Mixed LMI/Randomized Methods for Static Output Feedback Control Design

Denis Arzelier, Elena N. Gryazina, Dimitri Peaucelle and Boris T. Polyak

Abstract— **This paper addresses the problem of stabilization of LTI systems via static output feedback (sof). The objective is not only to compute a stabilizing sof but rather to compute a discrete set of stabilizing sof. Two complementary mixed LMI/randomized algorithms are defined for this purpose. The main idea is to combine a particular relaxed LMI parametrization of stabilizing sof with high efficiency of Hit-and-Run method for generating random points in a given domain. Their respective relevance is analysed on several examples of the COMPl$_e$ib library which is intended to be the reference library for evaluating performance of reduced-order controller synthesis algorithms. Dual of the approach presented in [30], the general methodology proposed here appears to be much more effective. Finally, the paper additionally provides an extensive evaluation of the different relevant instances of the COMPl$_e$ib library.**

I. Introduction

One of the most challenging open problems in control theory is the synthesis of static output-feedback (sof) controllers that meet desired performances and/or robustness specifications [33]. The static or reduced fixed-order dynamic output feedback problem is therefore always an active research area in the control literature. In the recent years, many attempts have been made to give efficient numerical procedures to solve related problems, [1], [21], [13], [11], [14], [15]. In [10], a numerical comparison was performed and classification into three categories (nonlinear programming, parametric optimization and convex programming approaches) was proposed. Even if these three classes may overlap, it gives a clear picture of the situation at that moment. Since then, new developments have been witnessed and the literature has been enriched by numerous contributions on the so-called nonlinear programming approach [22], [9], [8], [2] while pure convex programming methods or parametric optimization methods (which could be merged in a unique class) were scarcely still considered [26], [30]. The main reason mainly relies on the use of new powerful numerical tools based on nonsmooth optimization for the solution of static output feedback stabilization problems with closed-loop performance guarantees [9], [18]. The algorithms based on these techniques may be considered as the most numerically efficient ones at the moment as reported in different dedicated publications [3], [19], [17].

From a designer point of view, a possible limitation of these approaches is that the proposed algorithms rely on rather sophisticated theoretical tools (nonsmooth optimization) and are highly dedicated to the very purpose they have been designed for and without much room for the user to adapt the routine in a simple way. Additionally, if these last methods are rather efficient to find one solution, they seem far less competitive when dealing with the problem of the computation of sets of stabilizing sof. From a practical point of view, the designer is often more interested in the possibility of choosing a controller among many, guaranteing different levels of heterogenous performances in a given set than computing the optimal controller for a single performance which will often leads to poor alternative performances.

The objective of this paper is not to contest the present superiority of existing packages but rather to show that there is still room for alternative useful options on the specific issues mentioned above. In [27], a first step for designing sets of stabilizing static output feedbacks was proposed but the effective construction of samples in these ellipsoidal sets relied on usual deterministic local optimization algorithm. Recently, new randomized algorithms have been designed for the synthesis of feedback controllers in different contexts [16], [28], [12].

Our approach is based on Hit-and-Run (HR) method. It is a version of the Monte-Carlo method to generate points which are approximately uniformly distributed in a given set. The HR algorithm has been proposed by Turchin [34] and independently later by Smith [32]. It is aimed at approximately uniform generation of points in a body via random walks. Its properties have been studied in numerous works by Lovász and co-authors (see, e.g., the survey [7]).

Here, the idea is to propose a mixed LMI/randomized approach to tackle the problem of building sets of static output feedback stabilization. In [26], a new parametrisation of all stabilizing sof gains based on the introduction of additional variables was defined. This parametrization leads to a convex LMI relaxation of the original problem which is dual of the one proposed later in [30]. Revisiting these earlier works, two complementary mixed LMI/randomized algorithms are defined and evaluated on the relevant part of the benchmarks of the COMPl$_e$ib library [20], [24], [25].

The authors gratefully acknowledge the financial contribution of CNRS and RFBR via PICS No. 4281. One of the authors (D.A.) would like to thank Michael Overton and the Courant Institute of Mathematical Sciences of NYU, New York City, New York, USA, for hospitality, where this work was done. He would also like to thank Didier Henrion for helpful comments on early versions of this draft.

D. Arzelier and D. Peaucelle are with LAAS-CNRS ; Université de Toulouse ; 07 avenue Colonel Roche, F-31077 Toulouse, France, arzelier,peaucelle@laas.fr

E.N. Gryazina and B.T. Polyak are with the Institute of Control Science, Russian Academy of Sciences, Moscow, Russia, boris,gryazina@ipu.ru

A. Notations

For conciseness reasons, some abbreviations are used. $sym(A) = A + A'$, $[\star]'BA = A'BA$.

For symmetric matrices, $\succ$ ($\succeq$) is the Löwner partial order ($A \succ (\succeq) B$ if and only if $A - B$ is positive (semi)definite) defined in the cone of positive (semi)definite matrices $\mathbb{S}^{+*}$ ($\mathbb{S}^{+}$). $\mathrm{trace}(A)$ is the sum of diagonal elements of a (square) matrix. $\Lambda(A)$ is the spectrum of the matrix A (the set of all eigenvalues of the matrix A). $\mathbb{R}^{m\times n}$ is the linear space of rectangular $m \times n$ matrices with real entries and equipped with the inner product defined as the usual inner product of the vectors representing the matrices $< A, B >= \mathrm{trace}(A'B)$.

B. Statement of the problem

Let the state-space model of the system be given by its minimal state-space realization:

$$\begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t) \\ y(t) & = & Cx(t) \end{array} \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the control vector and $y \in \mathbb{R}^r$ is the output vector. All matrices are assumed to be of appropriate dimensions and it is assumed throughout the paper that $\mathrm{rank}(B) = m$ and $\mathrm{rank}(C) = r$.

The model (1) is stabilizable by static output-feedback (sof) if there exists a gain matrix $K_{sof} \in \mathbb{R}^{m\times r}$ such that the closed-loop matrix $A+BK_{sof}C$ is asymptotically stable ($A+BK_{sof}C$ is a Hurwitz matrix). Moreover, the system is stabilizable by state-feedback (sf) if there exists a gain matrix $K_{sf} \in \mathbb{R}^{n\times r}$ such that the matrix $A+BK_{sf}$ is Hurwitz. This last property is a special case of the former and corresponds to full state information output-feedback ($C = \mathbf{1}$) and has found a tractable solution through convex optimization and LMI formalism [6]. Let us define the set of stabilizing state-feedback matrices:

$$\mathcal{K}_{sf} = \left\{ K_{sf} \in \mathbb{R}^{m\times n} \ : \ \Lambda(A + BK_{sf}) \in \mathbb{C}^- \right\} \tag{2}$$

and the set of stabilizing static output-feedback matrices:

$$\mathcal{K}_{sof} = \left\{ K_{sof} \in \mathbb{R}^{m\times r} \ : \ \Lambda(A + BK_{sof}C) \in \mathbb{C}^- \right\} \tag{3}$$

The problem considered in this paper is to build non trivial sets of stabilizing sof for (1).

Problem 1: sof stabilization

Given the model (1), build a non trivial subset $\mathcal{K}_{sof}^{n_{sof}} \subset \mathcal{K}_{sof}$ of $n_{sof} \geq 1$ instances.

II. Construction of stabilizing sof sets via a mixed LMI/randomized approach

A. Parametrization of $\mathcal{K}_{sof}$

Two different mixed LMI/randomized algorithms may be defined. They both rely on the following parametrization of stabilizing sof matrices that has been first proposed in [26]. Let us define the following notation:

$$M(P) = \begin{bmatrix} A'P + PA & PB \\ B'P & \mathbf{0} \end{bmatrix} \tag{4}$$

A necessary and sufficient condition for the existence of a stabilizing sof for (1) is given in the following theorem.

Theorem 1: [26]

$\exists\ K_{sof} \in \mathcal{K}_{sof}$ for the model (1) if and only if there exist a stabilizing sf matrix $K_{sf} \in \mathcal{K}_{sf}$, a matrix $P \in \mathbb{S}^{+*}$ and matrices $F \in \mathbb{R}^{m\times m}$, $Z \in \mathbb{R}^{m\times r}$ solutions of the matrix inequality (5):

$$L(P, K_{sf}, Z, F) = M(P) + \mathrm{sym}\left(\begin{bmatrix} K_{sf}' \\ -\mathbf{1} \end{bmatrix} \begin{bmatrix} ZC & F \end{bmatrix} \right) \prec \mathbf{0} \tag{5}$$

Moreover, $K_{sof} = -F^{-1}Z \in \mathcal{K}_{sof}$.

Proof

Note that the sof stabilizability of (1) ($\exists\ K_{sof} \in \mathcal{K}_{sof}$) is equivalent to the existence of a matrix $K_{sof} \in \mathbb{R}^{m\times r}$ and a matrix $P \in \mathbb{S}^{+*}$ solutions to the following Lyapunov inequality:

$$\begin{array}{c} (A + BK_{sof}C)'P + P(A + BK_{sof}C) = \\ \begin{bmatrix} \mathbf{1} & C'K_{sof}' \end{bmatrix} M(P) \begin{bmatrix} \mathbf{1} \\ K_{sof}C \end{bmatrix} \prec \mathbf{0} \end{array} \tag{6}$$

Applying elimination lemma [31], this is also equivalent to the existence of matrices $K_{sof} \in \mathbb{R}^{m\times r}$, $F_1 \in \mathbb{R}^{r\times m}$, $F_2 \in \mathbb{R}^{m\times m}$ and a matrix $P \in \mathbb{S}^{+*}$ solutions to the following matrix inequality:

$$M(P) + \mathrm{sym}\left(\begin{bmatrix} C'K_{sof}' \\ -\mathbf{1} \end{bmatrix} \begin{bmatrix} F_1 & F_2 \end{bmatrix} \right) \prec \mathbf{0} \tag{7}$$

The matrix F_2 is always invertible since the block $(2, 2)$ reads $F_2 + F_2' \succ \mathbf{0}$. Factorizing F_2 in (7) leads to the result. ■

Even if the existence condition is expressed in terms of solution to a Bilinear Matrix Inequality (BMI), this parametrization has some interesting characteristics that will be useful for defining efficient numerical procedure for sof synthesis. First, thanks to the introduction of additional variables, there is a decoupling between the computation of $K_{sof} \in \mathcal{K}_{sof}$ and the Lyapunov certificate P. Moreover, the matrix K_{sf} at the origin of the non convexity of condition (5) must not be arbitrarily chosen but it has to be a stabilizing sf for (1), i.e. $K_{sf} \in \mathcal{K}_{sf}$. It is well-known that the computation of a stabilizing sf for (1) is a convex problem that may solved via the following LMI optimization problem [6]:

$$\begin{array}{ll} \min\limits_{X,R} & \mathrm{trace}(X) \\ \text{under} & \mathrm{trace(X)} > \alpha \\ & X \succ \mathbf{0} \\ & AX + XA' + BR + R'B' \prec \mathbf{0} \end{array} \tag{8}$$

for some $\alpha > 0$ and where $K_{sf} = RX^{-1} \in \mathcal{K}_{sf}$.

For a given $K_{sf} \in \mathcal{K}_{sf}$, the LMI convex set $\mathcal{L}_{sof}^{K_{sf}}$ defined by:

$$\begin{array}{rl} \mathcal{L}_{sof}^{K_{sf}} = & \{(P, Z, F) \in \mathbb{S}^{+*} \times \mathbb{R}^{m\times r} \times \mathbb{R}^{m\times m} \ : \\ & \exists\ K_{sf} \in \mathcal{K}_{sf} \mid L(P, K_{sf}, Z, F) \prec \mathbf{0}\} \end{array} \tag{9}$$

is a convex parametrization that approximates the set of all stabilizing sof $\mathcal{K}_{sof}$. Note that this convex approximation may be empty ($\mathcal{L}_{sof}^{K_{sf}} = \emptyset$) even if $\mathcal{K}_{sof} \neq \emptyset$ for a given

$K_{sf} \in \mathcal{K}_{sf}$. However, a complete parametrization of $\mathcal{K}_{sof}$ is obtained when K_{sf} covers the whole continuum of $\mathcal{K}_{sf}$.

$$\mathcal{K}_{sof} = \left\{ \bigcup_{i}^{\infty} \mathcal{L}_{sof}^{K_{sf}^i} \mid K_{sf}^i \in \mathcal{K}_{sf} \right\} \tag{10}$$

In fact, the problem of finding a stabilizing sof amounts to find a triplet composed of $(K_{sof}, K_{sf}, P) \in \mathcal{K}_{sof} \times \mathcal{K}_{sf} \times \mathbb{S}^{+*}$ verifying (5), meaning that a common Lyapunov certificate P has to be found for K_{sf} and K_{sof}. This last interpretation is reminiscent of a necessary and sufficient condition of sof stabilizability proposed in [5].

Remark 1: Note that the parametrization (5) of $\mathcal{K}_{sof}$ has an equivalent one for $\mathcal{K}_{sf}$ since:

$$\begin{aligned} L(P, K_{sf}, Z, F) = L(P, K_{sof}, H, F) = \\ M(P) + \mathrm{sym}\left(\begin{bmatrix} C'K'_{sof} \\ -\mathbf{1} \end{bmatrix} \begin{bmatrix} H' & F' \end{bmatrix} \right) \prec \mathbf{0} \end{aligned} \tag{11}$$

where $K_{sf} = -F^{-T}H' \in \mathcal{K}_{sf}$. Given $K_{sof} \in \mathcal{K}_{sof}$, one can easily get a convex parametrization approximating the set of all stabilizing sf. This seems of no consequence since the complete set $\mathcal{K}_{sf}$ may be parametrized by the LMI convex set defined in (8). Nevertheless, this alternate parametrization will appear useful in the sequel when looking for sets of sof.

For a given $K_{sof} \in \mathcal{K}_{sof}$, the LMI convex set $\mathcal{L}_{sf}^{K_{sof}}$ defined by:

$$\begin{aligned} \mathcal{L}_{sf}^{K_{sof}} = \{ (P, H, F) \in \mathbb{S}^{+*} \times \mathbb{R}^{n\times m} \times \mathbb{R}^{m\times m} \ : \\ \exists\ K_{sof} \in \mathcal{K}_{sof} \mid L(P, K_{sof}, H, F) \prec \mathbf{0} \} \end{aligned} \tag{12}$$

is a convex parametrization that approximates the set of all stabilizing sf $\mathcal{K}_{sf}$.

To make completely effective the condition of theorem 1, a precise description of the subset $\mathcal{K}_{sf}^g \subset \mathcal{K}_{sf}$ defined by:

$$\mathcal{K}_{sf}^g = \left\{ K_{sf} \in\subset \mathcal{K}_{sf} \ : \ \mathcal{L}_{sof}^{K_{sf}} \neq \emptyset \right\} \tag{13}$$

would be necessary. The idea is therefore to find at least one element belonging to this set by generating a sufficient number of random samples in $\mathcal{K}_{sf}$ via Hit-and-Run techniques.

B. Hit-and-Run (HR) for the stability set for matrices

We apply HR for generating the sets of stabilizing state-feedback matrices K_{sf} (2) and static output-feedback matrices K_{sof} (3). The approach is the same for both cases. We describe it for static output-feedback matrices (for state-feedback matrices take $C = I$, $K \in \mathbb{R}^{m\times n}$). Suppose matrices A, B, C are given and $K \in \mathbb{R}^{m\times r}$ is a variable, K^0 belongs to the bounded set of stabilizing gains:

$$\mathcal{K}_{sof} = \{ K : A + BKC \text{ is Hurwitz.} \} \tag{14}$$

The structure of this set is analyzed in [16]. It can be nonconvex and can consist of many disjoint domains.

In every step of the HR algorithm we generate matrix $D = Y/||Y||, Y =$ `randn(m,r)` which is uniformly distributed on the unit sphere in the space of matrices equipped with Frobenius norm. Matrix D is a random direction in the space of $m \times r$-matrices. We call *boundary oracle* an algorithm which provides $L = \{t \in R : K^0 + tD \in \mathcal{K}_{sof}\}$. We denote

$$\begin{aligned} A + B(K^0 + tD)C = F + tG, \\ F = A + BK^0C, \quad G = BDC \end{aligned}$$

for a matrix $K^0 \in \mathcal{K}_{sof}$, then $L = \{t : F + tG \text{ is Hurwitz}\}$. In the simplest case when $\mathcal{K}_{sof}$ is convex, this set is the interval $(-\underline{t}, \bar{t})$ where $\bar{t} = \sup\{t : K^0 + tD \in \mathcal{K}_{sof}\}, \underline{t} = \sup\{t : K^0 - tD \in \mathcal{K}_{sof}\}$. In more general situations boundary oracle provides all intersections of the straight line $K^0 + tD, -\infty < t < +\infty$ with $\mathcal{K}_{sof}$. L consists of finite number of intervals, the algorithm for calculating their end points is presented in [16], Section 4. However sometimes "brute force" approach is more simple. Introduce $f(t) = \max \Re \text{ eig}(F + tG)$, then the end points of the intervals are solutions of the equation $f(t) = 0$ and can be found by use of standard 1D equation solvers (such as command `fsolve` in Matlab).

HR method works as follows.

1. Find a starting point $K^0 \in \mathcal{K}_{sof}$; $i = 0$.
2. At the point $K^i \in \mathcal{K}_{sof}$ generate a random direction $D^i \in \mathbb{R}^{m\times r}$ uniformly distributed on the unit sphere.
3. Apply boundary oracle procedure, i.e., define the set
$$L_i = \{t \in \mathbb{R} : K^i + tD^i \in \mathcal{K}_{sof}\}.$$
4. Generate a point t_i uniformly distributed in L_i (we recall that L_i is, in general, a finite set of intervals), and compute a new point
$$K^{i+1} = K^i + t_i D^i.$$
5. Go to step 2 and increase i.

The simplest theoretical result on the behavior of HR method states that if $\mathcal{K}_{sof}$ does not contain lower dimensional parts, then the method achieves the neighborhood of any point of $\mathcal{K}_{sof}$ with nonzero probability and asymptotically the distribution of points k_i tends to uniform one. The rate of convergence strongly depends on geometry of $\mathcal{K}_{sof}$ and its dimension.

C. Two LMI/randomized algorithms for sof stabilization

Two algorithms using complementary advantages of degrees of freedom offered by parametrization (5) and Hit-and-Run numerical efficiency are built in order to generate non trivial sets of stabilizing sof. The first algorithm uses Hit-and-Run only for generating a large subset $\mathcal{K}_{sf}^{n_{sf}}$ of $\mathcal{K}_{sf}$ that will serve for checking if $\mathcal{L}_{sof}^{K_{sf}}$ is empty.

Algorithm 1: 1- Compute a stabilizing sf $K_{sf}^0 = RX^{-1} \in \mathcal{K}_{sf}$ via the solution of the LMI problem (8):

2- From K_{sf}^0, generate a set $\mathcal{K}_{sf}^{n_{sf}} \subset \mathcal{K}_{sf}$ which is the collection of n_{sf} samples of stabilizing sf matrices via H.R. ;

3- Compute a set $\mathcal{K}_{sof}^{n_{sof}^1} \subset \mathcal{K}_{sof}$ which is the collection of n_{sof}^1 samples of stabilizing sof matrices:
$\forall\ K_{sf}^i \in \mathcal{K}_{sf}^{n_{sf}}$, if the LMI set $\mathcal{L}_{sof}^{K_{sf}^i} \neq \emptyset$, add the solution $K_i = -F_i^{-1}Z_i$ to the collection set $\mathcal{K}_{sof}^{n_{sof}^1}$.

Remark 2: (8) is not the only way to initialize algorithm 1. K_{sf}^0 may also be computed as the LMI solution of the H_2 or H_∞ state-feedback problems (see [4] for more details).

Step 3 is performed by testing the realizability of the LMI set $\mathcal{L}_{sof}^{K_{sf}}$ for each instance $K_{sf} \in \mathcal{K}_{sf}^{n_{sf}}$ implying the solution of the homogeneous LMIs (7) with respect to the decision variables (P, Z, F). To avoid numerical problems, this step must be done by solving the following semidefinite programming problem:

$$\begin{array}{ll} \min\limits_{P,Z,F} & \mathrm{trace}(P) \\ \text{s.t.} & \mathrm{trace}(\mathrm{P}) > \alpha \\ & P \succ \mathbf{0} \\ & M(P) + \mathrm{sym}\left(\begin{bmatrix} K'_{sf} \\ -\mathbf{1} \end{bmatrix} \begin{bmatrix} ZC & F \end{bmatrix} \right) \prec \mathbf{0} \end{array} \tag{15}$$

for some $\alpha > 0$.

The interest of running this algorithm is twofold. First, it will be a good way to evaluate the conservatism of the convex approximation of $\mathcal{K}_{sof}$ induced by parametrization (5). Indeed, the percentage n_{sof}^1/n_{sf} may be defined as a quantitative measure of both the conservatism of the convex approximation as well as the difficulty to stabilize the plant via sof.

Remark 3: If K is a stabilizing sof *a priori* known, then the set $\mathcal{L}_{sof}^{KC} \neq \emptyset$ since the choice $Z = -FK$ will lead to the existence of $P \in \mathbb{S}^{+*}$ and $F \in \mathbb{R}^{m\times m}$ such that:

$$\begin{array}{l} M(P) + \mathrm{sym}\left(\begin{bmatrix} C'K' \\ -\mathbf{1} \end{bmatrix} \begin{bmatrix} -FKC & F \end{bmatrix} \right) = \\ M(P) + \mathrm{sym}\left(\begin{bmatrix} C'K' \\ -\mathbf{1} \end{bmatrix} [-F] \begin{bmatrix} KC & -\mathbf{1} \end{bmatrix} \right) \prec \mathbf{0} \end{array} \tag{16}$$

Applying elimination lemma, this last condition is equivalent to the existence of a matrix $P \in \mathbb{S}^{+*}$ such that:

$$(A + BKC)'P + P(A + BKC) \prec \mathbf{0} \tag{17}$$

which is obviously true since $K \in \mathcal{K}_{sof}$

A much more numerically efficient mixed LMI/randomized algorithm generating sets of stabilizing sof may be deduced from the previous one. It mainly avoids to run the computational burden of checking the emptyness of LMI set $\mathcal{L}_{sof}^{K_{sf}^i}$ for every instance $\forall\ K_{sf}^i \in \mathcal{K}_{sf}^{n_{sf}}$ but starts a hit and run generation of $\mathcal{K}_{sof}^{n_{sof}^2}$ as soon as an initial $K_{sof} \in \mathcal{K}_{sof}$ is found by LMI step. It proves to be much more efficient in practice to generate sets of stabilizing sof in almost every studied cases of the COMPl$_e$ib library.

Algorithm 2: 1- Compute a stabilizing state-feedback $K_{sf}^0 = RX^{-1} \in \mathcal{K}_{sf}$ via the solution of the LMI problem (8) ;

2- From K_{sf}^0, generate a set $\mathcal{K}_{sf}^{n_{sf}} \subset \mathcal{K}_{sf}$ which is the collection of n_{sf} samples of stabilizing sf matrices via H.R. ;

3- Find an initial stabilizing sof $K_{sof}^0 = -F_0^{-1}Z_0$: Check every $K_{sf}^i \in \mathcal{K}_{sf}^{n_{sf}}$ until $\mathcal{L}_{sof}^{K_{sf}^i} \neq \emptyset$;

4- From K_{sof}^0, compute a set $\mathcal{K}_{sof}^{n_{sof}^2} \subset \mathcal{K}_{sof}$ which is the collection of n_{sof}^2 samples of stabilizing sof matrices via H.R.

D. Results from the COMPl$_e$ib library

The COMPl$_e$ib library is composed of different LTI models (1) ranging from purely academic problems to more realistic industrial examples. The underlying systems that are already open-loop asymptotically stable have not been considered here for obvious reasons. With this last restriction, 53 different models mainly classified in six classes have been tested.

- Aerospace models: Aircraft models (AC), helicopter models (HE), jet engine models (JE)
- Reactor models (REA)
- Decentralized interconnected systems (DIS)
- Academic tests problems (NN)
- Various applications: Wind energy conversion model (WEC), binary distillation towers (BDT) and terrain following models (TF), string of high-speed vehicles (IH), strings (CSE), piezoelectric bimorph actuator (PAS),
- Second order models: A tuned mass damper (TMD), a flexible satellite (FS)
- 2D heat flow models (HF2D)

For precise details concerning each single example and benchmark, the interested reader may read [24] and [23]. The main results concerning the construction of stabilizing sof are presented in the tables I, II.

- First note that it has been possible to stabilize every model except `AC10`. In the first case, the HIFOO package which may be considered as one of the most effective tool for sof stabilization and for optimal H_∞ sof control is not able either to stabilize it.
- A quick look at the reference [29] clearly shows that algorithms 1 and 2 gives far better results than this last reference. In [29], the proposed algorithm was unable to stabilize plants `AC10`, `NN1`, `NN5`, `NN5`, `NN7`, `NN10` and `NN12`.
- Hard to stabilize examples may be identified as the ones for which less than 5% of the initializing state-feedback succeed in finding a stabilizing sof. `AC9`, `AC13`, `AC14`, `AC18`, `HE3`, `JE3`, `BDT2`, `TF3`, `NN9`, `NN12`, `NN14` are such plants for which n_{sof}^1 obtained with algorithm 1 is rather low. This is mainly due to the bad conditioning of numerical operations (matrix inversion) and LMI optimization rather than a failure of the method in itself. This is confirmed by the reference [25] where similar failures of SDP solvers were already noticed for some of the previous examples (`AC14`, `AC18`, `JE3`) for the problems of H_2 or H_∞ state feedback optimal control which are known to have a convex formulation. Moreover, the initialization $K_{sf} = KC$ where $K \in \mathcal{K}_{sof}$ is known *a priori*, does not perform well for all these examples, demonstrating that the plant matrices are poorly conditioned (see remark 3).
- It is not a surprise to note that $n_{sof}^1 \leq n_{sof}^2$ since if the initialization step succeeds $n_{sof}^2 = 1000$ while $\max(n_{sof}^1) = 1000$. More surprising is the easiness to get n_{sof}^2 stabilizing sof for almost all examples and considering also that the complete run is really fast in

general. Size seems to be a limiting factor for LMI step but not for Hit-and-Run step.

Figure 1 shows the population stabilizing gains with the boundary for example `AC7`. It is interesting to note that the exact shape of the set of stabilizing sof is easily obtained with the mixed LMI/randomized algorithm 1 (see [20] for comparison). Similar results have been obtained for all two-parameters examples of the database COMPl$_e$ib, labelled `AC4`, `NN1`, `NN5` (see below), `NN17` and `HE1`.

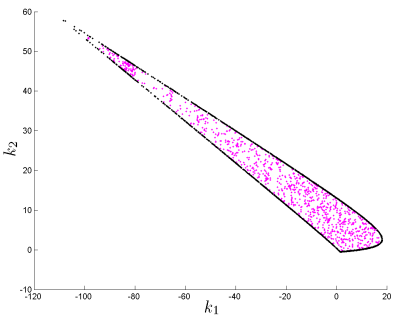

Fig. 1. Set of stabilizing sof generated with LMI/Randomized algorithm 2 for AC7 example

Figures 2 and 3 show a comparison of the populations respectively obtained via algorithms 1 and 2 for benchmark `NN5`.

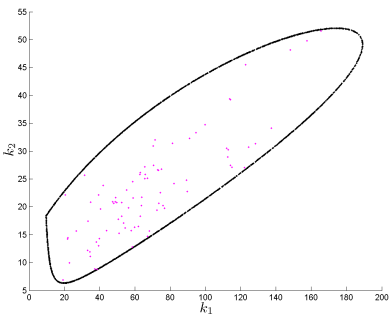

Fig. 2. Set of stabilizing sof generated with LMI/Randomized algorithm 1 for NN5 example

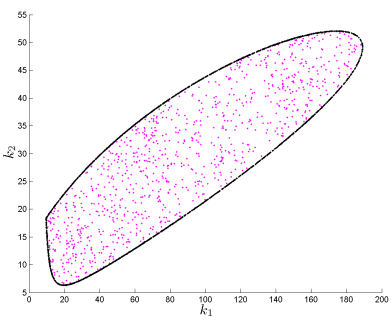

Fig. 3. Set of stabilizing sof generated with LMI/Randomized algorithm 2 for NN5 example

III. Conclusions

In this article, two different mixed LMI/Hit-and-run algorithms have been proposed for the stabilization of LTI systems via static output feedback. Evaluation of these algorithms on the 53 benchmarks that are not open-loop stable shows that both may be considered as effective tools for the construction of non trivial sets of stabilizing sof. These examples exhibit relatively different structural features and different size of the state-space realization as well. Due to the relative limitations of SDP solvers in terms of computation time for large scale LTI systems, these approaches cannot be considered as competitive packages with respect to the last dedicated packages based on nonsmooth optimization [3], [17]. Nevertheless, we strongly believe that a promising new perspective is possible in the continuity of the seminal work of [12]. In particular, the results presented here should pave the way for the development of similar routines for performance optimization via static output feedback. Preliminary promising results on optimal H_2 and H_∞ sof control are reported in the technical report [4].

Appendix

The following notations are used in tables I, II.

- OLS stands for Open-Loop Stable
- OLMS stands for Open-Loop Marginally Stable (a unique eigenvalue at 0 or multiple eigenvalue with 0 real part but scalar associated Jordan blocks)
- OLNS stands for Open-Loop Non Stable `max(real(eig(A)))>0`
- n^1_{sof}/n_{sf} stands for the portion of stabilizing sof found by the algorithm 1

Ex.	n_x	n_u	n_y	OLS	n^1_{sof}/n_{sf}
AC1	5	3	3	OLMS	240/1000
AC2	5	3	3	OLMS	334/1000
AC5	4	2	2	OLNS	283/1000
AC9	10	4	5	OLNS	34/1000
AC10	55	2	2	OLNS	*
AC11	5	2	4	OLNS	996/1000
AC12	4	3	4	OLNS	997/1000
AC13	28	3	4	OLNS	39/1000
AC14	40	3	4	OLNS	13/1000
AC18	10	2	2	OLNS	17/1000
HE1	4	2	1	OLNS	93/1000
HE3	8	4	6	OLNS	12/1000
HE4	8	4	6	OLNS	1000/1000
HE5	4	2	2	OLNS	13/1000
HE6	20	4	6	OLNS	321/1000
HE7	20	4	6	OLNS	327/1000
DIS2	3	2	2	OLNS	842/1000
DIS4	6	4	6	OLNS	1000/1000
DIS5	4	2	2	OLNS	725/1000
JE2	21	3	3	OLMS	94/1000
JE3	24	3	6	OLMS	44/1000
REA1	4	2	3	OLNS	999/1000
REA2	4	2	2	OLNS	554/1000
REA3	12	1	3	OLNS	965/1000

TABLE I

Numerical results for the COMPL$_e$IB library

Ex.	n_x	n_u	n_y	OLS	n^1_{sof}/n_{sf}
WEC1	10	3	4	OLNS	775/1000
BDT2	82	4	4	OLMS	43/1000
IH	21	11	10	OLMS	63/1000
CSE2	60	2	30	OLNS	10/10
PAS	5	1	3	OLMS	236/1000
TF1	7	2	4	OLMS	81/1000
TF2	7	2	3	OLMS	189/1000
TF3	7	2	3	OLMS	6/1000
NN1	3	1	2	OLNS	629/1000
NN2	2	1	1	OLMS	1000/1000
NN5	7	1	2	OLNS	84/1000
NN6	9	1	4	OLNS	983/1000
NN7	9	1	4	OLNS	620/1000
NN9	5	3	2	OLNS	7/1000
NN12	6	2	2	OLNS	28/1000
NN13	6	2	2	OLNS	77/1000
NN14	6	2	2	OLNS	44/1000
NN15	3	2	2	OLMS	821/1000
NN16	8	4	4	OLMS	61/1000
NN17	3	2	1	OLNS	125/1000
HF2D10	5	2	3	OLNS	991/1000
HF2D11	5	2	3	OLNS	993/1000
HF2D14	5	2	4	OLNS	1000/1000
HF2D15	5	2	4	OLNS	1000/1000
HF2D16	5	2	4	OLNS	998/1000
HF2D17	5	2	4	OLNS	1000/1000
HF2D18	5	2	2	OLNS	755/1000
TMD	6	2	4	OLNS	654/1000
FS	5	1	3	OLNS	977/1000

TABLE II

NUMERICAL RESULTS FOR THE COMPL$_e$IB LIBRARY

REFERENCES

[1] B.D.O. Anderson, N.K. Bose, and E.I. Jury. Output feedback stabilization and related problems - solutions via decision methods. *IEEE Trans. on Automat. Control*, 20(1), February 1975.

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

[3] P. Apkarian, D. Noll, and A. Rondepierre. Mixed H_2/H_∞ control via nonsmooth optimization. *SIAM J. on Control and Optimization*, 47(3):1516–1546, 2008.

[4] D. Arzelier, E. Gryazina, D. Peaucelle, and Boris Polyak. Mixed lmi/hit-and-run alorithms for static ouput stabilization: H_2 and h_∞ performances. Technical Report XXX, LAAS-CNRS, Toulouse, October 2009. URL: `http://www.laas.fr /~arzelier/`.

[5] A. Astolfi and P. Colaneri. Hankel/Toeplitz matrices and the static output feedback stabilization problem. *Mathematics of Control, Signals, and Systems (MCSS)*, 17(4):231–268, 2005.

[6] J. Bernussou, J.C. Geromel, and P.L.D. Peres. A linear programing oriented procedure for quadratic stabilization of uncertain systems. *Systems & Control Letters*, 13:65–72, July 1989.

[7] D. Bertsimas and S Vempala. Solving convex problems by random walks. *Journal of the AMC*, 51:540–556, 2004.

[8] J.V. Burke, D. Henrion, A.S. Lewis, and M.L Overton. Hifoo - a matlab package for fixed-order controller design and h-infinity optimization. In *IFAC Symposium On RObust CONtrol Design, ROCOND06*, Toulouse, France, June 2006.

[9] J.V. Burke, D. Henrion, A.S. Lewis, and M.L. Overton. Stabilization via nonsmooth, nonconvex optimization. *IEEE Trans. on Automat. Control*, 51:1760–1769, November 2006.

[10] M.C. de Oliveira and J.C. Geromel. Numerical comparison of output feedback design methods. In *American Control Conference*, Albuquerque, New Mexico, June 1997.

[11] L. El Ghaoui, F. Oustry, and M. AitRami. A cone complementarity linearization algorithm for static ouput-feedback and related problems. *IEEE Trans. on Automat. Control*, 42(8):1171–1176, August 1997.

[12] Y. Fujisaki, Y. Oishi, and R. Tempo. Mixed deterministic/randomized methods for fixed order controller design. *IEEE Trans. on Automat. Control*, 53(9):2033–2047, 2008.

[13] J.C. Geromel, P.L.D. Peres, and S.R. Souza. Convex analysis of output feedback control problems: Robust stability and performance. *IEEE Trans. on Automat. Control*, 41(7):997–1003, July 1996.

[14] J.C. Geromel, C.C. de Souza, and R.E. Skelton. Static output feedback controllers: Stability and convexity. *IEEE Trans. on Automat. Control*, 43(1):120–125, January 1998.

[15] K.M. Grigoriadis and E.B. Beran. *Advances in Linear Matrix Inequality Methods in Control*, chapter 13 : Alternating Projection Algorithms for Linear Matrix Inequalities Problems with Rank Constraints, pages 251–267. SIAM, Philadelphia, 2000. Edited by L. El Ghaoui and S.-I. Niculescu.

[16] E.N. Gryazina and B.T. Polyak. Stability domain in the parameter space: D-decomposition revisited. *Automatica*, 42(1):13–26, 2006.

[17] S. Gumussoy, D. Henrion, M. Millstone, and M.L. Overton. Multi-objective robust control with HIFOO 2.0. In *IFAC RObust CONtrol Design, ROCOND09*, Haifa, Israël, June 2009.

[18] S. Gumussoy and M.L. Overton. Fixed-order H_∞ controller design via HIFOO, a specialized nonsmooth optimization package. In *IEEE American Control Conference*, pages 2750–2754, Seattle, USA, June 2008.

[19] D. Henrion. Solving static output feedback problems by direct search optimization. In *Proceedings of the joint IEEE CCA/CACSD/ISIC Conference*, Munich, Germany, October 2006.

[20] D. Henrion and M. Sebek. Plane geometry and convexity of polynomial stability regions. In *Proceedings of the International Symposium on Symbolic and Algebraic Computations (ISSAC)*, Hagenberg, Austria, July 2008.

[21] T. Iwasaki and R.E. Skelton. The XY-centring algorithm for the dual LMI problem: a new approach to fixed-order control design. *Int. J. Control*, 62(6):1257–1272, 1995.

[22] 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.

[23] F. Leibfritz. COMPleib: COnstraint Matrix-optimization Problem library - a collection of test examples for nonlinear semidefinite programs, control system design and related problems. Technical report, University of Trier, Department of Mathematics, 2004. URL: `www.complib.de`.

[24] F. Leibfritz and W. Lipinski. Description of the benchmark examples in compleib 1.0. Technical report, University of Trier, Department of Mathematics, 2003. URL: `www.complib.de`.

[25] F. Leibfritz and W. Lipinski. COMPleib 1.0: COnstraint Matrix-optimization Problem - user manual and quick reference. Technical report, University of Trier, Department of Mathematics, 2004. URL: `www.complib.de`.

[26] D. Peaucelle and D. Arzelier. An efficient numerical solution for H_2 static output feedback synthesis. In *European Control Conference*, pages 3800–3805, Porto, Portugal, September 2001.

[27] D. Peaucelle and D. Arzelier. Ellipsoidal sets for resilient and robust static output-feedback. *IEEE Trans. on Automat. Control*, 50(6):899–904, June 2005.

[28] B.T. Polyak and E.N. Gryazina. Hit-and-run: new design technique for stabilization, robustness and optimization of linear systems. In *IFAC World Congress*, Seoul, July 2008.

[29] E. Prempain. A comparison of two algorithms for the static H_∞ loop shaping problem. In *IFAC World Congress*, Prague, July 2005.

[30] E. Prempain and I. Postlethwaite. Static h-infinity loop shaping control of a fly-by-wire helicopter. *Automatica*, 41(9), September 2005.

[31] R.E. Skelton, T. Iwasaki, and K. Grigoriadis. *A Unified Approach to Linear Control Design*. Taylor and Francis series in Systems and Control, 1998.

[32] R.L. Smith. Efficient monte carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

[33] V.L. Syrmos, C.T. Abdallah, P. Dorato, and K. Grigoriadis. Static output feedback: A survey. *Automatica*, 33(2):125–137, 1997.

[34] V.F. Turchin. On computation of multi-dimensional integrals by the monte-carlo method. *Theory of Probability and its Applications*, 16(4):720–724, 1971.