Nonsmooth Methods for Control Design with Integral Quadratic Constraints

Pierre Apkarian*, Dominikus Noll†, and Olivier Prot†

*ONERA-CERT, Centre d'études et de recherche de Toulouse, Control System Department, 2 av. Edouard Belin, 31055 Toulouse, France
†Université Paul Sabatier, Institut de Mathématiques, 118, route de Narbonne, 31062 Toulouse, France

Abstract

We develop an optimization technique to compute local solutions to synthesis problems subject to integral quadratic constraints (IQCs). We use the fact that IQCs may be transformed into semi-infinite maximum eigenvalue constraints over the frequency axis and approach them via nonsmooth optimization methods. We develop a suitable spectral bundle method and prove its convergence in the sense that, for an arbitrary starting point, the accumulation points of the sequence of serious iterates are critical. Our new way to look at IQCs is particularly suited for systems with large state dimension, because we avoid the use of Lyapunov variables. We present two numerical tests which validate our approach in the case of parametric uncertainties.

Keywords. Robust control, parametric uncertainty, integral quadratic constraint (IQC), structured controllers, NP-hard problems, nonsmooth optimization.

Notation

Let $\mathbb{R}^{n\times m}$ be the space of $n \times m$ matrices, equipped with the corresponding scalar product $\langle X, Y\rangle = X \bullet Y = \mathrm{Tr}(X^\top Y)$, where $X^\top$ is the transpose of the matrix X, $\mathrm{Tr}\,(X)$ its trace. For complex matrices X^{H} stands for its conjugate transpose. For Hermitian or symmetric matrices, $X \succ Y$ means that $X - Y$ is positive definite, $X \succeq Y$ that $X - Y$ is positive semidefinite. We use the symbol λ_1 to denote the maximum eigenvalue of a symmetric or Hermitian matrix. We use concepts from nonsmooth analysis covered by [11]. For a locally Lipschitz function $f : \mathbb{R}^n \to \mathbb{R}$, $\partial f(x)$ denotes its Clarke subdifferential at x.

1 Introduction

In many control applications stability and performance have to be guaranteed in the presence of various types of uncertainties in the system. These may be due to the fact that model equations are not perfectly known, or they may be attributed to idealizations in the model building, such as neglected nonlinearities, truncation of a high order components, or simply to incomplete knowledge of some physical parameters. The present paper discusses a numerically efficient way to analyse given controllers and to synthesize new ones in the presence of uncertainties, which we describe by so-called integral quadratic constraints (IQCs). While analysis with IQCs has been investigated from the late 1990s on, see e.g. [32, 27, 19, 18], there are fewer references devoted to synthesis under IQCs, due to its inherent difficulty [36, 10, 22, 13]. Note that in all existing approaches, synthesis problems involving IQCs are characterized by BMIs. This leads to a major challenge in numerical optimization, not only because these problems are nonconvex. More seriously, these BMIs are of large size due to the presence of Lyapunov variables, and are often very ill-conditioned due to the disparity between the Lyapunov variables and the controller gains. In this work, we have therefore adopted a different strategy, where Lyapunov variables are avoided. This leads to smaller and better tracktable optimization programs.

The organization of the paper is as follows. In Section 2 we show how IQCs arise and can be transformed into semi-infinite maximum eigenvalue constraints by introducing multipliers or scalings. In Sections 3 we show how to compute the function values of our nonsmooth objective. A specialized bisection algorithm to compute function values is given in Section 4. Section 5 discusses how to compute generalized derivative information. Computation of generalized gradients in several cases of practical interest is addressed in section 6. These include IQC analysis, positive real synthesis, parametric robust synthesis, and more generally, IQCs with dynamic multipliers. A central result of the paper is the nonsmooth algorithm introduced in section 4, whose details and practical aspects are discussed, and for which we have established convergence to critical points in [6]. Finally, several numerical experiments to validate our approach are given in section 8.

2 Integral quadratic constraints

Consider an uncertain plant in LFT (Linear Fractional Transformation) form:

$$\begin{array}{rcl} \begin{bmatrix} \dot{x} \\ z_\Delta \\ z \\ y \end{bmatrix} & = & \begin{bmatrix} A & B_\Delta & B_1 & B_2 \\ C_\Delta & D_{\Delta\Delta} & D_{\Delta 1} & D_{\Delta 2} \\ C_1 & D_{1\Delta} & D_{11} & D_{12} \\ C_2 & D_{2\Delta} & D_{21} & 0 \end{bmatrix} \begin{bmatrix} x \\ w_\Delta \\ w \\ u \end{bmatrix} \\ w_\Delta & = & \Delta(z_\Delta)\,, \end{array} \tag{1}$$

where Δ is an (unknown) uncertain continuous nonlinear operator, which varies in a known class $\mathbf{\Delta}$ of uncertainties. Here $x \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^{m_2}$ the control signal, $w \in \mathbb{R}^{m_1}$ the exogenous input, $z \in \mathbb{R}^{p_1}$ the performance variable, $y \in \mathbb{R}^{p_2}$ the measurement vector and $(w_\Delta, z_\Delta) \in \mathbb{R}^{m_\Delta} \times \mathbb{R}^{p_\Delta}$ represents the uncertainty channel.

We assume that performance of the system (1) is expressed by the channel (w, z) through the integral quadratic constraint (IQC) specified by the Hermitian matrix $\Pi_p(s) = \Pi_p^{\mathsf{H}}(s)$

$$\int_{-\infty}^{+\infty} \begin{bmatrix} z(j\omega) \\ w(j\omega) \end{bmatrix}^{\mathsf{H}} \Pi_p(j\omega) \begin{bmatrix} z(j\omega) \\ w(j\omega) \end{bmatrix} d\omega \leq 0 \tag{2}$$

for all square integrable signals w. Then the robust control problem is to find a linear time-invariant output feedback controller

$$K(s) = C_K(sI - A_K)^{-1}B_K + D_K, \qquad A_K \in \mathbb{R}^{k\times k}, \tag{3}$$

for the uncertain plant (1), such that the following conditions are satisfied:

(*i*) The closed-loop system (1), (3) is internally stable for all $\Delta \in \mathbf{\Delta}$.

(*ii*) The performance IQC (2) holds for all $\Delta \in \mathbf{\Delta}$.

Here k is the order of the controller, and the possibility $k = 0$ of a static controller $K(s) = D_K$ is included.

Following the IQC approach in [32], we solve the robust synthesis problem by assuming that we dispose of a class $\mathbf{\Pi}$ of Hermitian matrix functions $\Pi_\Delta(s) = \Pi_\Delta^{\mathsf{H}}(s)$, called multipliers or scalings, such that for every fixed $\Pi_\Delta(s) \in \mathbf{\Pi}$, all admissible uncertainties $\Delta \in \mathbf{\Delta}$ satisfy the integral quadratic constraint (IQC) defined by $\Pi_\Delta(s) = \Pi_\Delta^{\mathsf{H}}(s)$, that is,

$$\int_{-\infty}^{+\infty} \begin{bmatrix} z_\Delta(j\omega) \\ \Delta(z_\Delta)(j\omega) \end{bmatrix}^{\mathsf{H}} \Pi_\Delta(j\omega) \begin{bmatrix} z_\Delta(j\omega) \\ \Delta(z_\Delta)(j\omega) \end{bmatrix} d\omega \geq 0 \tag{4}$$

for all square integrable test signals z_Δ. In other words, we assume that the class class $\mathbf{\Pi}$ of multiplier matrix functions $\Pi_\Delta(s)$ is such that

$$\mathbf{\Delta} \subseteq \bigcap_{\Pi_\Delta(s)\in\mathbf{\Pi}} \{\Delta : \Delta \text{ satisfies (4) with } \Pi_\Delta(s)\}, \tag{5}$$

and we hope that the inclusion in (5) is as tight as possible in order to avoid conservatism. Note that multipliers $\Pi_\Delta(s)$ and $\Pi_p(s)$ are often restricted to constant Hermitian matrices, but in this work we will consider dynamic multipliers $\Pi_\Delta(\cdot)$.

Let us introduce the closed-loop transfer matrix

$$T(s,K) = \begin{bmatrix} T_{\Delta\Delta}(s,K) & T_{\Delta w}(s,K) \\ T_{z\Delta}(s,K) & T_{zw}(s,K) \end{bmatrix} := \left\{ \begin{array}{rcl} \dot{x}_{c\ell} & = & \mathcal{A}(K)x_{c\ell} + \mathcal{B}(K)\begin{bmatrix} w_\Delta \\ w \end{bmatrix} \\ \begin{bmatrix} z_\Delta \\ z \end{bmatrix} & = & \mathcal{C}(K)x_{c\ell} + \mathcal{D}(K)\begin{bmatrix} w_\Delta \\ w \end{bmatrix} \end{array} \right. , \tag{6}$$

where state-space data $\mathcal{A}(K)$, $\mathcal{B}(K)$, $\mathcal{C}(K)$ and $\mathcal{D}(K)$ determine the closed-loop system (1) and (3) with the Δ-loop $w_\Delta = \Delta(z_\Delta)$ still open. Here closed-loop data are given as:

$$\begin{gathered} \mathcal{A}(K) := \mathcal{A} + \mathcal{B}_2 K \mathcal{C}_2,\ \mathcal{B}(K) := \mathcal{B}_1 + \mathcal{B}_2 K \mathcal{D}_{21},\ \mathcal{C}(K) := \mathcal{C}_1 + \mathcal{D}_{12} K \mathcal{C}_2, \\ \mathcal{D}(K) := \mathcal{D}_{11} + \mathcal{D}_{12} K \mathcal{D}_{21}\,, \end{gathered} \tag{7}$$

with the standard dynamic augmentation

$$\begin{array}{lll} \mathcal{A} := \begin{bmatrix} A & 0 \\ 0 & 0_k \end{bmatrix} & \mathcal{B}_1 := \begin{bmatrix} B_\Delta & B_1 \\ 0 & 0 \end{bmatrix}, & \mathcal{B}_2 := \begin{bmatrix} 0 & B_2 \\ I_k & 0 \end{bmatrix} \\ \mathcal{C}_1 := \begin{bmatrix} C_\Delta & 0 \\ C_1 & 0 \end{bmatrix}, & \mathcal{D}_{11} := \begin{bmatrix} D_{\Delta\Delta} & D_{\Delta 1} \\ D_{1\Delta} & D_{11} \end{bmatrix} & \mathcal{D}_{12} := \begin{bmatrix} 0 & D_{\Delta 2} \\ 0 & D_{12} \end{bmatrix}, \\ \mathcal{C}_2 := \begin{bmatrix} 0 & I_k \\ C_2 & 0 \end{bmatrix}, & \mathcal{D}_{21} := \begin{bmatrix} 0 & 0 \\ D_{2\Delta} & D_{21} \end{bmatrix} & K := \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \end{array} \tag{8}$$

Then we have the following fundamental fact, see [26, 32] for details.

Theorem 1 *Suppose K is nominally closed-loop stabilizing, i.e. $\mathcal{A}(K)$ is Hurwitz. Assume (5) and suppose there exists a multiplier $\Pi_\Delta(s) \in \mathbf{\Pi}$ such that the following frequency domain inequality (FDI) is satisfied:*

$$\left[\frac{T(j\omega,K)}{I}\right]^{\mathsf{H}} \Pi(j\omega) \left[\frac{T(j\omega,K)}{I}\right] \prec 0 \quad \forall \omega \in [0,\, \infty] \tag{9}$$

where

$$\Pi := \left[\begin{array}{c|c} \Pi_{11} & \Pi_{12} \\ \hline \Pi_{12}^{\mathsf{H}} & \Pi_{22} \end{array}\right] := \left[\begin{array}{cc|cc} \Pi_{\Delta,11} & 0 & \Pi_{\Delta,12} & 0 \\ 0 & \Pi_{p,11}^{\mathsf{H}} & 0 & \Pi_{p,12} \\ \hline \Pi_{\Delta,12}^{\mathsf{H}} & 0 & \Pi_{\Delta,22} & 0 \\ 0 & \Pi_{p,12}^{\mathsf{H}} & 0 & \Pi_{p,22} \end{array}\right]. \tag{10}$$

Then robust stability (i) *and performance* (ii) *are satisfied for all* $\Delta \in \mathbf{\Delta}$.

Proof. We multiply the FDI in (9) with $[w_\Delta; w]$ from the left and from the right and expand. Using $[z_\Delta; z] = T[w_\Delta; w]$ in (6) we recover the sum of the two expressions in (4) and (2). Since by assumption (4) is satisfied, this term is nonnegative. The whole FDI being negative, dropping the ≥ 0 term leaves us with the remaining term in (2), which must then be negative. □

Remark 1 *Notice that we obtain conditions* (i) *and* (ii) *not just for the set* $\mathbf{\Delta}$, *but for the larger set* $\{\Delta : \Delta$ *satisfies (4) with* $\Pi_\Delta(s)\}$, *where* $\Pi_\Delta(s)$ *is the multiplier in the Theorem. This shows that the IQC approach is conservative, but as soon as equality holds in (5), the conservatism is mild.*

Inequality (9) is known as the robust performance FDI, see [32]. It strongly suggests introducing the nonsmooth function

$$\begin{aligned} f(K,\Pi) &= \max_{\omega\in[0,\,\infty]} \lambda_1 \left(\left[\frac{T(j\omega,K)}{I}\right]^{\mathsf{H}} \Pi(j\omega) \left[\frac{T(j\omega,K)}{I}\right] \right) \\ &= \lambda_{1,\infty} \left(\left[\frac{T(\cdot,K)}{I}\right]^{\mathsf{H}} \Pi(\cdot) \left[\frac{T(\cdot,K)}{I}\right] \right) \end{aligned} \tag{11}$$

and considering the optimization program

$$\begin{array}{ll} \text{minimize} & f(K,\Pi) \\ \text{subject to} & \Pi_\Delta \in \mathbf{\Pi} \\ & K \text{ is closed-loop stabilizing} \end{array} \tag{12}$$

which is then minimized until a value $f(K,\Pi) < 0$ is found. Here the performance multiplier Π_p in (10) is held fixed, and it is assumed that a convex cone $\mathbf{\Pi}$ of multipliers $\Pi_\Delta(s)$ satisfying (5) for the admissible uncertainties $\Delta \in \mathbf{\Delta}$ is available. Notice that $\mathbf{\Pi}$ may a priori be infinite dimensional, but we will later on have to restrict this class in order to make the problem computationally tractable. All that is required for the moment is that $\Pi v \in L^\infty$ for $v \in L^\infty$ for every $\Pi \in \mathbf{\Pi}$, because then the argument of $\lambda_{1,\infty}$ in (11) is an element of L^∞.

We will now address the following questions. How to compute the function value of f? How to compute Clarke subgradients of f? And finally, how to generate descent steps in order to decrease the value of f below 0? Notice that our approach to minimizing f until a negative value occurs is based on a local optimization paradigm. In consequence, we may occasionally end up with a local minimum of (12) whose value is ≥ 0, meaning failure to solve the control problem. In this case the method has to be restarted with a different initial seed.

A special case of program (12) is robustness analysis, where $K = K_0$ is held fixed, K_0 being referred to as the controller to be analysed. The question is then whether the closed loop system with feedback K_0 achieves the robust performance Π_p, that is, whether (i) and (ii) hold uniformly for all admissible $\Delta \in \mathbf{\Delta}$. This problem is easier, being convex in the decision variable $\Pi_\Delta \in \mathbf{\Pi}$. Currently analysis problems are solved by tailored interior point methods for LMIs arising from the Kalman-Yakubovich-Popov lemma [39, 19]. Even in that case we propose to proceed via (12), because this avoids Lyapunov variables. The reduction in the number of unknowns may be dramatic for systems with large state dimension n as well as for problems involving high order dynamic multipliers $\Pi_\Delta(s)$.

3 Computing $\lambda_{1,\infty}$

Introducing the semi-infinite objective function $f(K, \Pi)$ avoids the use of Lyapunov variables, but poses a new major problem. Namely, program (12) is now semi-infinite, and such programs are often difficult since discretization has to be used, which leads back to a large number of unknowns. Fortunately, our situation is different, because for a variety of choices of $\Pi(s)$ there exist efficient ways to compute the function value $f(x)$ for given data $x = (K, \Pi)$.

During the following we shall write $F(K, \Pi; j\omega)$ for the FDI in (9), so that $f(K, \Pi) = \max_{\omega \in [0,\infty]} \lambda_1 \left(F(K, \Pi; j\omega)\right)$. We start by explaining how the function value $f(K, \Pi)$ is computed for constant Π. This is based on an iterative procedure introduced by Boyd *et al.* [9, 8]. We start with the following

Lemma 1 *Let* Π *be constant and* $f(K, \Pi) = \max_{\omega \in [0,\infty]} \lambda_1 \left(F(K, \Pi; j\omega)\right)$. *Then for* $\lambda \in \mathbb{R}$*, the estimate* $f(K, \Pi) < \lambda$ *is equivalent to the following frequency domain test:*

$$\begin{bmatrix} (j\omega I - \mathcal{A}(K))^{-1}\mathcal{B}(K) \\ I \end{bmatrix}^{\mathsf{H}} \Psi[\lambda] \begin{bmatrix} (j\omega I - \mathcal{A}(K))^{-1}\mathcal{B}(K) \\ I \end{bmatrix} \prec 0 \quad \forall \omega \in [0, \infty] \,, \qquad (13)$$

where

$$\Psi[\lambda] := \begin{bmatrix} \mathcal{C}(K) & \mathcal{D}(K) \\ 0 & I \end{bmatrix}^{\mathsf{H}} \begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^{\mathsf{H}} & \Pi_{22} - \lambda I \end{bmatrix} \begin{bmatrix} \mathcal{C}(K) & \mathcal{D}(K) \\ 0 & I \end{bmatrix}. \tag{14}$$

Proof. Substituting $\Psi[\lambda]$ into (13) and expanding leads back to the expression (9). □

Lemma 2 *Let* Π *be constant and* $f(K,\Pi) = \max_{\omega\in[0,\infty]} \lambda_1 \left(F(K,\Pi; j\omega)\right)$*. Then for* $\lambda \in \mathbb{R}$*, the estimate* $f(K,\Pi) < \lambda$ *is satisfied if and only if* $\Psi_{22} \prec 0$ *and the Hamiltonian matrix*

$$\mathcal{H}[\lambda] = \begin{bmatrix} \mathcal{A}(K) - \mathcal{B}(K)\Psi_{22}^{-1}\Psi_{12}^{\mathsf{H}} & -\mathcal{B}(K)\Psi_{22}^{-1}\mathcal{B}(K)^{\top} \\ -(\Psi_{11} - \Psi_{12}\Psi_{22}^{-1}\Psi_{12}^{\mathsf{H}}) & -(\mathcal{A}(K) - \mathcal{B}(K)\Psi_{22}^{-1}\Psi_{12}^{\mathsf{H}})^{\mathsf{H}} \end{bmatrix} \tag{15}$$

has no eigenvalue on $j\mathbb{R}^+$*. Here* $\Psi = \Psi[\lambda]$ *is given by (14) in the previous Lemma.*

Proof. This may be inferred from spectral factorization theory [44, p. 350]. A different way to say it is by way of the Kalman-Yakubovic-Popov lemma, which applies directly to FDIs of the form (13). □

Remark 2 *We are at an important junction here. Having seen the KYP-lemma on scene, we might have the reflex to switch to its branch where a matrix inequality arises. If we put this up for (13), we end up with a bilinear matrix inequality (BMI) for synthesis, while of course for analysis, where* K *is fixed, a linear matrix inequality (LMI) is obtained. The current trend is to solve these LMIs with specially tailored interior point solvers [19]. The inconvenience is that a new unknown variable* X*, the Lyapunov matrix, arises. As its size is quadratic in the order of the system* $\mathcal{A}(K)$*, this leads to a large number of unknowns for large order systems. In practice, LMIs derived from the KYP lemma become quickly inefficient as systems get sizeable. We recommend our own approach instead, because it avoids the use of Lyapunov variables* X *and leads to much smaller convex programs.*

For BMIs the situation is even worse because the problem is in addition non-convex. Practical solutions for BMIs derived from the KYP lemma are therefore limited to small [24] or medium size [43] problems. A theoretical approach to such BMIs creates towers of approximations by LMIs with growing dimensions [29], similar to the idea of generating valid inequalities for integer programs. Theoretically, solving one of these LMIs is good enough to globally solve the BMI. Unfortunately, the size of the LMI in question, i.e. the one high enough in the tower, even if it was known, which is usually not the case, is now exponential in the order of $\mathcal{A}(K)$*. These methods could therefore* *hardly be expected to be functional in practical situations. Again we recommend our own method instead. It is less ambitious, because it does not aim at the globally optimal solution of the BMI, but it works in many cases.*

Suppose we want to check an FDI on a restricted frequency band $I = [\omega_1, \omega_2]$, where $0 \leq \omega_1 < \omega_2 \leq \infty$. Then the Hamiltonian test has to be modified. We write

$$f_I(K, \Pi) := \max_{\omega \in I} \lambda_1 \left(F(K, \Pi; j\omega) \right).$$

Checking an FDI on an interval $I = [\omega_1, \omega_2]$ with $\omega_2 \neq \infty$ can be reduced to the previous case by a one-to-one nonlinear transformation mapping the interval $[\omega_1, \omega_2]$ onto $[0, \infty]$ and applying the Hamiltonian test to the transformed system. Frequencies are related through

$$\omega' = \frac{\omega_1 - \omega}{\omega - \omega_2} \Longleftrightarrow \omega = \frac{\omega' \omega_2 + \omega_1}{\omega' + 1}$$

where $\omega \in [\omega_1, \omega_2]$ and $\omega' \in [0, \infty]$. Correspondingly, we infer the transformations in the Laplace variables

$$\frac{1}{s} = \frac{1 + j\frac{1}{s'}}{j\omega_2 - \omega_1 \frac{1}{s'}} \Longleftrightarrow \frac{1}{s'} = \frac{j\omega_2 - s}{\omega_1 + js}.$$

Using the Redheffer star product [41, 14] and posing $\alpha := \sqrt{\omega_2 - \omega_1}/\omega_2$ this can be written as

$$\frac{1}{s} = \frac{1}{s'} \star \begin{bmatrix} \frac{\omega_1}{j\omega_2} & \alpha \\ \alpha & \frac{1}{j\omega_2} \end{bmatrix}.$$

Finally, an FDI involving $T(s) = C(sI - A)^{-1}B + D = \frac{1}{s} \star \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ on the the interval $[\omega_1, \omega_2]$ is equivalent to an FDI involving $\widetilde{T}(s')$ on the interval $[0, \infty]$ with

$$\widetilde{T}(s') = \frac{1}{s'} I \star \begin{bmatrix} \frac{\omega_1}{j\omega_2} I & \alpha I \\ \alpha I & \frac{1}{j\omega_2} I \end{bmatrix} \star \begin{bmatrix} A & B \\ C & D \end{bmatrix}.$$

We deduce the state-space representation of $\widetilde{T}(s')$ from the right star product in the previous expression

$$\begin{bmatrix} \widetilde{A} & \widetilde{B} \\ \widetilde{C} & \widetilde{D} \end{bmatrix} = \begin{bmatrix} \frac{\omega_1}{j\omega_2} I + \alpha^2 A(I - A/(j\omega_2))^{-1} & \alpha(I - A/(j\omega_2))^{-1}B \\ C(I - A/(j\omega_2))^{-1}\alpha & D + C/(j\omega_2)(I - A/(j\omega_2))^{-1}B \end{bmatrix},$$

which is well-defined since A is Hurwitz. Altogether, we obtain the following

Lemma 3 *Let* Π *be static and* $f_I(K,\Pi) = \max_{\omega \in I} \lambda_1\left(F(K,\Pi; j\omega)\right)$. *Then for* $\lambda \in \mathbb{R}$, $f_I(K,\Pi) < \lambda$ *is satisfied if and only if the Hamiltonian*

$$\widetilde{\mathcal{H}} = \begin{bmatrix} \widetilde{\mathcal{A}}(K) - \widetilde{\mathcal{B}}(K)\widetilde{\Psi}_{22}^{-1}\widetilde{\Psi}_{12}^{\mathsf{H}} & -\widetilde{\mathcal{B}}(K)\widetilde{\Psi}_{22}^{-1}\widetilde{\mathcal{B}}(K)^{\top} \\ -(\widetilde{\Psi}_{11} - \widetilde{\Psi}_{12}\widetilde{\Psi}_{22}^{-1}\widetilde{\Psi}_{12}^{\mathsf{H}}) & -(\widetilde{\mathcal{A}}(K) - \widetilde{\mathcal{B}}(K)\widetilde{\Psi}_{22}^{-1}\widetilde{\Psi}_{12}^{\mathsf{H}})^{\mathsf{H}} \end{bmatrix} \tag{16}$$

has no eigenvalues in $j\mathbb{R}^{+}$ *and* $\widetilde{\Psi}_{22} \prec 0$, *where*

$$\widetilde{\Psi}_{22} = \widetilde{\mathcal{D}}(K)^{\top}\Pi_{11}\widetilde{\mathcal{D}}(K) + \widetilde{\mathcal{D}}(K)^{\top}\Pi_{12} + \Pi_{12}^{\mathsf{H}}\widetilde{\mathcal{D}}(K) + \Pi_{22} - \lambda I,$$

and $\widetilde{\mathcal{D}}(K) = \mathcal{D}(K) + \mathcal{C}(K)/(j\omega_2)\left(I - \mathcal{A}(K)/(j\omega_2)\right)^{-1}\mathcal{B}(K)$.

From the above we immediately obtain a procedure to compute $f(K,\Pi)$ when $\Pi(s)$ is an elementary function in the following sense. Let $I_1,\dots,I_\ell$ be sub-intervals of $[0,\infty]$, and let

$$\Pi(s) = \sum_{i=1}^{\ell} \Pi_i \chi_{I_i}(s), \tag{17}$$

where χ_{I_i} is the characteristic function of the ith frequency band. Then computing $f(K,\Pi)$ amounts to computing a finite number of values $f_{I_i}(K,\Pi_i)$.

We may go further and replace each f_{I_i} by a function f_i, where $\Pi_i(s)$ is dynamic on a frequency band I_i and has the special form

$$\Pi_i(s) = R_i(s)^{\mathsf{H}}\Phi_i R_i(s), \quad \Phi_i = \Phi_i^{\mathsf{H}}, \tag{18}$$

where $R_i(s)$ is assumed to have a state-space realization $R_i(s) = C_{iR}(sI - A_{iR})^{-1}B_{iR} + D_{iR}$ and where Φ_i, $(A_{iR}, B_{iR}, C_{iR}, D_{iR})$ and the controller K are unknown. In fact, substituting (18) into (9) gives a term of the same form on each I_i, so the KYP lemma may still be applied, and Φ_i plays the role previously given by Π_i.

4 Hamiltonian algorithm

Similarly to the line taken in [9, 8], where the computation of the H_∞ norm is discussed, Lemma 2 can be used to construct a quadratically convergent algorithm to compute $f(K,\Pi)$ for constant Π. The algorithm from [9, 8] not only computes the function value, it also provides the set of active frequencies. We use the notations $\mathcal{H}[\lambda]$ for (15) and $\Psi[\lambda]$ for (14) to highlight dependency on λ in the above notations. Dependency on K and Π is suppressed because they are fixed.

Computing $f(K,\Pi)$ for constant Π

```
Fix tolerance level $\varepsilon > 0$.
1.  Compute $\underline{\lambda} = \inf\{\lambda : \Psi_{22}[\lambda] \prec 0\}$. This is a lower bound for $f(K,\Pi)$.
    Initialize $\lambda$ with $\underline{\lambda} \leq \lambda \leq f(K,\Pi))$. Put $\Omega = \{\infty\}$.
2.  For given $\lambda$ compute imaginary eigenvalues $j\omega_1, \dots, j\omega_p$ of $\mathcal{H}[\lambda]$.
    If there are none, then return value $\lambda$ and active set $\Omega$. Otherwise
3.  Define $\omega_k^+ = \frac{1}{2}(\omega_k + \omega_{k+1})$, $k = 1, \dots, p-1$. Let $\Omega^+ = \{\omega_1^+, \dots, \omega_{p-1}^+\}$.
4.  Put $\lambda^+ = \max_{i=1,\dots,p-1} (1+\varepsilon)\lambda_1\left(F(K,\Pi; j\omega_i^+)\right) - \varepsilon\underline{\lambda}$. Go back to step 2.
```

Remark 3 *Notice that* $\Psi_{22}[\lambda] = \mathcal{D}(K)^\top \Pi_{11} \mathcal{D}(K) + \mathcal{D}(K)^\top \Pi_{12} + \Pi_{12}^{\mathsf{H}} \mathcal{D}(K) + \Pi_{22} - \lambda I = \Psi_{22}[0] - \lambda I$, *so* $\underline{\lambda} = \lambda_1(\Psi_{22}[0])$ *is the maximum eigenvalue of* $\Psi_{22}[0]$.

Definition. The set $\Omega(K,\Pi) = \{\omega \in [0,\infty] : \lambda_1(F(K,\Pi; j\omega)) = f(K,\Pi)\}$ is called the set of active frequencies at $x = (K,\Pi)$.

Remark 4 *The set of active frequencies* $\Omega(K,\Pi)$ *is estimated in step 3 of the algorithm. If the new* λ^+ *leads to a Hamiltonian* $\mathcal{H}[\lambda^+]$ *without eigenvalues on the imaginary axis, we keep* Ω^+ *from the previous step 3 as our estimation of* $\Omega(K,\Pi)$. *The error tolerance of the peak positions is* $\mathcal{O}(\varepsilon^{\frac{1}{2}})$. *More precisely, if* $\omega_1, \omega_2 \in \Omega$ *and* $[\omega_1, \omega_2]$ *contains the peak* $\bar{\omega}$, *then* $\omega_1^+ = (\omega_1 + \omega_2)/2 =: \bar{\omega}(\varepsilon)$ *is its estimation retained in* Ω^+. *Assume that the algorithm stops at this stage, so that we have* $\phi(\omega_1) = \phi(\omega_2)$ *and* $\phi(\omega_1) \leq \phi(\bar{\omega}) \leq (1+\varepsilon)\phi(\omega_1)$, *where* ϕ *denotes the frequency curve* $\omega \mapsto \lambda_1(F(K,\Pi; j\omega)) - \underline{\lambda}$. *Now recall that* ϕ *is twice differentiable at the peak* $\bar{\omega}$, *[8], so that* $\phi(\bar{\omega}(\varepsilon)) = \phi(\bar{\omega}) + \frac{1}{2}\phi''(\bar{\omega})(\bar{\omega}(\varepsilon) - \bar{\omega})^2 + \mathcal{O}((\bar{\omega}(\varepsilon) - \bar{\omega})^3)$. *Since* $\phi(\bar{\omega}(\varepsilon)) \geq \phi(\bar{\omega})/(1+\varepsilon)$, *we deduce*

$$|\bar{\omega}(\varepsilon) - \bar{\omega}| \leq \sqrt{\frac{2|\phi(\bar{\omega})|}{|\phi''(\bar{\omega})|}}\, \varepsilon^{\frac{1}{2}} + \mathcal{O}(|\bar{\omega}(\varepsilon) - \bar{\omega}|^{\frac{3}{2}}).$$

This means that we have $\varepsilon^{-\frac{1}{2}}|\bar{\omega}(\varepsilon) - \bar{\omega}| \to \sqrt{\frac{2|\phi(\bar{\omega})|}{|\phi''(\bar{\omega})|}}$ *as* $\varepsilon \to 0$.

5 Ingredients from nonsmooth analysis

In order to prepare our nonsmooth descent technique to minimize $f(K,\Pi)$, we need to show how to compute derivative information for f. To this aim, we shall repeatedly invoke the concept of active frequencies defined above. The following can be found in [8, 7]:

Lemma 4 *Let $\Pi(s)$ be rational and K closed-loop stabilizing. Then the set $\Omega(K, \Pi)$ of active frequencies is either finite, or $\Omega(K, \Pi) = [0, \infty]$.*

Proof. Notice that as K and Π are fixed, we are concerned with a one parameter family $\omega \mapsto F(K, \Pi(j\omega); j\omega)$ of $m_1 \times m_1$ Hermitian matrices. This family is analytic on $[0, \infty]$, where analyticity at ∞ follows from the fact that $T(j\omega, K)$ is stable. The bundle of the m_1 eigenvalues $\lambda_i\left(F(K, \Pi(j\omega); j\omega)\right)$ as functions of ω has therefore a very specific structure: it consists of m_1 real analytic functions $\phi_i(\omega)$, $\omega \in \mathbb{R}$; see e.g. [28, p. 82, p. 138], or [31]. More precisely, there are m_1 real analytic functions $\phi_1, \dots, \phi_{m_1}$ such that

$$\{\lambda_1(F(K, \Pi(j\omega); j\omega), \dots, \lambda_{m_1}(F(K, \Pi(j\omega); j\omega)\} = \{\phi_1(\omega), \dots, \phi_{m_1}(\omega)\}, \tag{19}$$

where $\lambda_i(F(K, \Pi(j\omega); j\omega))$ are the eigenvalues of $F(K, \Pi(j\omega); j\omega)$. These ϕ_i are also analytic at ∞, because we are dealing with a stable (proper) transfer matrix $T(j\omega, K)$, which is rational as a function of ω. (The statement may be made more precise: There exist functions ϕ_i which are analytic on a strip $\mathcal{S}$ on the Riemann sphere $\mathbb{S}^2 = \mathbb{C} \cup \{\infty\}$, such that $\mathcal{S}$ contains the meridian $\mathbb{S}^1 = \mathbb{R} \cup \{\infty\}$ passing through the north pole ∞, with ϕ_i taking real values on $\mathbb{S}^1$, such that (19) is satisfied for all $\omega \in \mathbb{S}^1$.)

Suppose now $\Omega(K)$ is infinite. Then one of these m_1 real analytic functions ϕ_i^2 has an infinity of maxima on $\mathbb{S}^1$ with the same value $f(K)$. Since $\mathbb{S}^1$ is compact, these maxima have an accumulation point $\bar{\omega}$ on the meridian. In terms of the analytic extensions on $\mathcal{S}$, that means the Taylor expansion of the ϕ_i^2 in question at $\bar{\omega}$ is the same as the Taylor expansion of the function with constant value $f(K)$. This implies indeed $\phi_i(\omega) = f(K)$ for all ω. □

Remark 5 *This result tells us more generally that an infinite dimensional $\Pi(s)$ of the form $\Pi(j\omega) = \sum_{i=1}^{\ell} \chi_{I_i}(\omega)\Pi_i(j\omega)$, where each $\Pi_i(j\omega)$ is analytic in a frequency band $I_i \subset [0, \infty]$, is open to the differential calculus presented below. In other words, what restricts the class of useful $\Pi(s)$ is not the computation of subgradients, but the computation of the function value.*

The following result is useful for computing Clarke subgradients of the nonsmooth $f = \lambda_{1,\infty} \circ F$.

Lemma 5 *$f = \lambda_{1,\infty} \circ F$ is regular in the sense of Clarke [11].*

Proof. $f = \lambda_{1,\infty} \circ F$ is a composite function of $\lambda_{1,\infty}$, which is nonsmooth but convex, and the smooth nonlinear operator F, mapping the space $\mathbb{R}^N$ with $N =$

$\dim(K)+\dim(\Pi)$ to the infinite dimensional space $C(j[0,\infty],\mathbb{H})$ of continuous functions $j[0,\infty] \to \mathbb{H}$, where $\mathbb{H}$ is the space of Hermitian matrices of appropriate dimension. f is therefore regular in the sense of Clarke [11]. □

These Lemmas make it possible to give a full description of the subdifferential of f. We start by characterizing the subdifferential $\partial\lambda_{1,\infty}(\pi)$ at a given $\pi \in C(j[0,\infty],\mathbb{H})$.

Proposition 1 *Let $\pi \in C(j[0,\infty],\mathbb{H})$, and suppose the set $\Omega(\pi)$ of active frequencies at π:*

$$\Omega(\pi) := \{\omega \in [0,\infty] : \lambda_{1,\infty}(\pi) = \lambda_1(\pi(j\omega))\}$$

is finite. For every active frequency $\omega \in \Omega(\pi)$ let Q_ω denote a matrix whose columns form an orthogonal basis of the eigenspace of $\pi(j\omega)$ associated with the largest eigenvalue $\lambda_1(\pi(j\omega))$. Then the subdifferential $\partial\lambda_{1,\infty}(\pi)$ of the mapping $\lambda_{1,\infty}$ at $\pi \in C(j[0,\infty],\mathbb{H})$ is the set of all linear functionals $\Phi_Y \in C(j[0,\infty],\mathbb{H})^*$ of the form

$$\Phi_Y(\mu) = \sum_{\omega\in\Omega(\pi)} \operatorname{Tr}\left(Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \mu(j\omega)\right), \qquad \mu \in C(j[0,\infty],\mathbb{H}), \tag{20}$$

indexed by the family $Y = (Y_\omega)_{\omega\in\Omega(\pi)}$, where $Y_\omega = Y_\omega^{\mathsf{H}} \succeq 0$ and $\sum_{\omega\in\Omega(\pi)} \operatorname{Tr}(Y_\omega) = 1$.

Proof. This is established using subdifferential formulas for λ_1 and the convex hull rule for max functions. The reader is referred to [25, 11] for details. □

Remark 6 *In the Proposition we avoided the case $\Omega(K,\Pi) = [0,\infty]$ because we never observed it in practice. However, it is possible to extend formula (20) to the case $\Omega(K,\Pi) = [0,\infty]$ by applying the maximum formula of [11] to that case. The families Y and Q are then replaced by infinite families Y_ω, Q_ω together with a probability measure $d\omega$ on $[0,\infty]$ so that $\int_{[0,\infty]} \operatorname{Tr}(Y_\omega)d\omega = 1$, and $\Phi_Y = \int_{[0,\infty]} \operatorname{Tr}(Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \mu(j\omega))d\omega$.*

Our next step is as follows. Given the subdifferential of $\lambda_{1,\infty}$ at $\pi = F(K,\Pi) \in C(j[0,\infty],\mathbb{H})$, we obtain the subdifferential of f at $x = (K,\Pi)$ using the chain rule

$$\partial f(K,\Pi) = F'(K,\Pi)^* \partial\lambda_{1,\infty}(\pi),$$

where $F'(K,\Pi)$ is the Fréchet derivative of F at (K,Π), and $F'(K,\Pi)^*$ its adjoint, which we now need to compute. This may seem arduous at first, since the Banach space dual $C(j[0,\infty],\mathbb{H})^*$ of $C(j[0,\infty],\mathbb{H})$ does not have an easy to manage representation. Fortunately, we only need to know the action of the adjoint $F'(K,\Pi)^*$ on functionals of the special form Φ_Y in (20), and this is easily found. Indeed, the definition of an adjoint gives

$$\langle F'(K,\Pi)(\delta K,\delta\Pi),\Phi_Y\rangle = \langle(\delta K,\delta\Pi),F'(K,\Pi)^*(\Phi_Y)\rangle,$$

where the right hand side is the standard scalar product in a suitable matrix space. Put differently, writing $F'(K,\Pi)^*(\Phi_Y) = (\Lambda_Y,\Sigma_Y)$, where Λ_Y is a matrix compatible with K and Σ_Y a matrix compatible with Π, we have

$$\langle(\delta K,\delta\Pi),(\Lambda_Y,\Sigma_Y)\rangle = \mathrm{Tr}(\delta K^\top\Lambda_Y) + \mathrm{Tr}(\delta\Pi^{\mathsf{H}}\Sigma_Y).$$

In order to pursue, we need the Fréchet derivative $F'(K,\Pi)$. For technical reasons we introduce the notations

$$\begin{bmatrix} T(K,s) & G_{12}(K,s) \\ G_{21}(K,s) & \star \end{bmatrix} \quad := \quad \begin{bmatrix} \mathcal{C}(K) \\ \mathcal{C}_2 \end{bmatrix} (sI-\mathcal{A}(K))^{-1} \begin{bmatrix} \mathcal{B}(K) & \mathcal{B}_2 \end{bmatrix} + \begin{bmatrix} \mathcal{D}(K) & \mathcal{D}_{12} \\ \mathcal{D}_{21} & \star \end{bmatrix}.$$

This simplifies the representation of F':

$$\begin{aligned} F'(K,\Pi)(\delta K,\delta\Pi) = {} & (G_{12}\delta K G_{21})^{\mathsf{H}}\Pi_{11}T(K) + T(K)^{\mathsf{H}}\Pi_{11}G_{12}\delta K G_{21} + \Pi_{12}^{\mathsf{H}}G_{12}\delta K G_{21} \\ & + (G_{12}\delta K G_{21})^{\mathsf{H}}\Pi_{12} + \left[\begin{array}{c} T(K) \\ I \end{array}\right]^{\mathsf{H}} \delta\Pi \left[\begin{array}{c} T(K) \\ I \end{array}\right] \end{aligned} \qquad (21)$$

where dependence on $j\omega$ has been omitted for simplicity.

With $\pi = F(K,\Pi)$ and $\mu = F'(K,\Pi)(\delta K,\delta\Pi)$, we use formula (20) to match coefficients in $\Phi_Y(\mu) = \mathrm{Tr}(\delta K^\top\Lambda_Y) + \mathrm{Tr}(\delta\Pi^{\mathsf{H}}\Sigma_Y)$. This gives

$$\Lambda_Y = 2\sum_{\omega\in\Omega(\pi)} \mathrm{Re}\left(G_{21}(j\omega)Q_\omega Y_\omega Q_\omega^{\mathsf{H}}\left(T(j\omega,K)^{\mathsf{H}}\Pi_{11} + \Pi_{12}^{\mathsf{H}}\right)G_{12}(j\omega)\right)^\top \qquad (22)$$

and

$$\Sigma_Y = \sum_{\omega\in\Omega(\pi)} \left[\begin{array}{c} T(j\omega,K) \\ I \end{array}\right] Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \left[\begin{array}{c} T(j\omega,K) \\ I \end{array}\right]^{\mathsf{H}}. \qquad (23)$$

We sum up our findings in the following

Theorem 2 *Consider a nominally stabilizing controller K, i.e $\mathcal{A}(K)$ Hurwitz, and a multiplier $\Pi(s)$. Assume the set of active frequencies $\Omega(K,\Pi)$ for the FDI in (9) is finite. Then the Clarke subdifferential $\partial f(K,\Pi)$ of f at (K,Π) is the set of subgradients $\left\{(\Lambda_Y,\Sigma_Y) : Y = (Y_\omega)_{\omega\in\Omega(K,\Pi)},\ \ Y_\omega = Y_\omega^{\mathsf{H}} \succeq 0, \sum_{\omega\in\Omega(K,\Pi)}\mathrm{Tr}\,(Y_\omega) = 1\right\}$, where Λ_Y and Σ_Y are given by (22) and (23).*

6 Applications

Note that the results in the previous section cover a large variety of IQCs, including those with dynamic multipliers. As long as f has a composite structure, $\lambda_{1,\infty} \circ F(K,\Pi)$, where Π gathers the multiplier or scaling, the latter represented in a suitable finite basis, it suffices to apply the chain rule to obtain subgradient information. It is also possible to extend the proposed framework to several synthesis FDIs, because the maximum of a finite family of FDIs, $\lambda_{1,\infty} \circ F_i(K,\Pi) < 0$, $i = 1,\dots,q$, can be written as a single FDI, $\lambda_{1,\infty} \circ \operatorname{diag}(F_1(K,\Pi),\dots,F_q(K,\Pi)) < 0\,.$

In IQC analysis the composite function $f = \lambda_{1,\infty} \circ F$ is convex as a function of Π alone and its subdifferential $\partial f(\Pi)$ is the usual subdifferential of convex analysis [25]. Subgradient information is again covered by (23). Subgradient information could in principle be used to find global linear lower bounds for FDI constraints via cutting planes [27], but we do not follow this route here.

In synthesis, multipliers and controller variables are updated simultaneously until satisfaction of the FDI in (9). In accordance with [23] we advocate not to use *D-K* type methods, where K and Π are updated alternatingly.

Finally, as already observed in our nonsmooth approach to H_∞ synthesis [2, 3], specific structural constraints on the controller can be easily incorporated in (12) by applying chain rules to the subgradients. The reader is referred to [5] for details.

In the following, we investigate practically interesting options for analysis and synthesis with IQCs.

6.1 IQC analysis

Nonsmooth results from Section 5 can be used to compute the L_2 gain or H_∞ norm of a system or perform a passivity test. In this situation, the uncertainty channel (w_Δ, z_Δ) is removed $T = T_{zw}$ and the multiplier $\Pi = \Pi_p$ must be selected as

$$\Pi_p = \begin{bmatrix} \gamma^{-1} I & 0 \\ 0 & -\gamma I \end{bmatrix} \text{ respectively } \Pi_p = \begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}. \qquad (24)$$

Assume now for stability analysis the system in (1) is subject to time-invariant parametric uncertainties $T = T_{\Delta\Delta}$, $w_\Delta = \Delta z_\Delta$, where $\Delta \in \mathbf{\Delta}$ is the class of multiplication operators with the block-diagonal structure

$$\Delta = \operatorname{diag}(\dots,\delta_i I,\dots,\Delta_j,\dots) \in \mathbb{R}^{m_\Delta \times m_\Delta} \qquad (25)$$

with normalization $\Delta^\top \Delta \preceq I$. Robust stability can be tested using μ-upper bound multipliers of the form

$$\Pi = \Pi_\Delta = \begin{bmatrix} S & (jG)^{\mathsf{H}} \\ jG & -S \end{bmatrix} \tag{26}$$

with $S = S^{\mathsf{H}} \succ 0$ and $G = G^{\mathsf{H}}$, where both S and G commute with Δ. A possible choice of the class $\mathbf{\Pi}$ are therefore constant multipliers of the form (26), where checking (5) uses that $\Delta \in \mathbf{\Delta}$ in (25) and S, G in (26) commute. The treatment of more complex (dynamic) multipliers is deferred to Section 6.4. The subgradient formulas in Theorem 2 must then be modified to cope with the particular structure of Π and with the extra constraint $S \succ 0$. We define $S = \Sigma\Sigma^{\mathsf{H}}$ where Σ is a lower-triangular Cholesky factor of S. This leads to

$$\delta\Pi = \begin{bmatrix} \Sigma(\delta\Sigma)^{\mathsf{H}} + (\delta\Sigma)\Sigma^{\mathsf{H}} & (j\delta G)^{\mathsf{H}} \\ j\delta G & -\Sigma(\delta\Sigma)^{\mathsf{H}} - (\delta\Sigma)\Sigma^{\mathsf{H}} \end{bmatrix}.$$

Substitution of this expression into the general formula (21), using Theorem 2 and the identities

$$\begin{array}{lcl} \operatorname{Tr}(MN^{\mathsf{H}} + M^{\mathsf{H}}N) & = & 2\operatorname{Tr}(\operatorname{Re} M \operatorname{Re} N^\top + \operatorname{Im} M \operatorname{Im} N^\top) \\ \operatorname{Tr}(ML^\top) & = & \operatorname{Tr}(\operatorname{tril}(M)L^\top) \end{array}$$

which hold for arbitrary complex M, N and lower-triangular L yields the sought formulas. Subgradients with respect to $\operatorname{Re}(\Sigma)$ and $\operatorname{Im}(\Sigma)$ are obtained respectively as $\operatorname{tril}(\operatorname{Re} U)$ and $\operatorname{tril}(\operatorname{Im} U)$ where

$$U := 2 \sum_{\omega \in \Omega(\Sigma, G)} T(j\omega, K) Q_\omega Y_\omega Q_\omega^{\mathsf{H}} T(j\omega, K)^{\mathsf{H}} \Sigma - Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \Sigma\,. \tag{27}$$

Subgradients with respect to $\operatorname{Re} G$ and $\operatorname{Im} G$ are obtained respectively as $-(\operatorname{Im} V + \operatorname{Im}(V)^\top)$ and $\operatorname{Re} V - \operatorname{Re}(V)^\top$ with the definition $V := \sum_{\omega \in \Omega(\Sigma, G)} T(j\omega, K) Q_\omega Y_\omega Q_\omega^{\mathsf{H}}$. As before, $\Omega(\Sigma, G)$ is the set of active frequencies for given Σ and G, the family Y is as in Theorem 2. For time-varying uncertainties with arbitrarily fast variations $w_\Delta = \Delta(t) z_\Delta$, the associate multiplier is real

$$\Pi := \begin{bmatrix} S & \Gamma \\ \Gamma^\top & -S \end{bmatrix}$$

where as before $S = \Sigma\Sigma^\top \succ 0$ and $\Gamma^\top = -\Gamma$. Subgradients are readily inferred from the complex case. For Σ, the subgradients are $\operatorname{tril}(\operatorname{Re} U)$ with U defined in (27). For Γ, we get the subgradients $\operatorname{Re} V - \operatorname{Re}(V)^\top$, where V is as described above.

6.2 H_∞ and positive real syntheses

H_∞ and positive real syntheses are special instances where the uncertainty channel is removed $T = T_{zw}$ and $\Pi = \Pi_p$ is chosen as in (24). For H_∞ synthesis we obtain the subgradients with respect to K

$$\Phi_Y = 2/\gamma \sum_{\omega\in\Omega(K)} \mathrm{Re}\,\left(G_{21}(j\omega)Q_\omega Y_\omega Q_\omega^{\mathsf{H}} T(j\omega,K)^{\mathsf{H}} G_{12}(j\omega)\right)^{\top}\,,$$

which is consistent with the results already derived in [3].

6.3 Robust synthesis

The above reasoning is easily generalized to robust L_2-gain synthesis with structured parametric uncertainties. Again with constant multipliers we have

$$\Pi := \left[\begin{array}{c|c} \Pi_{11} & \Pi_{12} \\ \hline \Pi_{12}^{\mathsf{H}} & \Pi_{22} \end{array}\right] := \left[\begin{array}{cc|cc} S & 0 & (jG)^{\mathsf{H}} & 0 \\ 0 & \gamma^{-1}I & 0 & 0 \\ \hline jG & 0 & -S & 0 \\ 0 & 0 & 0 & -\gamma I \end{array}\right]. \tag{28}$$

Clarke subgradients with respect to K are obtained from (22) and the definition in (28). Partial subgradients with respect to the multiplier are easily inferred from (27) and the partitioning in (6). They are $\mathrm{tril}(\mathrm{Re}\,U)$ and $\mathrm{tril}(\mathrm{Im}\,U)$ for the subgradients with respect to $\mathrm{Re}\,(\Sigma)$ and $\mathrm{Im}\,(\Sigma)$, where $S = \Sigma\Sigma^{\mathsf{H}}$ and

$$U := 2 \sum_{\omega\in\Omega(\Sigma,G,K)} \begin{bmatrix} T_{\Delta\Delta} & T_{\Delta w}\end{bmatrix} Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \begin{bmatrix} T_{\Delta\Delta} & T_{\Delta w}\end{bmatrix}^{\mathsf{H}} \Sigma - \begin{bmatrix} I & 0\end{bmatrix} Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \begin{bmatrix} I & 0\end{bmatrix}^{\mathsf{H}} \Sigma.$$

Subgradients with respect to $\mathrm{Re}\,G$ and $\mathrm{Im}\,G$ are obtained respectively as $-(\mathrm{Im}\,V + \mathrm{Im}\,V^{\top})$ and $\mathrm{Re}\,V - \mathrm{Re}\,V^{\top}$ with the definition

$$V := \sum_{\omega\in\Omega(\Sigma,G,K)} \begin{bmatrix} T_{\Delta\Delta} & T_{\Delta w}\end{bmatrix} Q_\omega Y_\omega Q_\omega^{\mathsf{H}} \begin{bmatrix} I & 0\end{bmatrix}^{\mathsf{H}}$$

Note that $\Omega(\Sigma, G, K)$ is the set of active frequencies for a given triple (Σ, G, K).

Clearly, the above analysis is applicable to more general hybrid block-diagonal operators Δ where each sub-block Δ_i satisfies an IQC defined by Π_i. The reader is referred to the IQCs listed in [32] to enrich the discussion along this line.

6.4 Dynamic multipliers

To reduce conservatism in IQC analysis or synthesis it is possible to use dynamic multipliers. In this context $\Pi(s)$ is an unknown function, and the problem is infinite-dimensional, which is not directly tractable. What hinders for infinite dimensional $\Pi(s)$ are not the subgradient formulas, but the computation of the function $\lambda_{1,\infty} \circ F$. The way out is to use finitely generated multipliers, possibly of the form

$$\Pi(s) = R(s)^{\mathsf{H}} \Phi R(s), \quad \Phi = \Phi^{\mathsf{H}}, \tag{29}$$

where $R(s)$ is assumed to have a state-space realization $R(s) = C_R(sI - A_R)^{-1}B_R + D_R$ and where Φ, the quadruple (A_R, B_R, C_R, D_R) and the controller K are unknown. The framework developed in Sections 3 and 5 applies and differential information can be obtained.

We present an interesting alternative to (29), which is covered by our technique, but in contrast is hardly accessible by state-space methods which call for LMIs or BMIs. Fix a finite partition $[0, \infty] = I_i \cup \cdots \cup I_\ell$ into frequency bands I_i and choose different constant multipliers Π_i on each I_i. In other words, write a dynamic piecewise constant multiplier $\Pi(s) = \sum_{i=1}^{\ell} \Pi_i \chi_{I_i}(s)$, where χ_{I_i} is the characteristic (or indicator) function of the ith band. Then robust performance can be expressed as

$$F_i(K, \Pi_i; j\omega) := \left[\frac{T(j\omega, K)}{I}\right]^{\mathsf{H}} \Pi_i \left[\frac{T(j\omega, K)}{I}\right] \prec 0, \quad \forall \omega \in I_i,\ \forall i = 1, \ldots, \ell\,.$$

This is equivalent to

$$\max_{i=1,\ldots,\ell} \max_{\omega \in I_i} \lambda_1(F_i(K, \Pi_i; j\omega)) < 0\,.$$

We get a max function for which the Clarke gradient is computed using the convex hull rule [11] and which is similar in structure to the Clarke gradient in Theorem 2. In this approach the finite basis is more natural than in (29). The procedure in Section 3 to compute function values and active frequencies works on prescribed frequency intervals I_i just as well as on the whole $[0, \infty]$. See [4] how this can be organized for multiband H_∞ synthesis.

7 Nonsmooth descent method

In this section we explain the basic mechanism of our descent method. Because of the complicated semi-infinite structure of the objective function $f = \lambda_{1,\infty} \circ F$, it will be helpful to look at pure eigenvalue optimization problems $\min_{x \in \mathbb{R}^n} \lambda_1(F(x))$ first.

This class has been studied by various authors, see e.g. [12, 30, 37, 38, 34]. Here we will expand on a method developed by Helmberg and Rendl [20] for semidefinite programming (SDP). In a first step we extend it to address non-convex maximum eigenvalue functions $f = \lambda_1 \circ F$. In a second step, we will go further to include the semi-infinite case $f = \lambda_{1,\infty} \circ F$.

Let $f = \lambda_1 \circ F$ with smooth $F : \mathbb{R}^n \to \mathbb{S}^m$ be a non-convex maximum eigenvalue function. At the current iterate x we consider the local model

$$\phi(y;x) = \lambda_1 \left(F(x) + F'(x)(y-x) \right),$$

which agrees with f at x, i.e., $\phi(x;x) = f(x)$, and is expected to be close to f in a neighbourhood of x, because $F(y) \approx F(x) + F'(x)(y-x)$ by Taylor's theorem. Notice that if F is affine, then $f(y) = \phi(y;x)$ is independent of x.

Recall that λ_1 is the support function of the compact convex set $\mathcal{G} = \{G \in \mathbb{S}^m : G \succeq 0, \mathrm{Tr}(G) = 1\}$. We consider approximations $\mathcal{G}_k \subset \mathcal{G}$ and introduce the family of functions

$$\phi_k(y;x) = \max\{G \bullet [F(x) + F'(x)(y-x)] : G \in \mathcal{G}_k\}. \tag{30}$$

Then $\phi_k(y;x) \leq \phi(y;x)$, and $\phi_k(x;x) = \phi(x;x) = f(x)$ as soon as we assure that $\mathcal{G}_k$ contains at least one matrix of the form $G = ee^\top$, where e is a normalized eigenvector of $F(x)$ associated with the maximum eigenvalue $\lambda_1(F(x))$. The rationale of using ϕ_k instead of ϕ is that ϕ_k is easier to compute than ϕ if the set $\mathcal{G}_k$ is reasonably small.

We use the model $\phi_k(\cdot;x)$ to generate trial steps y^{k+1} away from x which decrease the value $\phi_k\left(y^{k+1};x\right)$ below $\phi_k(x;x) = f(x)$. This is done by solving the unconstrained optimization program

$$\min_{y \in \mathbb{R}^n} \phi_k(y;x) + \frac{\tau_k}{2}\|y - x\|^2 \tag{31}$$

where $\tau_k > 0$ is the proximity control parameter, to be chosen anew at each step k. If the solution y^{k+1} of (31) gives sufficient decrease in f, we accept it as the new iterate x^+ and call this a serious step. Otherwise we refer to y^{k+1} as a null step. In that case we keep x unchanged, but update $\mathcal{G}_k$ and τ_k. More precisely, we try to replace $\mathcal{G}_k$ by a better suited approximation $\mathcal{G}_{k+1}$ containing additional information about the rejected y^{k+1}. This will allow us to do a better job at the next try $k+1$.

We require the following minimal assumptions on the sets $\mathcal{G}_k$:

1. $ee^\top \in \mathcal{G}_k$ for a normalized eigenvector e associated with $\lambda_1(F(x))$.

2. If the solution y^{k+1} of (31) is a null step, then keep $vv^\top \in \mathcal{G}_{k+1}$ for some normalized eigenvector v associated with $\lambda_1\left(F(x)+F'(x)(y^{k+1}-x)\right)$.

3. For y^{k+1} a null step, the necessary optimality condition for (31) gives $0 = F'(x)^\star G^k + \tau_k(y^{k+1}-x)$ and $\phi_k(y^{k+1};x) = G^k \bullet \left[F(x)+F'(x)(y^{k+1}-x)\right]$ for some $G^k \in \mathcal{G}_k$. Then keep $G^k \in \mathcal{G}_{k+1}$

Notice that it is not required that $\mathcal{G}_{k+1}$ includes the previous $\mathcal{G}_k$. Such a strategy would quickly lead to large sets $\mathcal{G}_k$ and would considerably slow down the solution of the subproblems (31). All that is needed is to retain $ee^\top$, $vv^\top$ and one element G^k called for by the optimality condition for (31) at y^{k+1} for the next instance $\mathcal{G}_{k+1}$. The element G^k in item 3. is sometimes referred to as an aggregate subgradient.

One way to assure the validity of these three rules is presented in [20]. A synthesis between the method of Helmberg and Rendl on the one hand and Cullum *et al.* [12] and Oustry [37] on the other hand is discussed in [21]. The elements of $\mathcal{G}_k$ are chosen of the form

$$\alpha\overline{G}^k + Q_k Y Q_k^\top \tag{32}$$

where $Y \in \mathbb{R}^r$ for some $r \ll m$, $Y \succeq 0$, $\alpha \geq 0$, $\alpha + \mathrm{Tr}(Y) = 1$, and where $\overline{G}^k \in \mathcal{G}$ is called the aggregate subgradient. The r columns of the $r \times m$ matrix Q_k form an orthonormal set. At least one normalized eigenvector e associated with $\lambda_1(F(x))$ is part of Q in order to comply with item 1. Similarly, a normalized eigenvector v associated with $F(x)+F'(x)(y^{k+1}-x)$ is included in Q_k to comply with item 2. Finally, to assure item 3, we use the following strategy. Suppose at the last instance $k-1$ the value $\phi_{k-1}(y^k;x)$ was attained at some G^{k-1} such that the optimality condition $0 = F'(x)^\star G^{k-1} + \tau_{k-1}(y^k-x)$ is satisfied. Then we choose $\overline{G}^k = G^{k-1}$ as the next aggregate subgradient.

For large scale problems, Helmberg and Rendl [20] recommend a more subtle way to propagate the information in G^{k-1} to the next sweep. Let $G^{k-1} = \alpha^*\overline{G}^{k-1} + Q_{k-1}Y^*Q_{k-1}^\top$ be the element where the supremum (30) was attained at the last sweep and the optimality condition held. Let $PD^*P^\top$ be a spectral decomposition of Y^*. Split $P = [P_1P_2]$ where P_1 is the basis associated with the important part D_1 of the spectrum, that is, the large eigenvalues, P_2 the basis of the remaining part D_2. Then put

$$\overline{G}^k = \alpha^*\overline{G}^{k-1} + Q_{k-1}P_2D_2P_2^\top Q_{k-1}^\top / \left(\alpha^* + \mathrm{Tr}(D_2)\right)$$

and keep the columns $Q_{k-1}P_1$ in the new matrix Q_k.

Remark 7 *The basic conditions 1. - 3. above guarantee convergence, but an intelligent choice of the basis Q_k of eigenvectors may improve the speed of convergence of the method.*

Spectral bundle algorithm for $\min_{x\in\mathbb{R}^n} f(x)$

```
Parameters: $0 < \gamma < \widetilde{\gamma} < \frac{1}{2} < \Gamma < 1$.
0. Initialize outer loop. Find initial iterate $x$.
1. Outer loop. Stop if $0 \in \partial f(x)$ at current outer iterate $x$. Otherwise
   goto inner loop.
2. Initialize inner loop. Initialize $\mathcal{G}_1 = \partial\lambda_1(F(x))$, $\overline{G}_1 = \frac{1}{m}I_m$, put inner
   loop counter $k = 1$, and choose $\tau_1 > 0$. If old value for $\tau$ from previous
   sweep is memorized, use it to initialize $\tau_1$.
3. Tangent program. At counter $k$ with given $\tau_k > 0$ and $\mathcal{G}_k$ solve
       $\min_{y\in\mathbb{R}^n} \phi_k(y;x) + \frac{\tau_k}{2}\|y-x\|^2$
   Solution is $y^{k+1}$. Find $G_k \in \mathcal{G}_k$ where optimality condition for (31) at
   $y^{k+1}$ is satisfied. Write $G_k^* = \alpha_k^*\overline{G}_k + Q_k Y_k Q_k^\top$ according to (32).
4. Acceptance test. Check whether
       $\rho_k = \frac{f(x) - f(y^{k+1})}{f(x) - \phi_k(y^{k+1};x)} \geq \gamma.$
   If this is the case put $x^+ = y^{k+1}$ (serious step). Compute new memory
   element $\tau^+$ as:
       $\tau^+ = \begin{cases} \frac{\tau_k}{2}, & \text{if } \rho_k > \Gamma \\ \tau_k, & \text{else} \end{cases}$
   Then go back to step 1 to commence a new sweep of outer loop. On the
   other hand, if $\rho_k < \gamma$ then continue inner loop with step 5.
5. Agreement test. Compute
       $\widetilde{\rho}_k = \frac{f(x) - \phi(y^{k+1};x)}{f(x) - \phi_k(y^{k+1};x)}$
   and put
       $\tau_{k+1} = \begin{cases} \tau_k & \text{if } \rho_k < \gamma \text{ and } \widetilde{\rho}_k < \widetilde{\gamma} \\ 2\tau_k & \text{if } \rho_k < \gamma \text{ and } \widetilde{\rho}_k \geq \widetilde{\gamma} \end{cases}$
6. Aggregate subgradient. Compute new set $\mathcal{G}_{k+1}$ according to (32).
   New aggregate subgradient is $\overline{G}_{k+1} = G_k^*$.
7. Inner loop. Increase counter $k \to k+1$ and go back to step 3.
```

Convergence analysis of this method is too lengthy and complex for the present work and is deferred to [6]. We state the main result:

Theorem 3 *Let x^1 be an initial point such that $\{x \in \mathbb{R}^n : f(x) \leq f(x^1)\}$ is bounded. Then every accumulation point $\bar{x}$ of the sequence x^j of serious iterates of the algorithm is a critical point of f, that is, satisfies $0 \in \partial f(\bar{x})$.*

Remark 8 *This may seem a weak convergence certificate, but in practice the sequence of serious steps x^j always converges to a local minimum of f.*

Let us now extend our approach to functions of the form $f = \lambda_{1,\infty} \circ F$. The key information to deal with this case is given by Lemma 4. Consider the case where $\Omega(x)$ is finite, that is

$$f(x) = f(x, \omega_1) = \cdots = f(x, \omega_p)$$

with $\Omega(x) = \{\omega_1, \dots, \omega_p\}$. Given a finite set $\Omega \subset \mathbb{R}_\infty = \mathbb{R} \cup \{\infty\}$ containing $\Omega(x)$, called an extension of $\Omega(x)$, we consider the function

$$f_\Omega(y) = \max_{\omega \in \Omega} f(y, \omega).$$

It is a well-known fact of nonsmooth analysis that the Clarke subdifferential of a maximum function f at x is entirely determined by the subdifferentials of the active elements $f(\cdot, \omega_i)$ at x. In other words,

$$\partial f(x) = \partial f_{\Omega(x)}(x) = \partial f_\Omega(x),$$

where $f_{\Omega(x)}(x) = f_\Omega(x) = f(x)$, $f_{\Omega(x)} \leq f_\Omega \leq f$. Now as Ω is finite, f_Ω is a maximum eigenvalue function. Indeed, it suffices to write $f(\cdot, \omega_i) = \lambda_1 \circ F_i$ for the finitely many $\omega_i \in \Omega$ and let F_Ω a block diagonal matrix with the diagonal blocks F_i arranged in any convenient order. Then $f_\Omega = \lambda_1 \circ F_\Omega$. The point is that we know how to generate descent steps for maximum eigenvalue functions. In fact, we use approximations $\phi_\Omega^k(y; x)$ relating to f_Ω as $\phi_k(y; x)$ relates to f in the earlier part of this section. Then we generate descent steps for f_Ω as in the previous part. If such a step does not give a satisfactory decrease in f, we update $\Omega \to \Omega'$ in such a way that the new $f_{\Omega'}$ gives a better approximation of f.

Spectral bundle algorithm for program $\min\limits_{x \in \mathbb{R}^n} \max\limits_{\omega \in \mathbb{R} \cup \{\infty\}} f(x, \omega)$

```
Parameters $0 < \gamma^\sharp < \gamma < \frac{1}{2}$.
0. Initialize outer loop. Choose initial $x$ such that $f(x) < \infty$.
1. Outer loop. If $0 \in \partial f(x)$ at current $x$ stop, else goto inner loop.
2. Initialize inner loop. Let $x_1 = x$ and choose finite $\Omega_1$ containing $\Omega(x_1)$.
   Put inner loop counter $\ell = 1$.
3. Sub-program. At inner loop counter $\ell$ and current $\Omega_\ell$, $f_{\Omega_\ell}$, $\phi_{\Omega_\ell}(\cdot; x)$ and
   $\phi^k_{\Omega_\ell}(\cdot; x)$ use inner loop of the first algorithm (with counter $k$) to generate
   trial step $x_\ell$ satisfying the test
       $\dfrac{f(x) - f_{\Omega_\ell}(x_\ell)}{f(x) - \phi^k_{\Omega_\ell}(x_\ell; x)} \geq \gamma.$
4. Reality check. Test whether
       $\dfrac{f(x) - f(x_\ell)}{f(x) - \phi^k_{\Omega_\ell}(x_\ell; x)} \geq \gamma^\sharp.$
5. Decision. If this is the case let $x^+ = x_\ell$ and go back to step 1. Otherwise
   add new frequencies to the set $\Omega_\ell$ to obtain $\Omega_{\ell+1}$ and go back to step 3.
```

Notice here that in step 2 a trial point y satisfying $\rho_{k,\ell} \geq \gamma$ is found by a finite procedure via the first algorithm. For fixed ℓ this may involve several inner iterations k updating $\mathcal{G}_k$ and τ_k. As f_{Ω_ℓ} is only an approximation of f satisfying $f_{\Omega_\ell} \leq f$, we need to do a reality check in step 4. Here we expect the weaker estimate $\rho^\sharp_{k,\ell} \geq \gamma^\sharp$ to be satisfied, and this is why we choose $\gamma^\sharp < \gamma$.

In order to assure that a finite number of steps ℓ is sufficient to locate a point y satisfying the second test $\rho^\sharp_{k,\ell} \geq \gamma^\sharp$, we have to guarantee $f_{\Omega_\ell} \to f$ as $\ell \to \infty$. This may for instance be arranged by fixing an increasing sequence Ω^ℓ of finite sets whose union is dense in $\mathbb{R}_\infty$, assuming $\Omega^\ell \subset \Omega_\ell$ at each step ℓ. However, in practice, there are more practical ways to select the sets Ω_ℓ.

Theorem 4 *Suppose the updating strategy $\Omega_\ell \to \Omega_{\ell+1}$ guarantees $f_{\Omega_\ell} \to f$ uniformly on bounded sets. Suppose the initial iterate x^1 is such that $f(x^1) < \infty$ and $\{x \in \mathbb{R}^n : f(x) \leq f(x^1)\}$ is bounded. Then the sequence of iterates x^j generated by the second algorithm is well-defined, i.e., $f(x^j) < \infty$ for all j, and every accumulation point $\bar{x}$ of the sequence x^j is a critical point of f.*

In practice, we select an enriched set of active frequencies as described in some detail in [2]. The convergence proof is again deferred to in [6].

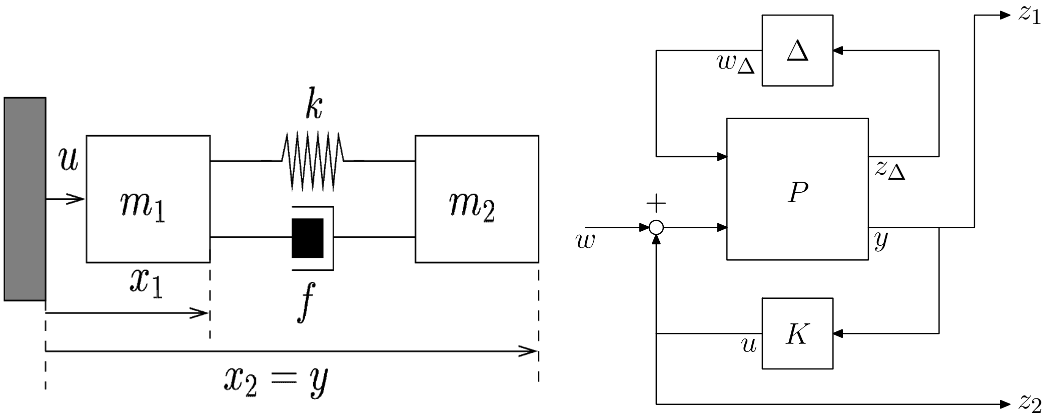

Figure 1: Left: schema of the mass-spring system. Right: interconnection structure for the controller design.

8 Numerical experiments

We have performed two studies in parametric robust H_∞ synthesis to validate our semi-infinite algorithm for IQCs. Our first example is the mass-spring system shown in Figure 1 (left) and described by the following system of second order differential equations

$$\begin{cases} m_1\ddot{x}_1 = -kx_1 + kx_2 - f\dot{x}_1 + f\dot{x}_2 + u \\ m_2\ddot{x}_2 = \ \ kx_1 - kx_2 + f\dot{x}_1 - f\dot{x}_2 \end{cases}$$

with $m_1 = m_2 = 0.5$kg, $k = 1$N/m, and $f = 0.0025$Ns/m. The interconnection structure used for control is shown on the right of Figure 1. Our goal is to synthesize a robust feedback controller K which stabilizes the position x_2 of mass m_2 in the presence of parameter uncertainties of p% on the parameters k and m_2,

$$|1\text{Nm}^{-1} - k|/1\text{Nm}^{-1} \leq p\%, \qquad |0.5\text{kg} - m_2|/0.5\text{kg} \leq p\%$$

where $p \in \{5, 10, 15, 20, 25, 30\}$.

Our second example, the Reichert missile model [42], has already been studied in [1], where the IQC framework has been used to assess robustness of a control system, and in [16, 15, 17, 35] via LPV robust synthesis. We refer to [42, 16] for a description of the model, LFT form, and control objectives. Uncertain parameters of this model are α, the angle of attack, and M, the Mach Number. The nominal value of α is $0°$

System	n	m_Δ	m_1	m_2	p_Δ	p_1	p_2
Mass-spring	4	2	1	1	2	2	1
Missile	9	10	1	1	10	3	2

Table 1: Matrix dimensions of uncertain plants in LFT form (1).

and the nominal Mach number is $M = 3$. We test the cases

$$\alpha \in [-p_1\% \cdot 30^\circ, +p_1\% \cdot 30^\circ], \qquad M \in \left[3 - p_2\% \cdot \frac{3}{2}, 3 + p_2\% \cdot \frac{3}{2}\right],$$

for the choices $(p_1, p_2) = (0.05, 5)$, $(p_1, p_2) = (0.1, 10)$, $(p_1, p_2) = (0.15, 15)$ and $(p_1, p_2) = (0.2, 20)$. Matrix dimensions are given in Table 1 for both mass-spring and missile model.

8.1 Implementation

Our implementation of the algorithm follows Section 7 and treats the case of multipliers (28) for parametric robustness. We shall therefore write the multiplier variable Π as (S, G, γ).

In phase I we require an initial nominal closed-loop stabilizing controller K_0. In the full order case, $k = n$, this is of course standard and can be obtained via AREs or LMIs as available in the MATLAB control toolbox. Computing the optimal H_∞ controller K_∞ provides a lower bound $\gamma_\infty = \|T_{wz}(K_\infty)\|_\infty$ for the optimal robust gain γ^*. We use K_∞ to initialize the K-variable in our algorithm.

Initializing K is more difficult if reduced-order ($k < n$) or structured controllers K are required. Finding $K_0 = K_\infty$ can then even be NP-hard in the sense of [33], but this is a theoretical result and pessimistic as a rule. In practice and for non-contrived examples a closed-loop stabilizing K_0 is either known or easy to compute. What is hard is to optimize its performance and enhance its robustness. Here we use the methods of [7] to compute initial closed-loop stabilizing guesses $K_0 = K_\infty$.

Any value $\gamma_0 > \gamma_\infty$ can be used to initialize the performance parameter in (24). We choose $S_0 = I$ and $G_0 = 0$ in (28). Altogether the following gain values are of interest and reported in Tables 2 and 3:

$$\gamma_\infty = \|T_{wz}(0, K_\infty)\|_\infty \leq \gamma^*_{\text{nom}} := \|T_{wz}(0, K^*)\|_\infty \leq \sup_{\Delta \in \boldsymbol{\Delta}} \|T_{wz}(\Delta, K^*)\|_\infty \leq \gamma^*,$$

where γ^* is the result of our algorithm, K^* the corresponding optimal robust controller, (S^*, G^*) the static multiplier certificate. $T_{wz}(\Delta, K)$ is the transfer matrix of the closed loop (1), (3) performance channel $w \to z$ for fixed $\Delta \in \mathbf{\Delta}$ and for a fixed controller K. Notice that typically $\|T_{wz}(\Delta, K_\infty)\|_\infty = \infty$ for some $\Delta \in \mathbf{\Delta}$, i.e., the nominal H_∞ controller K_∞ is not robust. (Otherwise a robust synthesis would not make sense). We also report

$$\alpha^*_{\text{nom}} = \max\{\operatorname{Re}\lambda : \lambda \in \operatorname{sp}(\mathcal{A}(K^*))\}$$

the closed-loop spectral abscissa of the nominal closed-loop system matrix $\mathcal{A}(K)$ at the robust controller K^*.

Once our algorithm is initialized successfully with $f(K_0, S_0, G_0, \gamma_0) < 0$, phase II starts and the following iterative procedure is used. At stage k, minimization of $(K, S, G) \mapsto f(K, S, G, \gamma_k))$ over (K, S, G) is performed. As soon as a new feasible point (K_k, S_k, G_k) has been reached, that is

$$f(K_k, S_k, G_k, \gamma_k) < 0,$$

we have a certificate $\Pi_{\Delta,k} = (S_k, G_k)$ that a robust controller K_k with robust performance at most γ_k has been found. This process is now repeated a few times in order to improve over the latest γ_k. For that we use extrapolation, as described in subsection 8.3, to compute a better γ_{k+1}, looping on until the optimal value γ^* with corresponding optimal (K^*, S^*, G^*) is approached.

In the case of parametric uncertainties it may be very conservative to use static multipliers $\Pi_\Delta(j\omega)$ in (11), because they characterize time-varying trajectories $\Delta(t)$. For parametric robustness, the uncertainties $\Delta \in \mathbf{\Delta}$ are rather constant, and to reduce conservatism of the IQC formulation, it is therefore preferable to employ dynamic multipliers $\Pi_\Delta(j\omega)$, which is what we do in phase III. We have implemented a routine to optimize our cost function $f(K, \Pi)$ for a dynamic multiplier $\Pi_\Delta(j\omega)$ with 3 frequency bands

$$\Pi_\Delta(j\omega) = \Pi_1 \chi_{[0,\omega_1]}(\omega) + \Pi_2 \chi_{[\omega_1,\omega_2]}(\omega) + \Pi_3 \chi_{[\omega_2,\infty]}(\omega),$$

where Π_1, Π_2 and Π_3 are matrices with the same structure as in (28). Here the cost function f is modified along the lines of subsection 6.4. To compute the value of the FDI on each band, Lemma 3 is applied. After initialization $K_0 = K^*$, $\gamma_0 = \gamma^*$ and with $S_0(j\omega) = S^*$, $G_0(j\omega) = G^*$ for all ω, the procedure $(K_k, S_k(j\omega), G_k(j\omega), \gamma_k) \to (K_{k+1}, S_{k+1}(j\omega), G_{k+1}(j\omega), \gamma_{k+1})$ is precisely the same as in the static case described before, and converges to the optimal triplet $(\bar{K}, \bar{S}(j\omega), \bar{G}(j\omega), \bar{\gamma})$. Again $\bar{K}$ has now a robust performance certificate, $\bar{\Pi}_\Delta(j\omega) = (\bar{S}(j\omega), \bar{G}(j\omega))$, and our experiments show that $\bar{\gamma} < \gamma^*$, so that conservatism is reduced. As in the static case we report

$$\gamma_\infty \leq \bar{\gamma}_{\text{nom}} := \|T_{wz}(0, \bar{K})\|_\infty \leq \sup_{\Delta \in \mathbf{\Delta}} \|T_{wz}(\Delta, \bar{K})\|_\infty \leq \bar{\gamma}$$

and we also list

$$\bar{\alpha}_{\text{nom}} = \max\left\{\operatorname{Re}\lambda : \lambda \in \operatorname{sp}\left(\mathcal{A}(\bar{K})\right)\right\}.$$

Use of a dynamic multiplier $\Pi_\Delta(j\omega) = (S(j\omega), G(j\omega))$ improves over the performance bound γ^* obtained with a static multiplier $\Pi_\Delta = (S, G)$. Two intermediate frequencies ω_1, ω_2 have to be specified in order to draw the 3 frequency bands. For the time being this is done by trial and error. In fact, we use the plot of the FDI at the optimal (K^*, Π^*) from the constant case to select these two values in a such way that each frequency band contains at least one active or nearly active peak. This choice is tricky in so far as one of these peaks may vanish during the optimization process.

Both for static and dynamic multipliers we use the following stopping tests. Let $\varepsilon_f, \varepsilon_x > 0$, (K, S, G) the actual iterate, (K^+, S^+, G^+) the next. Optimization of f is stopped if the descent is too small, that is

$$f(K, S, G, \gamma_k) - f(K^+, S^+, G^+, \gamma_k) < \varepsilon_f(1 + |f(K, S, G, \gamma_k)|),$$

and if the change in the controller parameters is negligible

$$\left\|K - K^+\right\| + \left\|S - S^+\right\| + \left\|G - G^+\right\| < \varepsilon_x(\|K\| + \|S\| + \|G\|).$$

Of course, a maximum number of iterations has been allowed for the algorithm. The tolerance values have been set to $\varepsilon_f = 1e-4$ and $\varepsilon_x = 1e-5$, the value of ε_f is also used as the tolerance when computing the value of the objective function f, see the bisection method on section 4.

8.2 Computing the sets $\Omega(K, \Pi)$ and $\Omega_e(K, \Pi)$

As we already indicated in section 4, computing the set of actives frequencies $\Omega(K, \Pi)$ uses the Hamiltonian technique of [8], which is generally used to compute the H_∞ norm of a plant. The situation is a bit more complicated in the case of IQCs, since we have to deal with complex Hamiltonian matrices. Indeed, the Hamiltonian $\mathcal{H}[\lambda]$ in equation (15) is constructed from the multiplier Π, which is complex in the case of robust synthesis (28). We have observed some numerical problems when computing the eigenvalues of the complex Hamiltonian matrix, especially for the missile model. If not dealt with, this may cause our algorithm to stop prematurely at non critical points, since the active frequency set $\Omega(K, \Pi)$ is in these cases known with low precision. The quality of the subgradients, which depend on $\Omega(K, \Pi)$, is then also bad and the tangent program computes solutions which are no longer descent directions. In order to overcome this difficulty, we have implemented a special routine which significantly improves the precision of existing complex Hamiltonian linear algebra software.

8.2.1 High precision computation of active frequencies

We will use the same notation as in section 3. For $\lambda \in \mathbb{R}$ fixed, the Hamiltonian algorithm requires computing the set

$$\Omega_\lambda := \{\omega \in \mathbb{R} \ : \ \lambda_1(F(K, \Pi, j\omega)) = \lambda\} .$$

It is well known [39] that if $j\omega$ is an eigenvalue of matrix $\mathcal{H} = \mathcal{H}[\lambda]$ in (15), then

$$\lambda_1(F(K, \Pi, j\omega)) = \lambda,$$

where $\lambda \in \mathbb{R}$ is as in the expression $\Psi = \Psi[\lambda]$ in equation (14). An approximation Θ_λ of the set Ω_λ is now obtained by computing the eigenvalues $\nu + j\omega$ of $\mathcal{H}$, and forming

$$\Theta_\lambda := \{\omega \in \mathbb{R} \ : \ \nu + j\omega \in \mathrm{sp}(\mathcal{H}), \nu \in \mathbb{R}, |\nu| < \varepsilon_\omega\} ,$$

where $\mathrm{sp}(\mathcal{H})$ is the spectrum of $\mathcal{H}$, and where $\varepsilon_\omega > 0$ is a suitable numerical tolerance.

Our algorithm uses Newton and bisection methods to improve the precision of these frequencies. Assuming Θ_λ finite and of cardinality $\#\Theta_\lambda$, its elements are $\omega_1 < \cdots < \omega_{\#\Theta_\lambda}$. (The procedure is initialized by those ω_i obtained from the standard linear algebra software). To measure to what precision the frequencies $\omega_i \in \Theta_\lambda$ approximate those in Ω_λ, we introduce

$$J(\Theta_\lambda) := \sum_{k=1}^{\#\Theta_\lambda} (\lambda_1(F(K, \Pi, j\omega_k)) - \lambda)^2,$$

whose best value is $J(\Omega_\lambda) = 0$. Let us also define the sign function $\sigma : \mathbb{R}^+ \to \{-1, 0, 1\}$ by

$$\sigma(\omega) := \mathrm{sign}(\lambda_1(F(K, \Pi, j\omega)) - \lambda),$$

and some useful transformations of set Θ_λ

$$\begin{array}{rcl} B(\Theta_\lambda) & := & \left\{\omega \in \mathbb{R}^+ \ : \ \omega = (\omega_k + \omega_{k+1})/2, k = 1, \cdots, \#\Theta_\lambda - 1\right\} \\ C(\Theta_\lambda) & := & \left\{\omega \in \mathbb{R}^+ \cap \Theta_\lambda \ : \ \sigma(\omega_{k+1})\sigma(\omega_k) = -1, k = 1, \cdots, \#\Theta_\lambda - 1\right\} , \end{array}$$

and an estimation of the number of frequencies to be computed

$$\iota(\Theta_\lambda) := \sum_{k=1}^{\#\Theta_\lambda - 1} \delta_{-1}(\sigma(\omega_k)\sigma(\omega_{k+1})) + \sum_{k=1}^{\#\Theta_\lambda} \delta_0(\sigma(\omega_k)).$$

We compute the set $O(\Theta_\lambda, m)$ defined by

$$\iota(O(\Theta_\lambda, m)) \leq \iota(\tilde{\Theta_\lambda}), \ \forall \tilde{\Theta} \in \{\tilde{\Theta} \subset \Theta_\lambda \ : \ \#\tilde{\Theta} = m\}.$$

Now we present our algorithm for high precision computation of frequencies:

Computing approximation Θ_λ of Ω_λ

```
Fix tolerance level $\varepsilon > 0$.
Initialize $\Theta_0$ and $\Theta'_0 = \Theta_0 \cup B(\Theta_0) \cup \{\frac{\omega_1}{2}, 2\omega_{\#\Theta_\lambda}\}$.
While $J(\Theta_k) \geq \varepsilon$
    Compute $m = \iota(\Theta'_k)$.
    $\Theta''_{k+1} = \Theta'_k \cup N(\Theta'_k) \cup B(\Theta'_k)$.
    $\Theta_{k+1} = O(\Theta''_k, m)$
    $\Theta'_{k+1} = C(\Theta''_k) \cup \Theta_{k+1}$
    $k = k + 1$.
End_while.
Return $\Theta_{k+1}$.
```

In this algorithm, $N(\Theta)$ is the set of frequencies computed using Newton's method. Computing these frequencies is based on the approximation,

$$\lambda_1(F(K, \Pi, j\omega')) - \lambda \approx Q_\omega^\top F(K, \Pi, j\omega') Q_\omega$$

for ω' in a neighbourhood of ω. More precisely, we define $\Lambda_\omega := (j\omega I - \mathcal{A}(K))^{-1}$ and

$$\zeta_{1,\omega} \;:=\; \Psi_{22}^{-1} \cdot (\Psi_{12}^\top \Lambda_\omega \mathcal{B}(K) + \mathcal{B}^\top(K) \Lambda_\omega^\top \Psi_{11} \Lambda \mathcal{B}(K) + \mathcal{B}^\top(K) \Lambda_\omega \Psi_{12}) Q_\omega, \tag{33}$$

$$\zeta_{2,\omega} \;:=\; \mathrm{Re}\,(Q_\omega^\top \Psi_{22}^{-1}(-\Psi_{12}^\top j \Lambda_\omega^2 \mathcal{B}(K) + \mathcal{B}^\top(K)(-\Lambda_\omega^{2T} j \Psi_{11} \Lambda_\omega \mathcal{B}(K) - \Lambda_\omega^{2T} j \Psi_{12})) Q_\omega). \tag{34}$$

Finally, set $N(\Theta)$ is

$$N(\Theta) = \left\{\omega \in \mathbb{R}^+ \;:\; \omega = \omega_k + \zeta_{2,\omega_k}^{-1}(1 + \mathrm{Re}\,(Q_{\omega_k}^\top \zeta_{1,\omega_k})), k = 1, \ldots, \#\Theta_\lambda\right\}.$$

8.2.2 Choice of $\Omega_e(K, \Pi)$

The choice of the extended set of frequencies $\Omega_e(K, \Pi)$ is of practical importance. On the one hand a bad choice with too few frequencies leads to a bad descent directions, which may stall the optimization process. Namely, if the number of frequency added in $\Omega_e(K, \Pi)$ is too small, then the descent direction is close to the direction of steepest-descent, which leads to zigzagging and produces very small steps. On the other hand, the number of added frequencies must not be too large either, because this may introduce numerical problems in the tangent program (31).

We have tested different rules to add frequencies to the active set $\Omega(K, \Pi)$. Numerical experiments show that there is no ideal choice, and for some problems we should switch from one rule to another. Our first strategy is to choose more frequencies around the peaks. Here we let

$$\Omega_e(K,\Pi) = \Omega(K,\Pi) \cup \left(\bigcup_{i=1}^{k} \Omega_{\lambda_i}(K,\Pi) \right)$$

where

$$\begin{aligned} \Omega_{\lambda_i}(K,\Pi) &:= \left\{ \omega \in \mathbb{R}^+ : \lambda_1(F(K,\Pi,j\omega)) = \lambda_i \right\}, \\ \lambda_{i+1} &:= \frac{\lambda_i + f(K,\Pi)}{2},\ \lambda_1 = \lambda \in \mathbb{R}^+. \end{aligned}$$

With this rule, we add a small number of frequencies around the peaks. When the peaks arise on a plateau, this approach will not add enough frequencies. We therefore use a second rule, based on a logarithmic discretization of the peaks. We choose a discrete subset D of

$$\{\log\omega : \omega \in \mathbb{R}^+ \ : \ \lambda_1(F(K,\Pi,j\omega)) \geq \lambda, \omega \in \Omega\},$$

in such a way that n_p equi-spaced points are chosen for each peak, then the extended set is defined as

$$\Omega_e(K,\Pi) = \Omega(K,\Pi) \cup \Omega_\lambda \cup \exp(D).$$

For the implementation we have used the value $k = 3$ for the first rule, $n_p = 20$ for the second rule. The value of λ has been set in both case to $\lambda = f(K,\Pi)(1-c) + bc$ with

$$b = \min\left\{ \lambda_1(\Psi_{22}[0]), \lambda_1\left(\begin{bmatrix} \mathcal{A}(K)^{-1}\mathcal{B}(K) \\ I \end{bmatrix}^{\mathsf{H}} \Psi[0] \begin{bmatrix} \mathcal{A}(K)^{-1}\mathcal{B}(K) \\ I \end{bmatrix} \right) \right\},$$

and $c = 0.05$.

8.3 Performance optimization

Minimizing the objective function f allows to reach a feasible point (K_k, S_k, G_k) for a fixed value of the performance upper bound γ_k. However, our goal is not only to find a robust controller K, we also have to optimize worst case performance $\gamma \geq \max_{\Delta\in\mathbf{\Delta}} \|T_{wz}(\Delta,K)\|_\infty$. An extrapolation method has been used to update the parameter $\gamma_k \to \gamma_{k+1}$ in order to drive it toward its optimal value γ^*. Let (K_k, S_k, G_k) with γ_k be the current feasible point reached by the algorithm, i.e. $f(K_k, S_k, G_k, \gamma_k) < 0$, so that K_k is a parametric robust controller for $\mathbf{\Delta}$ with robust performance $\leq \gamma_k$ and with the corresponding certificate $\Pi_{\Delta,k} = (S_k, G_k)$. The cost function can be written as

$$f(K_k, S_k, G_k, \gamma_k) = \max_{\omega \in [0,\infty]} \lambda_1(F(K_k, S_k, G_k, \gamma_k, j\omega)) = \max_{\omega \in [0,\infty]} Q_\omega^\top F(K_k, S_k, G_k, \gamma_k, j\omega) Q_\omega,$$

so that

$$f(K_k, \Pi_k) = \max_{\omega \in [0,\infty]} \frac{c_{1,\omega}}{\gamma} + \gamma c_{2,\omega} + r_\omega = \max_{\omega \in \Omega(K_k,\Pi_k)} \frac{c_{1,\omega}}{\gamma} + \gamma c_{2,\omega} + r_\omega,$$

where

$$c_{1,\omega} = Q_\omega^\top \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & I & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right] Q_\omega, \quad c_{2,\omega} = Q_\omega^\top \left[\begin{array}{cc|cc} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -I \end{array}\right] Q_\omega,$$

and

$$r_\omega = \lambda_1(F(K_k, \Pi_k, j\omega)) - \frac{c_{1,\omega}}{\gamma} + \gamma c_{2,\omega}.$$

Let $\varepsilon > 0$ be a fixed tolerance. We compute for each of the finitely many $\omega \in \Omega(K_k, \Pi_k)$ the number γ_ω^+ by:

- if $r_\omega < \varepsilon(|c_{1,\omega}| + |c_{2,\omega}|)$ then $\gamma_\omega^+ = \sqrt{-c_{1,\omega}/c_{2,\omega}}$,
- elseif $c_{2,\omega} < \varepsilon|r_\omega|$ then $\gamma_\omega^+ = c_{1,\omega}/r_\omega$,
- elseif $c_{1,\omega} < \varepsilon|r_\omega|$ then $\gamma_\omega^+ = -r_\omega/c_{2,\omega}$,
- else
$$\gamma_\omega^+ = \frac{-r_\omega - \sqrt{r_\omega^2 - 4c_{1,\omega}c_{2,\omega}}}{2c_{2,\omega}}.$$

Then parameter γ_k is updated to the value $\gamma_{k+1} = \max\{\gamma_\omega^+ \ : \ \omega \in \Omega(K_k, \Pi_k)\}$, and optimization is restarted with this new value. In some cases, the point (K_k, Π_k) is not feasible for the new γ_{k+1}. In that case we perform a backtracking linesearch to find the smallest value γ_{k+1}, for which the actual point is feasible. Optimization of γ_k is stopped if $\gamma_{k+1} \geq \gamma_k$, i.e., if we fail to reduce γ_k within the numerical precision. Yet another possible stopping criterion is

$$\frac{|\gamma_k - \gamma_{k+1}|}{\gamma_k} < \varepsilon,$$

where we have taken value $\varepsilon = 1e - 4$ in our code.

8.4 Results

For both case studies our robust synthesis algorithm has been run for different levels of uncertainty. As always, in phase I, the locally optimal H_∞ controller K_∞ is computed to have a lower bound γ_∞ for γ^*, and to initialize $\gamma_0 > \gamma_\infty$. In phase II our algorithm comes to play and K_∞ is used as a starting point K_0 to compute a locally optimal robust controller K^* using a static multiplier Π^* on $[0, \infty]$ and with robust performance at most γ^*. Then in phase III we use (K^*, Π^*) to initialize our algorithm again to compute a robust controller $\bar{K}$ and a dynamic multiplier $\bar{\Pi}(j\omega)$ in order to improve performance $\bar{\gamma} < \gamma^*$. In phase III our tests use dynamic multipliers with three frequency bands as described before. Results are presented in tables 2 and 3. For each value of p for the mass spring model, and of p_1, p_2 for the missile model, we have computed: (a) the worst case performance γ^* of the feedback controller with respect to the $p\%$ value, (b) the H_∞ performance of the nominal plant $\gamma^*_{\text{nom}} = \|T_{wz}(0, K^*)\|_\infty$ at the robust controller K^*, (c) the value of the spectral abscissa α_{nom} of the nominal plant at K^*, and (d) the criticality measure θ, which is Polak's optimality function value [40].

We observe in both studies that the performance bound γ^* increases as the level of uncertainty increases. We see in Table 2 that for small $p\%$, the value γ^* is very near the optimal H_∞ controller performance γ_∞. The same is seen in Table 3 for the missile model. Improvement of the performance bound of the controller is also observed when passing from the static case to the dynamic 3 bands case, i.e., $\gamma^* \to \bar{\gamma}$. Improvement from γ^* to $\bar{\gamma}$ is sometime dramatic, for example when $p = 25\%$ in the mass spring model. For the missile model this improvement is clear for all experiments. A gap is observed for parameter γ^*, for the missile model, when passing from $p_1 = 0.15, p_2 = 15$ to $p_1 = 0.20, p_2 = 20$.

We see in Tables 2 and 3 that for some experiments, the criticality value is rather large. For example the mass-spring model for $p = 20\%$ with 3 bands. The fact that optimization stops while criticality is not fully achieved can be explained by a bad detection of the actives frequencies $\Omega(K, \Pi)$, or by a bad choice of the extended frequency set. In Figure 3 we display the plot of the FDI for the last iteration of the mass-spring system 3 band controller synthesis with $p = 20\%$. We see on this figure that there is a very sharp peak on the right, and we have observed that the algorithm fails to compute the active frequency of this peak with sufficient accuracy. This causes the optimization process to stop prematurely.

A more diligent choice of the frequencies in the extended set $\Omega_e(K, \Pi)$, and restarting the optimization process, may overcome this difficulty. In the same vein, peak value precision may be increased by decreasing the tolerance parameter ε_f. For the example in question, optimization could be restarted for some iterations, by adding frequencies

Mass Spring - 1 Band					**Mass Spring - 3 Band**					
p%	γ^*	γ^*_{nom}	α^*_{nom}	θ	$\bar{\gamma}$	$\bar{\gamma}_{\text{nom}}$	$\bar{\alpha}_{\text{nom}}$	θ	ω_1	ω_2
5	1.3610	1.3551	-0.6172	-2.16e-04	1.3594	1.3553	-0.6206	-2.44e-03	0.76	1
10	1.6963	1.6807	-0.7869	-9.38e-04	1.6561	1.6270	-0.4136	-1.44e-04	0.76	2.30
15	2.0648	2.0106	-0.5615	-2.67e-03	1.6169	1.5168	-0.6114	-2.11e-01	2.40	4.10
20	2.5079	2.4064	-0.4501	-4.64e-03	1.7729	1.6544	-0.6835	-1.30e-01	2.40	4.10
25	3.0609	2.8979	-0.3646	-6.24e-03	1.9626	1.7751	-0.8070	-5.39e-02	2.40	4.10
30	3.7540	3.4569	-0.3146	-3.58e-05	3.6205	3.5405	-0.2272	-2.80e-04	0.76	45

Table 2: Results for the mass-spring model. Optimal full order $\gamma_\infty = 1.3261$ and $\alpha_\infty = -0.2129$

	Reichert's Missile - 1 Band				**Reichert's Missile - 3 Band**					
p_1%; p_2%	γ^*	γ^*_{nom}	α^*_{nom}	θ	$\bar{\gamma}$	$\bar{\gamma}_{\text{nom}}$	$\bar{\alpha}_{\text{nom}}$	θ	ω_1	ω_2
0.05; 5	2.9093	2.8257	-5.6525	-2.37e-02	2.6238	2.5755	-1.9448	-3.57e-04	36	250
0.1; 10	3.6214	3.1553	-9.8540	-1.81e-01	2.7443	2.6436	-2.3008	-3.91e-04	36	250
0.15; 15	3.7101	3.2804	-9.6331	-3.52e-02	2.8386	2.7301	-2.4953	-4.10e-04	36	250
0.2; 20	10.478	4.4420	-16.8509	-1.28e-02	9.7196	4.3821	-17.1033	-3.50e-02	69	400

Table 3: Results for the missile model. Optimal full order $\gamma_\infty = 2.4941$ and $\alpha_\infty = -0.4949$

by hand and decreasing ε_f value, until the algorithm stopped again prematurely due to bad detection of the sharp peak. An efficient rule for adding frequencies to set $\Omega_e(K, \Pi)$, when optimization gets stalled, remains to be defined.

Figure 2 shows the plot of the spectral abscissa, for the mass-spring model with $p = 25\%$, for the controllers obtained with static (left of figure) and dynamic (right of figure) multipliers. The spectral abscissa is computed for each value of perturbations δm_2 and δk of parameter m_2 and k. The left part of Figure 2 is an illustration of IQC conservatism. In the case of static Π, the boundary of the unstable region is very far from the square of admissible perturbations. The right part of Figure 2 shows that the boundary of unstable region gets closer to the square for the controller computed with a dynamic multiplier $\Pi(j\omega)$. Hence conservatism has been reduced.

9 Conclusion

We have proposed a new algorithm to minimize the semi-infinite maximum eigenvalue function $f = \lambda_{1,\infty} \circ F$ arising in IQC synthesis problems. Our method is nonsmooth

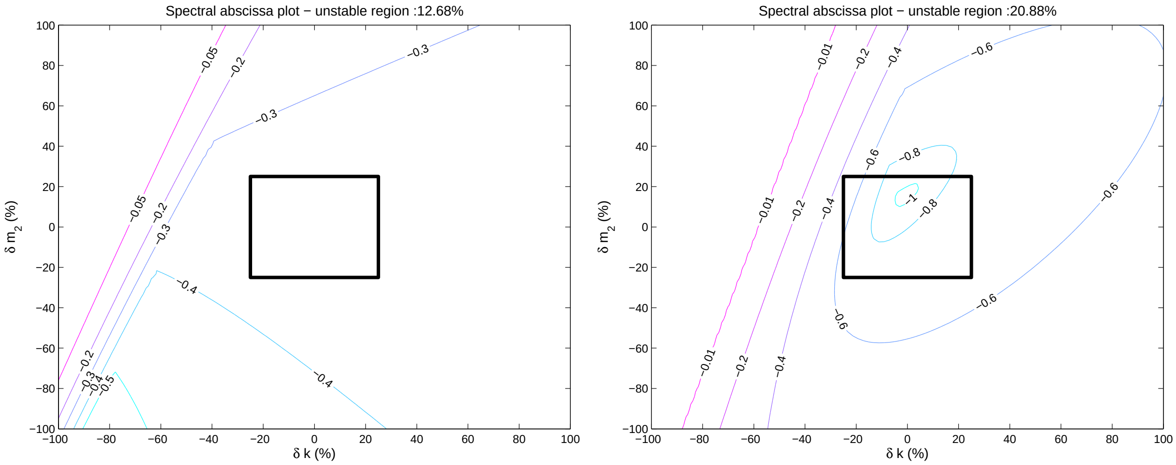

Figure 2: Reduction of conservatism for dynamic $\Pi(s)$. Contour plot of the perturbed closed-loop mass-spring system spectral abscissa $\Delta \mapsto \alpha\left(\mathcal{A}(K^*, \Delta)\right)$ (left) and $\Delta \mapsto \alpha\left(\mathcal{A}(\bar{K}, \Delta)\right)$ (right). The black square shows the $\pm p = 25\%$ uncertainty region about the nominal parameter values. Left: with the controller K^* computed using a static multiplier Π^*. Right: with the controller $\bar{K}$ synthesized using a dynamic three-band multiplier $\bar{\Pi}(s)$. The percentage on the top of the two figures measures the area of the unstable region within the frame.

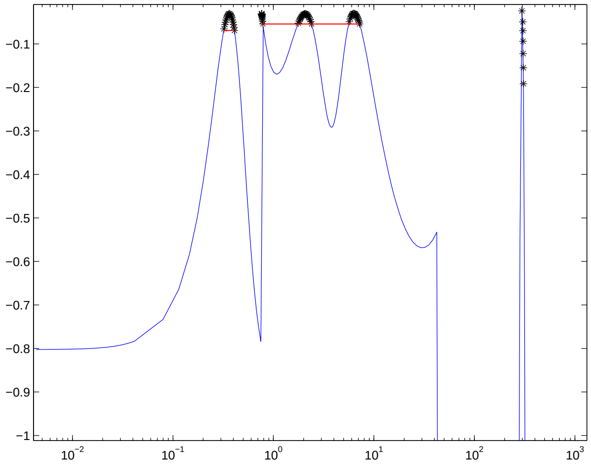

Figure 3: Plot of the FDI, last iteration of controller synthesis for the mass-spring system. We are on the dynamic 3 bands case and $p = 20\%$.

and uses a numerically tractable description of the Clarke generalized subdifferential $\partial f(x)$. A number of implementation details of the method have been discussed. Our design approach performs well on two case studies.

Acknowledgements

The present work was supported by research grants from *Fondation d'entreprise EADS* and from *Agence Nationale de Recherche* (ANR).

References

[1] I.-J. Adounkpé, E. Godoy, and J.-P. Harcaut, *Robustness analysis of a nonlinear missile autopilot : an integral quadratic constraint approach*, ASCC'2004, Melbourne, (2004).

[2] P. Apkarian and D. Noll, *Controller design via nonsmooth multi-directional search*, SIAM J. on Control and Optimization, 44 (2006), pp. 1923–1949.

[3] ——, *Nonsmooth H_∞ synthesis*, IEEE Trans. Aut. Control, 51 (2006), pp. 71–86.

[4] ——, *Nonsmooth optimization for multiband frequency domain control design.* To appear in Automatica, (2007).

[5] ——, *Nonsmooth optimization for multidisk H_∞ synthesis*, European J. of Control, 12 (2006), pp. 229–244.

[6] P. Apkarian, D. Noll, and O. Prot, *Trust region spectral bundle method for nonconvex eigenvalue optimization*, submitted, (2006).

[7] V. Bompart, D. Noll, and P. Apkarian, *Second-order nonsmooth optimization for H_∞ and H_2 syntheses*, submitted, (2005).

[8] S. Boyd and V. Balakrishnan, *A regularity result for the singular values of a transfer matrix and a quadratically convergent algorithm for computing its $\mathbf{L}_\infty$-norm*, Syst. Control Letters, 15 (1990), pp. 1–7.

[9] S. Boyd, V. Balakrishnan, and P. Kabamba, *A bisection method for computing the H_∞ norm of a transfer matrix and related problems*, Mathematics of Control, Signals, and Systems, 2 (1989), pp. 207–219.

[10] K. Chung-Yao, M. Ravuri, and A. Megretski, *Control synthesis with dynamic integral quadratic constraints - LMI approach*, in Proc. IEEE Conf. on Decision and Control, Sydney, NSW, Australia, Dec. 2000, pp. 1477–1482.

[11] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Canadian Math. Soc. Series, John Wiley & Sons, New York, 1983.

[12] J. Cullum, W. Donath, and P. Wolfe, *The minimization of certain nondifferentiable sums of eigenvalues of symmetric matrices*, Math. Programming Stud., 3 (1975), pp. 35 – 55.

[13] R. D'Andrea, *Generalized l_2 synthesis*, IEEE Trans. Aut. Control, 44 (1999), pp. 1145–1156.

[14] J. Doyle, A. Packard, and K. Zhou, *Review of LFT's, LMI's and μ*, in Proc. IEEE Conf. on Decision and Control, vol. 2, Brighton, Dec. 1991, pp. 1227–1232.

[15] B. Fares, P. Apkarian, and D. Noll, *An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory*, in Proc. American Control Conf., Chicago, Illinois, 2000, pp. 3702–3705.

[16] ——, *An Augmented Lagrangian Method for a Class of LMI-Constrained Problems in Robust Control Theory*, Int. J. Control, 74 (2001), pp. 348–360.

[17] B. Fares, D. Noll, and P. Apkarian, *Robust Control via Sequential Semidefinite Programming*, SIAM J. on Control and Optimization, 40 (2002), pp. 1791–1820.

[18] M. Fu and S. Dasgupta, *Robust stability analysis for linear and nonlinear systems with integral quadratic constraints.* submitted to CDC 1996.

[19] A. Hansson and L. Vandenberghe, *Efficient solution of linear matrix inequalities for integral quadratic constraints*, in Proc. IEEE Conf. on Decision and Control, vol. 5, Sydney, NSW, Australia, 2000, pp. 5033–5034.

[20] C. Helmberg and F. Rendl, *Spectral bundle method for semidefinite programming*, SIAM J. Optimization, 10 (2000), pp. 673 – 696.

[21] C. Helmberg, F. Oustry, *Bundle methods to minimize the maximum eigenvalue function.* Handbook of Semidefinite Programming. Theory, Algorithms and Applications. L. Vandenberghe, R. Saigal, H. Wolkowitz (eds.), vol. 27, 2000.

[22] A. Helmersson, *IQC synthesis based on inertia constraints*, in 14th World Congress of IFAC, Beijing, China, 1999, pp. 163–168.

[23] J. W. Helton and O. Merino, *Coordinate optimization for bi-convex matrix inequalities*, in Proc. IEEE Conf. on Decision and Control, San Diego, CA, 1997, pp. 3609–3613.

[24] D. Henrion, M.Kocvara, and M. Stingl, *Solving simultaneous stabilization BMI problems with PENNON*, in IFIP Conference on System Modeling and Optimization, vol. 7, Sophia Antipolis, France, July 2003.

[25] J.-B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms II: Advanced Theory and Bundle Methods*, vol. 306 of Grundlehren der mathematischen Wissenschaften, Springer Verlag, New York, 1993.

[26] U. Jönsson, *Robustness Analysis of Uncertain and Nonlinear Systems*, PhD thesis, Lund Institute of Technology, Lund, Sweden, 1996.

[27] C.-Y. Kao, A. Megretski, and U. Jönsson, *Specialized fast algorithms for IQC, feasibility and optimization problems*, Automatica, 40 (2004), pp. 239–252.

[28] T. Kato, *Perturbation Theory for Linear Operators*, Springer Verlag, New York (second edition), 1984.

[29] J.B. Lasserre, *Global optimization with polynomials and the problem of moments*, SIAM J. Optim. 11, no. 3, 2001, pp. 796 - 817.

[30] C. Lemaréchal and F. Oustry, *Nonsmooth algorithms to solve semidefinite programs, SIAM Advances in Linear Matrix Inequality Methods in Control series, ed. L. El Ghaoui & S.-I. Niculescu*, (2000).

[31] A. G. J. MacFarlane and Y. S. Hung, *Analytic properties of the singular values of a rational matrix*, Int. J. Control, 37 (1983), pp. 221–234.

[32] A. Megretski and A. Rantzer, *System analysis via integral quadratic constraints*, IEEE Trans. Aut. Control, 42 (1997), pp. 819–830.

[33] A. Nemirovskii, *Several NP-Hard problems arising in robust stability analysis*, Mathematics of Control, Signals, and Systems, 6 (1994), pp. 99–105.

[34] D. Noll and P. Apkarian, *Spectral bundle methods for nonconvex maximum eigenvalue functions: first-order methods*, Mathematical Programming Series B, 104 (2005), pp. 701–727.

[35] D. Noll, M. Torki et P. Apkarian. *Partially augmented Lagrangian method for matrix inequalities*, SIAM J. Optimization, vol. 15, no. 1, 2004, pp. 161 – 184.

[36] A. Ohara, K. Matsumoto, and M. Ide, *Robust controller synthesis for plants satisfying a certain classof IQC*, in Proc. American Control Conf., Arlington, VA, June 2001, pp. 3427–3432.

[37] F. Oustry, *A second-order bundle method to minimize the maximum eigenvalue function*, Math. Programming Series A, 89 (2000), pp. 1 – 33.

[38] M. Overton, *On minimizing the maximum eigenvalue of a symmetric matrix*, SIAM J. on Matrix Analysis and Applications, 9 (1988), pp. 256–268.

[39] P. Parrilo, *On the numerical solution of LMIs derived from the KYP lemma*, in Proc. IEEE Conf. on Decision and Control, Phoenix, AZ, 1999, pp. 2334–2338.

[40] E. Polak, *Optimization : Algorithms and Consistent Approximations*, Applied Mathematical Sciences, 1997.

[41] R. M. Redheffer, *On a certain linear fractional transformation*, J. Math. and Phys., 39 (1960), pp. 269–286.

[42] R. T. Reichert, *Robust Autopilot Design Using μ-Synthesis*, in Proc. American Control Conf., may 1990, pp. 2368–2373.

[43] J.-B. Thevenet, D. Noll, and P. Apkarian, *Nonlinear spectral SDP method for BMI-constrained problems: applications to control design*, in Informatics in Control, Automation and Robotics I, J. Braz, H. Araùjo, A. Vieira, and B. Encarnacão, eds., Springer Verlag, 2006, pp. 61–72.

[44] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice-Hall, 1996.