GENETIC SOLUTIONS TO MIXED $\mathcal{H}_2/\mathcal{H}_\infty$ PROBLEMS
Limits of Performance

Gustavo Sánchez, Miguel Strefezza
Universidad Simón Bolívar, Departamento de Procesos y Sistemas, Caracas, Venezuela
gsanchez@usb.ve, strefeza@usb.ve

Minaya Villasana
Universidad Simón Bolivar, Departamento de Cómputo Científico, Caracas, Venezuela
mvillasa@usb.ve

Keywords: Multi-objective control, Genetic algorithms, LMIs, Pole placement, *COMPl$_e$ib*.

Abstract: One of the most relevant problems for control engineers is the so-called "mixed $\mathcal{H}_2/\mathcal{H}_\infty$". To solve it, different convexifying strategies became popular in the later 1990s, mainly based on Linear Matrix Inequalities (LMIs). On the other hand, genetic algorithms have also been applied for $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis. Indeed, several authors agree that they are able to find good solutions to this important control problem. However, in most of the published papers, only low-order SISO models have been considered. In the present paper a LMI-based algorithm is compared against a genetic algorithm, with respect to three performance indicators: Set Coverage, Maximum Distance and Efficient Set Spacing. Five open-loop MIMO models extracted from *COMPl$_e$ib* are studied, for which the degree varies between 5 and 10. Based on numerical results, the genetic algorithm *is not able to improve* LMI solutions for problems with more than 42 variables, restricted to a budget of 20.000 function evaluations.

1 INTRODUCTION

One of the most important problems for control engineers is the so-called "mixed $\mathcal{H}_2/\mathcal{H}_\infty$". Typically, the $\mathcal{H}_\infty$ channel is used to enhance the robustness of the closed-loop system, whereas the $\mathcal{H}_2$ channel guarantees good performance (Apkarian et al., 2008).

To solve this problem, different convexifyng strategies became popular in the later 1990s, despite the inherent conservatism of this approach. For instance, in (Scherer et al., 1997) controllers are designed by solving a set of LMIs in tandem with nonlinear algebraic equalities. In fact, this design method (enhanced with many improvements over the years) remains as *state of the art* for this problem.

On the other hand, Multi-Objective Evolutionary Algorithms (MOEA) have also been applied for $\mathcal{H}_2/\mathcal{H}_\infty$ synthesis: in (Takahashi et al., 2001) and (Takahashi et al., 2004), a genetic approach is proposed to obtain $\mathcal{H}_2/\mathcal{H}_\infty$ solutions which are consistent with a Pareto set and less conservative compared to LMI solutions.

After these examples it may seen obvious that, under special circumstances, genetic algorithms are in fact able to find better solutions than LMI-based algorithms. However, in most of the published works which have been consulted for this paper, only low-order and SISO models have been considered, more appropriate to evaluate low-complexity controllers as PIDs (Astrom et al., 1998).

In this manner, the question arises as to whether the genetic algorithm advantage remains true when the open-loop models are high-order and MIMO (Multiple Input Multiple Output) as those proposed in *COMPl$_e$ib* (Leibfritz, 2004).

The rest of this paper is organized as follows. In section **2**, the controller design problem is formulated. Next, the two design methods to be compared are described in section **3**. In section **4**, numerical results are presented and conclusions are given in section **5**.

2 PROBLEM FORMULATION

The closed-loop system is shown in figure 1. Matrices $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times n_u}$ and $C \in \mathbb{R}^{n_y\times n}$ denote the corresponding open-loop state matrices.

The open-loop state-space equations are:

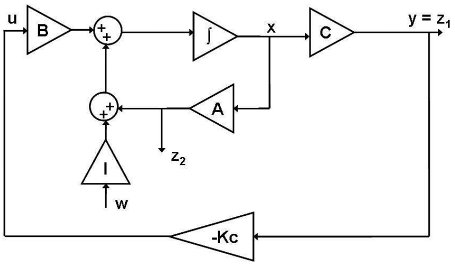

Figure 1: Continuous-time closed-loop design model.

$$\begin{cases} \dot{x} = Ax + Iw + Bu \\ z_1 = y = Cx \\ z_2 = Ax \end{cases} \qquad (1)$$

In this formulation, $w \in L_2^{n_w \times 1}$denotes the exogenous input, $z_1 \in L_2^{n_{z_1} \times 1}$ and $z_2 \in L_2^{n_{z_2} \times 1}$ represents the outputs to be regulated, while $u \in L_2^{n_u \times 1}$ and $y \in L_2^{n_y \times 1}$ represent the control input and the measured output respectively. It is assumed the open-loop model is strictly proper, stabilizable from u and detectable from y.

Consider a full-order linear controller K_c described by the state equations

$$K_c : \begin{cases} \dot{x}_c = A_K x_c + B_K y \\ u = C_K x_c + D_K y \end{cases} \qquad (2)$$

Finally, let

$$\begin{aligned} G_1(K_c) &= G_{z_1 w}(K_c) \\ G_2(K_c) &= G_{z_2 w}(K_c) \end{aligned} \qquad (3)$$

be the closed-loop transfer function from w to z_1 and z_2 respectively. The mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem is stated as:

$$P_{\mathcal{H}_2/\mathcal{H}_\infty} : \min_{K_c} \begin{pmatrix} \|G_1(K_c)\|_2 \\ \|G_2(K_c)\|_\infty \end{pmatrix} \qquad (4)$$

subject to
$G_1(K_c)$ and $G_2(K_c)$ are stable

In this formulation the term *min* should be interpreted as the search for the best possible *approximation of the corresponding Pareto-optimal set*.

3 DESIGN METHODS

In this section two methods to solve the multi-objective problem $P_{\mathcal{H}_2/\mathcal{H}_\infty}$ are described.

3.1 LMI-based Method

In this sub-section a design algorithm based on the results presented in (Scherer et al., 1997) is proposed. In fact, these authors demonstrated that given two positive scalars γ_2, γ_∞ the following equations hold

$$\begin{aligned} \|G_1(K_c)\|_2 &\leq \gamma_2 \\ \|G_2(K_c)\|_\infty &\leq \gamma_\infty \end{aligned} \qquad (5)$$

if there exist matrices $X > 0, Y > 0, \widehat{A}, \widehat{B}, \widehat{C}, \widehat{D}$ such that a certain set of LMIs are feasible.

To build the approximation of the Pareto-front, the following iterative procedure is proposed (see Algorithm 1): First the limit of one restriction is increased and the other one decreased (i.e. linearly). Then, the feasability problem is solved and the solution K_c is archived, *only in case it is non-dominated*. Otherwise it is rejected.

Algorithm 1: Algorithm to solve $P_{\mathcal{H}_2/\mathcal{H}_\infty}$ via LMIs.

```
Data: $G_1, G_2, N_{\mathcal{H}_2/\mathcal{H}_\infty}, \Delta\gamma_2, \Delta\gamma_\infty, \gamma_{\min}, \gamma_{\max}$
Result: $P$
$k = 1$;
while $k \leq N_{\mathcal{H}_2/\mathcal{H}_\infty}$ do
    Solve
        $\left\| G_2(K_c^k) \right\|_2 \leq \gamma_{\min} + k\Delta\gamma_2$
        $\left\| G_1(K_c^k) \right\|_\infty \leq \gamma_{\max} - k\Delta\gamma_\infty$
    $P_k = UpdateArchive(K_c^k, P_{k-1})$;
    $k = k + 1$;
end
```

3.2 Multi-objective Pole Placement with Evolutionary Algorithms (MOPPEA)

In the following we describe a design method, named Multi-Objective Pole Placement with Evolutionary Algorithms (MOPPEA).

In the general case, an output feedback controller can be designed by combining a full information controller with a state observer. The resulting output feedback sub-system is called “observer-based controller” and has the following state-equations:

$$\begin{cases} \dot{x}_c = (A + BK + LC)x_c - Ly \\ u = Kx_c \end{cases} \qquad (6)$$

where x_c is the estimated state.

Let $pk \in \mathbb{C}^{n_k}$ and $pl \in \mathbb{C}^{n_l}$ be the eigenvalues of $A + BK$ and $A + LC$ respectively. To assure closed-loop system stability, the gain matrix K and L must be calculated in such way that pk and pl belong to $\mathbb{C}^-$ (open left-half complex plan).

Thus, the key concept of the proposed design method is using an evolutionary process in order *to evolve matrices* $K \in \mathbb{R}^{n_u \times n}$ *and* $L \in \mathbb{R}^{n \times n_y}$. Thus, the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem is stated again as:

$$\widehat{P}_{\mathcal{H}_2/\mathcal{H}_\infty}: \min_{K\in\mathbb{R}^{n_u\times n}, L\in\mathbb{R}^{n\times n_y}} \begin{pmatrix} \|G_1(K,L)\|_2 \\ \|G_2(K,L)\|_\infty \end{pmatrix} \qquad (7)$$

subject to

$G_1(K,L)$, $G_2(K,L)$ are stable

Regarding the initial population, it can be generated using the algorithm proposed in (Sánchez et al., 2007). After that, SPEA2 is used to drive the design process, taking advantage of its ability to manage an archive of non-dominated solutions.

4 NUMERICAL RESULTS

In this section two algorithms are compared:

A1: SPEA2 - enhanced with special operators and restricted to 20.000 objective function evaluations, using the parameters shown in table 1. This quantity was fixed considering the total time available for computations.

A2: LMI-based design (see Algorithm 1).

Table 1: Setting parameters used for SPEA2.

Parameter	Value
Initial Population	Randomly generated
Representation	$K+L$
Cross-Over Recombination	Arithmetical
Cross-Over Rate	0.9
Mutation Operator	Gaussian
Mutation Rate	0.1
Population Size	200
Stop Condition	100 generations
Population Size	100
Offspring Size	100

Table 2 (at the top of the next page) presents the information related with the selected *COMPl$_e$ib* models, each one characterized by a particular nomenclature. Five models were selected: AC1, AC6, WEC1, NN10 and AC9, for which the number of decision varies between 30 and 90. Table 2 also presents, for each model, the parameters $N_{\mathcal{H}_2/\mathcal{H}_\infty}$,$\Delta\gamma_2$, $\Delta\gamma_\infty$, $\gamma_{\min}$ and $\gamma_{\max}$ used by A2.

Thirty executions were simulated for each algorithm and for each problem. Let $\mathcal{PF}_1$ and $\mathcal{PF}_2$ be the Pareto approximations found by two different algorithms. To compare their performance, the following indicators were computed:

- Set Coverage(C)
- Maximum Distance(MD)
- Efficient Set Spacing(ESS)

The values obtained for these indicators are shown in tables 3 and 4. For each indicator the mean value and the standard deviation within parentheses are presented. These results confirm that the LMI-based algorithm is able to produce dominating solutions with respect to the genetic algorithm, for problems with more than 42 decision variables. Note that A1 achieves better results than A2 with respect to *MD.* However, A2 achieves better results with respect to *ESS,* which can be explained given the deterministic nature of A1.

Table 3: Set coverage results.

$C(A_i,A_j)$	A1	A2

A1	–	AC1:0.8619(0.1326)
		AC6:0(0)
		NN10:0(0)
		WEC1:0(0)
		AC9:0(0)

A2	AC1:0(0)
	AC6:0(0)
	NN10:1(0)
	WEC1:1(0)
	AC9:1(0)
		–

Table 4: MD and ESS results.

	MD	ESS

A1	AC1:0.7589(0.0535)
	AC6:120.4263(67.0600)
	NN10:3.5940(0.7046)
	WEC1:45.3320(7.7541)
	AC9:564.2026(190.4456)
		AC1:0.0190(0.0069)
		AC6:3.5698(4.0164)
		NN10:0.1026(0.0407)
		WEC1:1.0784(0.4881)
		AC9:21.5037(17.8440)

A2	AC1:0.1618(0)
	AC6:2.8065(0)
	NN10:1.0200(0)
	WEC1:36.0976(0)
	AC9:59.2017(0)
		AC1:0.0122(0)
		AC6:0.0152(0)
		NN10:0.0188(0)
		WEC1:0.1540(0)
		AC9:3.3800(0)

5 CONCLUSIONS

In this paper, we analyze the performance of a genetic algorithm (SPEA2) to solve five mixed $\mathcal{H}_2/\mathcal{H}_\infty$ design problems, taking as reference a LMI-based iterative algorithm and considering a fixed budget of 20.000 evaluations.

Based on the obtained results, the following conclusions can be stated:

- Unlike other representations, the proposed (K,L) chromosome is able to efficiently explore the controller space, even for models with order greater

Table 2: Information related with the selected *COMPl$_e$ib* models.

COMPl$_e$ib	n	n_u	n_y	n_{MOPPEA}	$N_{\mathcal{H}_2/\mathcal{H}_\infty}$	$\Delta\gamma_2$	$\Delta\gamma_\infty$	$\gamma_{\min}$	$\gamma_{\max}$	$r_{\max}$	$i_{\max}$
AC1	5	3	3	30	100	0.01	0.01	1.5	5.5	10	10
AC6	7	2	4	42	100	0.1	1	1	110	20	20
NN10	8	3	3	48	100	0.01	0.02	3	15	10	10
WEC1	10	3	4	70	100	0.05	1	2	200	100	100
AC9	10	4	5	90	100	0.1	0.1	50	100	100	100

than 4 and with multiple inputs and outputs. The proposed variation operators allows to stay within the feasible region.

- The statistical tests show the genetic algorithm is not able to improve LMI solutions for problems with more than 42 variables, considering a fixed budget of 20.000 function evaluations.

As future work, more simulations need to be carried out, to find how many function evaluations are to be allowed in order the genetic algorithm is competitive again. It is also possible to test ”hybrid” design methods, based on both deterministic and stochastic strategies to find better Pareto approximations.

REFERENCES

Apkarian, P., Noll, D., and Rondepierre, A. (2008). Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control via nonsmooth optimization. *SIAM Journal on Control and Optimization*, 47(3):1516–1546.

Astrom, K., Panagopoulos, H., and Hagglund, T. (1998). Design of PI controllers based on non-convex optimization. *Automatica*, 34(5):585–601.

Leibfritz, F. (2004). COMPlib - Constrained Matrix-optimization Problem Library. Technical report, University of Trier. Department of Mathematics, Germany.

Sánchez, G., Villasana, M., and Strefezza, M. (2007). Multi-Objective Pole Placement with Evolutionary Algorithms. *Springer-Verlag. Berlin. Series: Lecture Notes in Computer Science*, 4403:417–427.

Scherer, C., Gahinet, P., and Chilali, M. (1997). Multiobjective output-feedback control via LMI optimization. *IEEE Transactions on Automatic Control*, 42(7):896–910.

Takahashi, R., Palhares, R., Dutra, D., and Goncalves, L. (2001). Synthesis and characterization of Pareto-optimal solutions for the mixed H2/Hinf problem. *Proceedings of the 40th IEEE conference on decision and control. Orlando,FL. USA*, pages 3997–4002.

Takahashi, R., Palhares, R., Dutra, D., and Goncalves, L. (2004). Estimation of Pareto sets in the mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem. *International Journal of System Science*, 35(1):55–67.