IQC library:
A frequency domain approach

User's guide

for version 1.00

Fabrice Demourant

ONERA
French aeronautics, space and defense research lab

Systems Control and Flight Dynamics Department
Toulouse

July 3, 2014

Contents

1 IQC based Robustness Analysis Technique **1**
- 1.1 Introduction . . . 1
- 1.2 IQC principle . . . 3
- 1.3 IQC Description . . . 4
  - 1.3.1 Slope restricted sector non linearity . . . 5
  - 1.3.2 Dynamic uncertainty . . . 8
  - 1.3.3 Real uncertainty . . . 8
  - 1.3.4 Time varying real parameter . . . 8
  - 1.3.5 Slowly time varying real parameter . . . 9
  - 1.3.6 Polytopic model . . . 10
- 1.4 IQC parametrization . . . 10
  - 1.4.1 Slope restricted sector non linearities . . . 10
  - 1.4.2 Dynamic uncertainty . . . 11
  - 1.4.3 Uncertain real scalar . . . 11
  - 1.4.4 Slowly time varying real parameter . . . 11
  - 1.4.5 Time varying parameters and polytopic form . . . 12
- 1.5 Resolution technique . . . 13
  - 1.5.1 State-space approach . . . 13
  - 1.5.2 Frequency domain approach . . . 14

2 Frequency domain resolution tool **19**
- 2.1 Data . . . 19
  - 2.1.1 LMI options . . . 21
  - 2.1.2 Resolution tool options . . . 22
- 2.2 Output arguments . . . 23
- 2.3 Termination condition . . . 24
- 2.4 Aircraft models . . . 27
  - 2.4.1 Aeroelastic model . . . 27
  - 2.4.2 Military aircraft . . . 28
- 2.5 Static non-linearity . . . 29
  - 2.5.1 Sector non-linearity . . . 29

2.5.2 Slope restricted non-linearity 30
2.6 Complex/Dynamic uncertainty 32
2.7 Real/parametric uncertainty 34
2.8 Time varying parameter 36
2.9 Polytopic model 38
2.10 Troubleshootings 39

Abstract

To analyze a large class of stability problems, typically non-linear, uncertain, time-varying, etc.. closed loop, an IQC approach is involved. In the context of this toolbox, we use standard IQC description and focus on the algorithmic issue. Usually the Kalman-Yakubovitch-Popov lemma based resolution is involved but with the consequence to add an auxiliary matrix P whose the size increases with the closed loop order. Finally this kind of approach leads to a strong increase in the number of optimization variables, which makes it untractable for high order models. Here a specific technique has been developed to solve the stability problem directly in the frequency domain with the guarantee that the solution is valid on the whole frequency domain. Some examples are given to illustrate the approach and a detailed description of the tool is provided.

Chapter 1

IQC based Robustness Analysis Technique

1.1 Introduction

The Integral Quadratic Constraint (IQC) technique, which appeared during the Nineties at least in its 'modern form' [10], can be viewed as the merging of two well-known robustness analysis techniques, namely the (scaled) small gain techniques, the best known being μ analysis [3], and the positivity/passivity techniques which study the interconnection of an linear time invariant (LTI) operator with a non-linearity (the famous "Lur'e problem"). As a consequence, the IQC technique enables to study a wide range of problems, namely the robust stability and performance properties of the interconnection $M(s) - \Delta$ of an LTI operator M(s) with a structured model uncertainty Δ containing non-linearities, LTI and/or linear time-varying (LTV) parameters, neglected dynamics, delays, specific non-linearities such as friction or hysteresis. . . .The principle is to replace each block of uncertainty by an IQC description of its inputs/outputs, i.e. the inputs/outputs of the block (e.g. a non-linearity inside a sector, possibly with a bound on its slopes) are supposed to satisfy a set of Integral Quadratic Constraints [10, 5]. The finer the IQC description of the block is, the less conservative the result is. This approach is very interesting for two reasons. It includes in the same formalism a large set of linear and non-linear stability theorems and finally the IQC approach can be described as an unified formalism. And secondly this unified formalism is based on inputs/outputs approach, namely a frequency domain approach.

In the context of this toolbox, we use standard IQC description and focus on the algorithmic issue. Let us remind that the stability criterion of this approach is based on Frequency Dependent Inequalities (FDI). Then the most classical way to solve an IQC analysis problem consists in solving the state-space LMI conditions derived from the KYP lemma, so that the optimization variables come from the IQC multipliers, but also from the Lyapunov matrix P. However, this solution becomes untractable when the order n of the state-space representation becomes too high, since the number of scalar optimization variables in P grows quadratically with n. Noting moreover that the initial state-space representation of $M(s)$ is augmented with the state-space representations of the dynamic multipliers, so that even if the order of the initial state-space representation is low, it may increase very fast when introducing dynamic multipliers. Different approaches based on Hamiltonian matrix as been developed ([8] for example and references therein) to avoid this problem. Here an alternative technique is implemented. The approach developed here is particularly attractive due to its conceptual simplicity. Besides the technique is based on very usual and common solver which is the LMI Toolbox of Matlab. In brief the implementation is straightforward, completely self-contained in Matlab environment, which might be useful for engineers in need for efficient and fast answers to analyze stability of complex problems.

A technique which consists in checking the validity of the solution on the whole frequency domain has been developed. This technique is based on a mathematical result on the singular value maximum of an LFT (Linear Fractional Transformation) structure. More precisely when a solution is obtained from a frequency domain griding the stability criterion which depends on frequencies is put under an LFT form to make appear the frequency ω as a real parameter in a Δ_r block of the LFT. Then the validity domain of the solution is computed thanks to an algebraic approach extracted from [12] and adapted to our problem [1]. If this domain is $[0, +\infty[$ the solution is valid on the whole frequency domain. Else, frequencies where the FDI are not satisfied are detected and are added to the initial frequency domain griding and a new solution is computed with the new griding and so on. If no solution is obtained on the griding the problem is considered as unfeasible. In brief the stability problem is recast as an LMI feasibility problem where the constraints (FDI) are added iteratively. Finally the number of optimization variables is completely independent of the closed loop order, which makes it very attractive for high dimensional systems [1].

1.2 IQC principle

An IQC describes a relation between inputs and outputs signals of an operator. Thus it is a mean to characterize this operator. These constraints can be defined in the time or frequency domain since these two formulations are completely equivalent. Nevertheless frequency domain constraints are more often used, which leads to obtain stability conditions easier to handle. So the definition of an IQC is given in the frequency domain:

Definition 1.2.1 *Two signals respectively of dimension m et p, square integrable on* $[0, \infty[$ *i.e. :*

$$v \in L_2^m[0, \infty[\,, \quad w \in L_2^p[0, \infty[$$

satisfy the IQC defined by Π *if and only if*

$$\int_{-\infty}^{\infty} \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix} d\omega \geq 0 \tag{1.1}$$

where $\hat{v}(j\omega)$ and $\hat{w}(j\omega)$ respectively correspond to Fourier transforms of v et w such as $w = \Delta v$.

A priori the operator Π, called multiplier, defined from jR in $C^{(m+p)\times(m+p)}$ can be any measurable Hermitian-valued function. In most situations it is sufficient to use rational operators that are bounded on imaginary ax.

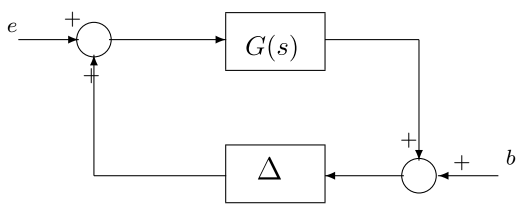

Figure 1.1: Analysis problem.

The linear system $G(s)$ is supposed stable and the problem consists in analyzing the closed loop of figure 1.1 with a causal and bounded operator Δ which can be non-linear and non-stationary. Let us suppose that inputs and output signals of Δ satisfy the IQC defined by Π. The following result is extracted from [10].

Theorem 1.2.2 *Let us suppose:*

- *the interconnection of G and $\tau\Delta$ is well posed for any $\tau \in [0, 1]$,*

- *$\tau\Delta$ satisfies the IQCs defined by Π, $\forall \tau \in [0,1]$,*
- *it exists $\epsilon > 0$ such as:*

$$\forall \omega \in R \quad \begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} \leq -\epsilon I \tag{1.2}$$

the closed loop system of the figure (1.1) is stable.

It is important to notice that if $\tau\Delta$ satisfies several IQC $\Pi_1, \dots, \Pi_n$, then a sufficient condition for the stability is that it exists $x_1, \dots, x_n \geq 0$ such as the inequality (1.2) is satisfied for $\Pi = x_1\Pi_1 + \dots + x_n\Pi_n$.

A global multiplier

The following proposition is very useful to consider the case with several multipliers [6].

Proposition 1.2.3 *Let us suppose a block diagonal structure $\Delta = diag(\Delta_1, \dots, \Delta_n)$ and each Δ_i satisfies the IQC defined by Π_i. Then Δ satisfies the IQC defined by $\Pi = daug(\Pi_1, \dots, \Pi_n)$ where the operator daug is defined as follows. If*

$$\Pi_i = \begin{bmatrix} \Pi_{i1} & \Pi_{i2} \\ \Pi_{i2}^* & \Pi_{i3} \end{bmatrix} \tag{1.3}$$

then

$$daug(\Pi_1, \Pi_2) = \left[\begin{array}{cc|cc} \Pi_{11} & 0 & \Pi_{12} & 0 \\ 0 & \Pi_{21} & 0 & \Pi_{22} \\ \hline \Pi_{12}^* & 0 & \Pi_{13} & 0 \\ 0 & \Pi_{22}^* & 0 & \Pi_{23} \end{array}\right] \tag{1.4}$$

1.3 IQC Description

We have presented in the previous section the definition and the main stability theorem. An important point is to define the multiplier Π explicitly. As we will see, by the IQC approach, it is possible to recover numerous stability theorems.

1.3.1 Slope restricted sector non linearity

Let us consider a non-linearity which is memoryless, static, and piecewise continuous in t. It is required that the non-linearity satisfies a sector condition.

Definition 1.3.1 *A memoryless non-linearity $\psi : [0,\infty) \times \mathbb{R}^p \rightarrow \mathbb{R}^p$ is said to satisfy a sector condition if*

$$(w - \underline{k}v)^T(w - \overline{k}v) \leq 0 \tag{1.5}$$

where $\underline{k}$ and $\overline{k}$ are gains which represent the limits of the sector and w and v represent the inputs/outputs of the non linearity.

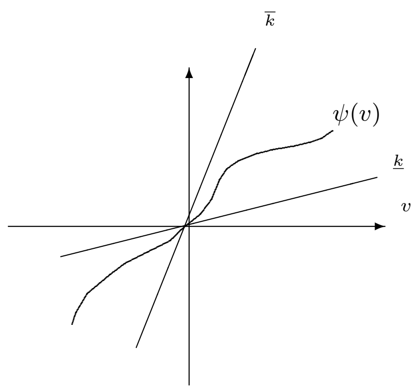

Figure 1.2: Global sector non-linearities.

This definition is illustrated by figure 1.2 in SISO case. Equation (1.5) can be formulated of the following way:

$$\forall t > 0, \quad \begin{bmatrix} v(t) \\ w(t) \end{bmatrix}^T \begin{bmatrix} -2\underline{k}\overline{k} & \underline{k}+\overline{k} \\ \underline{k}+\overline{k} & -2 \end{bmatrix} \begin{bmatrix} v(t) \\ w(t) \end{bmatrix} \geq 0 \tag{1.6}$$

As the integral of this function is positive and thanks to the frequency and time domain equivalence the following inequality is obtained:

$$\int_{\infty}^{\infty} \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix}^T \begin{bmatrix} -2\underline{k}\overline{k} & \underline{k}+\overline{k} \\ \underline{k}+\overline{k} & -2 \end{bmatrix} \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix} d\omega \geq 0 \tag{1.7}$$

In brief $\Pi_{sector} = \begin{bmatrix} -2\underline{k}\overline{k} & \underline{k}+\overline{k} \\ \underline{k}+\overline{k} & -2 \end{bmatrix}$. The stability of the closed loop which corresponds to the interconnection of $G(s)$ with a sector non-linearity $(\underline{k}, \overline{k})$

by a positive feedback is ensured if:

$$\forall\omega \in R, \quad \begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^* \begin{bmatrix} -2\underline{k}\overline{k} & \underline{k}+\overline{k} \\ \underline{k}+\overline{k} & -2 \end{bmatrix} \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} \leq 0 \tag{1.8}$$

We find again with inequality (1.8) the circle criterion. It suffices to take $\underline{k} = 0$, $\overline{k} = k$ and G in $-G$ (since the circle criterion is formulated with a negative feedback) and the well known relation $Re(G) \geq -\frac{1}{k}$ is obtained for a sector $(0, k)$.

Sector non linearity with Popov criterion

The Popov criterion is widely used for this kind of non-linear analysis. Let us remind that in SISO case the Popov criterion is written:

$$Re\left((1 + j\lambda\omega)G(j\omega)\right) > -\frac{1}{k} \tag{1.9}$$

If we use the Popov plot, i.e. the plot $Re[G(j\omega)]$ versus $\omega Im[G(j\omega)]$, the closed loop stability is ensured if the Popov plot of $G(j\omega)$ lies to the right of the line that intercepts the point $-1/k + 0j$ with a slope $1/\lambda$.

In brief let us consider a non-linearity with a sector $(0, k)$, if it exists λ such as equation (1.9) is satisfied then the non-linear closed loop which corresponds to the interconnection of G with the sector non-linearity is stable. The multiplier which corresponds to the criterion Popov is [10, 4]:

$$\Pi_{Popov} = \lambda \begin{bmatrix} 0 & j\omega \\ -j\omega & 0 \end{bmatrix} \tag{1.10}$$

With several multipliers $\Pi_1, \ldots, \Pi_n$ for one operator Δ, the final multiplier which corresponds to Δ is $\Pi = x_1\Pi_1 + \ldots + x_n\Pi_n$ with $x_i > 0$. Then the final multiplier for the sector non linearity is $\Pi = x\Pi_{sector} + \Pi_{Popov}$:

$$\Pi = x\Pi_{sector} + \Pi_{Popov} = \begin{bmatrix} -2x\underline{k}\overline{k} & x\overline{k} + x\underline{k} + j\lambda\omega \\ x\overline{k} + x\underline{k} - j\lambda\omega & -2x \end{bmatrix} \tag{1.11}$$

As previously it suffices to take $\overline{k} = k$, $\underline{k} = 0$ and $G = -G$, to find again in SISO case the Popov criterion of equation (1.9).

Remark 1: To compare Popov result with the IQC approach it is important to bear in mind that the IQC approach is involved with a positive feedback whereas the circle/Popov criterion is involved with a negative feedback. Then it is necessary to multiply the linear part $G(s)$ by -1 to switch from IQC approach to circle/Popov criterion and vice versa.

Remark 2: The case where a sector $(0, 1)$ is considered is not at all restrictive since any sector $(\underline{k}, \overline{k})$ can be transformed into a sector $(0, 1)$ by the loop shifting theorem [9].

Remark 3: For a sector $(0, +\infty)$ it is not judicious to replace k by a very high value as $1e6$, $1e9$, etc... Let us consider an LTI model interconnected by a negative feedback to a static non-linearity with a sector $(0, +\infty)$. The SISO analysis shows that the Nyquist/Popov plot must lies to the right of the point $-1/k$. In this case $-1/k = 0$. To recover an analysis problem with a sector $(0, 1)$ it suffices to replace the linear part $G(s)$ by $\tilde{G}(s) = G(s) - 1$, which is equivalent "to move" the nyquist plot of $G(s)$ to the point -1. To preserve the closed loop equivalence before and after this transformation (see loop shifting theorem in [9] for example), it is necessary to replace the static non-linearity by a static non-linearity interconnected to itself with a negative (due to the negative feedback of the closed loop) and unitary feedback. It is interesting to notice that this operation transforms a gain k into $\tilde{k} = \frac{k}{1+k}$. Thus it is easy to see that when $k = \infty$, $\tilde{k} = 1$. And finally the problem consists in analyzing the stability of the closed loop which corresponds to the interconnection by a negative feedback of a linear part $\tilde{G}(s)$ with a static non-linearity of sector $(0, 1)$. Let us remind that to involve the IQC approach it is necessary to multiply $\tilde{G}(s)$ by -1 to have a positive feedback (see remark 1). To be complete this kind of transformation remains valid in MIMO case, since it suffices to apply it on each channel.

Slope restricted sector non-linearities

A more general multiplier has been developed [11] since it includes the Popov multiplier case. The advantage is to take into account the slope restricted feature of a static non-linearity, and consequently to reduce the conservatism of the Popov multiplier. The multiplier used for a sloped restricted $(0, \beta)$ sector $(0, k)$ non-linearity φ is:

$$\Pi_\varphi = \begin{bmatrix} 0 & kx + j\omega\lambda + \omega^2\beta\gamma \\ kx - j\omega\lambda + \omega^2\beta\gamma & -2x - 2\omega^2\gamma \end{bmatrix} \tag{1.12}$$

with $x, \gamma \geq 0$, and $\lambda \in \mathbb{R}$. If $\gamma = 0$ the Popov multiplier is recovered.

1.3.2 Dynamic uncertainty

Let us consider an LTI dynamic uncertainty $\Delta(s)$ such as $\|\Delta(s)\|_\infty \leq 1$. The IQC which corresponds to dynamic uncertainties is:

$$\Pi_\Delta = \begin{bmatrix} X(j\omega) & 0 \\ 0 & -X(j\omega) \end{bmatrix} \tag{1.13}$$

with $X(j\omega) = x(j\omega)I = X(j\omega)^* = x^*(j\omega)I > 0$. It is interesting to notice the equivalence of this formulation with the upper bound of μ in the case of dynamic uncertainties. More precisely to ensure the stability of a $M - \Delta$ structure, it suffice to find $D = D^* > 0$ where D commutes with Δ such as:

$$\begin{array}{rl} & \|DMD^{-1}\|_\infty \leq 1 \\ \Leftrightarrow & \sqrt{\lambda_{max}(DMD^{-1})^*(DMD^{-1})} \leq 1 \\ \Leftrightarrow & (DMD^{-1})^* DMD^{-1} \leq I \\ \Leftrightarrow & MD^*DM \leq D^*D \\ \Leftrightarrow & M^*XM \leq X \end{array}$$

with $X = D^*D$ and $G = M$, inequality (1.2) with Π_Δ is find again.

1.3.3 Real uncertainty

Let us suppose that $\Delta = \delta I$ where δ represents a real uncertainty of absolute value ≤ 1. The multiplier which corresponds to this case is the following one:

$$\Pi_\delta = \begin{bmatrix} X(j\omega) & Y(j\omega) \\ Y^*j\omega) & -X(j\omega) \end{bmatrix} \tag{1.14}$$

with $X(j\omega) = X(j\omega)^* > 0$ and $Y(j\omega) = -Y(j\omega)^*$. If Π_δ is used in inequality (1.2) the mixed μ upper bound is recovered with $G = M$, $X = D$ and $Y = -jG_s$ [3] (let us notice that $Y^* = -Y \Leftrightarrow G_s = G_s^*$ which is in accordance with the mixed μ upper bound). Be aware that G_s is named G in the μ definition and corresponds to a scaling to take into account the real nature of δ.

1.3.4 Time varying real parameter

Let us consider $\Delta = \delta I$ with δ is a varying parameter and $\|\delta\|_\infty \leq 1$. In this case the multiplier is static:

$$\Pi_{\delta(t)} = \begin{bmatrix} X & Y \\ Y^T & -X \end{bmatrix} \tag{1.15}$$

where $X = X^T > 0$ and $Y = -Y^T$ are real matrices.

1.3.5 Slowly time varying real parameter

Here $|\delta(t)| \leq 1$ and $\dot{\delta(t)} \leq d$. Let $X = R^*R$ and $Y = S - S^*$. Besides let us define [7]:

$$\begin{aligned} R_C(s) &= C_R(sI - A_R)^{-1} \\ R_B(s) &= (sI - A_R)^{-1}B_R \\ S_C(s) &= C_S(sI - A_R)^{-1} \\ S_B(s) &= (sI - A_R)^{-1}B_S \end{aligned}$$

with the realizations $R(s) = C_R(sI - A_R)^{-1}B_R + D_R$ and $S(s) = C_S(sI - A_S)^{-1}B_S + D_S$. Then the following multiplier corresponds to LTV parameters:

$$\Pi_{\delta(t)} = \begin{bmatrix} R^*R + d\Gamma^*\Gamma & S - S^* \\ S^* - S & R^*R + d\Upsilon^*\Upsilon \end{bmatrix} \tag{1.16}$$

with $\Gamma = \begin{bmatrix} R_B \\ S_B \\ S_C^* \end{bmatrix}$ and $\Upsilon = R_C^* R$.

Let us notice 3 important points:

- The terms $d\Gamma^*\Gamma$ and $d\Upsilon^*\Upsilon$ can be interpreted as penalty to take into account the varying parameter nature.

- Consequently if $d = 0$ the multiplier which corresponds to a constant real scalar is recovered:

$$\Pi_\delta(s) = \begin{bmatrix} R(s)^*R(s) & S(s) - S(s)^* \\ S(s)^* - S(s) & -R(s)^*R(s) \end{bmatrix} \tag{1.17}$$

- In the case of an arbitrary rate of variation, i.e. $d \to \infty$, the stability criterion can be satisfied only if the multiplier Π is constant. Indeed $\Gamma = \Upsilon = 0$ which implies that $R(s) = D_R$ and $S(s) = D_S$, and then $\Pi_{\delta(t)} = \begin{bmatrix} R^TR & S - S^T \\ S^T - S & -R^TR \end{bmatrix}$ which corresponds to the multiplier for LTV parameter case.

1.3.6 Polytopic model

Let us consider a polytopic uncertainty $\Delta_p \in \mathbf{R}^{n\times n}$. Δ_p takes values in the polytope $D = hull\{\Delta_1, ..., \Delta_N\}$ where Δ_i represents the vertices of the convex hull D , $N = 2^n$ and $0 \in D$. The multiplier which corresponds to the polytopic representation is the following one:

$$\Pi_{\Delta_p} = \begin{bmatrix} X & Y \\ Y^T & -Z \end{bmatrix} \tag{1.18}$$

where $X = X^T$, $Z = Z^T \geq 0$ and Y are real matrices.

1.4 IQC parametrization

In this section the IQC parametrization w.r.t optimization variables to solve the feasibility problem is presented. From this parametrization the global multiplier is built and formulated under a factorized form to be implemented and solved thanks to the Matlab LMI toolbox.

1.4.1 Slope restricted sector non linearities

IQC for sector non-linearity with a sector $(0, k)$ is the following one:

$$\Pi_{sector} = \begin{bmatrix} 0 & k \\ k & -2 \end{bmatrix} \tag{1.19}$$

$$\Pi_{Popov} = \lambda \begin{bmatrix} 0 & j\omega \\ -j\omega & 0 \end{bmatrix}, \quad \forall\lambda \in R \tag{1.20}$$

where finally the multiplier:

$$\Pi = x\Pi_{sector} + \Pi_{Popov} = \begin{bmatrix} 0 & xk + j\omega\lambda \\ xk - j\omega\lambda & -2x \end{bmatrix}, x \geq 0, \quad \lambda \in R \tag{1.21}$$

It is possible to have a formulation with a sector $(\underline{k}, \overline{k})$. But if $\overline{k} > 0$ for example, conditions to involve the stability criterion (1.2) are not fulfilled, since the $\tau\Delta$ must satisfy IQC defined by Π for any $\tau \in [0, 1]$. Besides it is well known that any sector $(\underline{k}, \overline{k})$ can be transformed of equivalent way into a sector $(0, k)$ by the loop shifting theorem.

If the slope restriction $(0, \beta)$ is used we obtain the following multiplier:

$$\Pi_{\varphi} = \begin{bmatrix} 0 & kx + j\omega\lambda + \omega^2\beta\gamma \\ kx - j\omega\lambda + \omega^2\beta\gamma & -2x - 2\omega^2\gamma \end{bmatrix} \tag{1.22}$$

1.4.2 Dynamic uncertainty

As indicated previously the multiplier is:

$$\Pi_\Delta = \left[\begin{array}{cc} X(j\omega) & 0 \\ 0 & -X(j\omega) \end{array} \right] \tag{1.23}$$

The resolution of $X(j\omega)$ is based on the following parametrization:

$$X(j\omega) = \tilde{R}(j\omega)^* U \tilde{R}(j\omega) = R(j\omega)^* R(j\omega) \tag{1.24}$$

with $\Delta(j\omega) \in \mathbf{C^{n\times n}}$, $U = U^T$, $\tilde{R}(j\omega) = blkdiag(F, F, ..., F)$ where F is repeated n times, with $F = [1; filter_1; ...; filter_r]$. Typically these filters are first order low-pass filters.

1.4.3 Uncertain real scalar

For this case the multiplier is:

$$\Pi_\delta = \left[\begin{array}{cc} X(j\omega) & Y(j\omega) \\ Y^* j\omega) & -X(j\omega) \end{array} \right] \tag{1.25}$$

The parametrization used for $X(j\omega)$ and $Y(j\omega)$ are:

$$\begin{array}{rcl} X(j\omega) & = & \tilde{R}(j\omega)^* U \tilde{R}(j\omega) = R(j\omega)^* R(j\omega) \\ Y(j\omega) & = & V\tilde{S}(j\omega) - \tilde{S}(j\omega)^* V^T = S - S^* \end{array} \tag{1.26}$$

with $\Delta = \delta I_n$, $\tilde{R}(j\omega) = \tilde{S}(j\omega) = [I_n; filter_1 \times I_n; filter_2 \times I_n; \ldots; filter_r \times I_n]$, $U = U^T$. Let us notice that this parametrization is coherent with $X(j\omega) = X(j\omega)^*$ and $Y(j\omega) = -Y(j\omega)^*$.

As previously this parametrization preserves the convexity of the optimization problem since the stability criterion is affine with decision variables.

1.4.4 Slowly time varying real parameter

Let us consider the multiplier which corresponds to LTV parameters [7]:

$$\Pi_{\delta(t)} = \left[\begin{array}{cc} R^*R + d\Gamma^*\Gamma & S - S^* \\ S^* - S & R^*R + d\Upsilon^*\Upsilon \end{array} \right] \tag{1.27}$$

with Γ and Υ defined as previously. With the parametrization (1.26) the following multiplier is obtained:

$$\Pi_{\delta(t)} = \begin{bmatrix} \tilde{R}^*U\tilde{R} + d(\tilde{R}_B^*\tilde{R}_B + \tilde{S}_B^*\tilde{S}_B + V\tilde{S}_C\tilde{S}_C^*V^T) & V\tilde{S} - \tilde{S}^*V^T \\ \tilde{S}^*V^T - V\tilde{S} & -\tilde{R}^*U\tilde{R} + d\tilde{R}^*U\tilde{R}_C\tilde{R}_C^*U\tilde{R} \end{bmatrix}$$

We can see clearly that this multiplier is bilinear in U and V. Then to search for a multiplier $\Pi_{\delta(t)}$ such as:

$$\begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^* \left\{ \begin{bmatrix} dV\tilde{S}_C\tilde{S}_C^*V^T & 0 \\ 0 & d\tilde{R}^*U\tilde{R}_C\tilde{R}_C^*U\tilde{R} \end{bmatrix} + \Pi_{lin} \right\} \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} \leq 0 \tag{1.28}$$

with $\Pi_{lin} = \begin{bmatrix} \tilde{R}^*U\tilde{R} + d(\tilde{R}_B^*\tilde{R}_B + \tilde{S}_B^*\tilde{S}_B) & V\tilde{S} - \tilde{S}^*V^T \\ \tilde{S}^*V^T - V\tilde{S} & -\tilde{R}^*U\tilde{R} \end{bmatrix}$ is not a convex problem.

To make convex this problem let $\Lambda = \begin{bmatrix} G(j\omega) \\ I \end{bmatrix}$ and $W = \begin{bmatrix} \sqrt{d}V\tilde{S} & 0 \\ 0 & \sqrt{d}\tilde{R}^*U\tilde{R}_C \end{bmatrix}$, then inequality (1.28) becomes:

$$\Lambda^*\Pi_{lin}\Lambda + \Lambda^*WW^*\Lambda \leq 0 \tag{1.29}$$

From this relation it is possible to involve the Schur lemma which is based on the following property where $A(\xi)$, $B(\xi)$,$C(\xi)$ and $D(\xi)$ are affine in ξ:

$$\begin{array}{c} A(\xi) > 0 \\ C(\xi) - B^T(\xi)A^{-1}(\xi)B(\xi) > 0 \end{array} \Leftrightarrow \begin{bmatrix} A(\xi) & B(\xi) \\ B^T(\xi) & C(\xi) \end{bmatrix} > 0 \Leftrightarrow \begin{bmatrix} C(\xi) & B^T(\xi) \\ B(\xi) & A(\xi) \end{bmatrix} > 0$$

In brief the Schur lemma allows to transform a quadratic constraint into an affine one. Finally inequality (1.28) is equivalent to the following LMI constraint:

$$\begin{bmatrix} \Lambda^*\Pi_{lin}\Lambda & \Lambda^*W \\ W^*\Lambda & -I \end{bmatrix} < 0 \tag{1.30}$$

where Π_{lin} and W are affine in U and V. Of course if $d = 0$ then $W = 0$, and it remains only the first term of the LMI, $\Lambda^*\Pi_{lin}\Lambda$, which corresponds to LTI uncertainties case.

1.4.5 Time varying parameters and polytopic form

These multipliers are easily implemented since X, Y, Z are real matrices and independent of frequency. The general form is:

$$\Pi_\Delta = \begin{bmatrix} X & Y \\ Y^T & -Z \end{bmatrix} \tag{1.31}$$

where X, $Z =$ and Y are real matrices.

- if $X = X^T > 0$, $Y = -Y^T$ and $Z = X$ the multiplier corresponds to the time varying case with an arbitrary rate of variation;

- if $X = X^T$, $Z = Z^T \geq 0$ the multiplier corresponds to the polytopic case.

1.5 Resolution technique

1.5.1 State-space approach

The classical approach to solve the previous LMI feasibility problem (1.2) is based on the Kalman-Yakubovitch-Popov lemma.

Lemma 1.5.1 *Let us consider M a symmetric matrix, A, B, C, D a state-space representation of Φ such as $\Phi(s) = C(sI - A)^{-1}B + D$ and $\forall \omega \in \mathbf{R}$ $det(j\omega I - A) \neq 0$ then the two following propositions are equivalent: (i) the quadratic constraint*

$$\forall \omega \quad \Phi(j\omega)^* M \Phi(j\omega) < 0 \tag{1.32}$$

is satisfied
(ii) it exists $P = P^T > 0$ such as

$$\begin{bmatrix} C^T \\ D^T \end{bmatrix} M \begin{bmatrix} C & D \end{bmatrix} + \begin{bmatrix} I & 0 \\ A & B \end{bmatrix}^T \begin{bmatrix} 0 & P \\ P & 0 \end{bmatrix} \begin{bmatrix} I & 0 \\ A & B \end{bmatrix} < 0$$

The important point is that the second proposition can easily be solved since it is a feasibility problem under LMI constraints. We notice that inequality does not depend on frequency but a new optimization matrix P has appeared. In other terms an infinite set of constraints has been transformed into one constraint with a new optimization variable P. To involve the stability criterion (1.2) in theorem 1.2.2 it suffices to choose the multipliers as:

$$\sum_{i=1} x_i \Pi(j\omega) = \Psi(j\omega)^* M \Psi(j\omega) \tag{1.33}$$

where M is a symmetric matrix, structured according to the problem considered. This matrix contains all optimization variables. With

$$\Phi(j\omega) = \Psi(j\omega) \left[\begin{array}{c} G(j\omega) \\ I \end{array} \right] = C(j\omega I - A)^{-1}B + D \qquad (1.34)$$

the previous lemma allows to transform the infinite set of inequalities (1.2) into one LMI w.r.t. optimization matrices P and M.

1.5.2 Frequency domain approach

With the previous resolution technique an infinite number of LMI constraints has been replaced by one LMI constraint. Nevertheless this transformation is not "free" since a new optimization matrix P appears whose the size depends on the order of G plus dynamics of Ψ. More precisely the number of decision variables grows quadratically, which can lead to computational problem.
In this article the optimization problem is solved directly from frequency domain inequalities thanks to a griding. Of course the drawback to this approach is the lack of guarantee on the validity of the solution on the frequency domain continuum.
For this, a specific technique is involved to guarantee that the solution is valid on the whole frequency domain. In fact the basic idea is to represent the frequency as a real uncertain parameter in an LFT. The issue is to determine a complex matrix S, for a given strictly positive frequency ω_0 such as:

$$\Xi(j(\omega_0 + \delta\omega)) = F_l(S(\omega_0), \delta\omega I_m) \quad \forall \delta\omega \geq -\omega_0$$

where $\Xi(s)$ is dynamic system, $(A_\Xi, B_\Xi, C_\Xi, D_\Xi)$ a state-space representation of $\Xi(s)$ and m the order of $\Xi(s)$. Besides it is well known that:

$$\Xi(j(\omega_0 + \delta\omega)) = F_l(\Xi_0, \frac{I_m}{\omega_0 + \delta\omega}) \quad \forall \omega \geq -\omega_0$$

with

$$\Xi_0 = \left(\begin{array}{cc} D_\Xi & \frac{C_\Xi}{\sqrt{j}} \\ \frac{B_\Xi}{\sqrt{j}} & -jA_\Xi \end{array} \right)$$

Besides let us notice that:

$$\frac{I_m}{\omega_0 + \delta\omega} = F_l(T, \delta\omega I_m)$$

with:

$$T = \frac{1}{\omega_0} \left(\begin{array}{cc} I_m & I_m \\ -I_m & -I_m \end{array} \right)$$

In brief we have an LFT of an LFT:

$$\begin{aligned}\Xi(j(\omega_0+\delta\omega)) &= F_l(\Xi_0, F_l(T, \delta\omega I_m)) \quad \forall\omega \geq -\omega_0 \\ &= F_l(S(\omega_0), \delta\omega I_m)\end{aligned}$$

where $S(j\omega_0)$ can be written as follows:

$$S(\omega_0) = \begin{pmatrix} D_\Xi & \frac{C_\Xi}{\sqrt{j}} \\ \frac{B_\Xi}{\sqrt{j}} & -jA_\Xi \end{pmatrix} * \left(\frac{1}{\omega_0}\begin{pmatrix} I & I \\ -I & -I \end{pmatrix}\right)$$

where $*$ is the star product. Let us remind that the star product corresponds to the interconnection of two LFTs. This kind of result is widely used in μ-analysis.

The validation on the frequency domain continuum is based on following lemma [12]:

Lemma 1.5.2 *Let us consider the general case of a lower LFT $F_l(M, \Delta_r)$ where Δ_r is a real diagonal model perturbation, whereas M is a complex matrix which is partitioned as:*

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

Let:

$$\tilde{\Delta}_r = \begin{bmatrix} \Delta_r & 0 \\ 0 & \Delta_r \end{bmatrix}$$

$$H = \begin{bmatrix} M_{22} & 0 \\ 0 & M_{22}^* \end{bmatrix} - \begin{bmatrix} 0 & M_{21} \\ M_{12}^* & 0 \end{bmatrix} X^{-1} \begin{bmatrix} M_{12} & 0 \\ 0 & M_{21}^* \end{bmatrix}$$

$$X = \begin{bmatrix} I & M_{11} \\ M_{11}^* & I \end{bmatrix}$$

Let us suppose that $\overline{\sigma}(M_{11}) < 1$ and let $k < 1/\mu_{\Delta_r}(M_{22})$. Then

$$\overline{\sigma}(F_l(M, \Delta_r)) < 1 \quad \forall\Delta_r \in kB_{\Delta_r}$$

if and only if

$$det(I - \tilde{\Delta}_r H) \neq 0 \quad \forall\Delta_r \in kB_{\Delta_r}$$

where μ represents the structured singular value [2] and B_{Δ_r} the unit ball in the space of the structured perturbation Δ_r.

This technical result can be applied to $S(\omega_0)$ interconnected to $\delta\omega$ as a lower LFT where $\delta\omega$ is the real parameter in Δ_r.

Proposition 1.5.3 *if* $\overline{\sigma}(\Xi(\omega_0)) < 1$ *then* $\overline{\sigma}(F_l(S(\omega_0), \delta\omega I_m)) < 1$ *holds true for* $\omega_0 + \delta\omega \in [\underline{\omega}, \overline{\omega}]$ *where* $\underline{\omega}$ *and* $\overline{\omega}$ *are computed as follows:*

$$S(\omega_0) = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix}$$

and

$$X = \begin{bmatrix} I & S_{11} \\ S_{11}^* & I \end{bmatrix}$$

Let then:

$$H = \begin{bmatrix} S_{22} & 0 \\ 0 & S_{22}^* \end{bmatrix} - \begin{bmatrix} 0 & S_{21} \\ S_{12}^* & 0 \end{bmatrix} X^{-1} \begin{bmatrix} S_{12} & 0 \\ 0 & S_{21}^* \end{bmatrix} \tag{1.35}$$

Let η_n *be the real negative eigenvalue of* H *of maximal magnitude. Then:*

$$\underline{\omega} = \omega_0 + \frac{1}{\eta_n} \tag{1.36}$$

Let η_p *be the real positive eigenvalue of* H *of maximal magnitude. Then:*

$$\overline{\omega} = \omega_0 + \frac{1}{\eta_p} \tag{1.37}$$

This proposition is an application of lemma 1.5.2. It suffices to write $F_l(S(\omega_0), \delta\omega I_m) = S_{11} + \delta\omega S_{12}(I_m - \delta\omega S_{22})^{-1} S_{21}$. Besides condition $\overline{\sigma}(M_{11}) < 1$ of lemma 1.5.2 is equivalent to $\overline{\sigma}(S_{11}) < 1$ which corresponds to the condition $\overline{\sigma}(\Xi(\omega_0)) < 1$ of proposition 1.5.3. The assumption $k < 1/\mu_{\Delta_r}(M_{22})$ means that $F_l(S(\omega_0), \delta\omega I_m)$ is well-posed for $\delta\omega = -\omega_0$ and thus for all $\delta\omega \geq -\omega_0$. And finally lemma 1.5.2 can be applied by noting that $det(I - \delta\omega H) = 0$ can be rewritten as $det(I/\delta\omega - H) = 0$.

In brief if we consider a transfer matrix Ξ, in order to determine the frequency domain containing ω_0 such as the maximal singular value of $\Xi(j\omega)$ is inferior to 1, it suffices to evaluate $\overline{\omega}$ and $\underline{\omega}$ as above.
But it remains a problem since the IQC formalism involves a positivity condition (1.2) and not a weak gain condition as used here. To solve this problem a specific bilinear transformation is used. This transformation, named Cayley transform, is the following one:

$$\overline{\sigma}(\Xi) \leq 1 \Leftrightarrow Z + Z^* \geq 0 \tag{1.38}$$

with $\Xi = (I - Z)(I + Z)^{-1}$. In other terms if Z represents the stability criterion typically the positivity condition by taking the opposite of inequality (1.2), an inequality of type weak gain which is completely equivalent to the positivity condition is obtained by this transformation. This equivalence requires that $(I+Z)$ is invertible. Let us notice that in the IQC context $Z = Z^*$.

Remark 4: It is important to notice that Δ_r has no link with the Δ block of the closed loop, where Δ contains all uncertainties, non-linearities, etc... of the analysis problem. This Δ block is not used explicitly by the IQC approach, since it is replaced by a multiplier Π which "describes" more or less finely the input/output behavior of the Δ block thanks to a quadratic constraint. And finally a positivity condition (1.2) on transfer function matrix which corresponds to the stability criterion is got. When a solution is obtained, the Caley transform is involved to derive a weak gain condition and to evaluate the solution validity. This evaluation is based on an LFT realization of $\Xi(\omega)$ where $S(\omega_0)$ is interconnected to Δ_r block which contains the repeated real scalar $\delta\omega$.

Sketch of the algorithm

The algorithm can be summarized by the following steps:

1. A solution is searched for on a frequency domain griding (at worst only one frequency can be used at the beginning) to satisfy the stability criterion $\begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^* \Pi \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} < 0$;

2. If no solution is found stop. Else go to step 3;

3. A frequency domain, where the stability criterion is satisfied with the solution obtained previously, is computed. If the frequency domain is $[0\ +\infty[$ the solution is validated on the whole frequency domain and the stability is proved. Stop. Else go to the step 4;

4. Critical frequencies where the stability criterion is not satisfied are added to the frequency domain griding. Go to the step 1.

As indicated the resolution is based on the feasibility problem under LMI constraints. Consequently the Matlab function `feasp.m` is involved to solve the convex optimization problem represented by the stability criterion at each frequency domain point.

By this iterative approach the validation step is performed a priori and during the LMI optimization problem resolution. The choice of the initial

griding has no influence on the feasibility problem. It is possible to choose at the first iteration just a singleton. But to limit the number of iterations, and consequently the calculation time, without any knowledge a priori, it is recommended to take some frequencies roughly spread on the frequency domain. It is possible, when first solutions are obtained, to tune this initial frequency domain griding to decrease the number of iterations.
Concerning the choice of critical frequencies, more precisely 3 frequency domain points are chosen. Two frequencies, for instance named ω_a and ω_b, correspond to frequencies where the stability criterion intercepts the x/frequencies ax, and the last one corresponds to a frequency domain intermediate point. This point is chosen as follows: 10 uniformly distributed frequency responses are evaluated between ω_a and ω_b, and the frequency which corresponds to the highest domain frequency response among the 10 ones is chosen. This solution does not necessarily allow to detect the H_∞ norm in the interval $[\omega_a\ \omega_b]$ as illustrated in the figure 2.3. But it is usually sufficient to obtain a solution which satisfies the weak gain condition at the next iteration. The algebraic evaluation of a band-limited H_∞ norm, typically an Hamiltonian matrix based technique, is possible. But it is questionable for state-space representation of several hundreds states and/or several dozens inputs/outputs in term of numerical robustness and calculation time for a gain which seems very hypothetical. A last point is the case where the frequency domain is $[\omega_a, +\infty[$: only 2 frequency domain points are added (ω_a and $10\omega_a$).

Remark 5: It is necessary to check the positivity constraint of the multiplier $X(j\omega) = \tilde{R}(j\omega)^* U \tilde{R}(j\omega)$ for all ω. Of course it is possible to proceed of the same way: frequencies where the multiplier is negative are added in the optimization problem by an iterative approach. But this approach is useless and increases the computational burden. By noting that hardly always X is positive when the stability condition is satisfied for any frequency, the positivity condition of X is checked by the technique presented previously just once, i.e. when the stability criterion is satisfied on the whole frequency domain. In the exceptional case where it exists a frequency such as $X(j\omega) < 0$ then this frequency is added in the optimization problem and another solution is searched to satisfy the stability condition and the positivity of X.

Chapter 2

Frequency domain resolution tool

In this chapter we will illustrate the use of the IQC frequency domain resolution tool. We will see thanks to some simple examples how to involve the different multipliers which correspond to the analysis problem. This tool is based on Matlab file `data.m` which contains all necessary data for the analysis problem.

The following results have been obtained with Matlab 2012b, Windows 7 64bits, an Intel(R) Xeon(R) CPU W3530 @2.80GHz and 6.00 Go of RAM.

2.1 Data

The choice of different multipliers is done according to the different fields which are filled in the Matlab file `data.m`. You can find in the following lines the different fields which can be/must be filled.

```
wvalid=[0 logspace(-1,3,100)];
wopt=[1 5 10 20 100];

load closedloopmodel G

VectDelta=[];
PolesFilterDU=[];

VectdeltaRU=[];
PolesFilterRU=[];
```

```
VectdeltaLPV=[];
VarLPV=[];
```

```
NL.sector=[];
NL.slope=[];
```

```
VectdeltaPoly=[];
```

```
IQCoptions.LMIiter=100;
IQCoptions.LMIbound=1e6;
IQCoptions.LMILiter=100;
IQCoptions.LMIDisplay=1;

IQCoptions.TolReel.method1=1e-8;
IQCoptions.Visu=2;
IQCoptions.VisuIter=1;
```

The frequency domain griding represented by the vector `wvalid` is used for the frequency domain representation of the solution (see section 2.1.2). But it is important to keep in mind that these points are not used to valid the solution but just for the representation of this one. Let us remind that the validation is based on technique presented in section 1.5.2, and when a solution is obtained, necessarily this one is valid in the whole frequency domain.

The frequency domain griding represented by the vector `wopt` corresponds to the initial frequency domain griding (see section 1.5.2) used to determine a solution for the feasibility problem. Of course, if it is necessary to have several iterations, the length of this vector increases since new critical frequencies are added iteratively. As indicated previously, the choice of the initial griding has no influence on the feasibility problem. It is possible to choose at the first iteration just a singleton. But to limit the number of iterations, and consequently the calculation time, without any knowledge a priori, it is recommended to take some frequencies roughly spread on the frequency domain.

The Δ block has a specific ordered structure. The different inputs/outputs of the Δ block have to respect the following order:

- Non-linearities
- Dynamic uncertainties
- Time varying parameters
- Real uncertainties
- Polytopic models

For example if we have to consider 2 static non-linearities, 1 MIMO dynamic uncertainty 2×2 and 1 parametric uncertainty repeated 3 times, the first 2 inputs/outputs of Δ must correspond to the 2 non-linearities, the 2 following inputs/outputs of Δ must correspond to the dynamic uncertainty, and the last 3 inputs/outputs of Δ correspond to the real uncertainty. This order must be respected in any case.

A last point concerns the closed loop model. This model can be represented by a state-space realization or a transfer function. An easy way is to load the model from a Matlab file `mat`. The file name is completely free and can be chosen by the user. But the transfer function/state-space representation name **must be** G. It is possible "to build" the closed loop model directly in the Matlbab file `data.m`. **But in no case** the fields indicated in this section and described in the next sections must be removed.

With Matlab file `data.m` it is possible to define options which are used by the resolution algorithm. The default values are indicated in comment lines. It is possible to change these default values by uncommenting lines and by modifying values.

2.1.1 LMI options

These options correspond to the LMI toolbox options.

- `IQCoptions.LMIiter=100`: this option corresponds to the LMI toolbox option `OPTIONS(2)`, i.e. the maximum number of iterations;
- `IQCoptions.LMIbound=1e6`: this option corresponds to the LMI toolbox option `OPTIONS(3)`, i.e. the feasibility radius of the solution;

- `IQCoptions.LMILiter=100`: this option corresponds to the LMI toolbox option `OPTIONS(4)`, i.e. the termination condition according to the solution evolution;

- `IQCoptions.LMIDisplay=1`: this option corresponds to the LMI toolbox option `OPTIONS(5)`, i.e. when nonzero, the trace of execution is turned off.

Please refer to the help for the Matlab function `feasp` for more details concerning these options.

2.1.2 Resolution tool options

- `IQCoptions.TolReel.method1=1e-8`: this option is a threshold to detect the presence of real eigenvalues of H (1.35). Numerically it is unusual to obtain "pure real eigenvalues", i.e. an imaginary part strictly equal to 0. For simple models usually it is possible, but generally speaking for models with several dozens or hundreds states a "true real eigenvalues" appears as a complex eigenvalue with a very low imaginary part in comparison with the real part. But the "low" level of the imaginary part must be defined a priori as a threshold. By default an eigenvalue is considered as real in our tool if the ratio imaginary part/real part is inferior to $1e-8$;

- `IQCoptions.VisuIter=1`: this option allows to displays on the same figure (`figure(1000)`) the maximum singular value of $\Xi(s)$ (1.38) at each iteration. When a solution is obtained the stability criterion is represented by a green line. Furthermore the red stars '*' represents the frequency domain points (named critical frequencies previously) used by the algorithm to find a feasible solution. The frequency domain griding used for the representation is `wvalid`;

- `IQCoptions.Visu=2`: this option corresponds to the final display of results and can take different values:

  - `IQCoptions.Visu=-1`: no result is displayed;
  - `IQCoptions.Visu=0`: the SVD of $\Xi(s)$ (1.38) is represented in `figure(1)`. When a solution is obtained the maximum singular value is inferior to 0 db. The representation is based on the Matlab function `sigma`, consequently the frequency domain griding for the representation is chosen by `sigma` function;

– `IQCoptions.Visu=1`: figure(1) is the same one as `IQCoptions.Visu=0`. The maximum singular value of $\Xi(s)$ (1.38) and eigenvalues of the stability criterion (1.32) are represented in `figure(2)` and `figure(3)` respectively. Let us remind that equation (1.32) represents a factorized form of the stability criterion (1.2) where M is a matrix which contains all decision variables and $\Phi(s)$ all dynamics (closed loop + multipliers). Furthermore the frequency domain griding used for figure(2) and figure(3) is `wvalid`;

– `IQCoptions.Visu=2`: figure(1) and figure(2) are the same ones as `IQCoptions.Visu=1`. Figure(3) and figure(4) respectively represent the SVD and eigenvalues of the equivalent stability criteria (1.2) and (1.32). The continuous line represents the factorized stability criterion (1.32) which is explicitly used by the optimization program and '*' the original stability criterion (1.2). Furthermore the frequency domain griding used for figure(3) and figure(4) is `wvalid`. Of course the continuous line and '*' coincide.

2.2 Output arguments

When the execution of the tool is ended, it is possible do know the values of the different optimization variables/matrices. Theses differences variables are the following ones:

- `xsolSector`: this scalar decision variable corresponds to $x \geq 0$ for static non-linearities in section 1.4.1;
- `xsolPopov`: this scalar decision variable corresponds to the Popov scalar $\lambda \in \mathbf{R}$ for static non-linearities in section 1.4.1;
- `xsolPark`: this scalar decision variable corresponds to the Park scalar $\gamma \geq 0$ for static and slope restricted non-linearities in section 1.4.1;
- `xsolDU`: this matrix decision variable corresponds to the real and symmetric matrix $U = U^T > 0$ for dynamic uncertainty in section 1.4.2.
- `X_c`: this matrix decision variable corresponds to the transfer function matrix $X(s) = x(s)I = X(s)^* = x(s)^*I > 0$ for complex uncertainty in section 1.4.2;
- `xsolRUx`: this matrix decision variable corresponds to the real symmetric matrix $U = U^T > 0$ for real uncertainties/parameters in section 1.4.3 and 1.4.4;

- `X_r`: this matrix decision variable corresponds to the symmetric transfer function matrix $X(s) = X(s)^* > 0$ for real uncertainties/parameter in section 1.4.3 and 1.4.4;

- `xsolRUy`: this matrix decision variable corresponds to the real matrix V for real uncertainties/parameter in section 1.4.3 and 1.4.4;

- `Y`: this matrix decision variable corresponds to the anti-symmetric transfer function matrix $Y(s) = -Y(s)^*$ for real uncertainties/parameters in section 1.4.3 and 1.4.4;

- `Xp`: this matrix decision variable corresponds to the real and symmetric matrix $X = X^T$ for polytopic model in section 1.4.5;

- `Yp`: this matrix decision variable corresponds to the real matrix Y for polytopic model in section 1.4.5;

- `Zp`: this matrix decision variable corresponds to the real, symmetric and positive matrix $Z = -Z^T \geq 0$ for polytopic model in section 1.4.5;

- `Msol`: this matrix represents the solution matrix of the factorized stability criterion (1.38).

2.3 Termination condition

To solve the stability analysis problem the Matlab function `feasp` is used. Two kinds of results are obtained: the problem is feasible or not. Let us remind the help provided by Matlab for the function `feasp`:

```
Given an LMI  feasibility problem
              Find x such that L(x) < R(x),
feasp solves the auxiliary convex program:

Minimize   t    subject to   L(x) < R(x) + t*I

The system of LMIs is feasible iff. the global minimum TMIN is
negative. The current best value of t is displayed by feasp at
each iteration.
```

In our optimisation problem `R(x)=0`. If the best value of `t` is negative then a feasible solution is obtained at the current iteration. You can find in the following example a typical solution:

```
Solver for LMI feasibility problems L(x) < R(x)
   This solver minimizes  t  subject to  L(x) < R(x) + t*I
   The best value of t should be negative for feasibility

Iteration   :    Best value of t so far

    1                        0.282057
    2                        0.064808
    3                        0.064808
    4                        0.037917
    5                        0.037917
    6                        0.012852
    7                        0.012852
    8                        0.012852
    9                    6.110689e-03
   10                    6.110689e-03
   11                    6.110689e-03
   12                    3.713162e-03
   13                    3.713162e-03
   14                    2.292605e-03
   15                    2.292605e-03
   16                    1.348521e-03
   17                    1.348521e-03
   18                    9.791401e-04
   19                    9.791401e-04
   20                    6.292507e-04
   21                    6.292507e-04
   22                    9.828145e-05
   23                    4.871256e-05
   24                    6.046967e-06
   25                    3.558655e-06
   26                    6.411722e-07
   27                    -1.676272e-06

Result:  best value of t: -1.676272e-06
         f-radius saturation:  0.148% of R =  1.00e+06
```

Here `t=-1.676272e-06`, consequently, either the solution is valid on the continumm frequency domain and the stability is proved, or it is not the case, then it is necessary to add critical frequencies and to solve the optimisation problem with the new frequncy domain griding (see section 1.5.2).

In the example below, no solution is obtained:

```
 Solver for LMI feasibility problems L(x) < R(x)
    This solver minimizes  t  subject to  L(x) < R(x) + t*I
    The best value of t should be negative for feasibility

 Iteration   :    Best value of t so far

     1                        0.173464
     2                        0.046868
     3                        0.029950
     4                        0.025091
     5                    7.450290e-03
     6                    4.934380e-03
     7                    1.991907e-03
     8                    1.991907e-03
     9                    9.563769e-04
    10                    1.138023e-04
    11                    1.138023e-04
    12                    1.138023e-04
    13                    5.222925e-05
    14                    5.222925e-05
    15                    5.222925e-05
    16                    3.899955e-05
    17                    3.899955e-05
    18                    1.325824e-05
    19                    1.325824e-05
    20                    1.716911e-06
***                 new lower bound: -7.267890e-07
    21                    3.064212e-07
***                 new lower bound: -2.262369e-07
    22                    3.328994e-08
***                 new lower bound: -8.639019e-08
    23                    3.328994e-08
***                 new lower bound: 5.155947e-10

 Result:  best value of t: 3.328994e-08
          f-radius saturation:  0.000% of R =  1.00e+06

 Marginal infeasibility: these LMI constraints may be
          feasible but are not strictly feasible
```

Here the best value for `t=3.328994e-08` which is a positive value, consequently no feasible solution has been obtained. The algorithm stops since if it is not possible to find a solution on a frequency domain griding, a fortiori no solution can be valid on the continuum frequency domain (see section 1.5.2).
When a solution is obtained and valid on the whole frequency domain, for real/dynamic LTI uncertainties, it appears the line `Multiplier validation: OK`. It means the the positivity condition (see section 1.4.2, 1.4.3 and 1.4.4) of $X(s)$ on the whole frequency domain has been checked. The line `Elapsed time is` corresponds to the calculation time.
All results are saved in the Matlab file `M_SolAll.dat`.

2.4 Aircraft models

In the following section 2 models which are used to illustrate the approach are presented.

2.4.1 Aeroelastic model

An aeroelastic model with two freeplays on elevators has been considered. This model is characteristic of aeroelastic models used for load level evaluation or stability analysis. The main idea is to separate the condensed structural non-linearities from the rest of the linear aircraft model. The aeroelastic equation is written for the linear part, i.e. for the aircraft in nominal configuration except for its control surfaces (two elevators in our study), which are connected at their hinges but have no stiffness in rotation. Thus the resulting modal basis includes the rotation modes of both elevators which are at zero frequency. For each elevator, the non-linearity is modeled as external force, applied between its attachments. Then both linear and non-linear models are linked via the relative displacement of the attachments and the actuator forces. The study of the non-linear aeroelastic model can be viewed as a problem of closed loop [1].

To limit the computational effort for the frequency domain validation step, it is interesting to reduce the order of G(s). In brief the optimization problem is independent of the $G(s)$ order but the validation step is based on eigenvalues of H whose the size depends on $G(s)$ order. To ensure stability on the full order model a frequency domain error between the full and the reduced model is modelled. This error is seen as a neglected dynamic represented by $\Delta(s)$. Finally $G(s)$ is the linear part seen by the block $\Delta(s)$. $G(s)$ has four inputs/outputs which correspond to inputs/outputs of static non-linearities and Δ block. To reduce the model order a very classical approach has been involved. This one is based on Gramian-based balancing of state-space realizations. More precisely for stable systems, a balanced realization is a state-space representation for which the controllability and observability gramians are equal and diagonal. When this representation is obtained, the states with the lowest controllability/observability gramians are considered as negligible and then are eliminated. The full order aeroelastic model is of order 550. Finally a reduced order model of order 152 is obtained.

2.4.2 Military aircraft

The objective is to analyze the closed loop stability, closed loop which corresponds to the interconnection of a military aircraft model with a control law. In this analysis problem we have to consider:

- One critical static non-linearity which corresponds to a rate limiter. This rate limiter has been transformed into a normalized dead zone;
- Two LPV parameters which correspond to the mach number and the calibrated airspeed with a nominal rate of variation of 0.2 for both. These two time-varying parameters represent the flight case. The mach number and the calibrated airspeed respectively vary from 0 to 1 and from 150 to 275 kts.
- Five real LTI uncertainties. These real uncertainties are combination of different physical real uncertainties as mass, center of gravity position etc... This transformation is necessary to obtain a limited size for the LFT model. But the important think to keep in mind is the stability analysis is done for the maximum variation of real uncertainties and not a for restricted domain. In other words if the stability is guaranteed with all LTI uncertainties, the stability is guaranteed for the whole domain of physical parameters

The structure of Δ is the following one:

- One (0,1) sector non-linearity is considered;
- The Mach number is repeated twice;
- The calibrated airspeed is repeated eight times;
- Each real uncertainty is repeated once;
- Finally the number of inputs/outputs of Δ is sixteen.

2.5 Static non-linearity

2.5.1 Sector non-linearity

Let us consider a linear model interconnected with a static non-linearity Φ which has been normalized to a sector $(0, 1)$. The initial frequency griding is `wopt=[1 5 10 20 100]`. For this it suffices to write:

```
NL.sector=[1]
```

Let us notice that the sector inferior limit $\underline{k}$ is 0 by default and consequently this value is omitted. Let us consider two examples. The first one is extracted from [11] and corresponds to the 'Ex I':

$$\frac{s^2 - 0.2s - 0.1}{s^3 + 2s^2 + 1} \tag{2.1}$$

In this article a sector bounds are evaluated by a dichotomic approach. To modify the sector size by a k factor, either the linear model G is multiplied by k to analyze kG interconnected with a static non-linearity of sector (0,1), or the value k is directly sets in the field `NL.sector` such as `NL.sector=k` Finally the sector bound is obtained with $k = 1.7636$ in 6 iterations and in 0.5 s. The final frequency domain griding contains 20 frequencies. The maximum singular value of $\Xi(s)$ (1.38) obtained at each iteration is given by figure 2.1.

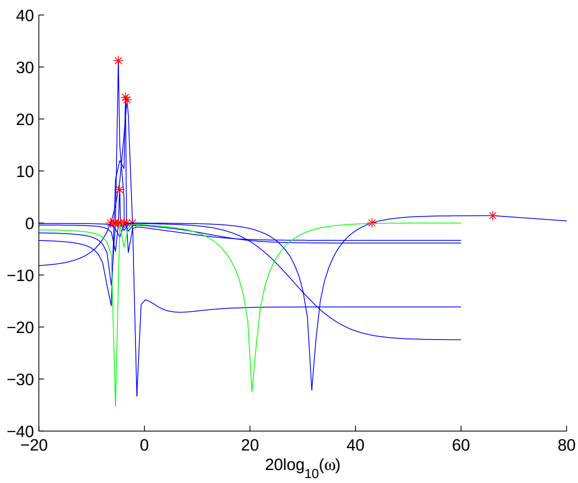

Figure 2.1: Stability criterion iterative representation of a SISO non-linear model

A second example is based on an high dimensional aeroelastic model (see section 2.4.1) interconnected with two static non-linearities which correspond to freeplays. As two non-linearities of sector $(0, 1)$ are considered it suffices to write:

```
NL.sector=[1 1]
```

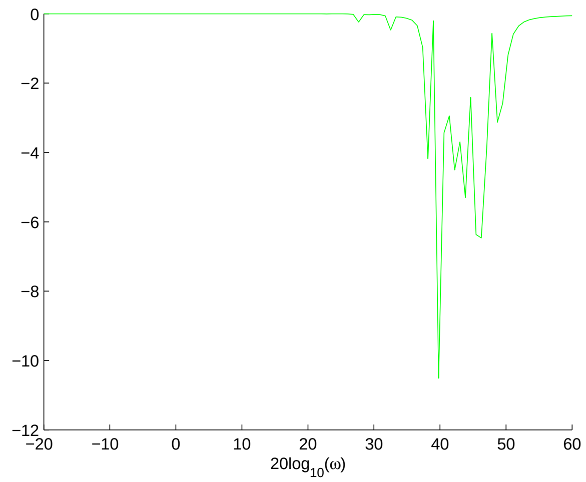

Figure 2.2: Stability criterion iterative representation of a high dimensional non-linear model

The result is obtained in 1 iteration and 1.5 s. Figure 2.2 represents the maximum singular value of $\Xi(s)$ (1.38) obtained at the first and alone iteration.
From this example we notice that the calculation time is very low and a state-space resolution is completely outperformed even with a very simple stability analysis problem. For this kind of problem the KYP lemma based resolution contains 11939 decision variables, which makes the optimization problem untractable.

More generally if n_Φ non-linearities with sector $(0, n_1), ..., (0, n_\Phi)$ are considered, then we write NL.sector=$[n_1 \quad n_2 ... \quad n_\Phi]$.

2.5.2 Slope restricted non-linearity

To involve the slope restricted feature the field `NL.slope=[]` must be modified as follows with $\beta = 1$ and a normalized sector $(0, 1)$ (see section 1.4.1):

```
NL.sector=1
NL.slope=1
```

As previously the slope inferior limit is 0 by default, consequently this value is omitted. The same examples are used to illustrate the non-linear analysis problem. The initial frequency griding is `wopt=[1 5 10 20 100]`. For the example named previsouly 'Ex I', a similar approach to determine sector bounds is used. With `NL.sector=4.5894` and `NL.slope=4.5894` or equivalently $G(s)$ multiplied by 4.5894 a solution is obtained in 6 iterations and 0.5 s. The stability criterion at each iteration is given by figure 2.3.

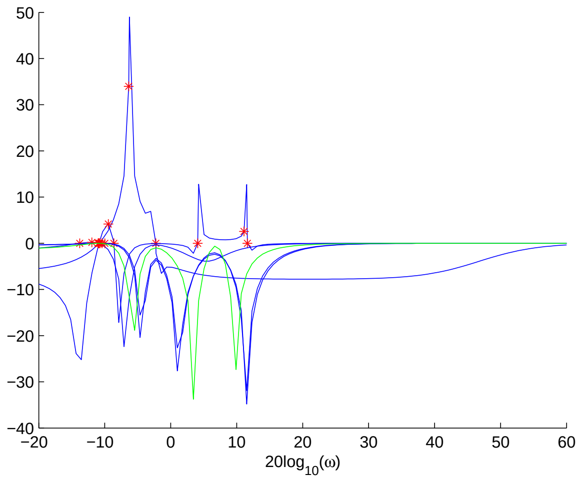

Figure 2.3: Stability criterion iterative representation of a SISO non-linear model

With the high order aeroelastic model it suffices to write in file *data.m*

```
NL.sector=[1 1]
NL.slope=[1 1]
```

Finally a solution is obtained very quickly in 1 iteration and 1.2 s.

2.6 Complex/Dynamic uncertainty

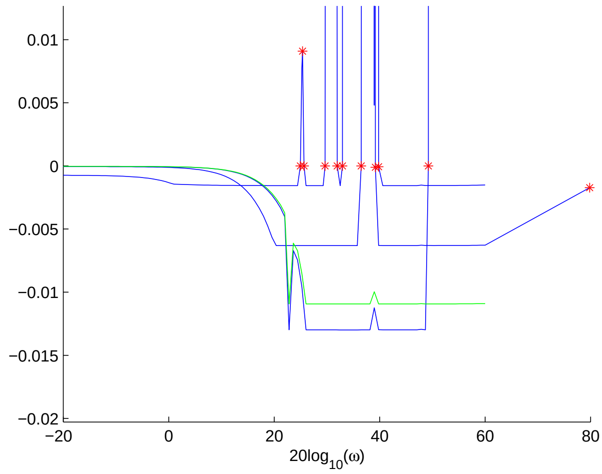

Figure 2.4: Iterative stability criterion of a high dimensional non-linear model with dynamic uncertainty

To take into account a dynamic uncertainty in the stability analysis problem, it suffices to fill the following field:

```
VectDelta=[];
PolesFilterDU=[];
```

`VectDelta` is a vector which contains the size of each dynamic uncertainty. For example if we have to consider two dynamic uncertainties $\Delta_1 \in \mathbf{C}^{\mathbf{2}\times\mathbf{2}}$ and $\Delta_2 \in \mathbf{C}^{\mathbf{3}\times\mathbf{3}}$ partitioned in Δ such as $\Delta = blkdiag(\Delta_1, \Delta_2)$ then `VectDelta=[2 3]`.
`PolesFilterDU` is a vector which contains the multipliers dynamic. As indicated in section 1.4.2, filters are low-pass filters of order 1 with the following structure $\frac{1}{s+s_0}$, where s is the Laplace variable. For example if two filters $filter_1$ and $filter_2$ are considered with respectively one pole at 1 rad/s and 2 rad/s we have to write `PolesFilterDU=[1 2]`.

Let us consider the aeroelastic model. A neglected dynamic is taken into account to ensure the stability on the full order model [1]. The neglected dynamic is represented by complex/dynamic matrix $\Delta(j\omega) \in \mathbf{C}^{2\times 2}$. One low-pass filter with a pole at 1 rad/s is chosen, then:

```
VectDelta=2;
PolesFilterDU=1;
```

As described in section 2.4.1 a dynamic uncertainty and two static non-linearities are considered, then finally in the Matlab file `data.m` we have:

```
NL.sector=[1 1];
NL.slope=[1 1];

VectDelta=2;
PolesFilterDU=1;
```

The initial frequency griding is `wopt=[1 5 10 20 100]`. A solution is obtained in 4 iterations and a calculation time of 8.9 s. As for previous examples the maximum singular value of $\Xi(s)$ is represented at each iteration by figure 2.4. Let us notice that a KYP lemma based resolution leads to consider 12889 decision variables, i.e. 12880 from P + 9 from M (which is slightly different from [1] since here Park multiplier is involved).

2.7 Real/parametric uncertainty

In this section real uncertainties are considered to take into account model parametric uncertainties. To illustrate this section the model presented in 2.4.2 is used. For this we have to consider the following lines:

```
VectdeltaRU=[];
PolesFilterRU=[];
```

`VectdeltaRU` represents a vector which contains the repetition of each real uncertainty. `PolesFilterRU` is a vector which contains the multipliers dynamic. `PolesFilterRU` can be different from `PolesFilterDU`. For example if two real uncertainties $\delta_1 I_2$ and $\delta_2 I_4$ are partitioned in δ such as $\delta = blkdiag(\delta_1 I_2, \delta_2 I_4)$ then `VectDelta=[2 4]`.

For the model described in section 2.4.2 five real uncertainties are taken into account and repeated once. One pole at 10 rad/s is chosen for $X(s)$, consequently we have:

```
VectdeltaRU=[1 1 1 1 1];
PolesFilterRU=10;
```

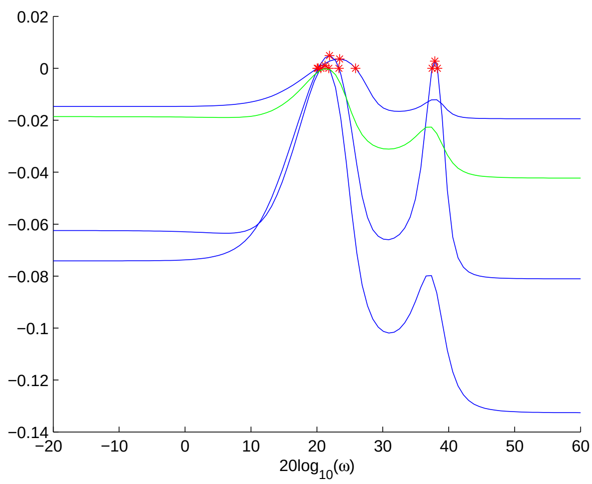

Figure 2.5: Iterative stability criterion of uncertain and non-linear model

Besides to take into account the rate limiter which is represented by a normalized dead-zone `NL.sector=1` and `NL.slope=1`. In brief a non-linear closed loop with parametric uncertainties is analyzed:

```
VectdeltaRU=[1 1 1 1 1];
PolesFilterRU=10;
NL.sector=1;
NL.slope=1;
```

The initial frequency griding is `wopt=[1 5 10 20 100]`. The solution is obtained in 4 iterations and 2.7 s. The maximum singular value of $\Xi(s)$ is represented at each iteration by figure 2.5.

2.8 Time varying parameter

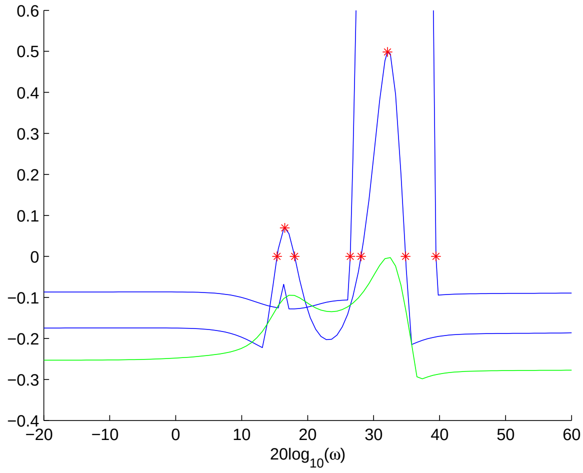

Figure 2.6: Iterative stability criterion of an LPV model

In this section time varying parameters are considered. These parameters are measurable physical parameters, typically, for aeronautical application, these ones correspond to the flight case (Mach number and true airspeed for example). The fields are the following ones:

```
VectdeltaLPV=[];
VarLPV=[];
PolesFilterRU=[];
```

The field `PolesFilterRU` is the same field used for LTI real uncertainties. When LPV parameters and/or parametric uncertainties are taken into account in the analysis problem this field must be filled. Consequently the multipliers dynamic used for these 2 cases is the same one. The second field must be filled as previously: `VectdeltaLPV` is a line vector which contains the repetition of each LPV parameters.

`VarLPV=[]` is a vector which contains the variation rate of each LPV parameter. The values order is important since `VarLPV(i)` corresponds to the variation rate of the parameter `VectdeltaLPV(i)`. Besides it is important to keep in mind that all uncertainties (real or dynamic) and LTV parameters are normalized to 1, consequently the variation rate must take into account this normalization.

Let us consider the military aircraft model with:

- 1 static slope restricted non linearity which corresponds to a rate limiter;
- 2 LPV parameters, Mach number and CAS (calibrated airspeed) repeated respectively 2 times and 8 times, which allow to define the flight case, with a variation rate of 0.2 for both;
- 5 real uncertainties repeated once for each one.

Globally a Δ block with 16 inputs/outputs is obtained. Finally we have to write:

```
NL.sector=[1];
NL.slope=[1];
VectdeltaRU=[1 1 1 1 1];
PolesFilterRU=10;
VectdeltaLPV=[2 8];
VarLPV=[0.2 0.2];
```

The initial frequency griding is `wopt=[1 5 10 20 100]`. A 310 decision variables feasibility problem is solved. The result is obtained in 3 iterations and 250 s. The stability criterion obtained at each iteration is given by figure 2.6

2.9 Polytopic model

In this section a polytopic representation is used to describe the behavior of a dynamic system. The following field has to be considered:

```
NbdeltaPoly=[];
```

`NbdeltaPoly` represents the polytope size. For example with $N = 3$ parameters `NbdeltaPoly=3`, these parameters describe the polytopic domain Δ which contains 2^3 vertices Δ_i such as $\Delta = hull\{\Delta_i, ..., \Delta_{2^N}\}$. Furthermore Δ_i are diagonal matrices with ± 1 in the diagonal. Just for illustration since no polytopic modelisation has been involved for the military aircraft model (section 2.4.2) if $N = 4$, i.e. 4 parameters are used to describe the polytopic model, plus 1 static non-linearity we have to write:

```
NL.sector=1;
NbdeltaPoly=4;
```

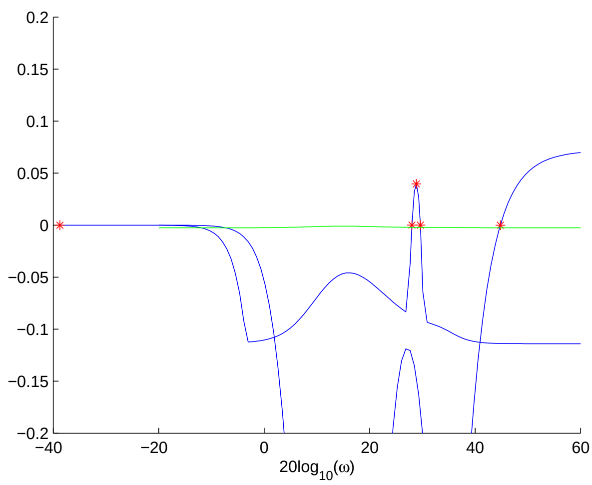

Figure 2.7: Maximum singular value plot polytopic model.

The initial frequency griding is `wopt=[1 5 10 20 100]`. Finally a solution is obtained in 2 s and 3 iterations. The stability criterion obtained at each iteration is given by figure 2.7

2.10 Troubleshootings

- A solution is obtained at the end of the LMI optimization but the representation of the solution, either the maximum singular value of (1.38) or the eigenvalues of (1.2), shows that this one is not valid on the frequency domain continuum. Generally speaking it suffices to decrease the option `IQCoptions.LMIbound` to a smaller value. By default this values is $1e6$ but it is possible and recommended in case of problem to reduce it to $1e3$ for example if necessary. This kind of problem can appear when the number of decision variables is very limited (< 10) and a large difference is obtained between the decision variables values;

- A solution is obtained at the end of the LMI optimization but the representation of the solution, either the maximum singular value of (1.38) or the eigenvalues of (1.2), shows that this one is not valid on the frequency domain continuum. The value of the option `IQCoptions.LMIbound` has been decreased but without noticeable improvement. In this case it can be necessary to modify the value of `IQCoptions.TolReel.method1` (see section 2.1.2). The default value is $1e-8$, it means that all eigenvalues of H (1.35) with a ratio between the imaginary part and the real part superior to $1e-8$ is considered as complex. Consequently these eigenvalues does not correspond to a violation of the stability criterion (see section 1.5.2). But it is possible that an eigenvalue with a ratio superior to $1e-8$ was a "true" real eigenvalue and corresponds to a critical frequency. This eigenvalue is not detected as a real eigenvalue due to the threshold defined by `IQCoptions.TolReel.method1`. In this case it can be necessary to increase the value of `IQCoptions.TolReel.method1` to $1e-6$. It is not recommended to increase beyond $1e-6$. The value of `IQCoptions.TolReel.method1` should vary between $1e-6$ and $1e-8$;

- Another problem can appear in the case where the number of iterations increases without any noticeable improvement of the solution between each iteration (for example by displaying on `figure(1000)` the solution at each iteration thanks to the option `IQCoptions.VisuIter=1`). It is not easy to give a threshold beyond which the number of iterations can be considered as unusual. But for a number of iteration superior to 20 with no noticeable improvement between each iteration, a convergence problem can be suspected. It can mean that some eigenvalues are considered as real whereas they are complex (see section 1.5.2). In this case it can be necessary to decrease the option

`IQCoptions.TolReel.method1`. The default value is $1e-8$. This value is sufficiently low for all tested problems. But if this option has been increased to $1e-6$ (see previous point) maybe it is necessary to decrease it to $1e-7$ or $1e-8$;

- Several tests are involved to check that the expected data in the Matlab file `data.m` are coherent, for example integer and positive values for the repetition of real uncertainties, real values for the sector size, etc...Error messages provided by the IQC tool should be sufficiently clear to allow the user to fix problems. But all possible tests are not involved, and it is possible that incoherent data lead to stop the IQC tool with a typical red error message of Matlab. In this case please check the data with the description given in the user's guide.

- Other options `IQCoptions.LMIiter`, `IQCoptions.LMILiter`, `IQCoptions.LMIDisplay` (see section 2.1.1) can be tuned in case of problem during the LMI resolution, but these options are specific to the LMI toolbox, consequently the user should refer to the `help` provided by Matlab concerning the function `feasp`;

- For any problem do not hesitate to contact the author at this email address: **fabrice.demourant@onera.fr**.

Bibliography

[1] F. Demourant. New algorithmic approach based on integral quadratic constraints for stability analysis of high order models. In *Control Conference (ECC), 2013 European*, pages 359–364, July 2013.

[2] J. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proceedings, Part D*, 129(6):260–265, 1982.

[3] M. Fan, A. Tits, and J. Doyle. Robustness in the presence of mixed parametric uncertainty and unmodeled dynamics. *IEEE Transactions on Automatic Control*, 36(1):25–38, 1991.

[4] U. Jonsson. Stability analysis with popov multiplieurs and integral quadratic constraints. *Systems and Control Letters*, 31:85–92, 1997.

[5] U. Jonsson. Lecture notes on integral quadratic constraints. 2001.

[6] U. Jonsson and A. Rantzer. A unifying format for multiplier optimization. In *American Control Conference, Proceedings of the 1995*, volume 5, pages 3859–3860 vol.5, Jun 1995.

[7] U. Jonsson and A. Rantzer. Systems with uncertain parameters - time variations with bounded derivatives. *International Journal of Robust and Nonlinear Control*, 6:969–982, 1996.

[8] C.-Y. Kao, A. Megretski, and U. Jonsson. Specialized fast algorithms for iqc feasibility and optimization problems. *Automatica*, 40(2):239 – 252, 2004.

[9] H. Khalil. *Nonlinear systems, 2nd edition*. Prentice Hall, 1995.

[10] A. Megretski and A. Rantzer. System analysis via integral quadratic constraints. *IEEE Transactions on Automatic Control*, 42(6):819–829, 1997.

[11] P. Park. Stability criteria of sector- and slope-restricted lur'e systems. *Automatic Control, IEEE Transactions on*, 47(2):308 –313, feb 2002.

[12] A. Sideris and R. Sanchez Pena. Robustness margin calculation with dynamic and real parametric uncertainty. *IEEE Transactions on Automatic Control,*, 35(8):970 –974, aug 1990.