$\mathcal{H}_2/\mathcal{H}_\infty$ Robust Static Output Feedback Control Design without solving Linear Matrix Inequalities

R. Toscano [1]

Laboratoire de Tribologie et de Dynamique des Systèmes CNRS UMR5513 ECL/ENISE, 58 rue Jean Parot 42023 Saint-Etienne cedex 2

Abstract. In this paper we investigate the problem of robust synthesis of a static output feedback controller, with guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, in the context of multiple parametric uncertainties. To solve this problem, it is proposed a random optimization technique based on a bisection method. The principle is as follows, for a given initial stabilizing controller of the nominal system, the proposed approach iteratively generates a sequence of matrices with a decreasing $\mathcal{H}_2/\mathcal{H}_\infty$ cost. By a bisection method, this procedure is stopped when the controller reaches the best possible nominal performance that satisfies a given guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost. Numerical examples show the practical applicability of the proposed method.

Keywords: Output feedback controller; $\mathcal{H}_2/\mathcal{H}_\infty$ cost function; Random optimization technique; Bisection method.

1 Introduction

The static output feedback is one of the most important issue in control theory and applications. The reason for this is that it represents the simplest closed-loop control that can be realized in practice. Another important reason is that many problems that require a dynamical controller can be rephrased as a static output feedback problem involving an augmented plant. Consequently, many researchers have addressed the problem of static output feedback control design. Give a complete state-of-the-art of this topic is not an easy task because there exist various unconnected approaches. However, among the existing results, we can distinguish the following methods (see also the survey papers [5, 26] for more details on this issue):

- Approaches based on solvability conditions expressed from structural properties of the open-loop system [33, 14, 11, 10, 3].
- Approaches based on the resolution of Riccati equations [13, 30, 17].
- Approaches based on optimization techniques [12, 4, 6, 23, 7, 15, 19, 21, 2].
- Approaches based on pole or eigenstructure assignment techniques [8, 16, 32].

1. E-mail address: toscano@enise.fr, Tel.:+33 477 43 84 84; fax: +33 477 43 84 99

However, despite these various efforts, this problem remains difficult to solve due to its non convex nature. In view of this difficulty some recent progress have been made in the use of heuristic methods for solving a given problem [29]. In particular, random search techniques have recently received more attention in the literature (see [28] and references therein). The idea of using a random search approach to solve a complex problem was first proposed, in the domain of automatic control, by Matyas [20]; a complete state of the art of this topic can be found in [24, 27]. Following this line of research, Sun, Chung and Chang [25] have proposed a novel synthesis of $\mathcal{H}_2/\mathcal{H}_\infty$ robust static output feedback control which combines a genetic algorithm (GA) and a LMI solver. Unfortunately, this approach requires the resolution of a set of LMIs and therefore cannot be used when the number of uncertain parameters is large. Indeed, in the context of robust control design, the LMI approach generally requires to simultaneously solve a number of convex inequalities, which is exponential in the number of parameters. Thus, the LMI approach is computationally critical for a large number of uncertain parameters. For example, considering 8 uncertain parameters, we need to solve $2 \times 2^8 = 512$ LMIs and this is not an easy task even for actual LMIs solvers. In addition, suppose that this set of LMIs can be solved, the result obtained can be conservative because the solution must satisfy a large number of LMIs, and consequently, this leads, generally, to a pessimistic result.

These various difficulties motivate the investigation of a new approach for solving the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ robust static output feedback control problem in the context of multiple parametric uncertainties, without using LMIs. For this, we propose a new iterative algorithm based on a bisection method, called random bisection algorithm (RBA). The principle is as follows, for a given initial stabilizing controller of the nominal system, the proposed approach iteratively generates a sequence of matrices with a decreasing nominal $\mathcal{H}_2/\mathcal{H}_\infty$ cost. By a bisection method, this procedure is stopped when the controller reaches the best possible nominal performance that satisfies a given guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost; i.e. the worst case performance is bounded by a given pre-specified value. The proposed control design method has the following interesting characteristics:

- Very general uncertainty structures and nonlinearities can easily be taken into account without introducing overbounding.
- The proposed method does not require the resolution of linear matrix inequalities (LMIs) or bilinear matrix inequalities (BMIs) [23, 31].

As aforementioned, this last point is very interesting from a practical point of view, notably in the context of multiple parametric uncertainties.

The paper is organized as follows. In section 2, the problem to be solved is stated. Section 3 shows that a robust static output feedback controller, satisfying a given guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, can be found via an appropriate random optimization technique based on a bisection method. Section 4 illustrates the validity of the proposed method via two numerical examples. Finally, section 5 concludes this paper.

2 Problem formulation

As in [25], we consider the following uncertain model:

$$M(\theta):\begin{cases}\dot{x}(t)=A(\theta)x(t)+B_w(\theta)w(t)+B_u(\theta)u(t)\\ y(t)=C_yx(t)\\ z_2(t)=C_2x(t)+D_{2u}u(t)\\ z_\infty(t)=C_\infty x(t)+D_{\infty w}w(t)+D_{\infty u}u(t)\end{cases}\tag{1}$$

where $x(t)\in\mathbf{R}^{n_x}$ is the state vector, $w(t)\in\mathbf{R}^{n_w}$ is the disturbance vector, $u(t)\in\mathbf{R}^{n_u}$ is the control vector, $y(t)\in\mathbf{R}^{n_y}$ is the output vector, and $z_2(t)\in\mathbf{R}^{n_{z_2}}$ and $z_\infty(t)\in\mathbf{R}^{n_{z_\infty}}$ are the controlled output vectors of the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance. All the matrices are assumed to have appropriate and known dimensions. The matrices $A(\theta)$, $B_u(\theta)$ and $B_w(\theta)$ are parametrized by the system parameters $\theta\in\mathbf{R}^{n_\theta}$. In the sequel, it is assumed that $M(\theta)$ is bounded and continuous. The representation (1) can be obtained by an appropriate identification of the real system around a given operating point. In the case where the nonlinear process model is known, this local representation can also be obtained by linearization via first order Taylor series expansion of the nonlinear model. Therefore, the local model (1) is valid only around the operating point where the system is identified or linearized, and is called the nominal linear model. For this nominal model we have $\theta=\theta_0$, where θ_0 is the nominal system parameters. When the system evolves on a certain domain including the operating point, the corresponding vector parameters θ vary. These variations can be seen as parameters uncertainties. In order to take into account these uncertainties, it is necessary to consider the set of possible vectors θ. It is assumed that the vector parameters θ lie in a bounded set Θ defined as follows:

$$\Theta=\left\{\theta\in\mathbf{R}^{n_\theta}:\underline{\theta}\leqslant_e\theta\leqslant_e\bar{\theta}\right\}\tag{2}$$

where the notation $\leqslant_e$ stands for an element-by-element inequality and the vectors $\underline{\theta}=[\underline{\theta}_1\cdots\underline{\theta}_{n_\theta}]^T$ and $\bar{\theta}=[\bar{\theta}_1\cdots\bar{\theta}_{n_\theta}]^T$ are the bounds of variation of θ. It is assumed that all pairs, $(A(\theta),B_u(\theta))$ and $(A(\theta),C_y)$, are both controllable and observable for all $\theta\in\Theta$. With this formulation, the matrices $A(\theta)$, $B_u(\theta)$ and $B_w(\theta)$ are assumed to be affected by parametric, possibly nonlinear, uncertainties. The entries of these matrices are then nonlinear functions of uncertain parameters which are bounded within intervals. In this paper, we do not make any specific assumption on the dependence of $A(\theta)$, $B_u(\theta)$ and $B_w(\theta)$ on θ, except for boundedness of the entries of these matrices for all $\theta\in\Theta$. Therefore, we have a family of models $\mathcal{M}(\theta)$, parametrized by θ. The set $\mathcal{M}(\theta)$ can be written as follows

$$\mathcal{M}(\theta)=\{M(\theta):\theta\in\Theta\subset\mathbf{R}^{n_\theta}\}\tag{3}$$

where $M(\theta)$ is given by (1), and Θ is defined by (2). We suppose that the full state is not measurable and only a partial information through $y(t)$ can be used for the control. Our main objective is to find a static output feedback (SOF) controller that works satisfactorily for almost all plants.

For this purpose let us consider the static output feedback

$$u(t) = -Ky(t) \tag{4}$$

where K is the constant output feedback gain. The consideration of the SOF case is not restrictive because the dynamic output feedback case can be rephrased as a SOF control problem involving an augmented plant. Applying the constant output feedback (4) to (1), the closed-loop system is given by

$$\left\{\begin{array}{l} \dot{x}(t) = A_c(\theta)x(t) + B_w(\theta)w(t) \\ z_2(t) = \bar{C}_2 x(t) \\ z_\infty(t) = \bar{C}_\infty x(t) + D_{\infty\omega} w(t) \end{array}\right. \tag{5}$$

where the matrices $A_c(\theta)$, $\bar{C}_2$, and $\bar{C}_\infty$ are defined as follows

$$\begin{array}{c} A_c(\theta) = A(\theta) - B_u(\theta)KC_y \\ \bar{C}_2 = C_2 - D_{2u}KC_y \\ \bar{C}_\infty = C_\infty - D_{\infty u}KC_y \end{array} \tag{6}$$

In this paper we consider the following robust synthesis problem.

Robust synthesis problem. *Consider the set of systems (3), and the following $\mathcal{H}_2/\mathcal{H}_\infty$ cost function:*

$$\left\{\begin{array}{l} J(\theta,K) = \alpha J_\infty(\theta,K) + \beta J_2(\theta,K) \\ \\ J_\infty(\theta,K) = ||G_\infty(s,\theta,K)||_\infty^2, \quad J_2(\theta,K) = ||G_2(s,\theta,K)||_2^2 \end{array}\right. \tag{7}$$

where $\alpha \geqslant 0$ and $\beta \geqslant 0$ are given weighting factors; $G_\infty(s,\theta)$ and $G_2(s,\theta)$ are the closed-loop transfer matrices from w to z_∞ and z_2, respectively; $||.||_\infty$ and $||.||_2$, denote the $\mathcal{H}_\infty$ and $\mathcal{H}_2$ norms, respectively. Our aim is to look for, from an initial stabilizing controller of the nominal system, a static output feedback K that reaches the best possible nominal performance and such that the worst-case performance, i.e. the guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, is bounded by a pre-specified value g_c.

The resolution of this problem requires the estimation of the guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, which can be formulated as follows.

Estimation of the guaranteed cost. *For a given controller K, a given set of systems $\mathcal{M}(\theta) = \{M(\theta) : \theta \in \Theta \subset \mathbf{R}^{n_\theta}\}$, for which the closed-loop is stable, and the $\mathcal{H}_2/\mathcal{H}_\infty$ cost function (7), determines the corresponding guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, i.e. the quantity $w_c(K)$ such that $J(\theta,K) \leqslant w_c(K)$ for all $\theta \in \Theta$.*

3 Main results

3.1 Estimation of the guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost

Let K be an output feedback gain for which the closed-loop system is asymptotically stable for all vector parameters $\theta \in \Theta$.

In this case, K is said an element of the set $\mathcal{K}_{RS}$ of robust stabilizing controllers which can be defined as follows

$$\mathcal{K}_{RS} = \left\{K \in \mathbf{R}^{n_y \times n_u} : A_c(\theta) \in \mathcal{H}, \forall \theta \in \Theta\right\} \tag{8}$$

where $A_c(\theta)$ is the state matrix of the closed-loop system, and $\mathcal{H}$ is the set of Hurwitz matrices:

$$\mathcal{H} = \left\{A \in \mathbf{R}^{n_x \times n_x} : \max_{1 \leqslant i \leqslant n_x} \mathrm{Re}[\lambda_i(H)] < 0\right\} \tag{9}$$

From a practical point of view, stability is necessary but often not sufficient. It is also very important to obtain a satisfactory performance level which can be evaluated by the mean of a given cost function. In the $\mathcal{H}_2/\mathcal{H}_\infty$ robust control problem studied in this paper, the performance function is the $\mathcal{H}_2/\mathcal{H}_\infty$ cost given by (7). For a given controller $K \in \mathcal{K}_{RS}$, we want to determine the guaranteed cost, i.e. the worst-case performance $w_c(K)$:

$$w_c(K) = \max_{\theta \in \Theta} J(\theta, K) \tag{10}$$

The problem of computing the exact value of the guaranteed cost is not an easy task especially for multiple parametric uncertainties. For instance, using an LMI formulation (see [25]), it is necessary to solve $2r+1$ LMIs with $r = 2^{n_\theta}$. Thus, for a large number of uncertain parameters (i.e. $n_\theta \geqslant 8$), the resolution can be very difficult if not impossible. To overcome this difficulty, a random estimation approach can be used to find an estimate $\hat{w}_c(K)$ of $w_c(K)$. Let $\theta^{(1)}, \cdots, \theta^{(\eta)} \in \Theta$ be i.i.d (independent and identically distributed) samples generated according to a uniform distribution on Θ, and define

$$\hat{w}_c(K) = \max_{1 \leqslant i \leqslant \eta} J(\theta^{(i)}, K) \tag{11}$$

For a given sample $\theta^{(i)}$, the quantities $J_\infty(\theta^{(i)}, K)$ and $J_2(\theta^{(i)}, K)$, can be easily computed with MatLab routines. For a detailed description on how to compute these quantities, see for instance [9]. In fact the main difficulty is how to determine the minimum number of samples η required so that $J(\theta, K)$ is bounded by $\hat{w}_c(K)$ for almost all $\theta \in \Theta$ with high probability. The following theorem answers this interrogation.

Theorem 1. *Let $\theta^{(1)}, \cdots, \theta^{(\eta)} \in \Theta$ be i.i.d samples generated according to a uniform distribution on Θ. For a given $K \in \mathcal{K}_{RS}$, $\alpha \geqslant 0$ and $\beta \geqslant 0$, define*

$$\hat{w}_c(K) = \max_{1 \leqslant i \leqslant \eta} \left\{\alpha J_\infty(\theta^{(i)}, K) + \beta J_2(\theta^{(i)}, K)\right\} \tag{12}$$

then, for a number of samples $\eta \geqslant \ln(1-\rho)/\ln(1-e)$, the $\mathcal{H}_2/\mathcal{H}_\infty$ cost function (7) is bounded by $\hat{w}_c(K)$ with a confidence at least ρ, except possibly for those θ belonging to a set of measure $\leqslant e$.

Proof. Let $\mathcal{E}$ be the set of ε-approximates of $w_c(K)$ defined as follows:

$$\mathcal{E} = \{\theta \in \Theta : w_c - \varepsilon \leqslant J(\theta,K) \leqslant w_c + \varepsilon\}$$

thus, any value $\hat{w}_c = J(\theta,K)$ with $\theta \in \mathcal{E}$, is said an estimate of the worst-case performance w_c with a precision level ε. The problem is how many samples are needed in order to obtain at least one sample belonging to the set $\mathcal{E}$ with a high probability ρ (e.g $\rho = 0.99$), and thus to obtain a reliable estimate of w_c. Let e be the probability so that θ belong to the set $\mathcal{E}$, this probability is defined as $e = \text{vol}(\mathcal{E})/\text{vol}(\Theta)$, where the notation vol(.) stands for the volume of the set (.). Let us consider a succession of η drawings on the set Θ, the probability to obtain one sample belonging to $\mathcal{E}$, is given by $e + \sum_{k=2}^{\eta} e(1-e)^{k-1}$. The problem is then to determine η such that $e + \sum_{k=2}^{\eta} e(1-e)^{k-1} \geqslant \rho$. This inequality can be rewritten as follows:

$$1 - e - e\sum_{k=2}^{\eta}(1-e)^{k-1} \leqslant 1 - \rho$$

let us pose $\zeta = 1 - e$, the inequality becomes:

$$\begin{array}{l}
\zeta - (1-\zeta)\sum_{k=2}^{\eta}\zeta^{k-1} \leqslant 1-\rho \\
\zeta - (1-\zeta)(\zeta + \zeta^2 + \cdots + \zeta^{\eta-1}) \leqslant 1-\rho \\
\zeta - (\zeta + \zeta^2 + \cdots + \zeta^{\eta-1} - \zeta^2 - \zeta^3 - \cdots - \zeta^{\eta}) \leqslant 1-\rho \\
\zeta^{\eta} \leqslant 1-\rho \\
(1-e)^{\eta} \leqslant 1-\rho
\end{array}$$

therefore, taking the logarithm, we obtain:

$$\eta \geqslant \ln(1-\rho)/\ln(1-e) \tag{13}$$

Finally, if for a given ρ close to 1 and a given e close to zero we compute $\hat{w}_c(K)$ using (12) with a number of samples $\eta \geqslant \ln(1-\rho)/\ln(1-e)$, it can be asserted, with a confidence ρ, that $J(\theta,K)$ is bounded by $\hat{w}_c(K)$ for all $\theta \in \Theta$, except possibly for those θ belonging to the set $\mathcal{E}$ which have a probability measure $\text{vol}(\mathcal{E})/\text{vol}(\Theta)$ no larger than e.

Remark 1. Inequality (13) shows that the number of samples required for each uncertain parameter does not depend upon the number of these uncertain parameters. This fact allows the treatment of multiple parametric uncertainties without curse of dimensionality, nevertheless it depends on the value of e, i.e. the reliability of the estimate. Indeed, a given value of e implies a certain level of precision ε, which is unknown. However, ε is closely related to e ($e = \text{vol}(\mathcal{E})/\text{vol}(\Theta)$), and then it can be interpreted as the reliability of the estimation process.

3.2 Robust synthesis with guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost

We have a set of plants $\mathcal{M}(\theta)$ parametrized by $\theta \in \Theta$. The objective is to find a single fixed controller K that ensures reasonable performance for a large variety of plants.

More precisely, consider the $\mathcal{H}_2/\mathcal{H}_\infty$ cost function (7) which reflects the performance of the closed-loop system. Our objective is to find, from an initial stabilizing controller K_{init} of the nominal system, a static output feedback K that reaches the best possible nominal performance and such that the worst-case performance, i.e. the guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost, is bounded by a pre-specified value g_c. For this purpose, we consider the following normalized cost function:

$$\psi(\theta,K) = \begin{cases} 1 & \text{if } A_c(\theta) \notin \mathcal{H} \\ \dfrac{J(\theta,K)}{1+J(\theta,K)} & \text{otherwise} \end{cases} \tag{14}$$

where $A_c(\theta)$ is the state matrix of the closed-loop system and $J(\theta,K)$ is given by (7). Finally, the problem that we want to solve can be formulated as follows. From a given K_{init}, with $\psi(\theta_0,K_{init}) < 1$, find K such that $\psi(\theta_0,K)$ is as small as possible, satisfying:

$$\psi(\theta,K) \leqslant \max_{\theta\in\Theta} \psi(\theta,K) \leqslant g_c \tag{15}$$

where g_c is the specified level of robust performance. For the same reasons as aforementioned, the exact resolution of this problem, is very difficult especially in the case of multiple parametric uncertainties. To overcome this drawback, in this paper we adopt a random optimization technique that approximately solves this problem.

As we will see further, the proposed solution to solve this problem requires an initial stabilizing controller and a procedure able to generate a sequence of controllers with a decreasing cost function. Of course, this can be done only under certain feasibility conditions. This is the reason why, we consider the class of problems satisfying the following assumptions:

Assumption 1. *Let $\mathcal{K}_S$ be the set of nominal stabilizing controllers and $\mathcal{K}$ the set of bounded controllers:*

$$\begin{aligned} \mathcal{K}_S &= \{K \in \mathbf{R}^{m\times p} : \psi(\theta_0,K) < 1\} \\ \mathcal{K} &= \left\{K \in \mathbf{R}^{m\times p} : \underline{k}_{ij} \leqslant [K]_{i,j} \leqslant \bar{k}_{ij}\ \forall i,j\right\} \end{aligned} \tag{16}$$

Note that the entries $[K]_{i,j}$ of the matrix K are constrained to lie in some known intervals $\underline{k}_{ij} \leqslant [K]_{i,j} \leqslant \bar{k}_{ij}$. The solution set $\mathcal{K} \cap \mathcal{K}_S$ has a nonempty interior.

Assumption 2. *For an initial stabilizing controller $K^{(1)}$, we define the set of possible γ-nominal-stabilizing controllers as follows:*

$$\mathcal{K}(\gamma) = \{K \in \mathcal{K}_S : \psi(\theta_0,K) \leqslant \gamma\} \tag{17}$$

where γ is such that $\gamma_{min} < \gamma \leqslant \gamma_{max}$, with $\gamma_{min} = \inf_{K\in\mathcal{K}_S} \psi(\theta_0,K)$ and $\gamma_{max} = \psi(\theta_0,K^{(1)})$. For a given $K_0 \in \mathcal{K}(\gamma)$, we consider the following set:

$$\mathcal{S} = \{K \in \mathcal{K}_S : K = K_0 + \Delta_K,\, \Delta_K \in \mathcal{K}_d\} \tag{18}$$

where $\mathcal{K}_d = \{\Delta_K \in \mathbf{R}^{m\times p} : |[\Delta_K]|_{i,j} \leqslant d\}$*. For any* γ *with* $\gamma_{min} < \gamma \leqslant \gamma_{max}$*, the set* $\mathcal{S} \cap \mathcal{K}(\gamma - \Delta_\gamma)$ *has a non empty interior.*

Assumption 1 guarantee that a stabilizing controller can be found in $\mathcal{K}$. Assumption 2 guarantee that in a vicinity of a given stabilizing controller, it is always possible to find a best new controller. Based on these feasibility assumptions, the following section presents random search procedures allowing the obtention of the aforementioned requirements.

3.2.1 Preliminary

The problem of finding a stabilizing controller which belongs to $\mathcal{K}$ can be solved using the random search procedure described in Algorithm 1.

Algorithm 1.

1. *Generate a sample* $K \in \mathcal{K}$ *according to a uniform probability distribution on the set* $\mathcal{K}$*.*
2. *If* $\psi(\theta_0,K) = 1$*, go to step 1, otherwise stop.*

The convergence of this algorithm is stated in the following theorem, which is proved in the appendix.

Theorem 2. *Let Assumption 1 be satisfied, then the Algorithm 1 converges, with high probability, to an element of* $\mathcal{K} \cap \mathcal{K}_S$*. Moreover, the number of iterations necessary to obtain a solution with a probability at least* $1-\delta$ *with* $\delta \in (0,1)$*, is given by* $\eta \geqslant \ln(\delta)/\ln(1-\xi)$*, where* $\xi = \Pr\{K \in \mathcal{K} \cap \mathcal{K}_S\}$*, (the notation* $\Pr\{.\}$ *represents the probability of the event* $\{.\}$*).*

Note that the convergence of this algorithm can be very slow if the probability ξ is too small. In this case, the problem of finding an initial stabilizing controller can be solved using the gradient-based method presented in appendix.

This random search procedure can be extended in order to find a stabilizing controller which in addition ensures a desired level of performance. Let γ_0 be a specified level of nominal performance. For a given stabilizing controller (which, for instance, can be found using Algorithm 1), the problem is now to find a γ-nominal-stabilizing controller, that is, a controller K such that $\psi(\theta_0,K) \leqslant \gamma_0 < 1$. A solution of this problem can be found using Algorithm 2.

Algorithm 2.

1. *Select a desired level of nominal performance* γ_0*, an initial controller* $K^{(1)} \in \mathcal{K}_s$*, a domain of exploration* $[-d,d]$ *with* $d > 0$*, a decreasing step* Δ_γ *with* $0 < \Delta_\gamma \ll 1$ *(e.g.* $\Delta_\gamma = 10^{-3}$*), and let* $i = 1$*.*
2. *Generate a sample* $\Delta_K^{(i)} \in \mathcal{K}_d = \{\Delta_K \in \mathbf{R}^{m\times p} : |[\Delta_K]|_{i,j} \leqslant d\}$*, according to a uniform probability distribution on* $\mathcal{K}_d$*, and such that* $K^{(i)} + \Delta_K^{(i)} \in \mathcal{K}_s$*.*

3. *If* $\psi(\theta_0,K^{(i)} + \Delta_K^{(i)}) \leqslant \psi(\theta_0,K^{(i)}) - \Delta_\gamma$ *let* $K^{(i+1)} = K^{(i)} + \Delta_K^{(i)}$ *and* $i = i+1$*, otherwise go to step 2.*

4. *If* $\psi(\theta_0,K^{(i)}) > \gamma_0$*, go to step 2, otherwise stop.*

Let us consider the i^{th} iteration of this algorithm, the principle is to find, by random search, in a neighbourhood of the matrix $K^{(i-1)}$ obtained at the preceding iteration, a new matrix $K^{(i)}$ such that $\psi(\theta_0,K^{(i)}) \leqslant \psi(\theta_0,K^{(i-1)}) - \Delta_\gamma$. This procedure is repeated until $\psi(\theta_0,K^{(i)}) \leqslant \gamma_0$, we obtain thus a γ-nominal-stabilizing controller. The convergence of algorithm 2 is stated in theorem 3, which is proved in appendix.

Theorem 3. *Let Assumption 2 be satisfied. For a given level of nominal performance* γ_0*, the sequence* $\{K^{(i)}\}$ *obtained using Algorithm 3 converges, with high probability, to a solution* $K \in \mathcal{K}(\gamma_0)$*. Moreover, the number of trials necessary to obtain a solution is bounded.*

3.2.2 The RBA algorithm

We consider now the problem of guaranteed quadratic cost (15). Let K be the nominal controller such that $\psi(\theta_0,K) \leqslant \gamma_0 < 1$, where γ_0 is the nominal level of performance. Let $g_c < 1$ be the specified (tolerate) quadratic cost. For the nominal controller, it is necessary to verify that the worst-case performance is such that $w_c(K) = \max_{\theta\in\Theta} \psi(\theta,K) \leqslant g_c$. If it exists $\theta \in \Theta$ such that $w_c(K) > g_c$, the controller must be rejected because the desired level of performance is not satisfied. Following Theorem 1, the worst-case performance $w_c(K)$ can be estimated by:

$$\hat{w}_c(K) = \max_{1\leqslant i\leqslant \eta} \psi(\theta^{(i)},K) \tag{19}$$

where η is the number of samples (13). If for the nominal controller K we have $\hat{w}_c(K) > g_c$, another controller must be found with a new nominal level of performance γ_n, such that $\gamma_0 < \gamma_n \leqslant g_c$. This procedure must be repeated until the desired $\mathcal{H}_2/\mathcal{H}_\infty$ cost is satisfied.

More precisely we want to find a nominal level of performance γ_n and the corresponding nominal controller K_n such that the constraint $\hat{w}_c(K_n) \leqslant g_c$ is satisfied. A solution of this problem can be found using the following random optimization procedure, based on a bisection method, called random bisection algorithm (RBA).

Algorithm 3 (RBA).

1. *For a given* ρ *close to 1 and a given* ϵ *close to zero, generate* η *i.i.d samples* $\theta^{(1)},\cdots,\theta^{(\eta)} \in \Theta$*, according to a uniform probability distribution on the set* Θ*, with* $\eta \geqslant \ln(1-\rho)/\ln(1-e)$*.*

2. *Select* $\gamma_I = \gamma_0$*,* $\gamma_S = g_c$ *and an accuracy* ϵ*.*

3. *Using Algorithm 1, find a stabilizing controller* K_{init}*.*

4. *Compute* $\gamma = (\gamma_I + \gamma_S)/2$.

5. *From* K_{init}, *and using the Algorithm 2, find a controller* K_n *such that* $\psi(\theta_0, K_n) \leqslant \gamma$.

6. *If* $\hat{w}_c(K_n) > \gamma_S$ *then* $\gamma_I = \gamma$ *otherwise* $\gamma_S = \gamma$.

7. *If* $\gamma_S - \gamma_I > 2\epsilon\gamma_I$ *goto step 4.*

8. *If* $\hat{w}_c(K_n) > g_c$ *goto step 2 otherwise stop.*

Theorem 4. *Let Assumptions 1 and 2 be satisfied, and let* γ_n *be the smallest nominal level of performance implying that the worst-case performance is less than or equal to the specified guaranteed cost* g_c. *Then, Algorithm 3 converges to a controller* K_n *satisfying* $\hat{w}_c(K_n) \leqslant g_c$ *and* $\psi(\theta_0, K_n) = \hat{\gamma}_n$, *where* $\hat{\gamma}_n$ *is an approximation of* γ_n *with a relative accuracy of* ϵ.

Proof. This algorithm requires:

- a stabilizing controller K_{init} (step 3),
- a γ-nominal-stabilizing controller K_n (step 5),
- an estimation of the worst-case performance $\hat{w}_c(K_n)$ (step 6).

As is easily seen from Theorems 1, 2 and 3, these requirements are satisfied. Now, assume that there exists a controller K_n for which the corresponding nominal performance γ_n is such that: $\gamma_0 \leqslant \gamma_n \leqslant g_c$ and $w_c(K_n) < g_c$. After N iterations, Algorithm 3 produces the following sequences: $\{\gamma_1, \gamma_2, \cdots, \gamma_N\}$ and $\{K_1, K_2, \cdots, K_N\}$, where γ_i $(i = 1, \cdots, N)$ is the nominal performance of the controller K_i at the i^{th} iteration. The bisection method ensures that $\gamma_I \leqslant \gamma_i \leqslant \gamma_S$, $\gamma_I \leqslant \gamma_n \leqslant \gamma_S$ and $\gamma_S - \gamma_I = 2^{-i}(g_c - \gamma_0)$, thus the quantity $|\gamma_i - \gamma_n|$, decreases as the number of iterations increases. On exit, $\gamma_N = \frac{\gamma_I + \gamma_S}{2} = \hat{\gamma}_n$ is guaranteed to approximate γ_n within a relative accuracy of ϵ, that is $|(\gamma_I + \gamma_S)/2 - \gamma_n| \leqslant \epsilon\gamma_n$. Since $\psi(\theta_0, K_N) = \hat{\gamma}_n$, it follows that the corresponding controller K_N satisfies $\hat{w}_c(K_N) \leqslant g_c$ because $\hat{\gamma}_n$ is an approximation of the smallest nominal level of performance satisfying the guaranteed cost g_c.

4 Numerical examples

In this section, the effectiveness of the presented method to deal with a large number of uncertain parameters is tested on two numerical examples. The first one gives a comparison with the GA-LMI method described in [25]. The second example is dedicated to the robust synthesis of a SOF controller for the lateral motion of an aircraft. All the experiments are performed using a 1.2 Ghz Celeron personal computer.

4.1 Robust design with guaranteed $\mathcal{H}_2/\mathcal{H}_\infty$ cost

As in [25], consider the three-state unstable plant with the following equation:

$$\dot{x}(t) = \begin{bmatrix} 0 & 10 & 2 \\ -1 & 1+\Delta a_{22} & 0 \\ 0+\Delta a_{31} & 2 & -5+\Delta a_{33} \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \\ 0+\Delta b_{31} \end{bmatrix} u(t) + \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} w(t)$$

$$y(t) = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} x(t)$$

$$z_2(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u(t)$$

$$z_\infty(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

where $|\Delta a_{22}| \leqslant 0.5$, $|\Delta a_{31}| \leqslant 0.8$, $|\Delta a_{33}| \leqslant 1$, $|\Delta b_{31}| \leqslant 0.5$. Using Algorithm 3 with: $g_c = 0.8$, $\alpha = \beta = 1$, $|[K]_{i,j}| \leqslant 5$, $\eta = 1200$ ($\rho = 0.995$, $e = 0.005$), $\epsilon = 0.001$, $d = 0.025$ and $\Delta_\gamma = 0.001$, we obtain $K = 4.889$. The estimate of the worst-case performance is then $\hat{w}_c(K) = 0.6349$.

Performance	Method of [25], GA-LMI (Pentium 4, 1.8 Ghz CPU)	Proposed method RBA (Celeron(TM), 1.2 Ghz CPU)
K	4.5398	4.889
$\|\|G_2\|\|_2^2$	1.488	0.9886
$\|\|G_\infty\|\|_\infty^2$	2.024	0.7502
Searching time	77.11 sec	84.64 sec

Tab. 1. Comparison of the proposed method with the GA-LMI.

A comparison between the proposed method (RBA), and the GA-LMI approach [25], is shown in Table 1. Clearly, this comparison shows the pessimistic results obtained with LMI approach. In addition, since the computer used is less powerful than a 1.8 Ghz Pentium 4, one can conclude that the searching time of the RBA is comparable with GA-LMI method.

For practical use of the algorithm, Table 1 summarize the influence of the parameters ϵ, d and η on the accuracy and the searching time.

Parameters	Accuracy	Searching time
$\epsilon \searrow$	of $\hat{\gamma}_n \nearrow$	$\nearrow$
$d \searrow$	-	$\nearrow$
$\eta \nearrow$	of $\hat{w}_c \nearrow$	$\nearrow$

Tab. 1. Effect of the parameters on accuracy and searching time.
($\nearrow$: increase, $\searrow$: decrease, - : no effect)

4.2 Robust design with guaranteed quadratic cost

In this section, we consider the synthesis of a robust static output feedback controller for the lateral motion of an aircraft. The multivariable model consists of four states, two inputs and three outputs. The state space equation is given by [1]:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & L_p & L_\beta & L_r \\ g/V & 0 & Y_\beta & -1 \\ (g/V)N_{\dot\beta} & N_p & N_\beta + N_{\dot\beta}Y_\beta & N_r - N_{\dot\beta} \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ 0 & -3.91 \\ 0.035 & 0 \\ -2.53 & 0.31 \end{bmatrix} u(t)$$

The inputs u_1 and u_2 represent the rudder and aileron deflections, while the state variables x_1, x_2, x_3 and x_4 are the bank angle, its derivative, the side-slip angle, and the yaw rate, respectively. The output equation (measured variables) is given by:

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} x(t)$$

Let θ be the vector of uncertain parameters, and θ_0 its nominal value:

$$\theta = [L_p \;\; L_\beta \;\; L_r \;\; g/V \;\; Y_\beta \;\; N_{\dot\beta} \;\; N_p \;\; N_\beta \;\; N_r]^T$$

$$\theta_0 = [-2.93 \;\; -4.75 \;\; 0.78 \;\; 0.086 \;\; -0.11 \;\; 0.1 \;\; -0.042 \;\; 2.601 \;\; -0.29]^T$$

These uncertain parameters are allowed to vary 85% around the nominal values. To show the flexibility of the proposed approach, we consider here the resolution of the robust static output feedback problem with guaranteed quadratic cost, i.e. we want to find K such that:

$$J(\theta,K) = \int_0^\infty \left[x(t)^T Q x(t) + u(t)^T R u(t)\right] dt \leqslant g_c \tag{20}$$

where the weighting factors Q and R are positive symmetric matrices. This problem can be solved by the proposed method, without modification except obviously for the computation of the criterion. Using Algorithm 3 with: $R = I_2$, $Q = I_4$, $g_c = 0.97$, $|[K]_{i,j}| \leqslant 15$, $\eta = 1200$ ($\rho = 0.995$, $e = 0.005$), $\epsilon = 0.001$, $d = 0.025$ and $\Delta_\gamma = 0.001$, we obtain the following controller:

$$K = \begin{bmatrix} 1.1682 & 6.9827 & -10.1368 \\ -1.0936 & -1.8573 & 3.5859 \end{bmatrix}$$

The estimate of the worst-case performance is then $\hat{w}_c(K) = 0.962$. For comparison with the work [22], we consider now the robust stabilization by state feedback controller (i.e. $C = I$). The vector of parameters is allowed to vary 85% around its nominal value. Using Algorithm 3 and letting $Q = 0.01I$ as in [22], we obtain the following robust controller:

$$K = \begin{bmatrix} -4.8142 & -12.5158 & -2.0299 & -7.8334 \\ -6.1027 & -15.6165 & -2.7077 & -9.3598 \end{bmatrix}$$

The worst-case performance is then $0.903 < \gamma_{gc}$. In [22], a probabilistic LMI approach was used to solve this problem and it was concluded that the largest uncertainty for which there exists a robust controller is about 15%. This comparison shows that the method presented in this paper is less conservative.

5 Conclusion

The intrinsic complexity of the static output feedback controller (SOF) is increased by considering the inevitable multiple parametric uncertainties in the process model. Consequently, there is no systematic procedure, both analytic and numeric to find a satisfying static output feedback controller. In these conditions, heuristic approaches seem a good alternative for this task. In this framework, a random optimization technique, based on a bisection method, has been proposed to find a robust static output feedback controller in the context of multiple, possibly non linear, parametric uncertainties. Numerical examples, have been presented which demonstrate the flexibility and the practical applicability of the proposed design method.

A possible direction for future work is to improve the efficiency of the proposed algorithms. For example, the convergence properties of Algorithm 2 can be improved by including a parabolic line search method. Another future work is to extend the proposed method in order to solve the problem of finding a robust SOF-controller that realizes the best compromise between $\mathcal{H}_2$ and $\mathcal{H}_\infty$ performance, i.e. to find a Pareto-optimal controller.

Références

[1] B. D. O. Anderson and J. B. Moore. *Optimal control, linear quadratic methods.* Prentice-Hall, Englewood Cliffs, NJ, 1989.

[2] D. Arzelier and D. Peaucelle. An iterative method for mixed h_2/h_∞ synthesis via static output-feedback. *41st IEEE Conference on Decision and Control (CDC'2002), Las Vegas (USA)*, pages 3464–3469, 2002.

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

[4] R. E. Benton and D. Jr. Smith. Static output feedback stabilization with prescribed degree of stability. *IEEE Transactions on Automatic Control*, 43(10):1493–1496, 1998.

[5] D. Bernstein. Some open problems in matrix theory arising in linear systems and control. *Linear Algebra and its Applications*, 162-164:409–432, 1992.

[6] Y. Cao, J. Lam, and Y.-X.M. Sun. Static output feedback stabilization: An ilmi approach. *Automatica*, 34(12):1641–1645, 1998.

[7] M. Chilali, P. Gahinet, and P. Apkarian. Robust pole-placement in lmi regions. *IEEE Transactions on Automatic Control*, 44(12):2257–2270, 1999.

[8] E.J. Davison and S.H. Wang. On pole assignment in linear multivariable systems using output feedback. *IEEE Transactions on Automatic Control*, (8):516–518, 1975.

[9] J. Doyle, B. Francis, and A. Tannenbaum. *Feedback Control Theory.* Macmillan Publishing Co., 1990.

[10] G. Garcia, B. Pradin, S. Tabouriech, and F. Zeng. Robust stabilization and guaranteed cost control for discrete-time linear systems by static output feedback. *Automatica*, 39(9):1635–1641, 2003.

[11] J. C. Geromel, C. C. Souza, and R. E. Skelton. Static output feedback controllers: stability and convexity. *IEEE Transactions on Automatic Control*, 43:120–125, 1998.

[12] L. El Ghaoui, F. Oustry, and M. AitRami. A cone complementarity linearization algorithm for static output-feedback and related problems. *IEEE Transactions on Automatic Control*, 42:1171–1176, 1997.

[13] G. Gu. On the existence of linear optimal control with output feedback. *SIAM Journal Control and Optimization*, 28:711–719, 1990.

[14] P. Hagander and B. Bernhardsson. On the notion of strong stabilization. *IEEE Transactions on Automatic Control*, pages 927–929, 1990.

[15] D. Henrion, S. Tabouriech, and G. Garcia. Output feedback robust stabilization of uncertain linear systems with saturating controls. an lmi approach. *IEEE Transactions on Automatic Control*, 44(11):2230–2237, 1999.

[16] H. Kimura. A further result on the problem of pole assignment by output feedback. *IEEE Transactions on Automatic Control*, 25(3):458–463, 1977.

[17] V. Kucera and C. E. De Souza. A necessary and sufficient condition for output feedback stabilization. *Automatica*, 31(9):1357–1359, 1995.

[18] V. B. Larin. Stabilization of the system by static output feedback. *Applied and Computational Mathematics*, 2(1):2–12, 2003.

[19] F. Leibfritz. An lmi-based algorithm for designing suboptimal static h2/hinf output-feedback controllers. *SIAM J. Control. Optim.*, 39(6):1711–1735, 2001.

[20] J. Matyas. Random optimization. *Automation and Remote control, translated from Avtomatika i Telemekhanika*, 26(2):246–253, 1965.

[21] D. Peaucelle and D. Arzelier. An efficient numerical solution for h_2 static output feedback synthesis. *European Control Conference (ECC'01), Porto (Portugal)*, pages 3800–3805, 2001.

[22] B.T. Polyak and R. Tempo. Probabilistic robust design with linear quadratic regulators. *Systems & Control Letter*, 43:343–353, 2001.

[23] E. Feron S. Boyd, L. El Ghaoui and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* Studies in Applied Mathematics. SIAM, Philadelphia, PA, 1994.

[24] J. C. Spall. *Introduction to stochastic search and optimization.* John Wiley and Sons, 2003.

[25] C. C. Sun, H.Y. Chung, and W. J. Chang. h_2/h_∞ robust static output feedback contol design via mixed genetic algorithm and linear matrix inequalities. *Journal of Dynamic Systems, Measurement and Control*, 127:715–722, 2005.

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

[27] R. Tempo, G. Calafiore, and F. Dabbene. *Randomized algorithms for analysis and control of uncertain systems.* Springer-Verlag, 2004.

[28] R. Toscano. A simple method to find a robust output feedback controller by random search approach. *ISA Transactions*, 45(1):35–44, 2006.

[29] R. Toscano and P. Lyonnet. Stabilization of systems by static output feddback via heuristic kalman algorithm. *Journal of Applied and Computational Mathematics*, 5:1–12, 2006.

[30] A. Trofino-Neto and V. Kucera. Stabilization via static output feedback. *IEEE Transaction on Automatic Control*, pages 764–765, 1993.

[31] J. G. VanAntwerp and R. D. Braatz. A tutorial on linear and bilinear matrix inequalities. *Journal of Process Control*, 10:363–385, 2000.

[32] X. A. Wang. Grassmannian, central projection, and output feedback pole assignment of linear systems. *IEEE Transaction on Automatic Control*, 41(6):786–794, 1996.

[33] D. Youla, J. Bongiorno, and C. Lu. Single loop feedback stabilization of linear multivariable dynamical systems. *Automatica*, 10:159–173, 1974.

Appendix

A1. Proof of theorem 2

Let us consider n iterations of the Algorithm 2, the probability so that $K \notin \mathcal{K} \cap \mathcal{K}_S$ is given by the binomial probability distribution

$$P\{K \notin \mathcal{K} \cap \mathcal{K}_S\} = \left[\frac{n!}{r!(n-r)!}\xi^r(1-\xi)^{n-r}\right]_{r=0} = (1-\xi)^n \qquad (21)$$

where r is the number of success (i.e. the number of times that $K \in \mathcal{K} \cap \mathcal{K}_S$) and ξ the probability of success i.e. $\xi = \Pr\{K \in \mathcal{K} \cap \mathcal{K}_S\} = \mathrm{vol}(\mathcal{K} \cap \mathcal{K}_S)/\mathrm{vol}(\mathcal{K})$. The notation vol(.) stands for the volume of the set (.). By Assumption 1 we have $\xi > 0$, it is then clear that $\lim_{n\to\infty}(1-\xi)^n = 0$, the algorithm converges then, with high probability, to a solution. From (21), the number of iterations necessary to obtain a solution with a probability at least $1-\delta$ is such that $(1-\xi)^\eta \leqslant \delta$ which gives $\eta \geqslant \ln(\delta)/\ln(1-\xi) < \infty$.

A2. Proof of theorem 3

For a specified decreasing step Δ_γ, the maximum number of success necessary to obtain a controller K such that $\psi(\theta_0, K) \leqslant \gamma_0$, is given by:

$$N = \left\lceil \frac{\psi(\theta_0, K^{(1)}) - \gamma_0}{\Delta_\gamma} \right\rceil \qquad (22)$$

Where $K^{(1)}$ is the initial stabilizing controller of the sequence $\{K^{(i)}\}$, and $\lceil . \rceil$ is the minimum integer greater or equal of the argument. Our objective is now to show that each success is obtained after a finite number of trials with high probability. Let $K^{(i)}$ be the controller obtained after a number of i iterations (i.e. after i success), we have $\psi(\theta_0, K^{(i)}) = \gamma^{(i)}$. Consider the following sequence:

Sequence 1.

1. *Generate a sample* $\Delta_K \in \mathcal{K}_d = \{\Delta_K \in \mathbf{R}^{m\times p} : |[\Delta_K]|_{i,j} \leqslant d\}$*, according to a uniform probability distribution on* $\mathcal{K}_d$*, and such that* $K^{(i)} + \Delta_K \in \mathcal{K}_s$*.*

2. *If* $\psi(\theta_0,K^{(i)}+\Delta_K) \leqslant \gamma^{(i)} - \Delta_\gamma$ *then stop, otherwise go to step 1.*

when this sequence is stopped (i.e. when we have a success), the cost function decreases at least of Δ_γ.

Consider n iterations of this sequence, the probability so that $K^{(i)}+\Delta_K \notin \mathcal{K}(\gamma^{(i)}-\Delta_\gamma)$ is given by the binomial probability distribution:

$$\begin{aligned}\Pr\left\{(K^{(i)}+\Delta_K) \notin \mathcal{K}(\gamma^{(i)}-\Delta_\gamma)\right\} &= \left[\frac{n!}{r!(n-r)!}(\xi^{(i)})^r(1-\xi^{(i)})^{n-r}\right]_{r=0} \\ &= (1-\xi^{(i)})^n \qquad (23)\end{aligned}$$

where r is the number of success and $\xi^{(i)}$ the probability to find $K^{(i)}+\Delta_K$ such that $\psi(\theta_0,K^{(i)}+\Delta_K) \leqslant \gamma^{(i)}-\Delta_\gamma$, that is:

$$\begin{aligned}\xi^{(i)} &= \Pr\left\{(K^{(i)}+\Delta_K) \in \mathcal{K}(\gamma^{(i)}-\Delta_\gamma)\right\} \\ &= \frac{\mathrm{vol}\left(\mathcal{S}_i \cap \mathcal{K}(\gamma^{(i)}-\Delta_\gamma)\right)}{\mathrm{vol}(\mathcal{S}_i)} \qquad (24)\end{aligned}$$

where vol(.) is the volume of the set (.), and $\mathcal{S}_i$ is the searching set at the step i, that is:

$$\mathcal{S}_i = \{K \in \mathcal{K}_S : K = K^{(i)}+\Delta_K,\, |[\Delta_K]_{i,j}| \leqslant d\} \qquad (25)$$

By Assumption 2, we have $\xi^{(i)} > 0$, for $i = 1,2,\ldots,N$, it is then clear that $\lim_{n\to\infty}(1-\xi^{(i)})^n = 0$, thus the Sequence 1 is stopped with a probability one as $n \to \infty$. From (23), the number of trials τ_i, at the iteration number i, necessary to obtain $\psi(\theta_0,K^{(i)}+\Delta_K) \leqslant \gamma^{(i)}-\Delta_\gamma$ with a probability at least $1-\delta$, with $\delta \in (0,1)$, is such that $(1-\xi^{(i)})^{\tau_i} \leqslant \delta$ which gives:

$$\tau_i \geqslant \ln(\delta)/\ln(1-\xi^{(i)}) < \infty \qquad (26)$$

the total number of trials τ, necessary to obtain a solution is then given by $\tau = \sum_{i=1}^{N}\tau_i$, consequently we have:

$$\tau < N\tau_{max}, \quad \text{with:} \quad \tau_{max} = \max_{1\leqslant i\leqslant N}\tau_i \qquad (27)$$

from (26) we have:

$$\tau_{max} \geqslant \frac{\ln(\delta)}{\ln\left(1-\min\limits_{1\leqslant i\leqslant N}\xi^{(i)}\right)} < \infty \qquad (28)$$

it follows that there exists a worst-case probability ξ_{wc} such that:

$$\tau_{max} = \frac{\ln(\delta)}{\ln(1-\xi_{wc})} < \infty, \quad \text{with:} \quad 0 < \xi_{wc} \leqslant \min_{1\leqslant i\leqslant N}\xi^{(i)} \qquad (29)$$

consequently, from (27) and (29) we have:

$$\tau < \frac{\ln(\delta^N)}{\ln(1-\xi_{wc})} < \infty \tag{30}$$

Of course, the worst-case probability ξ_{wc} is unknown, thus the total number of trials cannot be evaluated in advance. However, the result (30) shows that the Algorithm 2 leads to a solution in a finite number of trials. Indeed, we have:

$$\tau < \frac{\ln(\delta^N)}{\ln(1-\xi_{wc})} < \beta z^2 < \infty \tag{31}$$

where β is the constant $-\ln(\delta^N)$ and $z = 1/\xi_{wc}$.

A3. The initial stabilizing controller problem

The implementation of the Random Bisection Algorithm requires an initial stabilizing controller. As seen previously, this can be done using Algorithm 1. In the case where this algorithm does not converge sufficiently fast, this means that the probability ξ is too small. To solve this problem it is proposed a gradient-based method inspired to the work of Vladimir Larin [18]. Consider the nominal system described by:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) \\ y(t) = Cx(t) \end{cases} \tag{32}$$

and the following modified system:

$$\begin{cases} \dot{x}(t) = (A + \mu I)x(t) + qBu(t) \\ y(t) = Cx(t) \end{cases} \tag{33}$$

where the scalar μ is chosen such that $(A + \mu I) \in \mathcal{H}$. The factor q will be adjusted in order to obtain $K \in \mathcal{K}$. The main idea is then to find a stabilizing SOF-controller of the modified system which stabilizes also the initial system. This problem can be solved iteratively by decreasing the parameter μ. More precisely, consider the following cost function:

$$J = \int_0^\infty x(t)^T x(t) dt \tag{34}$$

The gradient of this function with respect to K is given by (see [18]):

$$\frac{\partial J}{\partial K} = 2qB^T P^2 C^T, \quad A_c^T P + PA_c + I = 0 \tag{35}$$

with $A_c = A + \mu I + qBKC$. The principle is then to compute via gradient-based method a stabilizing matrix K with a decreasing factor μ. This procedure is stopped when $(A + BKC) \in \mathcal{H}$. Note that the choice of μ such that $(A + \mu I) \in \mathcal{H}$ makes trivial the choice of the initial matrix K required by the gradient-based method. Indeed, since $(A + \mu I)$ is Hurwitz, it is possible to accept as initial value $K = 0$. The following algorithm is based on this principle.

Algorithm 4.

1. *Select* μ *such that* $(A + \mu I) \in \mathcal{H}$, $q = 1$ *(default),* $K = 0$.
2. *Using a gradient-based method with (35), compute a new stabilizing matrix* K.
3. *If* $(A + BKC) \notin \mathcal{H}$, $\mu = \mu/2$ *goto step 2, otherwise stop.*

Step 2 can be performed using the Newton-Gauss method i.e.:

$$K_{k+1} = K_k - \rho_k \left\{ \left[\frac{\partial J}{\partial K} \left(\frac{\partial J}{\partial K} \right)^T \right]^{-1} \frac{\partial J}{\partial K} \right\}_{K=K_k} \tag{36}$$

where the factor ρ_k can be chosen constant, or can be adjusted at each iteration by solving the following problem:

$$\rho_k = \arg \min_{\rho \geqslant 0,\ A_c \in \mathcal{H}} \|T\|_2 \|T^{-1}\|_2 \tag{37}$$

where T is the eigenvector matrix of A_c. In other words, the factor ρ_k is determined in order to minimize the sensitivity of the closed-loop state matrix to unstructured perturbations [28].

Example. Consider the problem of finding an initial stabilizing controller for the nominal system given in section 4.2. Using Algorithm 4 with $\mu = -5$, $q = 1$, $\rho_k = 0.1$ gives the following matrix:

$$K = \begin{bmatrix} 0.2445 & -0.3954 & 2.2139 \\ 0.8750 & -2.7858 & 0.4883 \end{bmatrix}$$

in only one iteration. Assume now that it is required to find a stabilizing matrix K such that:

$$K \in \mathcal{K} = \left\{ K \in \mathbf{R}^{m \times p} : -1 \leqslant [K]_{i,j} \leqslant 1\ \forall i,j \right\} \tag{38}$$

if there exists one such matrix, it can be obtained by increasing q until $K \in \mathcal{K}$. Using Algorithm 4 with $\mu = -5$, $q = 3$, $\rho_k = 0.1$ gives the following matrix:

$$K = \begin{bmatrix} 0.0815 & -0.1318 & 0.7380 \\ 0.2917 & -0.9286 & 0.1628 \end{bmatrix}$$

in only one iteration. For comparison, using Algorithm 1, gives the following matrix $K \in \mathcal{K}$:

$$K = \begin{bmatrix} 0.2662 & 0.5548 & 0.0184 \\ 0.9605 & 0.2369 & -0.7146 \end{bmatrix}$$

in 21 iterations.