Nonsmooth optimization algorithm for mixed H_2/H_∞ synthesis

Pierre Apkarian, Dominikus Noll, Aude Rondepierre

***Abstract*— The mixed H_2/H_∞ synthesis problem is addressed via nonsmooth mathematical programming. The proposed algorithm is of first order and can handle any controller structure of practical interest. Since computations are carried out in the frequency domain, the method does not suffer dimensional restrictions like LMI or BMI methods. Global convergence is established and several numerical tests are presented.**

***Index Terms*— Mixed H_2/H_∞ synthesis, multi-objective control, structured controllers design, nonsmooth optimization.**

I. Introduction

Mixed H_2/H_∞ output feedback control is a multi-objective design problem, where the feedback controller has to respond favorably to two concurring performance specifications. Typically in H_2/H_∞ synthesis, the H_∞-channel is used to enhance the robustness of the design, whereas the H_2-channel guarantees the performance of the system.

Due to its importance in practice, mixed H_2/H_∞ control has been addressed in various ways. First approaches are based on coupled Riccati equations in tandem with homotopy methods, but the numerical success of these strategies remains to be established. With the rise of LMIs in the later 1990s, different strategies which convexify the problem became increasingly popular. The price to pay for convexifying the problem is either a considerable conservatism, or that controllers have large state dimension [11], [10].

In [15], [16], [17], Scherer develops characterizations for the H_2/H_∞ synthesis problem with full-order or Youla parameterized controllers. The problem is reduced to LMIs involving Lyapunov and controller matrix variables together with multipliers. The drawback of this approach is the presence of Lyapunov variables, which grow quadratically in the system size. The consequence is that current BMI and LMI solvers quickly succumb when plants get sizable.

Following [2], [3], [5], [4], we address H_2/H_∞ synthesis by a new strategy which avoids Lyapunov variables. This leads to a nonsmooth and semi-infinite optimization program.

The paper is organized as follows. The H_2/H_∞ synthesis problem is introduced in section II. In sections III and IV we successively present our method and a nonsmooth algorithm for solving the H_2/H_∞ problem. After detailing some technical elements in section V, we discuss numerical examples to validate our algorithm in the last section.

Pierre Apkarian is with ONERA - 2, avenue Edouard Belin, 31055 Toulouse, France - and Université Paul Sabatier, Toulouse, France

Dominikus Noll and Aude Rondepierre are with Université Paul Sabatier, Institut de Mathématiques, 118, route de Narbonne, 31062 Toulouse, France.

II. Problem settings

We consider a plant in state space form

$$P: \quad \left[\begin{array}{c} \dot{x} \\ \hline z_\infty \\ z_2 \\ y \end{array}\right] = \left[\begin{array}{c|ccc} A & B_\infty & B_2 & B \\ \hline C_\infty & D_\infty & 0 & D_{\infty u} \\ C_2 & 0 & 0 & D_{2u} \\ C & D_{y\infty} & D_{y2} & 0 \end{array}\right] \left[\begin{array}{c} x \\ \hline w_\infty \\ w_2 \\ u \end{array}\right] \qquad (1)$$

where $x \in \mathbb{R}^{n_x}$ is the state, $u \in \mathbb{R}^{n_u}$ the control, $y \in \mathbb{R}^{n_y}$ is the measured output, $w_\infty \to z_\infty$ is the H_∞ channel, $w_2 \to z_2$ the H_2 channel. We seek an output feedback controller

$$K: \quad \begin{bmatrix} \dot{x}_K \\ u \end{bmatrix} = \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix} \begin{bmatrix} x_K \\ y \end{bmatrix} \qquad (2)$$

with state $x_K \in \mathbb{R}^{n_K}$ such that the closed-loop system (1)-(2) satisfies the following properties:

1) **Internal stability.** K stabilizes P exponentially in closed-loop.
2) **Fixed H_∞ performance**. The H_∞ channel has a pre-specified performance level $\|T_{w_\infty \to z_\infty}(K)\|_\infty \leq \gamma$.
3) **Optimal H_2 performance**. The H_2 performance $\|T_{w_2 \to z_2}(K)\|_2$ is minimized among all K satisfying 1. and 2.

We will solve the H_2/H_∞ synthesis problem by way of the following mathematical program

$$\begin{array}{ll} \text{minimize} & f(K) := \|T_{w_2 \to z_2}(K)\|_2^2 \\ \text{subject to} & g(K) := \|T_{w_\infty \to z_\infty}(K)\|_\infty^2 \leq \gamma^2 \end{array} \qquad (3)$$

where $T_{w_2 \to z_2}(K,s)$ denotes the transfer function of the H_2 closed-loop performance channel, while $T_{w_\infty \to z_\infty}(K,s)$ stands for the H_∞ robustness channel.

Notice that $f(K)$ is a smooth function, whereas $g(K)$ is not, being an infinite maximum of maximum eigenvalue functions. The unknown K is in the space $\mathbb{R}^{(n_K+n_u)\times(n_K+n_y)}$, so the dimension $n = (n_K+n_u)(n_K+n_y)$ of (3) is usually small, which is particularly attractive when small or medium size controllers for large systems are sought.

For brevity, we set $T_2 := T_{w_2 \to z_2}$ and $T_\infty := T_{w_\infty \to z_\infty}$ in (1). The performance measures H_2 and H_∞ are defined as:

$$f(K) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} \operatorname{Tr}\left[T_2(K,j\omega)^H T_2(K,j\omega)\right] d\omega$$
$$g(K) = \max_{\omega \in [0,\infty]} g(K,\omega) = \max_{\omega \in [0,\infty]} \lambda_1\left(T_\infty(K,j\omega)^H T_\infty(K,j\omega)\right)$$

where the transfer matrices T_2 and T_∞ are stable and T_2 has to be strictly proper to ensure finiteness of the H_2 norm. For later use we define the set $\mathbb{B}_m = \{Y \in \mathbb{S}_m : Y \succeq 0, \operatorname{Tr}(Y) = 1\}$ and the spectraplex $\mathbb{B}_m^q = \left\{(Y_1,\ldots,Y_q) : Y_i \in \mathbb{S}_m, Y_i \succeq 0, \sum_{i=1}^q \operatorname{Tr}(Y_i) = 1\right\}$ where $\mathbb{S}_m$ denotes the space of $m \times m$ Hermitian matrices.

III. Nonsmooth optimization method

In this section, we present our main result, a nonsmooth optimization method for the mixed program (3).

A. Local model and optimality conditions

Following an idea in [14], we address the mixed program (3) by introducing the progress function: for $(K,\widetilde{K}) \in (\mathbb{R}^n)^2$,

$$F(\widetilde{K};K) = \max\Big\{ f(\widetilde{K}) - f(K) - \mu[g(K)-\gamma^2]_+; \quad g(\widetilde{K}) - \gamma^2 - [g(K)-\gamma^2]_+ \Big\}$$

where $\mu > 0$ is fixed. Its relation with (3) is given by

Lemma 1: If $\hat{K} \in \mathbb{R}^n$ is a local minimizer of (3), then $\hat{K}$ is a local minimizer of $F(.;\hat{K})$ and $0 \in \partial_1 F(\hat{K};\hat{K})$.

Here $\partial_1 F(\widetilde{K},K)$ stands short for $\partial F(\cdot,K)(\widetilde{K})$. Conversely we have the following:

Lemma 2: If $0 \in \partial_1 F(\hat{K};\hat{K})$, then we have two possibilities:

i. Either $g(\hat{K}) > \gamma^2$, then $\hat{K}$ is a critical point of g alone, called a critical point of constraint violation.
ii. Or $g(\hat{K}) \le \gamma^2$, then $\hat{K}$ satisfies the Fritz John necessary optimality conditions for (3). In addition, when $g(\hat{K}) < \gamma^2$ or $0 \notin \partial g(\hat{K})$, then $\hat{K}$ is even a KKT point of (3).

These two lemmas explain why we should search for points $\hat{K}$ satisfying $0 \in \partial_1 F(\hat{K};\hat{K})$. It also indicates that minimizing the progress function F leads to a phase I-phase II algorithm. Namely, as long as the iterates remain unfeasible, the dominant term in F is on the right, and minimizing F reduces constraint violation. When phase I terminates successfully, iterates become and stay feasible. Phase II begins and the objective f is minimized. Notice that failure of phase I may occur when iterates accumulate in the neighborhood of a local minimizer of the constraint g (see statement i. in lemma 2).

B. Optimality function and tangent program

We introduce the set $\Omega(K) = \{\omega \in [0,\infty] : g(K) = g(K,\omega)\}$ of active frequencies, or peaks. It can be shown [8] that $\Omega(K)$ is either finite, or coincides with $[0,\infty]$. Since the latter never occurs in practice, we consider the finite case from now on. Consider a finite extension $\Omega_e(K)$ of $\Omega(K)$, which is built in such a way that it depends continuously on K (see [3] for more details). Procedures based on thresholding and discretization as in [3] guarantee this property. Using $\Omega_e(K)$, we build a first order estimation of the progress function F:

$$\widetilde{F}(K+H;K) = \max\left\{ f'(K)H - \mu[g(K)-\gamma^2]_+; \max_{\omega\in\Omega_e(K)} \max_{\substack{Y_\omega \succeq 0,\\ Tr(Y_\omega)=1}} g(K,\omega) - \gamma^2 - [g(K)-\gamma^2]_+ + \langle \Phi_{Y_\omega}, H\rangle \right\}$$

where Φ_{Y_ω} stands for the subgradients of $g(K,\omega)$ as obtained in [3]. We observe that $\partial_1 \widetilde{F}(K;K) = \partial_1 F(K;K)$. Now for some fixed $\delta > 0$, we introduce the optimality function:

$$\theta_e(K) = \min_{H\in\mathbb{R}^n} \widetilde{F}(K+H;K) + \tfrac{1}{2}\delta\|H\|^2.$$

The concept of optimality functions was introduced by E. Polak [14] for finite and infinite families of smooth functions. Its interest stems from the fact that for any stabilizing K, $\theta_e(K) \le 0$, and that $\theta_e(K) = 0$ implies that K satisfies $0 \in \partial_1 F(K;K)$. As we know from Lemma 2, in all cases of practical interest, this implies that K is a critical point of (3).

Proposition 1 (Dual form of θ_e):

$$\theta_e(K) = -\min_{\substack{\tau_0,\tau_\omega \ge 0\\ \tau_0 + \sum_{\omega\in\Omega_e(K)} \tau_\omega = 1}} \min_{\substack{Y_\omega \succeq 0,\\ Tr(Y_\omega)=1}} \Bigg\{ \tau_0 \mu [g(K)-\gamma^2]_+ + \sum_{\omega\in\Omega_e(K)} \tau_\omega \left([g(K)-\gamma^2]_+ - [g(K,\omega)-\gamma^2]\right) + \frac{1}{2\delta}\|\tau_0 f'(K) + \sum_{\omega\in\Omega_e(K)} \tau_\omega \Phi_{Y_\omega}\|^2 \Bigg\} \quad (4)$$

The solution $H(K)$ attaining $\theta_e(K) = \widetilde{F}(K+H(K);K) + \frac{1}{2}\delta\|H(K)\|^2$ is given by:

$$H(K) \;=\; -\frac{1}{\delta}\left[\tau_0 f'(K) + \sum_{\omega\in\Omega_e(K)} \tau_\omega \Phi_{Y_\omega}\right] \quad (5)$$

where τ_0, $(\tau_\omega)_{\omega\in\Omega_e(K)}$, $(Y_\omega)_{\omega\in\Omega_e(K)}$ are solution to (4).

Proof: Let $\Sigma_e^0(K)$ be the set of $\tau \in [0,1]^{\text{card}\,\Omega_e(K)+1}$ such that $\tau_0 + \sum_{\omega\in\Omega_e(K)} \tau_\omega = 1$. Expanding the supremum $\widetilde{F}$ and replacing the first outer and first inner maxima by a maximum over the convex hull with $\tau \in \Sigma_e^0(K)$ as convex coordinates, the optimality function θ_e could be rewritten as

$$\theta_e(K) = \min_{H\in\mathbb{R}^n} \max_{\tau\in\Sigma_e^0(K)} \max_{Y_\omega\in\mathbb{B}} \Big\{ \tau_0 f'(K).H - \tau_0\mu[g(K)-\gamma^2]_+ + \sum_{\omega\in\Omega_e(K)} \tau_\omega \left[g(K,\omega) - \gamma^2 - [g(K)-\gamma^2]_+ + \langle \Phi_{Y_\omega}, H\rangle\right] + \tfrac{1}{2}\delta\|H\|^2 \Big\}$$

We now use Fenchel duality to swap the outer minimum and the inner double maximum (see for example [14, corollary 5.5.6]) to obtain the following dual expression:

$$\theta_e(K) = \max_{\tau\in\Sigma_e(K)} \max_{Y_\omega\in\mathbb{B}} \Bigg\{ -\tau_0\mu[g(K)-\gamma^2]_+ + \sum_{\omega\in\Omega_e(K)} \tau_\omega \left[g(K,\omega) - \gamma^2 - [g(K)-\gamma^2]_+\right] + \min_{H\in\mathbb{R}^n} \left[\tau_0 f'(K).H + \sum_{\omega\in\Omega_e(K)} \tau_\omega \langle \Phi_{Y_\omega}, H\rangle + \tfrac{1}{2}\delta\|H\|^2\right] \Bigg\}$$

The inner minimum is now unconstrained and attained at $H(K)$ given by (5). Substituting (5) back into $\theta_e(K)$ yields the expected dual program (4).

■

Useful properties of the optimality function exploited in algorithmic constructions are as follows:

Proposition 2: For all stabilizing $K \in \mathbb{R}^{(n_K+n_u)\times(n_K+n_y)}$,

i. $\theta_e(K) \le 0$.
ii. $d_1 F(K;K;H(K)) \le \theta_e(K) - \frac{1}{2}\delta\|H(K)\|^2 \le \theta_e(K)$.

iii. $g(K,\omega)-\gamma^2-[g(K)-\gamma^2]_+ + \langle\Phi_{Y_\omega},H(K)\rangle \leq \theta_e(K)-\frac{1}{2}\delta\|H(K)\|^2 \leq \theta_e(K)$ for all $\omega\in\Omega_e(K)$, $Y_\omega\succeq 0, \mathrm{Tr}(Y_\omega)=1$.

iv. $-\mu[g(K)-\gamma^2]_+ + f'(K).H(K) \leq \theta_e(K)-\frac{1}{2}\delta\|H(K)\|^2 \leq \theta_e(K)$.

v. Computing $\theta_e(K)$ via its dual (4) is equivalent to a SDP, and reduces to a convex QP when max singular values are simple over $\Omega_e(K)$ [1].

We now infer from the dual formula (4) that equality $\theta_e(K)=0$ can only occur when

$$\begin{aligned} \tau_0[g(K)-\gamma^2]_+ &= 0\\ \forall\omega\in\Omega_e(K)\backslash\Omega(K), \tau_\omega &= 0\\ \forall\omega\in\Omega(K), \tau_\omega\left([g(K)-\gamma^2]_+ - [g(K)-\gamma^2]\right) &= 0\\ \tau_0 f'(K) + \sum_{\omega\in\Omega(K)}\tau_\omega\Phi_{Y_\omega} &= 0 \end{aligned}$$

Under these conditions, we distinguish three alternatives:

- If $g(K)<\gamma^2$, then $\tau_\omega=0$ for all $\omega\in\Omega_e(K)$ and the condition $\theta_e(K)=0$ is equivalent to:
$$f'(K)=0$$
which means K is a critical point of f.
- If $g(K)>\gamma^2$, then $\tau_0=0$ and the condition $\theta_e(K)=0$ is equivalent to:
$$\sum_{\omega\in\Omega(K)}\tau_\omega\Phi_{Y_\omega}=0$$
which means: $0\in\partial g(K)$; K is a critical point of g.
- If $g(K)=\gamma^2$, the condition $\theta_e(K)=0$ is equivalent to:
$$\tau_0 f'(K)+\sum_{\omega\in\Omega(K)}\tau_\omega\Phi_{Y_\omega}=0$$
which means K is a F. John critical point.

In conclusion, we obtain the following result:

Theorem 1: A stabilizing controller $K\in\mathbb{R}^{(n_K+n_u)\times(n_K+n_y)}$ is a Fritz John critical point of the mixed H_2/H_∞ program (3) if and only if $\theta_e(K)=0$ and $g(K)\leq\gamma^2$. Whenever $\theta_e(K)<0$, the direction $H(K)$ defined by (5) is a qualified descent direction of $F(.;K)$ at K.

Theorem 1 follows from statement ii. in proposition 2. Moreover, assertions iii. and iv. allow us to say that

- if $g(K)<\gamma^2$, $H(K)$ is a descent direction of f at K.
- if $g(K)>\gamma^2$, $H(K)$ is a descent direction of g at K.
- if $g(K)=\gamma^2$, $H(K)$ is a descent direction of both f and g at K.

These observations lead us to develop a nonsmooth descent algorithm for solving the mixed program (3).

IV. Nonsmooth descent algorithm

In this section, we first propose a nonsmooth algorithm for the synthesis of locally optimal controllers for the mixed H_2/H_∞ program and then establish its global convergence.

The algorithm below is a first order descent method applied step by step to the progress function F. The principle is as follows: at each iteration of our algorithm, we compute a descent direction of the progress function $H\mapsto F(K_j+H;K_j)$ around the current iterate K_j. According to theorem 1, we therefore solve the tangent program:

$$\min_{H\in\mathbb{R}^n}\widetilde{F}(K_j+H;K_j)+\frac{\delta}{2}\|H\|^2. \tag{6}$$

whose solution $H(K_j)$ is (5) and is a qualified descent direction for $F(\cdot;K_j)$ at K_j. Performing a backtracking line search, we compute a step s such that $K_j+sH(K_j)$ remains stabilizing and satisfies:

$$F\left(K_j+sH(K_j);K_j\right)\leq s\alpha\theta(K_j) \tag{7}$$

where $\alpha\in(0,1)$ is the minimum fraction required of the directional derivative along H_j at K_j. The algorithm stops as soon as the optimality condition $0\in\partial_1F(K_j;K_j)$ is satisfied.

Algorithm 1 Nonsmooth algorithm for H_2/H_∞ synthesis

Require: γ the performance level, n_K the controller order, $\mu>0$, $\delta>0$ and $\alpha\in(0,0.25]$.

1: **Initialization**. Find initial closed loop stabilizing controller K_0. Put main loop counter to $j=0$.
2: **while** K_j does not satisfy the optimality condition **do**
3: **Frequency generation**. Construct finite extension $\Omega_e(K_j)$ of the set of active frequencies $\Omega(K_j)$ at K_j.
4: **Tangent program**. Solve tangent program:
$$\min_{H\in\mathbb{R}^n}\widetilde{F}(K_j+H;K_j)+\frac{\delta}{2}\|H\|^2.$$
Solution is $H_j=H(K_j)$. Compute $\theta_j=\theta_e(K_j)$.
5: **Line search**. Backtrack to compute a step s such that:
$$F(K_j+sH_j;K_j)\leq s\alpha\theta_j$$
and K_j+sH_j remains stabilizing.
6: **Update**. $K_{j+1}:=K_j+sH_j$; $j:=j+1$.
7: **end while**

We now prove global convergence of algorithm 1 in the sense that every accumulation point of a sequence of iterates generated by the algorithm is a critical point of the mixed H_2/H_∞ program. Consider:

(H_1) The set $\{K\in\mathbb{R}^n:\gamma_\infty^2\leq g(K)\leq g(K_0)\}$ is bounded.

(H_2) f is weakly coercive on the level set $\{K\in\mathbb{R}^n:g(K)\leq\gamma_\infty^2\}$ in the following sense: if K_j is a sequence of feasible iterates with $\limsup_{j\to\infty}\|K_j\|=\infty$, then $f(K_j)$ is not monotonically decreasing.

Under these assumptions, any sequence of steps generated by our algorithm is bounded (see [6] for details) and we are now ready to show the convergence of our algorithm:

Theorem 2: Assume (H_1), (H_2) at K_0, and let K_j the sequence generated by algorithm 1. Then every accumulation point $\hat{K}$ of K_j is either a F. John critical point of the mixed H_2/H_∞ problem, or a critical point of the constraint violation.

Proof: We have to show that $0\in\partial_1F(\hat{K};\hat{K})$. There are two cases to be discussed. Either K_j are feasible from some index onwards, or K_j remain unfeasible all the time. Let us discuss the first case. Assume contrary to the statement that $\theta_e(\hat{K})<0$. Then $H(\hat{K})$ gives qualified descent at $\hat{K}$

in the sense that $F(\hat{K}+tH(\hat{K});\hat{K}) \leq \alpha t \theta_e(\hat{K})$ for all $0 < t \leq t(\hat{K})$, where $t(\hat{K})$ is the largest step such that every $t \in (0, t(\hat{K})]$ satisfies the Armijo condition. Now observe that a practical backtracking line search does not compute $t(K)$, but some $t^\sharp(K) \in (0, t(K)]$. For instance Polak [14] advocates $t^\sharp(K) = \max\{\beta^\nu : \nu \in \mathbb{N}, F(K+\beta^\nu H(K);K) \leq \alpha \beta^\nu \theta_e(K)\}$ with some fixed $0 < \beta < 1$. Then $K_{j+1} = K_j + t^\sharp(K_j)H(K_j)$.

Now recall that the $\Omega_e(K)$ depend continuously on K, hence $\theta_e(K)$ and $H(K)$ also depend continuously on K. Suppose $t^\sharp(K_j) \to t^\sharp$, then $t^\sharp \in \{\beta t^\sharp(\hat{K}), t^\sharp(\hat{K})\}$ and $t^\sharp \leq t(\hat{K})$. Since $K_j \to \hat{K}$ for a subsequence, we have $t^\sharp(K_j)H(K_j) \to t^\sharp H(\hat{K})$, hence $F(K_j + t^\sharp(K_j)H(K_j);K_j) \leq \frac{1}{2}\alpha t^\sharp \theta_e(\hat{K}) \leq \frac{1}{2}\alpha\beta t^\sharp(\hat{K})\theta_e(\hat{K}) \leq \frac{1}{2}\alpha\beta^2 t(\hat{K})\theta_e(\hat{K}) < 0$ for $j \geq j_0$. This contradicts the fact that $F(K_{j+1};K_j) \to 0$ and settles the first case. The proof of the second case is similar. ■

V. Some practical aspects

Algorithm 1 has been implemented for both structured and unstructured H_2/H_∞ synthesis. In practice it is often required that some controller gains be put to zero, while others can be freely assigned. This is e.g. the case when the controller has to be strictly proper to ensure finiteness of the H_2 norm.

A. Stopping criteria

Since our algorithm is a first order method, it may be slow in the neighborhood of a local solution of (3). As in [3], we have therefore implemented termination criteria which ensure that unnecessary iteration with marginal progress near the local optimum can be avoided.

Our first stopping test is based on $0 \in \partial_1 F(K;K)$ and checks whether the algorithm has reached a critical point of (3) by computing

$$\inf\{||\Phi||; \Phi \in \partial_1 F(K;K)\} < \varepsilon_1.$$

We also define two additional tests that compare the relative progress of the local model around the current iterate and the step length to the controller gains:

$$|F(K^+;K)| \leq \varepsilon_2 \qquad ||K^+ - K|| \leq \varepsilon_3(1+||K||).$$

For stopping, either the first or the last two tests are required.

B. Performance level

For all test examples, we compute the locally optimal H_2 controller K_2^* for channel T_2, the locally optimal H_∞ controllers K_∞^* for channel T_∞ and then: $\gamma_2 := ||T_\infty(K_2^*)||_\infty$ and $\gamma_\infty^* = \|T_\infty(K_\infty^*)\|_\infty$. It is now trivial (see e.g. [7]) that the performance level γ in (3) has to satisfy

$$\gamma_\infty^* \leq \gamma < \gamma_2. \tag{8}$$

Indeed the H_2/H_∞ problem is unfeasible for $\gamma < \gamma_\infty^*$, while for $\gamma \geq \gamma_2$, the optimal H_2 controller K_2^* is also optimal for (3).

Disregarding complications due to (multiple) local minima, it would make sense in a specific case study, to consider the entire one parameter family $K(\gamma)$ of solutions of (3) as a function of the gain value γ in the range (8), as this transforms K_∞^* continuously into K_2^* (see Fig. 1).

In our tests we only compute $K(\gamma)$ for a fixed value γ in order to compare our method to existing approaches.

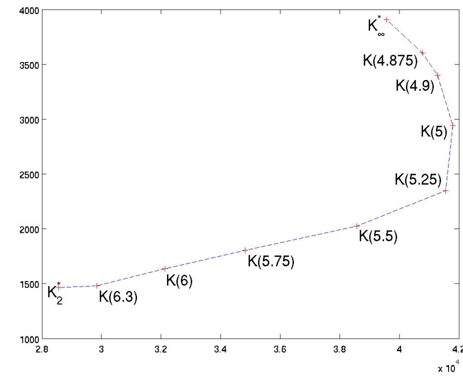

Fig. 1. H_2/H_∞ optimal static controllers $K(\gamma) = (K_1(\gamma), K_2(\gamma)) \in \mathbb{R}^2$ for the vehicular suspension control problem (see VI-B); $\gamma \in [\gamma_\infty^*, \gamma_2] \mapsto K(\gamma)$ continuously transforms the H_∞ optimal gain K_∞^* into the H_2 optimal gain K_2^*

VI. Numerical tests

In this section we present numerical tests of algorithm 1 on a variety of H_2/H_∞ synthesis problems. In all tests, we use the techniques in [8] to compute an initial stabilizing K^0, which is not necessarily feasible for (3). This allows to test phase I of the method. In some cases K_∞ might be chosen as a feasible initial iterate, so that phase I can be avoided. We choose $\gamma \in [\gamma_\infty^*, \gamma_2)$, see Tab. I.

Problem	(n_x, n_y, n_u)	n_K	α_2^* / γ_2	γ_∞^*
Academic ex [7]	$(2,1,1)$	0	$6\frac{1}{4} / \frac{3}{\sqrt{5}}$	1
Academic ex [18]	$(3,1,1)$	3	7.748 / 23.586	9.5237
Vehicular [19] suspension pb	$(4,2,1)$	0	32.416 / 6.3287	4.8602
		2	32.299 / 6.1828	4.8573
		4	32.267 / 6.3260	4.6797
From *COMPl_eib* 'BDT2'	$(82,4,4)$	0	0.79389 / 1.3167	0.67421
		10	0.78877 / 1.1386	0.72423
		41	0.77867 / 1.1302	0.77405
'HF1'	$(130,1,2)$	0	5.8193e-2 / 0.4611	0.44721
		10	5.8198e-2 / 0.4600	0.44721
		25	5.8174e-2 / 0.4605	0.44721
'CM4'	$(240,1,2)$	0	9.2645e-1 / 1.6546	0.81650
		50	9.3844e-1 / 4.2541	0.81746

TABLE I

Results of non-constrained H_2 and H_∞ synthesis where $\alpha_2^* = ||T_2(K_2^*)||_2$, $\gamma_2 = ||T_\infty(K_2^*)||_\infty$ and $\gamma_\infty^* = ||T_\infty(K_\infty^*)||_\infty$.

Next, the parameter δ is arbitrarily choosen as 0.1. Inspired from trust region techniques [6], a way to improve the approximation of the progress function $F(\cdot;K)$ by the model $\widetilde{F}(\cdot;K) + \frac{\delta}{2}\|\cdot - K\|^2$, would be to evaluate the progress of the descent algorithm at each iteration and then to readjust δ.

A. Two academic examples

We start with two academic examples whose models are given in [7] and [18, example 1]. The first one is simple enough to allow explicit computation of static output feedback controllers for H_2, H_∞ and H_2/H_∞ synthesis.

For the purpose of testing, we first apply our algorithm for a performance level $\gamma > \gamma_2 = \|T_\infty(K_2^*)\|_\infty$, so that it finds the optimal H_2 controller K_2^*. See Table I.

Problem	Academic ex. [7]		
(n_x, n_y, n_u)	$(2,1,1)$		
μ	10		
n_K	0	0	1
γ	2	1.2	1.2
Iter	10	11	21
H_2 norm	1.5651	1.5735	1.5394
H_∞ norm	1.3416	1.2	1.2
Final K	$[\,-0.8165\,]$	$[\,-0.9458\,]$	K_{f_1}
(LMI) H_2 norm	-	1.5778	-
(Th.) H_2 norm	-	1.5735	-
Problem	**Academic ex. [18]**		
(n_x, n_y, n_u)	$(3,1,1)$		
μ	1		
n_K	3		
γ	23.6	12	
Iter	83	150	
H_2 norm	7.7484	10.4552	
H_∞ norm	23.5675	12.0000	
Final K	K_{f_2}	K_{f_3}	
(LMI) H_2 norm	8.07	-	
(Th.) H_2 norm	7.748	-	

TABLE II

MIXED H_2/H_∞ SYNTHESIS FOR TWO ACADEMIC EXAMPLES

$$K_{f_1} = \begin{bmatrix} -1.4286 & -0.8013 \\ 0.8194 & -0.5003 \end{bmatrix}$$

$$K_{f_2} = \begin{bmatrix} -2.5016 & 2.4625 & -0.4895 & -1.6314 \\ -1.9540 & -1.1773 & 0.8868 & 1.9661 \\ 0.1410 & -2.9072 & -4.0565 & 0.9222 \\ 2.7108 & -0.6513 & 0.0935 & 0 \end{bmatrix}$$

$$K_{f_3} = \begin{bmatrix} -3.6188 & -2.3240 & 2.0119 & -1.3994 \\ 3.4490 & -2.8053 & 1.0488 & 2.1526 \\ 0.7189 & 1.8672 & -3.7921 & -3.0990 \\ 2.0573 & -3.5362 & 0.3347 & 0 \end{bmatrix}$$

We then perform the H_2/H_∞ synthesis on the two considered examples (see Table II). We not only improve the results computed by LMI approaches in [7] and [18], but also obtain the theoretically best values of the H_2 and H_∞ norms.

B. Vehicular suspension controller design

The model of the vehicular suspension is described in [9] and [19]. We first focus on static H_2/H_∞-synthesis. The H_∞ performance level in (3) is chosen as $\gamma = 5.225$ and the optimal solution we obtain is

$$K^* = \begin{bmatrix} 4.1586 & 0.2393 \end{bmatrix}$$

The H_2 norm computed by our algorithm is $||T_2(K^*)||_2 = 34.446$ instead of 35.8065 obtained by [19] and the related H_∞ performance is $||T_\infty(K^*)||_\infty = 5.2250$ instead of 5.0506 in [19]. This highlights the conservatism of the LMI approach in [19]. In contrast our algorithm attains the H_∞ performance constraint, as it should. Results are given in Table III.

We also present numerical results of the H_2/H_∞ synthesis for dynamic order controllers of orders $n_K = 2, 4$.

Problem	Vehicular suspension controller design [19]		
(n_x, n_y, n_u)	$(4,2,1)$		
n_K	0	2	4
μ	10^2	10^2	10^2
γ	5.225	5.225	5.225
Iter	264	300 (max.)	157
H_2 norm	34.446	33.318	33.313
H_∞ norm	5.2250	5.2232	5.5953
K_{final}	$[\,41599 \;\; 2393\,]$	K_{f_1}	K_{f_2}

TABLE III

MIXED H_2/H_∞ SYNTHESIS FOR THE VEHICULAR SUSPENSION PROBLEM

$$K_{f_1} = \begin{bmatrix} 0.0407 & 0.0228 & 0.5902 & 0.0211 \\ -0.5602 & -0.1051 & -0.0227 & -0.0784 \\ 0.1779 & -0.5606 & 0.0880 & 1.7435 \end{bmatrix} e+03$$

$$K_{f_2} = \begin{bmatrix} -1.967 & -0.149 & 0.633 & -0.196 & 0.065 & -3.959 \\ -0.295 & -0.153 & 0.094 & 0.038 & 0.003 & 0.835 \\ 0.341 & 0.090 & -0.118 & 0.030 & -0.009 & 0.433 \\ 0.140 & -0.078 & -0.031 & -0.118 & -0.011 & 0.697 \\ -3.943 & 0.938 & 0.646 & 0.461 & 0.094 & 12.282 \end{bmatrix} e+02$$

C. *COMPl$_e$ib* examples

Models in this section are from the *COMPl$_e$ib* collection [13]: distillation tower 'BDT2', heat flow in a thin rod 'HF1' and cable mass model 'CM4'. These problems are originally H_∞ synthesis problems. As proposed by F. Leibfritz in [12], we have added a H_2 channel by setting $B_2 = B_\infty$ and $D_{y2} = 0$.

In each example, the H_∞ performance constraint is first chosen as $\gamma > \gamma_2$ to obtain an upper bound of the optimal H_2 performance and an approximation of the related H_∞ performance γ_2. Our results are presented in Tab. IV and V.

Problem (n_x, n_y, n_u)	n_K	γ	Iter	H_2 norm	H_∞ norm
'BDT2' $(82,4,4)$	0	10	192	8.0510e-01	9.5010e-01
	10	10	543	7.6480e-01	1.1438
	0	0.8	115	8.1892e-01	7.9994e-01
	10	0.8	300	7.7021e-01	7.9976e-01
	41	0.8	300	8.4477e-01	7.9998e-01
'HF1' $(130,1,2)$	0	10	7	5.8193e-02	4.6087e-01
	0	0.45	13	5.8808e-02	4.4972e-01
	25	0.45	25	5.8700e-02	4.4993e-01
'CM4' $(240,1,2)$	0	10	5	9.2645e-01	1.6555
	0	1	19	9.8436e-01	1
	50	1	49	9.4216e-01	9.9977e-01

TABLE IV

MIXED H_2/H_∞ SYNTHESIS FOR TEST EXAMPLES FROM *COMPl$_e$ib*

As an illustration, Figs. 2 and 3 show the evolution of the H_2 and H_∞ norms for 'BDT2' example during first iterations. In Fig. 2, phases I and II clearly appear: while the current iterate is unfeasible, descent steps to minimize constraint violation are generated. When the H_∞ constraint is met, the technique privileges minimization of the H_2 objective. Fig. 3 shows the evolution of the max singular value associated with the H_∞ constraint in the first 5 iterations. Stars indicate frequencies selected to build the extension $\Omega_e(K)$. We observe that max singular values are simple at selected frequencies which seems valid as a rule in most applications.

VII. CONCLUSION

Mixed H_2/H_∞ is a practically important problem for which successful numerical methods are lacking. In response we

Problem	γ	K_{final}
'BDT2'	10	$\begin{bmatrix} -0.7629 & -0.6087 & 12.1193 & 1.5088 \\ -0.4776 & -1.4756 & 7.1627 & 27.1322 \\ -0.5665 & 0.0612 & 10.5079 & 7.5538 \\ -1.0256 & -0.4369 & 19.6955 & 32.3475 \end{bmatrix}$
	.8	$\begin{bmatrix} 0.8398 & -3.8674 & -6.7992 & -6.9955 \\ -1.8563 & 4.8118 & 17.4900 & 12.8540 \\ 0.4544 & -1.4443 & 1.4876 & -1.3709 \\ -1.3602 & 4.0823 & 19.4811 & 10.0646 \end{bmatrix}$
'HF1'	10	$[\,-0.1002 \quad -1.1230\,]$
	.45	$[\,-0.2399 \quad -1.1334\,]$
'CM4'	10	$[\,-0.5448 \quad -1.3322\,]$
	1	$[\,-0.5219 \quad -0.8070\,]$

TABLE V

STATIC H_2/H_∞ OUTPUT FEEDBACK CONTROLLERS FOR EXAMPLES EXTRACTED FROM $COMPL_eib$

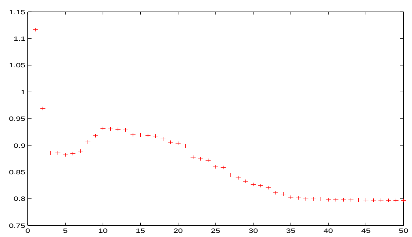

Fig. 2. 'BDT2' example - H_2 norm during the first 50 iterations

have proposed an algorithm based on nonsmooth optimization, which improves systematically over numerical results from the literature, and in particular, over conservative results obtained by LMI techniques. Our approach seems promising since it is capable to handle large size problems with up to 240 states. Extensions to problems involving a mixture of time- and frequency-domain constraints as well as to nonlinear systems are currently under investigation.

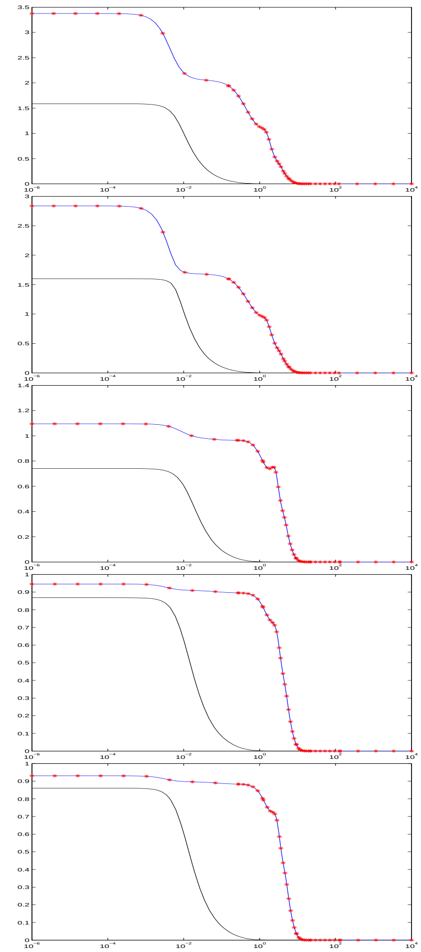

Fig. 3. 'BDT2' example - singular value plot versus frequency over the first 5 iterations. Max singular value is simple

REFERENCES

[1] P. Apkarian, V. Bompart, and D. Noll. Nonsmooth structured control design with applications to PID loop-shaping of a process. *International Journal of Robust and Nonlinear Control*, 14:1320–1342, 2007.

[2] P. Apkarian and D. Noll. Controller design via nonsmooth multidirectional search. *SIAM Journal on Control and Optimization*, 44(6):1923–1949, 2006.

[3] P. Apkarian and D. Noll. Nonsmooth H_∞ control. *IEEE Transaction on Automatic Control*, 51(1):71–86, 2006.

[4] P. Apkarian and D. Noll. Nonsmooth optimization for multiband frequency domain control design. *Automatica*, 2006.

[5] P. Apkarian and D. Noll. Nonsmooth optimization for multidisk H_∞ synthesis. *European Journal of Control*, 12(3):229–244, 2006.

[6] P. Apkarian, D. Noll, and A. Rondepierre. Mixed H_2/H_∞ control via nonsmooth optimization. (Submitted).

[7] D. Arzelier and D. Peaucelle. An iterative method for mixed H_2/H_∞ synthesis via static output-feedback. In *Proceedings of IEEE Conference on Decision and Control*, 2002.

[8] V. Bompart, D. Noll, and P. Apkarian. Second-order nonsmooth optimization for H_∞ synthesis. *Numerische Mathematik*, 107(3):433–454, 2007.

[9] J. Camino, D. Zampieri, and P. Peres. Design of a vehicular suspension controller by static output feedback. In *Proceedings of American Control Conference*, pages 3168–3172, 1999.

[10] J.C. Geromel, P.L.D. Peres, and S.R. Souza. A convex approach to the mixed H_2/H_∞ control problem for discrete time uncertain systems. *SIAM Journal on Control and Optimization*, 33:1816–1833, 1995.

[11] H.A. Hindi, B. Hassibi, and S. Boyd. Multiobjective H_2/H_∞-optimal control via finite dimensional Q-parametrization and linear matrix inequalities. In *Proceedings of American Control Conference*, pages 3244–3248, 1998.

[12] F. Leibfritz. An LMI-based algorithm for designing suboptimal static H_2/H_∞ output feedback controllers. *SIAM Journal on Control and Optimization*, 39(6):1711 – 1735, 2001.

[13] F. Leibfritz. *$COMPl_eib$*, COnstrained Matrix-optimization Problem library - a collection of test examples for nonlinear semidefinite programs, control system design and related problems. Technical report, Universitat Trier, 2003.

[14] E. Polak. *Optimization: Algorithms and Consistent Approximations*, volume 124 of *Applied Mathematical Sciences*. Springer, 1997.

[15] C. Scherer. Multiobjective H_2/H_∞ control. *IEEE Transactions on Automatic Control*, 40:1054 – 1062, 1995.

[16] C. Scherer. Lower bounds in multi-objective H_2/H_∞ problems. In *Proceedings of the 38th IEEE Conference on Decision and Control*, 1999.

[17] C. Scherer. An efficient solution to multi-objective control problems with lmi objectives. *Systems and Control Letters*, 40:43–57, 2000.

[18] C. Scherer, P. Gahinet, and M. Chilali. Multi-objective output-feedback control via LMI optimization. *IEEE Transaction on Automatic Control*, 42:896–911, 1997.

[19] J. Yu. A new static output feedback approach to the suboptimal mixed H_2/H_∞ problem. *International Journal of Robust and Nonlinear Control*, 14:1023–1034, 2004.