International Journal of Systems Science
volume 35, number 1, 15 January 2004, pages 55–67

Taylor & Francis
Taylor & Francis Group

Estimation of Pareto sets in the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem

RICARDO H. C. TAKAHASHI†, REINALDO M. PALHARES‡*, DANIEL A. DUTRA† and LEILA P. S. GONÇALVES§

Multiobjective design problems give rise to a well-defined object: the Pareto-set. This paper proposes some verifiable conditions that are applicable to sets with finite number of elements, to corroborate *or* falsify *the* hypothesis *of the elements of that set being samples of the Pareto set. These conditions lead to several generic criteria that can be employed in the evaluation of algorithms as multiobjective optimization mechanisms. A conceptual multiobjective genetic algorithm is proposed, exploiting the group properties of the intermediate Pareto-set estimates to generate a consistent final estimate. The methodology is applied to the case of a mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control design. Recent dedicated multiobjective algorithms are evaluated under the proposed methodology, and it is shown that they can generate sub-optimal or non-consistent solution sets. It is shown that the proposed synthesis methodology can lead to both enhanced objectives and enhanced consistency in the Pareto-set estimate.*

1. Introduction

The statement of system design problems in terms of multiobjective optimization formulations is known to be more suitable to describe real-life design desiderata than single-objective formulations (Takahashi *et al.* 2000). In abstract contexts, multiobjective design methods can be seen as a way of generating design alternatives that vary along some sets of the solution space that are known to have 'good solutions'. In fact, multiobjective solutions can have the definable property of producing large enhancements in some objectives for small debasements in others (Chankong and Haimes 1983).

Consider a generic design problem:

Problem 1 (design problem): *Consider the design parameter vector $x \in \mathbb{R}^n$, the feasible solution set $\mathcal{X}_f \subset \mathbb{R}^n$, and the design objective vector $f(\cdot) : \mathbb{R}^n \mapsto \mathbb{R}^m$. Consider that there is the connotation that $f_i(a) < f_i(b)$ implies that solution a is better than solution b in the objective $f_i(\cdot)$. Find a* solution vector *$x = x^* \in \mathcal{X}_f$ such that all $f_i(x^*), i = 1, \ldots, m$ attain the smallest possible values.* □

The problem of finding optimal design solutions can be posed inside the stream of single-objective optimization methods as:

Problem 2 (single-objective setting): *Given problem 1, find $x^* \in \mathbb{R}^n$ such that:*

$$x^* = \arg\min f_i(x)$$
$$\textit{subject to:} \quad \begin{cases} x \in \mathcal{X}_f \\ f_j(x) \leq \gamma_j, \quad (j = 1, \ldots, m,\ j \neq i). \end{cases} \tag{1}$$

□

Problem 1 does not uniquely define a corresponding problem 2. In this formulation, objective f_i has been arbitrarilly chosen as the reference, and the others have been taken as constraints.

Another point of view can be adopted: the multiobjective approach. This approach works over a solution set,

Received 31 October 2001. Revised 17 October 2003. Accepted 3 December 2003.

† Department of Mathematics, Federal Univesity of Minas Gerais, Av. Antônio Carlos 6627, Pampulha, 31270-010, Belo Horizonte—MG—Brazil.

‡ Department of Electronics Engineering, Federal University of Minas Gerais, Av. Antônio Carlos 6627, Pampulha, 31270-010, Belo Horizonte—MG—Brazil.

§ Tecnometal Engenharia e Construçẽs Mecânicas Ltda, Av. das Naçôes, 3801—Distrito Industrial de Vespasiano, 33200-000, Vespasiano—MG—Brazil.

* To whom correspondence should be addressed.
e-mail: palhares@cpdee.ufmg.br

International Journal of Systems Science ISSN 0020–7721 print/ISSN 1464–5319 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/00207720310001657081

instead of working over a single solution. In this way, a well-defined object emerges from the definition of problem 1: the *Pareto set*.

Definition 1 [Pareto set (1)]: *Define $\mathcal{S}_i$ as the set of all solutions to problem 2 when objective f_i is fixed as the reference, and vector $\gamma \in \mathbb{R}^{m-1}$ assumes all possible values. The set $\mathcal{P}$ defined by:*

$$\mathcal{P} = \mathcal{S}_1 \cap \mathcal{S}_2 \cap \ldots \cap \mathcal{S}_m \tag{2}$$

is called the efficient solution set, *or the* Pareto set *of problem 1.* □

This set should be recognized as the set of all 'reasonable solutions' for problem 1. An equivalent definition for the same set can be stated in a 'relative' fashion:

Definition 2 [Pareto set (2)]: *Consider $\mathcal{X} \subset \mathbb{R}^n$ and $f : \mathbb{R}^n \mapsto \mathbb{R}^m$. Then:*

$$\mathcal{P} \triangleq \{x^* \in \mathcal{X}_f \mid \nexists x \in \mathcal{X}_f \text{ such that } f(x) \leq f(x^*) \text{ and } f(x) \neq f(x^*)\} \tag{3}$$

defines the set $\mathcal{P}$, that is called the Pareto set *or efficient solution set of Problem* 1. □

The equivalence of definitions 1 and 2 can be demonstrated (Chankong and Haimes 1983). Although these definitions lead to the same solution set $\mathcal{P}$, the first is conceptually supported in a series of independent solutions of a single-objective optimization mechanism. The second, instead, links any solution belonging to the set $\mathcal{P}$ to all others. The group properties that characterize Pareto sets are emphasized with this definition, which will be employed, therefore, as the basis for the development of analysis and synthesis tools in this work. With this definition, the design problem becomes:

Problem 3 (multiobjective setting): *Given problem 1, find the set $\mathcal{P}$.* □

Like the optimal solution of single-objective optimization problems, the Pareto set of multiobjective optimization problems is a conceptual object that should be approached numerically via suitable algorithms. In single-objective optimization, some optimality criteria (such as Kuhn–Tucker conditions) should be employed for testing each single candidate solution. In multiobjective optimization, similar criteria for testing single points that are candidates for belonging to the Pareto set also exist. However, there are also group criteria that can be employed in the whole set of points that are candidates for belonging to set $\mathcal{P}$.

This paper deals with the construction of such group criteria that are intended to be applied to sets of candidate solutions of multiobjective problems. With these criteria, existing algorithm solution sets can be evaluated and compared with the solution sets of other algorithms.

A genetic algorithm is built on the basis of these criteria, and it is shown to perform better than algorithms based on single-point criteria.

As a case study, this paper approaches one of the most traditional multiobjective design problems that appear in control theory literature: the $\mathcal{H}_2/\mathcal{H}_\infty$ problem (Khargonekar and Rotea 1991a, 1991b, Scherer 1995, Scherer *et al.* 1997, Takahashi *et al.* 1997), taken here in the contexts of state-feedback and static output feedback. An extra reference about multiobjective design techniques in control theory can be found in Dorato (1991). Recently developed algorithms for this problem (Cao *et al.* 1998, Shimomura and Fujii 2000) that employ formulations with the form of Problem 2 are analysed, and their outputs are shown to not be consistent from the viewpoint of the group properties of a Pareto set.

The multiobjective genetic algorithm proposed here leads to solution sets that enhance the aforementioned algorithm solution sets. However, the proposed methodology is not restricted to this particular problem. It is known that several control objectives have been stated by means of matrix inequalities like Riccati or LMIs (Linear Matrix Inequalities). These formulations lead to sufficient descriptions of these objectives, which means that, in most of cases, there is a gap between necessity and sufficiency. The reduction of this gap has been exploited in two research lines: Lyapunov parameter-dependent formulations (Leite and Peres 2003) and LMI relaxation of variables (de Oliveira *et al.* 2002). Nevertheless, these formulations have not completely filled that gap in the context of multiobjective design problems (Scherer *et al.* 1997). The approach proposed in this paper is the usage of the multiobjective genetic algorithm to search for better solutions inside that gap. A further discussion on this issue can be found in Fleming and Purshouse (2002).

2. Multiobjective analysis

For the purpose of discussing the Pareto-set properties that are relevant here, figure 1 presents typical structures that emerge after the execution of any computation to estimate some points belonging to $\mathcal{P}$. A problem with only two objectives is considered, for the purpose of visualization. The analysis, however, is valid for any number of objective functions.

The 'exact' Pareto set $\mathcal{P}$ is the continuous curve in figure 1. This 'exact' set, however, is in principle unavailable, for generic functionals f_1 and f_2, in the sense that, even if one has a set of points that belong

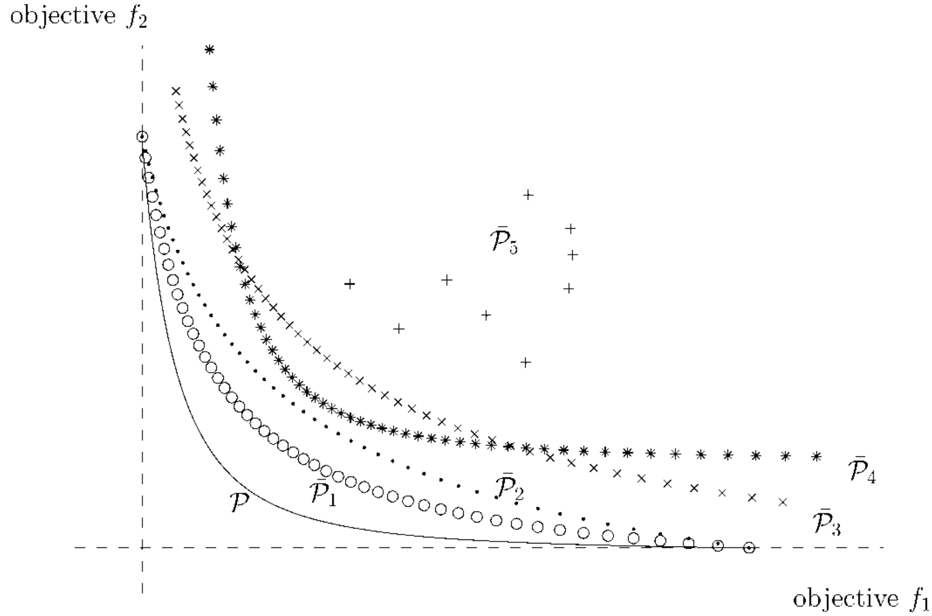

Figure 1. Typical computational estimates of the Pareto-set $\mathcal{P}$, in the space of objectives. Exact Pareto-set $\mathcal{P}$ (continuous line); estimated set $\bar{\mathcal{P}}_1$ (○); estimated set $\bar{\mathcal{P}}_2$ (·); estimated set $\bar{\mathcal{P}}_3$ (×); estimated set $\bar{\mathcal{P}}_4$ (∗); estimated set $\bar{\mathcal{P}}_5$ (+). Dashed lines denote the optimal values of the objectives f_1 and f_2.

to it, there have been no means, up to now, for proving that fact. To characterize solutions that are 'likely' to belong to $\mathcal{P}$ (the *Pareto candidates*), another relation is defined here, employing only points that are 'available':

Definition 3 (Pareto candidate set): *Let $f(\cdot)$ be a vector of objectives and $\mathcal{K} \subset \mathrm{Dom}(f(\cdot))$ be a set with a finite number of elements: $\mathcal{K} = \{x_1, \ldots, x_v\}$, where Dom$(f(\cdot))$ denotes the domain of $f(\cdot)$. The set $\Psi(\mathcal{K})$, defined by:*

$$\Psi(\mathcal{K}) \triangleq \{x_p \in \mathcal{K} |\ \nexists\ x_i \in \mathcal{K} \text{ such that } f(x_i) \le f(x_p) \text{ and } f(x_i) \ne f(x_p)\} \quad (4)$$

is called the Pareto candidate set, *associated with the 'sample set'* $\mathcal{K}$. □

In fact, owing to the unavailability of the set $\mathcal{P}$, the characterization of solution sets $\Psi(\cdot)$ as *Pareto candidates* is performed with *falsification* procedures, which can show that some sets are not candidates but cannot ever show that any set is in fact a Pareto set. This is the role of the *Pareto candidate set* concept.

2.1. *Consistency*

Given any set $\mathcal{X}$, it can be considered as a *Pareto candidate* only if $\Psi(\mathcal{X}) = \mathcal{X}$. If this occurs, the set is said to be *auto-consistent*. Otherwise, the possibility of $\mathcal{X}$ being a subset of the Pareto set $\mathcal{P}$ is *falsified*.

2.2. *Ordering and dominance*

Given two sets, $\mathcal{X}_1$ and $\mathcal{X}_2$, ordering relations between these sets are defined:

$$\mathcal{X}_1 \prec \mathcal{X}_2 \Leftrightarrow \{\Psi(\mathcal{X}_1 \cup \mathcal{X}_2) \supset \mathcal{X}_1 \text{ and } \Psi(\mathcal{X}_1 \cup \mathcal{X}_2) \not\supset \mathcal{X}_2\}$$
$$\mathcal{X}_1 \preceq \mathcal{X}2 \Leftrightarrow \{\Psi(\mathcal{X}_1 \cup \mathcal{X}_2) \supset \mathcal{X}_1\}. \quad (5)$$

In the case of $\mathcal{X}_1 \prec \mathcal{X}_2$, the possibility of set $\mathcal{X}_2$ being a subset of the Pareto set $\mathcal{P}$ is *falsified*, while the set $\mathcal{X}_1$ remains being a *Pareto candidate*. In this case, $\mathcal{X}_1$ is said to *dominate* $\mathcal{X}_2$. There are two possibilities of *non-dominance*: if both relations $\mathcal{X}_1 \preceq \mathcal{X}_2$ and $\mathcal{X}_2 \preceq \mathcal{X}_1$ do not hold, then both sets become *falsified* as *Pareto candidates*; and if both relations $\mathcal{X}_1 \preceq \mathcal{X}_2$ and $\mathcal{X}_2 \preceq \mathcal{X}_1$ hold, both sets continue to be *Pareto candidates*.

With these concepts, figure 1 is analysed. The set of estimates $\bar{\mathcal{P}}_5$ is not *auto-consistent* and therefore is *falsified* as a *Pareto candidate*. The sets $\bar{\mathcal{P}}_1$ to $\bar{\mathcal{P}}_4$ are each one auto-consistent and could be considered, therefore, as *Pareto candidates* if only one of them were available. There is an ordering relation among these Pareto-set estimates:

$$\mathcal{P} \prec \bar{\mathcal{P}}_1 \prec \bar{\mathcal{P}}_2 \prec \{\bar{\mathcal{P}}_3, \bar{\mathcal{P}}_4\} \prec \bar{\mathcal{P}}_5. \quad (6)$$

This ordering corresponds to a *dominance* ordering. If all these sets were available, the only *Pareto candidate* would be $\bar{\mathcal{P}}_1$, since:

$$\bar{\mathcal{P}}_1 = \Psi(\bar{\mathcal{P}}_1 \cup \bar{\mathcal{P}}_2 \cup \bar{\mathcal{P}}_3 \cup \bar{\mathcal{P}}_4 \cup \bar{\mathcal{P}}_5). \tag{7}$$

The only sets that are not 'ordered', in figure 1, are $\bar{\mathcal{P}}_3$ and $\bar{\mathcal{P}}_4$. If they were both available, they would be both *falsified* as *Pareto candidates*, without need for any additional information.

2.3. *Extension*

Another kind of analysis that is useful for evaluation of a Pareto-set estimate is determining to what extent it covers the Pareto surface. For this purpose, some additional definitions are necessary.

Consider the space $\mathcal{Y}$ of the objective vectors. Let $\epsilon > 0$ be a fixed real number and $h \in \mathcal{Y}$ a solution point. The set $\delta(\cdot, \cdot)$ is defined by:

$$\delta(\epsilon, h) = \left\{ g \in \mathcal{Y} \text{ such that } |g - h| \leq \epsilon \right\} \tag{8}$$

Take the set $\mathcal{X} = \{x_1, \ldots, x_v\}$, $\mathcal{X} \subset \mathcal{Y}$.

Definition 4 (ϵ-extension): *The set* $\Theta(\epsilon, \mathcal{X})$ *defined by*

$$\Theta(\epsilon, \mathcal{X}) = \bigcup_{i=1}^{v} \delta(\epsilon, x_i) \tag{9}$$

is the ϵ*-extension of the set* $\mathcal{X}$. □

For any set $\mathcal{X}$, $\Theta(\epsilon, \mathcal{X}) \supset \mathcal{X}$ trivially.

Consider now two sets $\mathcal{X}_1$ and $\mathcal{X}_2$ such that $\mathcal{X}_1 \preceq \mathcal{X}_2$ and $\mathcal{X}_2 \preceq \mathcal{X}_1$, i.e. both sets are *Pareto candidates*, and let $\epsilon > 0$. The following relations become defined:

$$\begin{aligned} \Theta(\epsilon, \mathcal{X}_1) \supset \mathcal{X}_2 \quad &\Leftrightarrow \quad \mathcal{X}_1 \overset{\epsilon}{\supset} \mathcal{X}_2 \\ \Theta(\epsilon, \mathcal{X}_1) \not\supset \mathcal{X}_2 \quad &\Leftrightarrow \quad \mathcal{X}_1 \overset{\epsilon}{\not\supset} \mathcal{X}_2. \end{aligned} \tag{10}$$

The following situations can occur:

(1) $\mathcal{X}_1 \overset{\epsilon}{\supset} \mathcal{X}_2$ and $\mathcal{X}_2 \overset{\epsilon}{\supset} \mathcal{X}_1$: In this case, the sets $\mathcal{X}_1$ and $\mathcal{X}_2$ are said to be *extent-equivalent*.

(2) $\mathcal{X}_1 \overset{\epsilon}{\supset} \mathcal{X}_2$ and $\mathcal{X}_2 \overset{\epsilon}{\not\supset} \mathcal{X}_1$: In this case, the set $\mathcal{X}_2$ is said to be an *extent subset* of set $\mathcal{X}_1$.

(3) $\mathcal{X}_1 \overset{\epsilon}{\not\supset} \mathcal{X}_2$ and $\mathcal{X}_2 \overset{\epsilon}{\not\supset} \mathcal{X}_1$: In this case, the sets are said to be *extent-incommensurable*.

If a set is an *extent subset* or if it is *extent-incommensurable* when compared with another set, it becomes *falsified* as a *Pareto candidate*.

2.4. *Extremal data*

There are other consistency data that can be known *a priori* in some cases: the individual optima of some or all the objectives can be known, for instance by using analytical tools. This means that the extremal points of the Pareto set are known. In figure 1, if this were the case, the sets $\bar{\mathcal{P}}_3$ and $\bar{\mathcal{P}}_4$ would become both *falsified* as *Pareto candidates*. The sets $\bar{\mathcal{P}}_1$ and $\bar{\mathcal{P}}_2$, taken individually, would remain as *candidates*.

3. Multiobjective decision and synthesis

The multiobjective analysis that has been presented can be used in two distinct ways:

(1) As a validation procedure for the analysis of specific algorithm outputs. In this case, any algorithm that claims to 'solve' a multiobjective problem should produce outputs that pass the analysis being still a *Pareto candidate*. Summarizing the analysis procedure, the following criteria can be employed in the evaluation of an algorithm $\mathbb{A}$ against a reference algorithm[1] $\mathbb{B}$:

   Criterion 1: Any set of output points $\bar{\mathcal{P}}_a$, generated by $\mathbb{A}$, should be *auto-consistent*.

   Criterion 2: Given a set of algorithm $\mathbb{B}$ output points $\bar{\mathcal{P}}_b$, the ordering relation $\bar{\mathcal{P}}_a \preceq \bar{\mathcal{P}}_b$ should hold.

   Criterion 3: The relation $\bar{\mathcal{P}}_a \overset{\epsilon}{\supset} \bar{\mathcal{P}}_b$ should hold.

   Criterion 4: If the problem has individual optima known *a priori*, the relation $\Psi(\bar{\mathcal{P}}_a \cup \mathcal{O}) \supset \bar{\mathcal{P}}_a$ should hold, with $\mathcal{O}$ denoting the set of individual solutions of the objective functions. Additionally, the algorithm $\mathbb{A}$ should be equipped with a parameter that allows the generation of $\mathcal{O}(2, \infty)$, at least asymptotically.

(2) As an aggregation procedure that takes a set of points (that could be generated by a single algorithm or by any combination of different algorithms) and produces a subset that is a Pareto candidate. Let $\bar{\mathcal{P}}_1, \ldots, \bar{\mathcal{P}}_N$ be the output sets of N different algorithms. The estimate of the Pareto set, given these sets, is the set: $\bar{\mathcal{P}}_e = \Psi(\bar{\mathcal{P}}_1 \cup \ldots \cup \bar{\mathcal{P}}_N)$.

In usage (1), a set of solution points that are derived from a given algorithm are taken as representative of that algorithm, and are analysed using criteria 1–4. Each set of points, originating from a single algorithm, is viewed as an entire object. The comparison between two such objects is performed as a representative comparison between the algorithms that have given rise to them. When applied to the analysis of algorithms that are built on the basis of

[1] An arbitrary algorithm that is used as a comparison basis for the evaluation of $\mathbb{A}$.

pointwise criteria (formulated as Problem 2), the above criteria 1 and 4 can introduce some consistency information in the solution set. Any two different algorithms can be compared with each-other, using the comparison criteria 2 and 3.

Usage (2), instead, will take a set of solution points that can be obtained anyway, and will apply the criteria 1 and 4 to eliminate several non-consistent points. As an instance of this usage, a multiobjective genetic algorithm is proposed here, employing this procedure to 'borrow' the output set of other algorithms, thereby becoming 'better than or equal to' any of these algorithms, in a tautological manner. There is, however, an 'information gain' that leads the multiobjective genetic algorithm output beyond the trivial tautology, because the algorithm operators usually lead its output set strictly better than the initial input set. Even in the case of the output becoming 'equal to' the initial set, there is an information gain in the form of some 'corroboration' to the hypothesis of the solution set being a Pareto-set sample.

4. Multiobjective genetic algorithm

The problem of optimization of arbitrary functionals has been, since the early development of the optimization theory, a main goal. However, each different method that was developed was built with several assumptions on the structure of the function to be optimized: linearity, convexity, differentiability, etc. The class of methods that has attained the best approximation to the problem of 'arbitrary functional optimization' is the family of 'stochastic optimization methods'. A group of methods that has attained a high applicability from this class is the family of 'genetic algorithms'.

Owing to the 'global optimization' properties of the genetic algorithms, they have become a natural tool for problems like the $\mathcal{H}_2/\mathcal{H}_\infty$ design (Chen *et al.* 1995). Another potential reason for this suitability is pointed out here: since the genetic algorithms work with populations of candidate solutions, instead of a single candidate solution like other optimization methods, they are able to incorporate operators that exploit the group properties of the Pareto-set estimates (Coello 2001).

4.1. *Multiobjective genetic algorithm construction*

The multiobjective genetic algorithm can be built through the modification of any mono-objective genetic algorithm (Fonseca and Fleming 1995).

A genetic algorithm can be defined as the successive application of the following operations to a set of tentative solutions of the problem (called a 'population'):

(1) **Crossover:** The population is divided into pairs, and each pair of solutions is replaced by a new pair, that is generated employing information retained from the original pair;

(2) **Mutation:** Some solutions ('individuals') are chosen randomly to receive a perturbation in their parameters;

(3) **Selection:** The population that arises after the crossover and mutation operations is modified, with the exclusion of some 'individuals' and the replication of others, while maintaining the total size of the population. The probability of being replicated is greater for the greater optimization functional values (for maximization problems);

(4) **Elitism:** Some individuals (the 'best' ones) are deterministically maintained in the population.

After several applications of these operations, the 'population' converges to solutions that, in some sense, are 'good approximations' of the global solutions of the problem.

Any mono-objective genetic algorithm can be adapted through the following guidelines, to produce a multiobjective genetic algorithm:

(1) Select, from the initial population $\mathcal{Q}_0$, the group of individuals that form the maximal consistent subset $\bar{\mathcal{P}}_0$. This operation is defined by: $\bar{\mathcal{P}}_0 = \Psi(\mathcal{Q}_0)$.

(2) At each iteration, a new population $\mathcal{Q}_i$ is generated by the application of the genetic operators. Recalculate the estimate $\bar{\mathcal{P}}_i = \Psi(\mathcal{Q}_i)$, eventually excluding some individuals and including others. The set $\bar{\mathcal{P}}_i$ is employed as the 'elite set' in the 'elitism' operation.

(3) A 'niche' technique should be employed, to avoid the inclusion of points that are much close to each-other in the set $\bar{\mathcal{P}}$. In this way, the solution set of the multiobjective genetic algorithm (MGA), denoted as $\bar{\mathcal{P}}_{MGA}$ pressure for covering the whole set $\mathcal{P}$.

(4) The functional that guides the selection operation should be composed with the individual functionals that compose the objective vector. In the specific implementation employed here, they are scaled and then aggregated with the operator *max*.

In this way, instead of searching for *single* solutions, the whole set $\mathcal{P}$ is searched, as an object with intrinsic properties. The design procedure starts with any *non-consistent* or *conservative* algorithm, which furnishes an initial solution set $\bar{\mathcal{P}}_a$ that is further refined by a Multiobjective Genetic Algorithm (MGA).

Denote by $\bar{\mathcal{P}}_i$ the Pareto-set candidate produced by MGA at the ith iteration. The multiobjective genetic operators have been tailored such that:

(1) **niche+selection** produces a pressure that leads the Pareto estimate $\bar{\mathcal{P}}_{MGA}$ to increase its 'extension';

(2) **elitism** deterministically guarantees that: (i) $\bar{\mathcal{P}}_{i+1} \preceq \bar{\mathcal{P}}_i$ and (ii) $\bar{\mathcal{P}}_{i+1} \overset{\epsilon}{\supset} \bar{\mathcal{P}}_i$.

(3) **selection** produces the enhancement pressure that eventually allows $\bar{\mathcal{P}}_{i+1} \prec \bar{\mathcal{P}}_i$ and $\bar{\mathcal{P}}_i \overset{\epsilon}{\not\supset} \bar{\mathcal{P}}_{i+1}$.

The set operators $\preceq$ and $\overset{\epsilon}{\not\supset}$ have been defined in (5) and (10).

The multiobjective genetic algorithm, therefore, extends the initial algorithm solutions in the sense that some 'failures' in the estimate of $\mathcal{P}$ are 'repaired' by the MGA. Some 'holes' in the Pareto-set estimates can be filled by the MGA, thereby enhancing the Pareto estimate.

The MGA used here is the real-biased genetic algorithm described in Ramos *et al.* (2003).

5. $\mathcal{H}_2/\mathcal{H}_\infty$ control problem statement

To verify the applicability of the proposed methodology, a design problem that is longstanding in the field of control theory is examined here: the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem. Several current approaches for the approximation of this problem solutions are studied, and then the proposed methodology is applied.

Consider the following linear time-invariant dynamic system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + \sum_{k=1}^{N} B_k w_k(t) \\ z_k(t) = C_k x(t) + D_k u(t), \quad k = 1, \dots, N \\ y(t) = Cx(t) + \sum_{k=1}^{N} E_k w_k(t), \end{cases} \tag{11}$$

in which $x \in \mathbb{R}^n$ is the system state vector, $u \in \mathbb{R}^m$ is the control input vector, and $w_k \in \mathbb{R}^p_k$, $k = 1, \dots, N$ are the exogenous disturbance vector, $z_k \in \mathbb{R}^q_k$, are the controlled output, and $y \in \mathbb{R}^r$ is the measurement output. This system configuration describes N channels from the disturbance input w_k to the controlled variable output z_k, and associated with each channel, a performance index can be defined to be minimized or upper-bounded.

For control purposes, the static output control law is considered: $u(t) = Ky(t)$. As a special case, in the standard setting of the static state feedback design problem, the state vector x is considered to be available for control law synthesis with $C = I$ and $u(t) = Kx(t)$.

In particular, the closed-loop transfer functions from w_k to z_k are denoted[2] by

$$H_{z_k w_k}(s) = C_{cl}^{(k)} (s\boldsymbol{I} - A_{cl})^{-1} B_{cl}^{(k)}, \tag{12}$$

in which

$$\begin{aligned} A_{cl} &= A + BKC \\ B_{cl}^{(k)} &= B_k + BKE_k \\ C_{cl}^{(k)} &= C_k + D_k KC \end{aligned} \tag{13}$$

for static output feedback. For static state feedback, just consider $C = I$ and $E_k = 0,\ k = 0, \dots, N$.

Several different performance criteria could be defined for the closed-loop transfer matrix $H_{z_k w_k}(s)$. In this paper, the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms are used.

The main multiobjective problem to be addressed in this paper is stated for $k = 2$.

Problem 4 (the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ controller set computation): *Let γ_2 denote the value of the $\mathcal{H}_\infty$ norm of the closed-loop system with optimal $\mathcal{H}_2$ norm, and γ_∞ the optimal $\mathcal{H}_\infty$ norm. Let $\mathcal{S}$ denote the set of stabilizing static controllers with compatible dimensions. Determine the set $\mathcal{K}_{2\infty}$ such that:*

$$\mathcal{K}_{2\infty} \triangleq \left\{ K_{2\infty} \middle| \begin{bmatrix} K_{2\infty} = \arg_K \inf_K \|H_{w_1 z_1}\|_2 \\ \text{subject to} \begin{cases} \|H_{w_2 z_2}\|_\infty \le \gamma \\ K \in \mathcal{S} \end{cases} \end{bmatrix}, \gamma_\infty \le \gamma \le \gamma_2 \right\}. \tag{14}$$

□

The research tradition in the problem of $\mathcal{H}_2/\mathcal{H}_\infty$ control synthesis has dealt with the question of finding *one* controller that is expected to belong to the set $\mathcal{K}_{2\infty}$, or at least approximates it (Khargonekar and Rotea 1991b, Scherer 1995, Scherer *et al.* 1997, Thakahashi *et al.* 1997). Formulation of Problem 4 is, therefore, similar to the process in Problem 2. The approach that is proposed here, on the other hand, is based on a search for a representative set of solutions describing the Pareto-set $\mathcal{P}$, to take advantage of the cross-validation possibilities of the solutions belonging to this set.

In a multiobjective setting, Problem 4 is stated in terms of the objectives $\|H(K)\|_2$, the $\mathcal{H}_2$ norm, and $\|H(K)\|_\infty$, the $\mathcal{H}_\infty$ norm of the closed-loop system for controller K, considered in the appropriate channels.

[2] The following assumption is made: $D_{cl} = D_k K E_k = 0$, $k = 1, \dots, N$.

These norms define the control objectives, which are organized in the objective vector:

$$f(K) = \left[\, \|H(K)\|_2 \quad \|H(K)\|_\infty \,\right]^T. \tag{15}$$

6. Current approaches

The following constrained mono-objective optimization problem is the usual formulation for the generation of solutions for the multiobjective $\mathcal{H}_2/\mathcal{H}_\infty$ problem:

Problem 5 (the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ single-run design): *Let the disturbance attenuation level $\gamma > 0$ be assigned with a fixed value. Let $\mathcal{S}$ denote the set of stabilizing static controllers of compatible dimensions. Find $K_{2\infty} \in \mathcal{S}$ such that:*

$$\begin{cases} K_{2\infty} \text{ minimizes } \|H_{w_1 z_1}(K)\|_2 \\ \text{subject to } \|H_{w_2 z_2}(K)\|_\infty \leq \gamma \end{cases} \tag{16}$$

□

This is a constrained non-linear, non-smooth and non-convex mono-objective optimization problem with a possibly non-convex and unbounded feasible set. The set of solutions of the multiobjective problem is obtained by varying the constraint parameter γ. Note that, because of these characteristics, the set of exact solutions for different γs cannot be affirmatively characterized—this implies the need for a falsification procedure for solution characterization.

Consider any optimization algorithm to solve Problem 5 several times, with different γs, to generate an estimate of the Pareto set. The estimated set is likely to be not only a *Pareto sub-optimum*, but even *non-auto-consistent*, because:

(1) a single solution is found in each optimization algorithm run;
(2) the solutions are not taken as a set with set properties;
(3) the mono-objective optimization algorithms that are employed are likely to find only local minima of Problem 5;
(4) these minima are not necessarily related, from one run to another.

These is one exception for the earlier Linear Matrix Inequalities (LMI) formulation of the mixed objective problem, in terms of conservative convex algorithms. It is based on sufficient, but not necessary, conditions (Khargonekar and Rotea 1991b), which means that Problem 5 is modified in LMI formulation, being only approximately solved. Therefore, these algorithms lead to solutions that do not belong to the Pareto set and, in fact, can be significantly far from it. However, since the LMI formulation becomes convex, any single run of the optimization problem leads to its global solution. Because of this, the LMI algorithm does furnish points that are *auto-consistent*. However, it is an easy task to find other solutions that lie below the curve $\bar{\mathcal{P}}_{LMI}$ found with the LMI algorithm.

Different algorithms have been employed as optimization engine instances for solving Problem 5. Recently, an iterative non-convex algorithm that solves a sequence of LMI problems that approximate the exact Bilinear Matrix Inequalities (BMI) form of Problem 5 has been proposed to furnish less conservative solutions to $\mathcal{H}_2/\mathcal{H}_\infty$ problems (Shimomura and Fujii 2000). Other heuristic solutions have been proposed for these problems, sometimes employing genetic algorithms (Chen *et al.* 1995) or other non-convex optimization schemes (Takahashi *et al.* 1997), with the aim of approaching solutions belonging to the set $\mathcal{P}$. All these algorithms can furnish solutions that are not *auto-consistent*.

The old, popular, and conservative LMI formulation and its succedaneum, the BMI formulation, are studied here as reference solutions that will initialize the multiobjective algorithm. Any other solutions could be employed for the same purpose.

6.1. *Matrix inequalities formulations*

In a matrix inequality setting, the exact mixed control problem, as formulated above, is the direct combination of the actual $\mathcal{H}_2$ norm computation with the Bounded Real Lemma. That is, assuming that the closed-loop system is asymptotically stable, the optimal $\mathcal{H}_2$ norm computation is performed by

$$\|H_{w_1 z_1}\|_2^2 = \inf_{X_2, J} \{\mathrm{Tr}\,(J)\} \tag{17}$$

$$\text{s.t.} \quad \begin{bmatrix} J & C_{cl}^{(1)} \\ (C_{cl}^{(1)})' & X_2 \end{bmatrix} > 0 \tag{18}$$

$$\begin{bmatrix} A_{cl}' X_2 + X_2 A_{cl} & X_2 B_{cl}^{(1)} \\ (B_{cl}^{(1)})' X_2 & -I \end{bmatrix} < 0. \tag{19}$$

However, the Bounded Real Lemma is stated in the following way: $\gamma > 0$, A_{cl} is asymptotically stable, and $\|H_{z_2 w_2}\|_\infty < \gamma$ if, and only if, there is a symmetric definite positive matrix X_∞ such that

$$\begin{bmatrix} A_{cl}' X_\infty + X_\infty A_{cl} & X_\infty B_{cl}^{(2)} & (C_{cl}^{(2)})' \\ (B_{cl}^{(2)})' X_\infty & -I & 0 \\ C_{cl}^{(2)} & 0 & -\gamma^2 I \end{bmatrix} < 0. \tag{20}$$

Thus, the exact mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem can be completely restated as in the following, with the simple substitution of the closed-loop matrices given in (13) into (18), (19), and (20):

Problem 6 (exact matrix inequalities formulation): *Determine a stabilizing static feedback control K that achieves*

$$\Gamma = \min_{X_2, X_\infty, J, K} \{\mathrm{Tr}(J)\}$$

$$\text{s.t.} \begin{bmatrix} J & C_1 + D_1KC \\ (C_1 + D_1KC)' & X_2 \end{bmatrix} > 0$$

$$\begin{bmatrix} (A+BKC)'X_2 + X_2(A+BKC) & X_2(B_1 + BKE_1) \\ (B_1 + BKE_1)'X_2 & -I \end{bmatrix} < 0$$

$$\begin{bmatrix} (A+BKC)'X_\infty + X_\infty(A+BKC) & X_\infty(B_2 + BKE_2) & (C_2 + D_2KC)' \\ (B_2 + BKE_2)'X_\infty & -I & 0 \\ C_2 + D_2KC & 0 & -\gamma^2 I \end{bmatrix} < 0$$

$$X_\infty > 0.$$

□

6.2. *Standard LMI formulation*

For the state feedback case, the conventional strategy adopted in the literature is based on the simple change of variables of type $K = ZW^{-1}$ (Bernussou *et al.* 1989), with the imposition $W = X_2^{-1} = X_\infty^{-1}$, $C = I$ and $E_k = 0$, $k = 1, 2$ in Problem 6. From this, the following optimization LMI control synthesis description can be obtained:

Problem 7 (standard LMI formulation):

$$\Upsilon = \min_{Z, W, J} \mathrm{Tr}\{J\}$$

$$\text{s.t.} \begin{bmatrix} J & C_1W + D_1Z \\ (C_1W + D_1Z)' & W \end{bmatrix} > 0$$

$$\begin{bmatrix} AW + WA' + Z'B' + BZ & B_1 \\ B_1' & -I \end{bmatrix} < 0$$

$$\begin{bmatrix} AW + WA' + Z'B' + BZ & B_2 & (C_2W + D_2Z)' \\ B_2' & -I & 0 \\ C_2W + D_2Z & 0 & -\gamma^2 I \end{bmatrix} < 0,$$

where $\|H_{z_1w_1}\|_2^2 \le \Upsilon$, $\|H_{z_2w_2}\|_\infty < \gamma$ *and the static state feedback gain is given by* $K = ZW^{-1}$. □

6.3. *BMI formulation*

This formulation is derived from Shimomura and Fujii (2000). The key idea is to handle the non-affine characteristics introduced by non-positive quadratic terms when one substitutes (13) in (18)–(20) by means of matrix upper bounds. Completing the square relative to the non-affine terms in Problem 6, and upper-bounding the non-positive quadratic terms generated, the following mixed problem (Shimomura and Fujii 2000) can be obtained:

Problem 8 (BMI formulation):

$$\Omega = \min_{X_2, X_\infty, J, K} \{\mathrm{Tr}(J)\}$$

$$\text{s.t.} \begin{bmatrix} J & C_1 + D_1KC \\ (C_1 + D_1KC)' & X_2 \end{bmatrix} > 0$$

$$\begin{bmatrix} \Phi_{21} & X_2B_1 & X_2B + C'K' & X_2B \\ B_1'X_2 & \Phi_{22} & 0 & E_1'K' \\ B'X_2 + KC & 0 & -I & 0 \\ B'X_2 & KE_1 & 0 & -I \end{bmatrix} < 0$$

$$\begin{bmatrix} \Phi_{\infty 1} & X_\infty B_2 & X_\infty B + KC & X_\infty B & (C_2 + D_2KC)' \\ B_2'X_\infty & \Phi_{\infty 2} & 0 & E_2'K' & 0 \\ B'X_\infty + KC & 0 & -I & 0 & 0 \\ B'X_\infty & KE_2 & 0 & -I & 0 \\ C_2 + D_2KC & 0 & 0 & 0 & -\gamma^2 I \end{bmatrix} < 0$$

$$X_\infty > 0, \tag{21}$$

where

$$\Phi_{21} = A'X_2 + X_2A - 2X_2BL_1 - 2L_1'B'X_2 + 2L_1'L_1 - C'K'M - M'KC + M'M \tag{22}$$

$$\Phi_{\infty 1} = A'X_\infty + X_\infty A - 2X_\infty BL_2 - 2L_2'B'X_\infty + 2L_2'L_2 - C'K'M - M'KC + M'M \tag{23}$$

$$\Phi_{22} = -I - E_1'K'N_1 - N_1'KE_1 + N_1'N_1 \tag{24}$$

$$\Phi_{\infty 2} = -I - E_2'K'N_2 - N_2'KE_2 + N_2'N_2 \tag{25}$$

$$M = KC, \quad L_1 = B'X_2, \quad L_2 = B'X_\infty, \quad N_1 = KE_1, \quad N_2 = KE_2. \tag{26}$$

□

Note that Problem 8 can be easily restated as a static state feedback mixed problem with $C = I$ and $E_k = 0$, $k = 1, 2$.

Adopting the above formulation, the following iterative algorithm is proposed in Shimomura and Fujii (2000). For simplicity, Step 1 is characterized for the state feedback case, and then the static output feedback problem is discussed.

Iterative algorithm:

Step 1. Set $K^{(0)} = K$, where K is the optimal solution of Problem 7 as well as $(X_2^{(0)})^{-1} = (X_\infty^{(0)})^{-1} = W$ and $\Gamma^{(0)} = \Omega^{(0)} = \Upsilon$. Set $i = 1$.

Step 2A. In Problem 6 set $X_2 = X_2^{(i-1)}$, $X_\infty = X_\infty^{(i-1)}$ and $\Gamma = \Gamma^{(i-1)}$.

Step 2B. With $K = K^{(i-1)}$ fixed, solve Problem 6 with respect to $X_2^{(i-1)} > 0$, $X_\infty^{(i-1)} > 0$ and $\Gamma^{(i-1)}$.

Step 2C. In Problem 8, set $M = K^{(i-1)}C$, $L_1 = B'X_2^{(i-1)}$, $L_2 = B'X_\infty^{(i-1)}$, $N_1 = K^{(i-1)}E_1$ and $N_2 = K^{(i-1)}E_2$ (for the particular case of state feedback $E_k = 0$, $k = 1, 2$ and $C = I$).

Step 3. Solve Problem 8 for $X_2^{(i)} > 0$, $X_\infty^{(i)} > 0$, $K^{(i)}$ and $\Omega^{(i)}$.

Step 4. If $\|\Omega^{(i-1)} - \Omega^{(i)}\| < \epsilon$ for a sufficiently small positive scalar ϵ, then stop. Else, set $i = i + 1$ and return to Step 2A.

For the static output feedback problem, the algorithm can be started with any feasible controller K that ensures a disturbance attenuation level γ, i.e. $\|H_{z_2 w_2}\|_\infty < \gamma$ and with finite $\|H_{z_1 w_1}\|_2^2$. In this case, the approach proposed by Cao *et al.* (1998), for example, can be used.

7. Examples of $\mathcal{H}_2/\mathcal{H}_\infty$ design

In this section, two examples of algorithm combination with the multiobjective genetic algorithm are presented. Single-channel cases are employed, for simplicity. Both examples adopt 50 generations, and the population size is dynamically varied inside the MGA from 75 to 150 individuals. The figures show the results of typical runs.

7.1. *Full state feedback*

A very simple system is presented in the first place, to visualize both the objective space (of $\mathcal{H}_2$ and $\mathcal{H}_\infty$ closed-loop norms) and the controller parameter space.

The system equations are:

$$\dot{x} = \begin{bmatrix} -0.3868 & 0.0751 \\ 0 & -0.0352 \end{bmatrix} x + \begin{bmatrix} -0.6965 \\ 1.6961 \end{bmatrix} u + \begin{bmatrix} 0.0591 & 0 \\ 0 & 1.7971 \end{bmatrix} w$$

$$z = \begin{bmatrix} 0.0346 & 0.0535 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0.5297 \end{bmatrix} u.$$

This system is controlled with a state-feedback controller:

$$u = \begin{bmatrix} K_1 & K_2 \end{bmatrix} x$$

The controller design problem is solved through: (1) the standard (conservative) LMI formulation defined in **Problem 4**; (2) the 'less conservative' BMI formulation defined in **Problem 5**; and (3) the multiobjective genetic algorithm, starting from both the solution set of **Problem 4** and that of **Problem 5**. The closed-loop norms obtained are ploted in figure 2, and the controller parameters in figure 3. Figure 3 shows the frontier in the space of the optimization parameters (the controller parameters) for which the LMI and BMI approaches can lead. Note that the genetic algorithm starting from both solutions sets, derived from LMI and BMI, reaches a single frontier in the controller parameter space. The region, in the controller parameter space, between the genetic algorithm frontier and the LMI and BMI frontiers is a good illustration of the gap that the multiobjective genetic algorithm can transpose.

These figures show that, in this case, the BMI formulation has generated some solutions that are even more conservative than those generated with the LMI formulation. Such behaviour can be explained by the non-convex nature of the BMI solutions. Both initial condition sets for the multiobjective genetic algorithm have led to the same solution set (in the sense of the frontier obtained in the objective space) that is less conservative and consistent with the characteristics of a Pareto set. The MGA solution sets cover all the extension between the two individual optima, while the BMI solution set leaves some spaces unfilled.

7.2. *Static output feedback*

The following unstable system of order three, with two control inputs and two measured outputs, is investigated:

$$\dot{x} = \begin{bmatrix} 1.9574 & -0.3398 & 1.1902 \\ 0.5045 & 0 & -1.1162 \\ 1.8645 & -0.2111 & 0.6353 \end{bmatrix} x + \begin{bmatrix} -0.6014 & 0 \\ 0.5512 & -2.0046 \\ -1.0998 & -0.4931 \end{bmatrix} u + \begin{bmatrix} 0.4620 & 0 & 0 \\ 0 & -0.3210 & 0 \\ 0 & 0 & 1.2366 \end{bmatrix} w$$

$$y = \begin{bmatrix} 0 & 0.2311 & 0 \\ -2.3252 & 0 & 0 \end{bmatrix} x$$

$$z = \begin{bmatrix} 0.1372 & 0.4374 & 0.7258 \\ 0.5216 & 0.4712 & 0 \\ 0.8952 & 0 & 0.3584 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.6264 & 0.9781 \\ 0.2412 & 0.6405 \end{bmatrix} u.$$

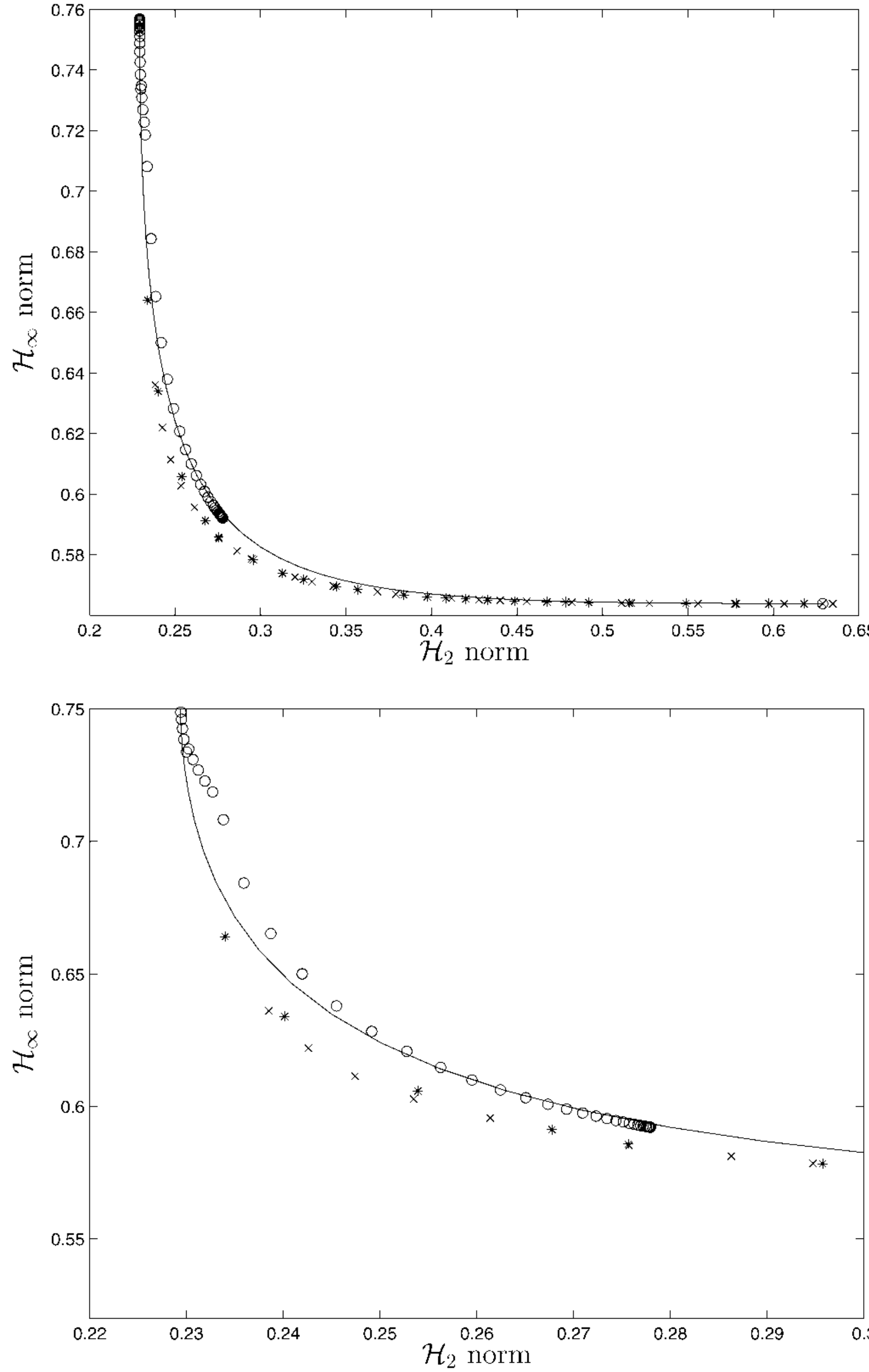

Figure 2. Pareto-set estimates, in the space of objectives, obtained from: LMI standard formulation (continuous line); BMI 'less conservative' formulation ($\circ$); multiobjective genetic algorithm starting from the LMI solutions ($\times$); multiobjective genetic algorithm starting from the BMI solutions ($*$). In the bottom, a detail of the figure is shown.

The control structure that is employed, in this case, is the static output feedback:

$$u = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}.$$

Some stabilizing controllers for this system can be found using the $\mathcal{H}_\infty$-algorithm presented in Cao *et al.* (1998). The different controllers are obtained by varying the $\mathcal{H}_\infty$-norm upper-bound parameter γ. These stabilizing controllers are then employed as starting points to the iterative algorithm of the BMI formulation in

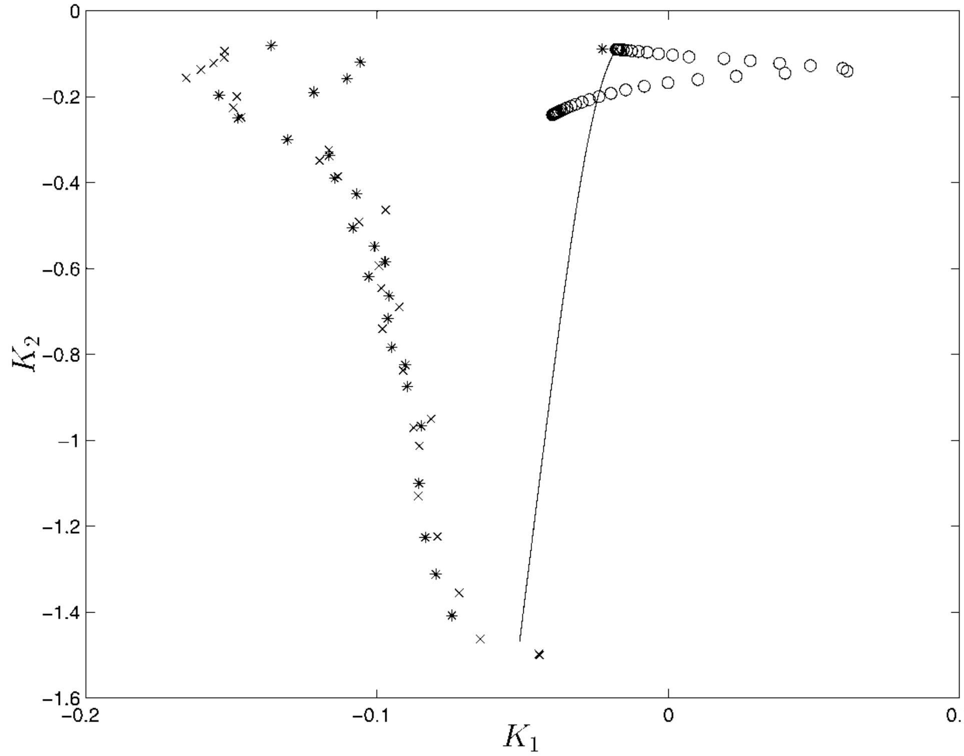

Figure 3. Pareto-set estimates, in the space of controller parameters, obtained from: LMI standard formulation (continuous line); BMI 'less conservative' formulation (○); multiobjective genetic algorithm starting from the LMI solutions (×); multiobjective genetic algorithm starting from the BMI solutions (∗).

Problem 8. The $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms associated with each controller derived from the two algorithms have been evaluated and are shown in figure 4.

The combination of all these controllers is used as an initial 'population' for the multiobjective genetic algorithm, used to search for an estimate $\bar{\mathcal{P}}$ of the Pareto set. The results are also shown in figure 4.

Note that none of the two first groups of solutions have reached the Pareto-set estimate that has been found by the multiobjective genetic algorithm. Neither the solution sets of the $\mathcal{H}_\infty$ algorithm or the BMI formulation are auto-consistent (this is due to the non-convexity nature of these formulations). Moreover, both algorithms have points that falsify some points of the other algorithm. Then, under the analysis stated in Section 2, both algorithms are shown to be unable to generate the Pareto set. However the multiobjective genetic algorithm estimated set is auto-consistent, dominant, and has a larger extension.

Notice that in these examples of small dimension, random initial population can generate the same estimate for the Pareto set. However, for larger cases, even recovering the LMIs or BMIs solution performances could be very difficult for the MGA. In this case, starting from the LMIs or BMIs solutions is still a natural choice that can help find better solutions.

8. Conclusions

By construction, it becomes tautologous that the output $\bar{\mathcal{P}}_{MGA}$ of the proposed multiobjective genetic algorithm is the best estimate available for the Pareto set in mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control design problems. The main consequences of this are:

(1) Any algorithm that claims to find 'the least conservative' solutions for this problem should have its output set $\bar{\mathcal{P}}_{lc}$ such that $\bar{\mathcal{P}}_{lc} \preceq \bar{\mathcal{P}}_{mga}$ and $\bar{\mathcal{P}}_{lc} \overset{\epsilon}{\supset} \bar{\mathcal{P}}_{mga}$. The proposed scheme can be seen, therefore, as a strong validation procedure for any mixed-criteria controller design algorithm.

(2) Otherwise, any algorithm should be coupled to MGA, to be able to generate the best approximation to the Pareto-set $\mathcal{P}$. Any algorithm that does not intend to solve the mixed problem completely, but only finds tentative solutions can be aggregated in this way.

Up to now, the best design for the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ problem is always finished by the application of MGA, for finding better solutions, or for corroborating the conjecture (that cannot be proved) that some solution is already the best possible solution.

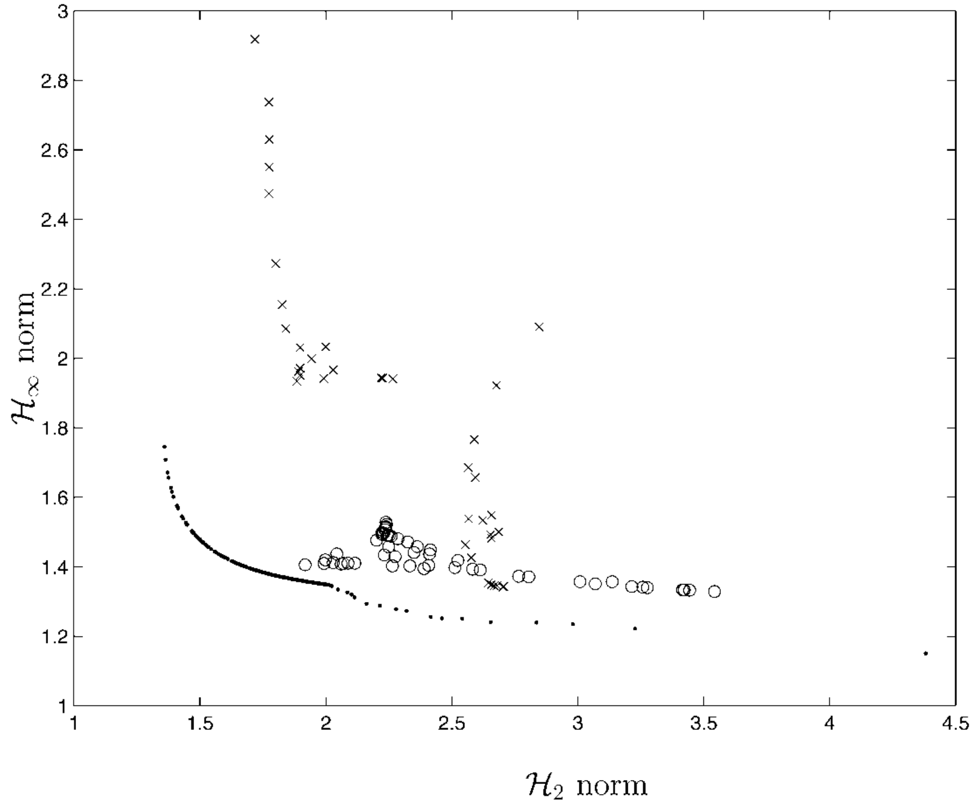

Figure 4. Pareto-set estimates, in the space of objectives, obtained from: $\mathcal{H}_\infty$ stabilizing static output feedback formulation (○); BMI formulation (×); and multiobjective genetic algorithm starting from both the BMI formulation and $\mathcal{H}_\infty$-algorithm solutions (·).

The methodology presented here, although presented in the context of the $\mathcal{H}_2/\mathcal{H}_\infty$ control problem, is not specific for this domain. Any design problem with multiple objectives could be analysed using the proposed tools, with minor adaptations.

Acknowledgements

This research has been supported in part by grants from CNPq and CAPES, Brazil.

References

Bernussou, J., Peres, P. L. D., and Geromel, J. C., 1989, A linear programming oriented procedure for quadratic stabilization of uncertain systems. *System & Control Letters*, **13,** 65–72.

Cao, Y.-Y., Lam, J., and Sun, Y.-X., 1998, Static output feedback stabilization: an ILMI approach. *Automatica*, **34,** (12), 1641–1645.

Chankong, V., and Haimes, Y. Y., 1983, *Multiobjective Decision Making: Theory and Methodology* (New York: North-Holland).

Chen, B. S., Cheng, Y. M., and Lee, C. H., 1995, A genetic approach to mixed $\mathcal{H}_\infty/\mathcal{H}_2$ optimal PID control. *IEEE Control Systems Magazine*, **15,** (5), 51–60.

Coello, C. A. C., 2001, A short tutorial on evolutionary multiobjective optimization. *Lecture Notes in Computer Science*, 1993, pp. 21–40.

de Oliveira, M. C., Geromel, J. C., and Bernussou, J., 2002, Extended $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norm characterizations and controller parametrizations for discrete-time systems. $\mathcal{H}_\infty/\mathcal{H}_2$ optimal PID control. *International Journal of Control*, **75,** (9), 666–679.

Dorato, P., 1991, A survey of robust multiobjective design techniques. In S. P. Battacharyya and L. H. Kell (eds.) *Control of Uncertain Dynamic Systems* (Boca Raton, FL: CRC Press), pp. 249–259.

Fleming, P. J., and Purshouse, R. C., 2002, Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice*, **10,** (11), 1223–1241.

Fonseca, C. M., and Fleming, P. J., 1995, Multiobjective genetic algorithms made easy: selection sharing and mating restriction. *Proceedings of the First International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications* (Sheffield, UK), pp. 45–52.

Khargonekar, P. P., and Rotea, M. A., 1991a, Controller synthesis for multiobjective optimal control. In S. P. Battacharyya and L. H. Kell (eds.) *Control of Uncertain Dynamic Systems* (Boca Raton, FL: CRC Press), pp. 261–279.

Khargonekar, P. P., and Rotea, M. A., 1991b, Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control: a convex optimization approach. *IEEE Transactions on Automatic Control*, **36,** (7), 824–837.

Leite, V. J. S., and Peres, P. L. D., 2003, An improved LMI condition for robust D-stability of uncertain polytopic systems. *IEEE Transactions on Automatic Control*, **48,** (3), 500–504.

Ramos, R. M., Saldanha, R. R., Takahashi, R. H. C., and Moreira, F. J. S., 2003, The real-biased multiobjective genetic algorithm and its application to the design of wire antennas. *IEEE Transactions on Magnetics*, **39,** (3), 1329–1332.

Scherer, C., 1995, Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control. In A. Isidori (ed.) *Trends in Control* (Berlin: Springer-Verlag), pp. 173–216.

Scherer, C., Gahinet, P., and Chilali, M., 1997, Multiobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, **42,** (7), 896–911.

Shimomura, T., and Fujii, T., 2000, Multiobjective control design via sucessive over-bounding of quadratic terms. *Proceedings of the 39th IEEE Conference on Decision and Control* (Sydney, Australia), pp. 2763–2768.

Takahashi, R. H. C., Camino, J. F., Zampieri, D. E., and Peres, P. L. D., 2000, Multiobjective weighting selection for optimization-based control design. *ASME Journal of Dynamic Systems, Measurement and Control*, **122,** (3), 567–570.

Takahashi, R. H. C., Peres, P. L. D., and Ferreira, P. A. V., 1997, Multiobjective $\mathcal{H}_2/\mathcal{H}_\infty$ guaranteed cost PID design. *IEEE Control Systems Magazine*, **17,** (5), 37–47.