation $mu\_max\_1f.m$, the computation is based on $mu.m$. Consequently, this routine may still be applied for largely repeated uncertainties.

- Routine $mu\_max\_3.m$ implements a more sophisticated algorithm (see [1] for more details), which uses an improved frequency elimination technique. Thus, larger frequency intervals may be eliminated at each step and the computational-time is further reduced. Note that the main purpose of this routine is to compute the maximum value of a μ upper-bound over a frequency range. Therefore, secondary peaks will not be necessarily detected. If such an information is required, tuning options can be used (see the help of the function). Moreover, the accuracy may also be improved by switching to LMI techniques on the most critical frequency segments. This option is to be avoided in case of largely repeated uncertainties.

Note that these new routines are not applicable to skew μ problems, since the routine mu.m only computes classical μ bounds, not skew μ ones.

4 Summary of the routines

	M or $M(s)$	μ or skew μ	LHP or sector	μ lower or up. bound	real, complex or mixed	polynomial or exp. time
mu_lb_with_freq.m	$M(s)$	μ	sector	lower	real	polynomial
mu_max_1.m	$M(s)$	skew μ	sector	upper	mixed	polynomial
mu_max_1f.m	$M(s)$	μ	sector	upper	mixed	polynomial
mu_max_2.m	$M(s)$	skew μ	LHP	upper	mixed	polynomial
mu_max_3.m	$M(s)$	μ	sector	upper	mixed	polynomial
mu_elim.m	$M(s)$	skew μ	LHP	upper	mixed	polynomial
mixed_mu_lb_freq.m	$M(s)$	skew μ	sector	lower	mixed	polynomial
mixed_mu_lb.m	M	skew μ	sector	lower	mixed	polynomial
mixed_mu_ub.m	M	skew μ	sector	upper	mixed	polynomial
mu_dailey.m	M	skew μ	sector	lower	real	exponential
mu_zd.m	M	skew μ	sector	upper	real	exponential
worst_case_margin.m	$M(s)$	skew μ	LHP	upper	mixed	polynomial

Table 1: characteristics of the (skew) μ methods.

In a few words, the toolbox contains the following computational methods for μ:

- With a frequency gridding: classical mixed μ lower and upper bounds (*mixed_mu_ub.m*, *mixed_mu_lb.m*), exponential-time real μ upper and lower bounds (*mu_zd.m*, *mu_dailey.m*). All these routines can also be called via *mu_frequency_gridding.m*. See also the function *mixed_mu_lb_freq.m*.

- Computation of a guaranteed (skew) mixed μ upper bound on a frequency interval: *mu_max_1.m*, *mu_max_1f.m*, *mu_max_2.m* and *mu_max_3.m*.

- Elimination of frequency intervals, inside which the (skew) s.s.v. is guaranteed to be less than a given value, with *mu_elim.m*.

- Computation of a real μ lower bound with a mixed frequency/state-space approach: *mu_lb_with_freq.m*.

- With a frequency gridding, computation of worst-case values of MIMO gain, phase and delay margins in the presence of parametric uncertainties and neglected dynamics: *worst_case_margin.m*.

5 Calls to the routines: examples

5.1 Classical μ problems (see also the file *calls_mu.m*)

This Toolbox includes many different routines to solve classical μ problems (see Table 1). These routines may be called on some specific examples via *calls_mu.m* or using the interactive demo : *demo_mu.m*. In this subsection, we propose to focus on the newly developed function *mu_max_3.m* which is devoted to classical μ problems involving high-order plants with numerous and possibly largely repeated uncertainties.

The aim of this subsection is to illustrate two ways of using the routine to compute the robustness margin of a flight control system applied to a flexible aircraft.

5.1.1 Standard call of mu_max_3.m

In this standard call, neither any initial frequency gridding, nor any option is specified. The Matlab command lines are given below :

```
>> load flexible_airplane_m_delta;
>> [mub,tab_mu,tab_puls] = mu_max_3(sys_M,blk);
```

Here, a frequency gridding with 20 points is generated automatically (using the function *gen_grid.m*). Note that the gridding is automatically refined near flexible modes. Then, standard options are used :

- μ upper-bounds are computed using *mu.m*. Then, a switch to LMI iterations is performed on critical segments,
- the frequency elimination phase attempts to eliminate all frequency intervals where the μ can be proved to remain below the highest upper-bound computed so far.

While execution is beeing performed, the following informations are displayed :

- Iteration number,
- Number of remaining intervals. Note that this number may increase, when the elimination technique fails. In that case indeed, an additional frequency point has to be considered,
- The μ upper-bound : This the highest value of μ which has been computed so far,

- Reliability of the upper-bound : The highest μ upper-bound which has been computed after n iterations is clearly reliable only if all intervals can be eliminated. Consequently the reliablity is simply given as the ratio between the total length of intervals which could be eliminated and the length of the whole frequency range.

On our proposed example we obtained :

```
Computations will be performed on one frequency segment :
  --> SEGMENT = [ 0.00 , 100.00 ]  -  standard stability constraint

----------------------------------------------------------------------
Iteration | Remaining Intervals | mu Upper-bound so far | reliability
----------------------------------------------------------------------
        1 |                20   |                0.000  |     0.00%
        2 |                21   |                0.000  |     0.00%
        3 |                22   |                1.766  |     0.14%
  ...............................................................
       35 |                 8   |                4.502  |    18.50%
  ...............................................................
       40 |                 6   |                4.816  |    19.38%
       41 |                 5   |                4.816  |   100.00%
       41 |                 0   |                4.816  |   100.00%
----------------------------------------------------------------------


                  #############################
                    Switched to LMI iterations
                  #############################

Number of decision variables in LMI step : 20

  ****** Recompute peak   1  =   4.816  on   [ 13.35 , 13.41 ] ******
----------------------------------------------------------------------
Iteration | Remaining Intervals | mu Upper-bound so far | reliability
----------------------------------------------------------------------
        1 |                 1   |                4.501  |     0.00%
        2 |                 2   |                4.566  |     0.00%
  ...............................................................
        5 |                 0   |                4.586  |   100.00%
----------------------------------------------------------------------
```

Remarks:
(i) Note that a switch to LMI iterations was possible here, since the number of decision variables associated to scaling matrices D and G was limited ($N_{var} = 20$). This step has permitted

to further reduce the μ upper-bound (from 4.8 to 4.6) without any large impact on the global computational-time since only 5 LMI iterations were required. This low number of iterations can be easily explained since the main iterations (based on *mu.m*) have permitted to identify a very small-size critical segment.
(ii) As indicated before iterations start, the computation is performed on one frequency segment, along the imaginary axis. We recall that the routine may also be applied on the borderline of a truncated sector. In that case, computations are performed along two segments. Standard stability constraint is considered first. Then, the damping constraint is treated.

The results are displayed on figure (1). Clearly, if the highest value of the upper-bound was computed with a high accuracy level. But, this is not the case of secondary peaks.

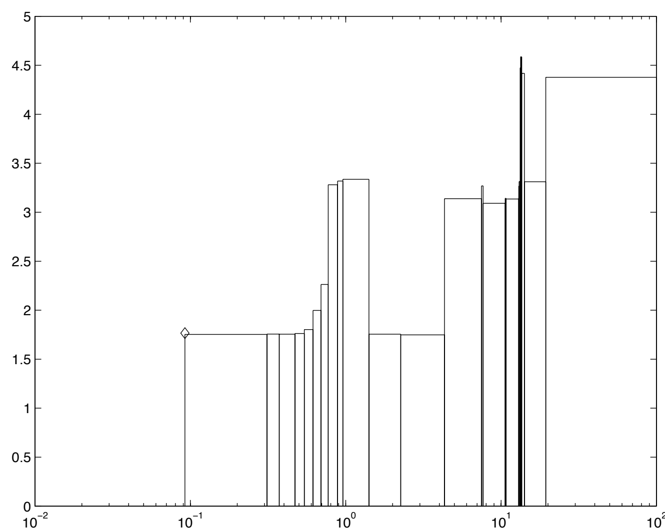

Figure 1: Fast computation of the robustness margin

5.1.2 Refined computations using mu_max_3.m

Refined computations may be achieved by specifying an initial frequency gridding, and setting options in order to put limitations on the frequency elimination technique. The Matlab command lines now read (see the help of the function for more informations) :

```
>> puls=logspace(-1,2,100); % initial frequency gridding
>> options(1)=0;            % initial value for upper-bound
>> options(2)=2;            % mixed approach (switch to LMIs on critical segment)
>> options(3)=1;            % refined computation of secondary peaks
```

```
>> options(4)=-1;           % plots results on a new figure
>> [mub,tab_mu,tab_puls] = mu_max_3(sys_M,blk,puls,options);
```

Of course, the number of iterations (and thus the computational-time) will be much higher in this case. More that 100 iterations are needed here. The corresponding results are plotted on figure (2). In this case, secondary peaks are much more precisely detected.

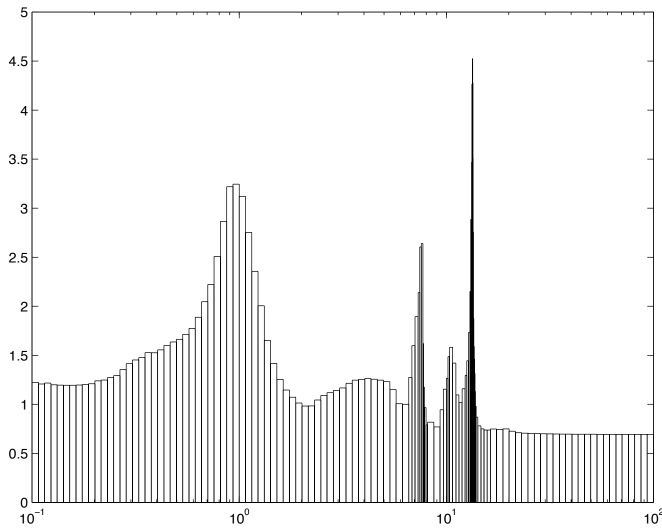

Figure 2: Refined computation of secondary peaks

5.1.3 Frequency elimination using mu_max_3.m

While it is not mainly devoted to such tasks, the routine may also be used to check wether the maximum value of the μ upper-bound is below a specified target. For example if we want to check that the μ upper-bound between 0 and $100\, rad/s$ is below 5, then the following Matlab sequence is to be considered :

```
>> puls=[0 100];    % very rough gridding :  only extremal points
>> options(1)=5;    % initial upper-bound to be checked
>> options(2)=1;    % mu.m based computation without switching to LMIs
>> options(3)=0;    % fast computation (secondary peaks ignored)
>> options(4)=0;    % no plot
>> [mub,tab_mu,tab_puls] = mu_max_3(sys_M,blk,puls,options);
```

and we obtain the following result :

```
----------------------------------------------------------------------
Iteration | Remaining Intervals | mu Upper-bound so far | reliability
----------------------------------------------------------------------
        1 |                 1   |                5.000  |    78.02%
        2 |                 1   |                5.000  |    78.02%
        3 |                 2   |                5.000  |    87.69%
        4 |                 3   |                5.000  |    88.15%
        5 |                 2   |                5.000  |    91.26%
        6 |                 2   |                5.000  |    99.66%
        7 |                 1   |                5.000  |    99.66%
        8 |                 2   |                5.000  |   100.00%
        8 |                 0   |                4.964  |   100.00%
----------------------------------------------------------------------
```

which shows that the frequency segment [0 100] has been cleared after 8 iterations, without increasing the initial μ upper-bound.

Note that specific tools also exist in the Toolbox to perform such a test (see subsection 3.1). These tools are more general since they handle the case of skew uncertainties.

5.1.4 The case of largely repeated uncertainties

To conclude this subsection, we now focus on the case of largely repeated uncertainties for which the routine *mu_max_3.m* was specifically developed. A challenging example is proposed in the Toolbox (see in Applications directory : *generic_rep_unc.mat*). The plant has 60 states which correspond to badly damped flexible modes. Moreover, it contains 4 repeated uncertainties :

```
>> load generic_rep_unc;
>> blk

blk =

   -10     0     1
   -10     0     1
    -8     0     1
    -2     0     1
```

The routine is then applied with default options on this example, by the following commande line :

```
>> [mub,tab_mu,tab_puls] = mu_max_3(sys_M,blk);
```

The following result is obtained :

```
Computations will be performed on one frequency segment :
  --> SEGMENT = [ 0.00 , 100.00 ]  -  standard stability constraint

----------------------------------------------------------------------
Iteration | Remaining Intervals | mu Upper-bound so far | reliability
----------------------------------------------------------------------
        1 |               20    |                0.000  |     0.00%
        2 |               21    |                0.000  |     0.00%
        3 |               22    |                0.950  |     2.61%
        4 |               25    |                0.950  |    34.13%
        5 |                6    |                0.950  |    34.13%
  ..............................................................
       34 |               10    |               25.384  |    37.56%
       35 |                9    |               25.384  |    37.56%
       36 |                7    |               25.384  |    99.99%
       37 |                1    |               62.296  |   100.00%
       37 |                0    |               62.296  |   100.00%
----------------------------------------------------------------------


                  #############################
                    Switched to LMI iterations
                  #############################

Number of decision variables in LMI step : 268
Computational-time is estimated too high
Switch to LMI has then been aborted.
You may however perform this refined computation by calling mu_max_3 again
by specifying the most critical segments in puls, and using options(2)=0
```

The main iterations (based on $mu.m$) were performed in $20\,s$ (on a Sun Blade 1500 Workstation) which is extremely fast despite the complexity of the problem. Although the LMI iterations could not be performed here, it should be emphasized that the computed μ upper-bound is not very conservative. A lower-bound computation (using $mu\_lb\_wit_freq.m$) produced the following result : $\underline{\mu} = 61.06$. The gap between the two bounds is then approximately 2%

5.2 Skew μ problems (see also the file *calls_skew_mu.m*)

References

[1] J-M. Biannic and Gilles Ferreres. Efficient computation of a guaranteed robustness margin. Submitted to IFAC World Congress, Pragues.

[2] J.M. Biannic and G. Ferreres. Efficient computation of a guaranteed robustness margin. *submitted*, 2004.

[3] G. Ferreres, J.F. Magni, and J.M. Biannic. Robustness analysis of flexible structures: practical algorithms. *International Journal of Robust and Nonlinear Control*, 13:715–734, 2003.

[4] J.F. Magni, C. Doll, C. Chiappa, B. Frapard, and B. Girouart. Mixed μ–analysis for flexible systems, Part I: Theory. July 1999. Proc. of the IFAC World Congress.