MULTIOBJECTIVE CONTROLLER DESIGN: OPTIMISING CONTROLLER STRUCTURE WITH GENETIC ALGORITHMS

A. Molina-Cristóbal[a] I. A. Griffin[b] P. J. Fleming[c] D. H. Owens[d]

Dept. Automatic Control and Systems Engineering,
The University of Sheffield, Mappin Street, Sheffield, S1 3JD, UK.
[a]Email: a.molina-cristobal@sheffield.ac.uk, [b]Email:i.griffin@sheffield.ac.uk
[c]Email: p.fleming@sheffield.ac.uk, [d]Email: d.h.owens@sheffield.ac.uk

Abstract: This paper studies the open problem of reduced- and fixed-order H_∞ synthesis. Often, this non-convex constraint is tackled with iterative convex optimisation procedure over LMI constraints. In this paper, an evolutionary approach is proposed such that the trial and error approach involved in LMI techniques might be overcome. The order of the controller is optimised as a multiobjective problem over a set of controller structures, H_∞, and time-domain specifications. Numerical results are presented with its counterpart the LMI procedure design, that show the advantage of investigating the Pareto optimal set resulting from the design procedure proposed.

Keywords: Multiobjective optimisation, genetic algorithms, linear matrix inequalities, fixed-order systems, order reduction, computer-aided control system design.

1. INTRODUCTION

Within the field of embedded control systems, it is desirable to implement a low order controller. When a state-space H_∞ technique is used, the order of the controller must be at least the same as the order of the plant. There are three alternatives to reduce the order of such controllers:

1. A reduced order approximation of the plant can be generated before designing the controller.
2. A reduced order approximation of the controller can be generated after controller design.
3. The order of the controller can be constrained during the design.

This paper examines the third alternative. Using linear matrix inequalities (LMIs) the intention is to include a constraint dim (A_c) < k for some k that is smaller than the dimension of A where A and A_c are the state matrices of the plant and controller respectively. The corresponding LMIs can be derived. However, they include rank constraints that are non-convex and difficult, if not impossible, to treat by current optimisation techniques (Scherer et al. 1997). However, recent research has developed some solutions to this problem; e.g. an algorithm in state space realization (Xin 2004) and a sufficient LMI condition in polynomial systems (Henrion 2003) have been proposed to overcome this non-convexity and can be used to fix the order of the controller.

Alternatively, the multiobjective genetic algorithm (MOGA) has been shown capable of optimizing controller structure and controller parameters (Chipperfield and Fleming 1996; Schroder 1998). A controller is designed by optimising the *controller structure* (order of the controller) and multiple design objectives in both the time domain (overshoot, undershoot and settling time) and frequency domain. In section 4, two examples of flexible structures have been solved numerically to illustrate the proposed procedure. The results and the procedure are compared with those obtained by LMI techniques.

2. PROBLEM STATEMENT

The mathematical framework employed in this work is polynomial systems. Consider the plant P of order n where

$$P = \frac{b_n}{a_n}, \tag{1}$$

and the SISO feedback configuration shown in Figure 1,

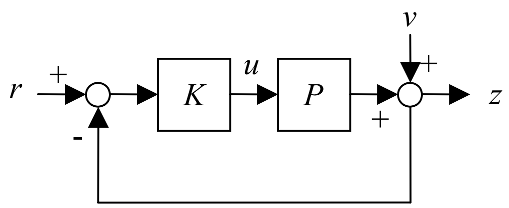


Fig. 1. Feedback system.

with coprime polynomials a_n and b_n, and where K is the controller of order m with coprime polynomials y_n and x_n,

$$K = \frac{y_n}{x_n}, \tag{2}$$

r is the reference signal and v is the disturbance and z is the control system output. Given a plant, P, the design of feedback control system consists of designing the controller, K, such that the resulting feedback system exhibits the desired performance. Another desired specification is that the system be internally *bounded-input bounded-output* (BIBO) stable. This means that for any bounded exogenous input in r and v, the internal signal u will be bounded too. This can be achieved by using the Youla-Kučera (YK) parametrization of stabilizing controllers Kučera (1979). For a given continuous-time plant, $P=b/a$, a stabilizing controller exists in a feedback configuration and all the controllers that stabilize the given plant are generated by all pairs of $\hat{x}$, $\hat{y}$ that solve the Bézout equation (3).

$$a\hat{x} + b\hat{y} = 1 \tag{3}$$

Traditionally, controller design with YK parametrization leads to high order controllers; however, recent research has shown that this problem can be overcome in the scalar case. The following results in this section are a brief description of this achievement on YK parametrization and the reader is referred to Lemma 1 in paper Henrion et al. (2004) for a more formal statement. The procedure starts by calculating an initial proper controller of order m. Such a controller can be obtained by placing the poles at arbitrary locations and solving the Diophantine equation in eq. (4):

$$a_n\hat{x}_n + b_n\hat{y}_n = a_d x_d, \tag{4}$$

where the degree of the polynomials a_d and x_d are n and $m=n$-1 respectively and $\hat{x}_n$ and $\hat{y}_n$ define the initial controller. Defining $a = a_n/a_d$, $b = b_n/a_d$, $\hat{x} = \hat{x}_n/x_d$ and $\hat{y} = \hat{y}_n/x_d$, equation (4) can be transformed into the Bézout equation (3) and provides the parametrization shown in eq (5):

$$K = \frac{\hat{y}_n - aq}{\hat{x}_n + bq}, \tag{5}$$

where $q = a_d q_n / x_d q_d$ is an arbitrary proper stable rational parameter. This parametrization can be expressed with the controller polynomials x_n and y_n in the arrangement of equation (6) indicating that the vector $N = [x_n \quad y_n \quad q_n \quad q_d]^T$ belongs to the null-space of a given polynomial matrix A.

$$\underbrace{\begin{bmatrix} a_d x_d & 0 & -b_n & -\hat{x}_n \\ 0 & a_d x_d & a_n & -\hat{y}_n \end{bmatrix}}_{A} \begin{bmatrix} x_n \\ y_n \\ q_n \\ q_d \end{bmatrix} = 0 \tag{6}$$

Assuming that q_n, q_d are polynomials and noting also that polynomial matrix A has full row rank, the dimension of the null-space will be equal to two. Let

$$N = \begin{bmatrix} x_{1n} & x_{2n} \\ y_{1n} & y_{2n} \\ q_{1n} & q_{2n} \\ q_{1d} & q_{2d} \end{bmatrix} \tag{7}$$

be a minimal basis for the null-space of matrix A, i.e. such that $AN = 0$ and the column degree of N are minimal among all possible null-space bases. By extracting a minimal polynomial basis N for the null-space of the polynomial matrix A defined in (6), all the stabilizing controllers with denominator and numerator polynomials x_n and y_n can be generated with the parametrization shown in eq. (8),

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} x_{1n} & x_{2n} \\ y_{1n} & y_{2n} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}, \tag{8}$$

where λ_1, λ_2 are polynomials such that the YK denominator polynomial, q_d, is stable and the corresponding YK numerator polynomial is given by

$$\begin{bmatrix} q_d \\ q_n \end{bmatrix} = \begin{bmatrix} q_{1d} & q_{2d} \\ q_{1n} & q_{2n} \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}. \tag{9}$$

Then a controller, K, of fixed order m can be found if there exist polynomials λ_1 and λ_2 of order m. The remainder of the controller design procedure using LMI techniques is fully explained in Henrion et al. (2004) and used later in this paper to find the LMI controllers. Alternatively, YK parametrization can be used to structure the controller (13), which is then optimised with the MOGA-based method (Section 3). The suitability of MOGA is due to a trade off between the controller order and the performance of the closed-loop system (Chipperfield and Fleming 1996). Hence, rather than a single optimal solution, this multiobjective problem results in a family of non-dominated or Pareto optimal solutions.

3. MULTIOBJECTIVE OPTIMISATION USING GENETIC ALGORITHMS

MOGA is an evolutionary algorithm that uses standard genetic algorithm (GA) operators (selection, crossover and mutation), Pareto ranking, fitness sharing and mating restriction. The design philosophy of MOGA is to develop a population of Pareto optimal or near Pareto-optimal solutions whilst maintaining the independence of the objectives throughout the optimisation process Fonseca and Fleming (1998).

Within MOGA, the initial population is randomly generated within a defined range and then decoded (in case of a non-real chromosome) to produce the corresponding vectors of decision variables. A set of objective function values is then evaluated for each individual within the population. The sequence of genetic operators is then applied, resulting in the subsequent generation of further potential solutions. MOGA employs multiobjective preference articulation extensions to the standard GA. Individuals are ranked, based on the objective vector and the designer preference's (goal, priority). The consideration of goal and priority selectively excludes objectives according to their priority and whether or not they meet their goals.

Although the population is potentially able to search many local optima, a finite population will tend to evolve towards a small region of the search space even if other equivalent optima exist. This phenomenon is known as genetic drift. A remedy to this problem has been proposed by Fonseca and Fleming (1995) with *Fitness Sharing*. This is a technique involving the estimation of the population density at the points defined by each individual by so called *kernel* methods, and is used to penalize individuals according to the proximity of other individuals. *Mating Restriction* specifies how close individuals should be in order to mate and it is used following *fitness sharing*. In addition, population diversity is encouraged by applying a mutation operator to a small number of the existing individuals.

3.1 Evaluation function

To define the control problem, an evaluation function is encoded, which contains all the objectives functions that the controller, *K,* has to satisfy. The performance evaluation function used in this work has seven objectives. The first two objectives are the H_∞ norm of the sensitivity function (10) and the complementary sensitivity function (11).

$$\|S\|_\infty = \left\| \frac{a_n x_n}{a_n x_n + b_n y_n} \right\|_\infty \tag{10}$$

$$\|T\|_\infty = \left\| \frac{b_n y_n}{a_n x_n + b_n y_n} \right\|_\infty \tag{11}$$

Because of the nature of the problems in this paper (flexible structures), a third objective called Total Variation (*TV*) (Owens and Chotai 1981) was included. This objective provides a measure of the oscillations (differences between peak and valley). The *TV* of a function *f*, is defined by

$$TV(f) = \max_{0 \le t_1 \le t_N} \sum_{i=1}^{N} \left| f(t_{i+1}) - f(t_i) \right| \tag{12}$$

In order to ensure good time-response performance, three time-domain objectives were also included: Undershoot, Overshoot, Settling time. An additional objective was included that measured the order of the controller. The chromosome structure shown in Figure 2 was implemented along the lines of Schroder (1998), where it was used for optimisation of weighting functions.

C	α_0	α_1	β_0	α_2	α_3	α_4	β_1
β_2	α_5	α_6	α_7	α_8	β_3	β_4	β_5

Fig. 2. Structure of the chromosome.

The variable C shown in Figure 2 selects the controller structure, and then the controller $K(\lambda_1,\lambda_2)$ is decoded according to equations (13). Finally, the controller $K=y_n/x_n$ can be obtained using equation (8).

$$K(\lambda_1,\lambda_2) = \begin{cases} 0 \le C < 1, & \lambda_1 = \alpha_0 + \alpha_1 s \\ & \lambda_2 = \beta_0 + s \\ 1 \le C < 2, & \lambda_1 = \alpha_2 + \alpha_3 s + \alpha_4 s^2 \\ & \lambda_2 = \beta_1 + \beta_2 s + s^2 \\ 2 \le C \le 3, & \lambda_1 = \alpha_5 + \alpha_6 s + \alpha_7 s^2 + \alpha_8 s^3 \\ & \lambda_2 = \beta_3 + \beta_4 s + \beta_5 s^2 + s^3 \end{cases} \tag{13}$$

4. NUMERICAL EXAMPLES

Both MATLAB Genetic Algorithm Toolbox with MOGA extension by Fonseca and Fleming (1998) and the LMI Algorithms developed by Henrion (2003); Henrion et al. (2004) were used to solve the following numerical problems.

4.1 Example 1: Low-order damping mode

$$P = \frac{b_n}{a_n} = \frac{1}{s(s^2+s+10)} \tag{14}$$

The control problem is to design a controller, *K*, so that the resulting feedback system exhibits a step response with no undershoot and the minimal possible overshoot and settling time as well as a high damping signal response. The first step is to obtain an initial controller, for example by solving the Diophantine equation (4) with arbitrary placement of the poles at $s = -1$, giving $a_d x_d = (s+1)^5$. The result is an initial controller with order $m=2$ and polynomials $\hat{x}_n = -4 + 4s + s^2$, $\hat{y}_n = 1 + 45s - 26s^2$. The only requirement of the initial controller is to be stabilizing. The minimal polynomial basis N for the nominal matrix A is given by

$$N = \begin{bmatrix} 0 & -1 & -4+4s+s^2 & -1 \\ 1 & -26 & -103+149s & -26+10s+s^2+s^3 \end{bmatrix}^T$$

By solving equations (8) and (9), all the stabilizing controllers can be computed from polynomials λ_1 and λ_2 in eq. (15) and it may be possible to find polynomials x_n and y_n (controllers *K*) that have low degree with a stable YK polynomial q_d (9),

$$K = \begin{cases} x_n = \lambda_2 \\ y_n = -\lambda_1 + 26\lambda_2 \end{cases}. \tag{15}$$

The H_∞ design procedure, LMI approach proposed in Henrion et al. (2004) was used to tackle this problem. The formulation to solve the control problem is as follows: Given the polynomials a_n and b_n from plant

(14) and the bound γ_s, the frequency domain specification on the sensitivity function (16) and the linear constraint (17), expressed as an LMI constraint, have to be simultaneously optimised. The H_∞ design algorithm was set up to solve the optimisation control problem, and then the polynomials x_n, q_d and q_n of a given degree were sought such that (16) and (17) are satisfied.

$$\|S\|_\infty = \left\| \frac{a_n x_n}{a_n x_n + b_n y_n} \right\|_\infty = \left\| \frac{a_n x_n}{q_d} \right\|_\infty < \gamma_s , \qquad (16)$$

$$a_d x_d x_n = b_n q_n + \hat{x}_n q_d . \qquad (17)$$

In each single run, a controller was found by using the concept of the central polynomial, *c(s),* which is the key design step. The H_∞ design procedure consists of iteratively adjusting the roots of *c(s),* while lowering the upper bound γ_s. After a series of trials a set of controllers of orders *m=1, 2* and *3* were obtained and the closed-loop step responses of the best controllers are shown in Figure 3.

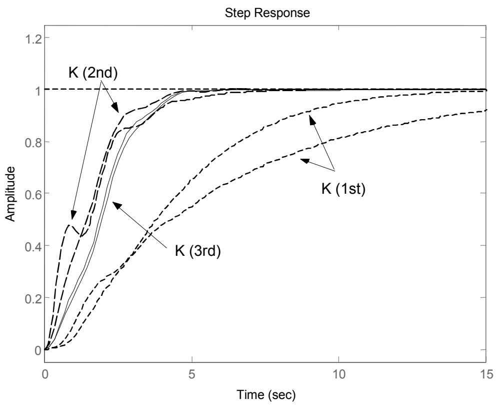
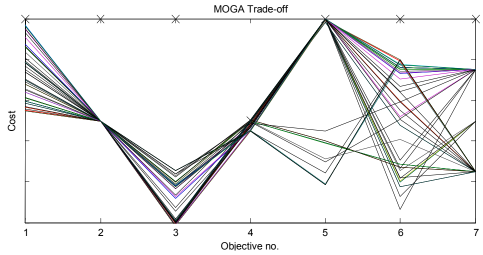
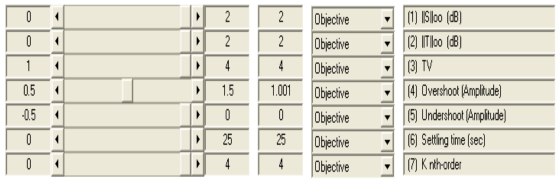


Fig. 3 Step response of: first-order controllers ('dotted line'), second-order controllers ('dashed line') and third order controllers ('solid line'):LMI approach.

The controller design problem was then tackled using the MOGA-based method. The decision variables used were the controller parameters defined in (13) in the chromosome structure from Figure 2. An evaluation function was encoded with seven objectives, as defined in Section 3.1. The MOGA parameters used were: 16 bit resolutions Gray coding and linear scaling, the crossover operator employed was shuffle with reduced surrogate with probability 0.7. The mutation operator was modified such that only decision variables that are currently active are mutated; the probability of applying mutation is 7e-4 and a population of 100 individuals. The algorithm was iterated for 100 generations. The results are illustrated in Figures 4 and 5. It can be seen that one single solution does not exist, rather a family of solutions. The trade-off of the objectives is shown in Figure 4. Each line represents a Pareto optimal solution, the *y-axis* shows the performance and the *x-axis* represents the objective. The cross-marks on the plot correspond to the goals and are connected by a dotted line. The objective 6 settling time, the objective 7 order of the controller and the objective 1 H_∞ norm of the sensitivity function appear to compete quite heavily. Here is where the control engineer can select depending on the specifications of the applications. Figure 5 illustrates the most representative step responses. Most of the step responses that have large settling times are first-order controllers. The second-order and third order controllers show better settling time and this is confirmed by looking at the trade-off graph in Figure 4a.



(a)



(b)

Fig. 4. (a) Trade-off graph for the controllers K_{MOGA} (b) Graphical User Interface (GUI) for preference articulation and design objectives.

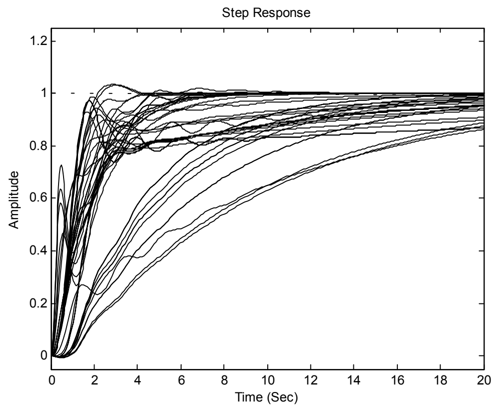


Fig. 5. Family of preferable step responses (MOGA).

By changing the values of the goals, the search is forced to examine other areas of the trade-off surface. Note for example, different values of TV produce range of responses from smooth to oscillatory. Taking advantage of the initial information, some of the goals (objective 1, 2, 3 and 6) were tightened (new goal vector = [1.5 1.5 1.2 1.001 0 4 4]) and the controller order was reduced to 2. The new results are displayed in Figure 6, which reveals the controllers that meet the new design requirements. Figure 7 illustrates the "best" closed-loop step responses obtained by the MOGA. The step response produced by the MOGA controller (18) has time-domain characteristics *TV* =1 which means no variations, no overshoot, no undershoot and settling time = 3.3 sec.

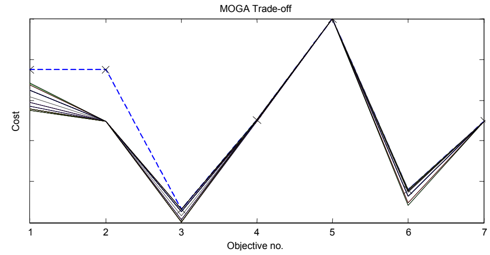


Fig 6. New trade-off graph for the controllers K_{MOGA}.

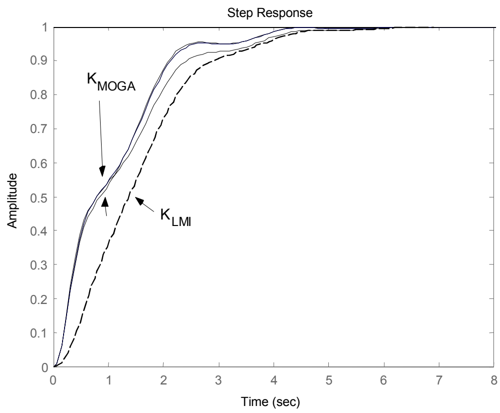


Fig 7. Flexible mode closed-loop step responses of MOGA controllers ('solid line') and the best LMI controller ('dashed line').

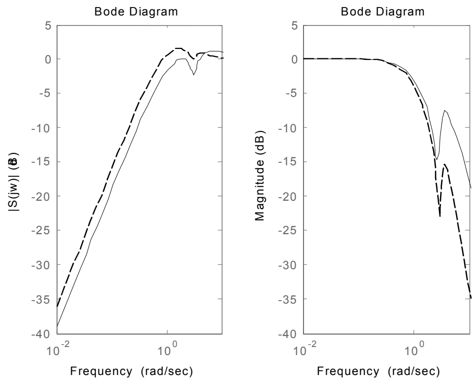


Fig 8 Flexible mode closed-loop frequency-domain characteristics of the MOGA controller ('solid line') and the LMI controller ('dashed line').

The numerical results in Figure 7 also show the "best" step response produced by the LMI controller with no overshoot, no undershoot and settling time = 3.8 sec. This LMI controller can be found in (Henrion et al. 2004). The resulting controller from the MOGA method is:

$$K(s)_{MOGA} = \frac{y_n}{x_n} = \frac{1226 + 107.436s + 172.24s^2}{135.13 + 15.169s + s^2} \quad (18)$$

In Figure 8, the closed-loop frequency domain performance characteristics of the MOGA controller are $\|T\|_\infty$= 0 dB and $\|S\|_\infty$ =1.2 dB and those by the LMI controller are $\|T\|_\infty$= 0 dB and $\|S\|_\infty$ =1.5 dB

4.2 Example 2: Flexible Beam (Doyle et al. 1992):

$$P = \frac{-6.475s^2 + 4.0302s + 175.77}{s(5s^3 + 3.5682s^2 + 139.5021s + 0.0929)} \quad (19)$$

The control problem is to design a controller, K, so that the resulting feedback system exhibits a step response with overshoot no greater than 10% and settling time approximately 8 sec. The procedure is the same as in Example 1. A new initial controller was calculated by solving the Diophantine equation (4) with arbitrary placement of the poles at s=-1, then $a_d x_d = (s+1)^7$. The controller design problem was then tackled using the MOGA-based method. All the settings of the MOGA were the same as for Example 1. The goal vector was set up to search for controllers of first, second and third order and the remaining objectives were left free. In this way a Pareto optimal front was found. This information allows the control engineer to visualise which specification can be achieved and if a reduction in the order of the controller will achieve good performance (see Figure 9).

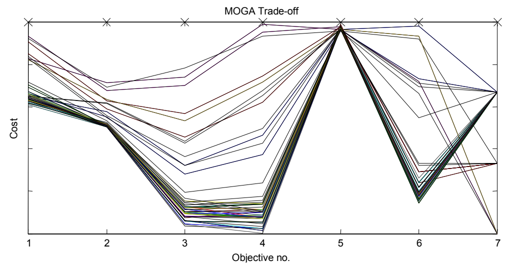


Fig. 9. Trade off graph.

According to the time domain specifications and the information from Figure 9, all the goals were tightened except the undershoot and the new goal vector was set to [1.5 1.5 1.3 0.1 0 8 2]. The new Trade-off results are shown in Figure 10.

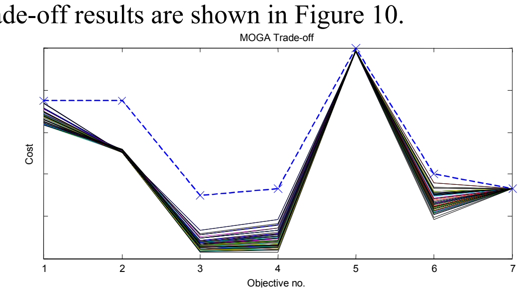


Fig 10. New trade-off graph for the controllers K_{MOGA} for example 2.

Figure 11 illustrates the "best" closed-loop step responses obtained by the MOGA and LMI techniques (Henrion 2003). Both controllers produced very similar step responses. In Figure 12 the closed-loop frequency domain performance measures of the MOGA controller in eq. (20) are $\|T\|_\infty$ = 0 dB and $\|S\|_\infty$ =2.07 dB and those of the LMI controller are $\|T\|_\infty$ = 0 dB and $\|S\|_\infty$ =1.5 dB. It is important to mention that similar specifications, in Doyle et al. (1992), were achieved with H_∞ state space techniques. However, the controller was of order eight. The resulting controller is:

$$K_{MOGA} = \frac{0.56229 \cdot 10^{-2} + 1.3052s - 1.3691 \cdot 10^{-2} s^2}{3.851 + 3.73s + s^2} \quad (20).$$

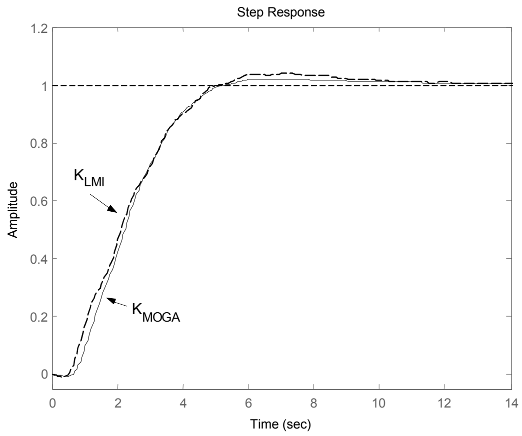


Fig 11. Flexible Beam closed-loop step responses of MOGA controller ('solid line') and the best LMI controller ('dashed line').

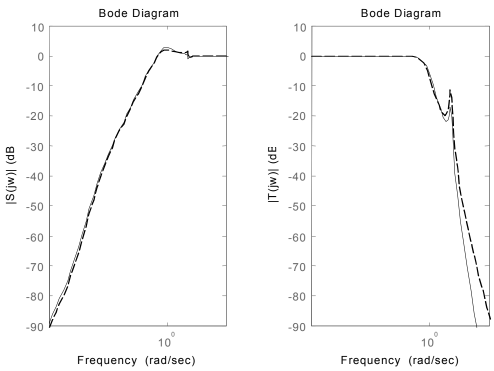


Fig 12 Flexible Beam closed-loop frequency-domain characteristics of the MOGA controller ('solid line') and the LMI controller ('dashed line').

The algorithms were run on a standard PC with Pentium IV, 1.7GHz processor. The computational cost of solving the examples with MOGA was CPU time 15min to 25min. A single run of LMI algorithms was CPU time 15 sec. Some experiments showed that MOGA with YK parametrization enhances the population of closed-loop stable controllers providing a reduction in the computational burden more so than the same formulation without YK parametrization. The MOGA approach has been used to deals with MIMO systems. However, in this paper, the controller structure (13) relies on the preliminary results of YK parametrization by Henrion et al. (2004), and thus only applies to the SISO case.

5. CONCLUSIONS

The LMI optimisation for fixed-order H_∞ controller design can be computed quickly. However, the whole procedure of designing a controller requires several attempts. Multiple trials changing the central polynomial, bound on the frequency domain specification and the order of the controller have to be carried out until the specification seems to be satisfied or no progress is noticed. Despite the computational cost of the MOGA, the Pareto surface between controller order and closed-loop performance can be investigated in a single run. Thus, the control engineer has a choice from among the family of Pareto optimal solutions, and design time is saved. In general it is rather difficult with LMI techniques to cope with time-domain specifications (non-convex specifications), whereas the MOGA approach affords a straightforward way of dealing with them. Moreover, a mixture of objectives with a considerable mathematical complexity can be included in the evaluation function of MOGA with excellent flexibility.

ACKNOWLEDGMENT

The authors gratefully acknowledge useful comments on this paper by D. Henrion.

REFERENCES

Chipperfield, A. and Fleming, P. J. (1996). Multiobjective gas turbine engine controller design using genetic algorithms. *IEEE Transactions on Industrial Electronics* **43**(5): 583-587.

Doyle, J. C., Francis, B. A. and R., T. A. (1992). *Feedback Control Theory*. New York, MacMillan.

Fonseca, C. M. and Fleming, P. J. (1995). Multiobjective Genetic Algorithms Made Easy: Selection, Sharing and Mating Restriction. *Evolutionary Computation. Proc 1st IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems*, Galesia: 45-52.

Fonseca, C. M. and Fleming, P. J. (1998). Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation. *IEEE Transactions on Systems, Man and Cybernetics-Part A: systems and humans* **28**(1): 26-37.

Henrion, D. (2003). LMI optimization for fixed-order H-infinity controller design. *Proc of the IEEE Conference on Decision and Control*, Maui, Hawaii.

Henrion, D., Kučera, V. and Molina-Cristobal, A. (2004). Optimizing simultaneously over the numerator and denominator polynomials in the Youla-Kučera parametrization. *Proc of the 43rd IEEE Conference on Decision and Control*, Paradise Island, Bahamas.

Kučera, V. (1979). *Discrete Linear Control: The Polynomial Equation Approach*. Chinchester, UK, John Wiley and Sons.

Owens, D. H. and Chotai, A. (1981). Norm Bounds for Linear Systems. Sheffield, Department of Control Engineering, University of Sheffield: Research Report No. 159.

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

Schroder, P. (1998). Multivariable Control of Active Magnetic Bearings. Department of Automatic Control and Systems Engineering. PhD thesis, The University of Sheffield, Sheffield UK.

Xin, X. (2004). Reduced-order controllers for the H-Infinity control problem with unstable invariant zeros. *Automatica* **40**(2): 319-326.