GAS TURBINE ENGINE CONTROLLER DESIGN USING MULTIOBJECTIVE GENETIC ALGORITHMS

A Chipperfield and P Fleming

University of Sheffield, UK

email: A.Chipperfield@Sheffield.ac.uk

ABSTRACT

This paper describes the use of multiobjective genetic algorithms (MOGAs) in the design of a multivariable control system for a gas turbine engine. It is shown how the MOGA confers an immedeate advantage over conventional multiobjective optimization methods by evolving a family of Pareto-optimal solutions allowing the control engineer to examine the trade-offs between the different design objectives. In addition, the paper demonstrates how the genetic algorithm can be used to search in both controller structure and parameter space thereby offering a potentially more general approach to optimization in controller design than traditional numerical methods.

INTRODUCTION

Modern gas turbine engines are highly complex systems that require equally complex controllers in order to remain stable whilst satisfying the demands placed on them by the pilot and operating conditions. Because of these complexities, extensive use is made of computer aided control system design (CACSD) methods to design controllers to meet the desired performance specifications. In particular, optimization based methods in CACSD have been shown to be a valuable tool in assisting the control engineer in selecting suitable controller parameters [1].

However, parametric optimization methods are numerically intensive and require repeated application to identify the trade-offs between different design objectives. In this paper, we consider the application of multiobjective genetic algorithms (MOGAs) to the design of gas turbine engine control systems. It is shown that the MOGA confers an immediate advantage over conventional multiobjective optimization methods by evolving a family of Pareto-optimal solutions. Thus, the relative trade-offs between design objectives may be easily identified and a more informed choice made for the final controller structure.

MULTIOBJECTIVE OPTIMIZATION

The use of multiobjective optimization (MO) recognises that most practical problems require a number of design criteria to be satisfied simultaneously, viz:

$$\min_{x \in \Omega} \quad F(x)$$

where $x = [x_1, x_2, \ldots, x_n]$ and Ω define the set of free variables, x, subject to any constraints and $F(x) = [f_1(x), f_2(x), \ldots, f_n(x)]$ are the design objectives to be minimised.

Clearly, for this set of functions, $F(x)$, it can be seen that there is no one ideal "optimal" solution, rather a set of Pareto-optimal solutions for which an improvement in one of the design objectives will lead to a degradation in one or more of the remaining objectives. Such solutions are also known as non-inferior or non-dominated solutions to the MO problem.

Conventionally, members of the Pareto-optimal solution set are sought through solution of an appropriately formulated nonlinear programming problem. A number of approaches are currently employed including the ε-constraint, weighted sum and goal attainment methods [2]. However, such approaches require precise expression of a, usually not well understood, set of weights and goals. If the trade-off surface between the design objectives is to be better understood, repeated application

Genetic Algorithms in Engineering Systems: Innovations and Applications
12-14 September 1995, Conference Publication No. 414,

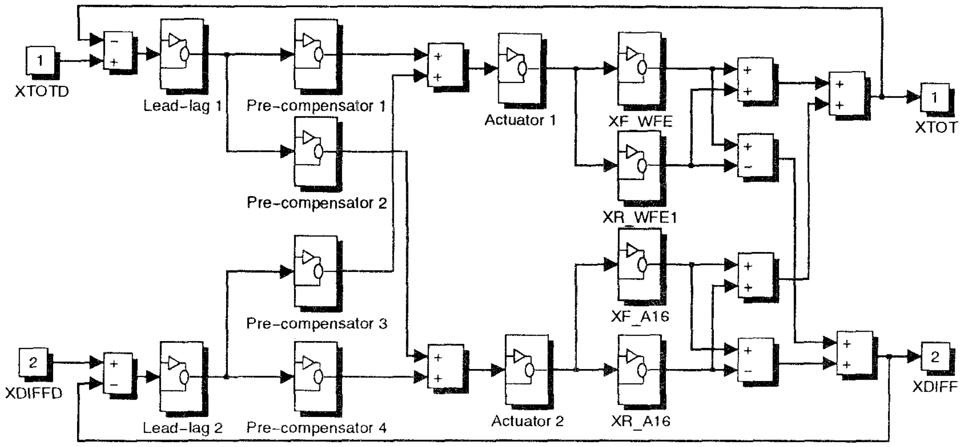


Figure 1: SIMULINK model of the ASTOVL engine

of such methods will be necessary. In addition, nonlinear programming methods cannot handle multimodality and discontinuities in function space well and can thus only be expected to produce local solutions.

Multiobjective GAs [3] evolve a population of solution estimates thereby conferring an immediate benefit over conventional MO methods. Using rank-based selection and nicheing techniques, it is feasible to generate populations of non-dominated solution estimates without combining objectives in some way. This is advantageous because the combination of non-commensurate objectives requires precise understanding of the interplay between those objectives if the optimization is to be meaningful. The use of rank-based fitness assignment permits different non-dominated individuals to be sampled at the same rate thereby according the same preference to all Pareto-optimal solutions.

Because MOGAs are susceptible to unstable converged populations, due to the potential for very different genotypes to result in non-dominated individuals, a particular problem is the production of lethals when fit members of the population are mated. The search then becomes inefficient and the GA is likely to converge to some suboptimal solution. However, the use of mating restrictions, to reduce the production of lethals, enhances the stability of the population whilst allowing a wide diversity in genetic material.

ASTOVL EXAMPLE

This example application demonstrates how GAs may be used to select the controller structure and suitable parameter sets for a multivariable flight control system. The system considered is a propulsion unit for an Advanced Short Take-Off, Vertical Landing (ASTOVL) aero-engine [4], shown in Fig. 1. There are two inputs to the system, XTOTD and XDIFFD, and it is required that the pilot have control of the fore-aft differential thrust (XDIFF) and the total engine thrust (XTOT). The design problem is to find a set of pre-compensators that satisfy a number of time-response design specifications whilst minimizing the interaction between the loops of the system.

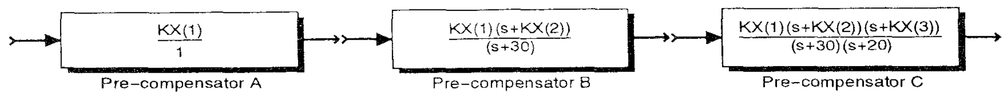
The time-domain performance requirements, in response to a step in demand at one of the inputs, are:

(i) 70% rise-time ≤ 0.35 seconds
(ii) 10% settling-time ≤ 0.5 seconds
(iii) maximum overshoot ≤ 10%

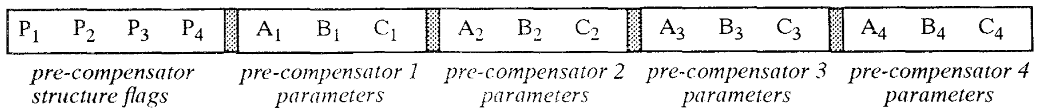
at the associated output. The amount of interaction, or cross-coupling, between modes is measured as:

$$\int_0^\infty (XTOT)^2 dt,$$

when excited by a step input to XDIFFD, and vice-versa, and should be less than 0.05 for this example.



(a) pre-compensator structures



(b) chromosome structure

Figure 2: Representing the pre-compensator structures and parameter sets

IMPLEMENTATION

The ASTOVL propulsion unit was modelled directly using the SIMULINK package as shown in Fig. 1. The objective functions were written as m-files using commands from the Control Systems Toolbox. The pre-compensators for this problem were allowed to be either first or second order or simple gains, Fig. 2(a).

Using a structured chromosome representation [5], Fig. 2(b), it is possible to allow the free parameters for each possible pre-compensator configuration to reside in all individuals. Here, high-level genes, labelled P_1 to P_4 in Fig. 2(b) and encoded as integers, are used to determine which pre-compensator structures are active in a particular chromosome. Associated with each pre-compensator, P_i, are three sets of real-valued parameters, A_i, B_i and C_i, corresponding to the gains and time constants of the permissible pre-compensator structures. Thus, the values of the pre-compensator structure flags select which set of parameters are valid with each pre-compensator and therefore the order of the pre-compensators. In this way, a chromosome may contain a number of possibly good representations at any one time, although only the set defined by the values of the high-level genes will be active.

In addition, as the overall structure of the controller varies with the set of active pre-compensators, an additional objective was included that measured the complexity of the controller. This was calculated by summing the values of the pre-compensator flags thus:

$$\sum_{i=1}^{4} P_i \leq 9.$$

Thus, a total chromosome length of 28 elements was used and nine design objectives should be satisfied.

The MATLAB Genetic Algorithm Toolbox [6], was used to implement the GA with additional extensions to accommodate multiobjective ranking, sharing and mating restrictions in the objective domain [3].

Multiobjective ranking is based upon the dominance of an individual, how many individuals outperform it in objective space, combined with goal and priority information. In this example, the goals were set to the values given in the previous Section and all objective were assigned the same priority. In cases where objectives are assigned different priorities, higher priority objectives are optimized in a Pareto fashion until their goals are met at which point the remaining objectives are optimized (see, [7]).

Niche induction techniques [8] provide a mechanism for uniform sampling of non-dominated individuals in the region of the trade-off surface relevant to the optimization. Fitness sharing, implemented in the objective domain [9] [3], favours sparsely populated regions of the trade-off surface and may be combined with mating restrictions to reduce the production low performance individuals by encouraging the mating of individuals similar to one another.

The crossover operator employed was intermediate recombination [10] applied with probability 0.7. As the chromosome contains many inactive elements, the probability of applying breeder GA mutation [10] was set to 0.1. The use of adaptive mutation rates may have been more appropriate for this example and representation, although the (seemingly) high mutation rate is consistent with the use of real-valued operators and the average number of active parameters. No fine-tuning of operator rates was attempted.

Finally, in order to reduce the computational burden of evaluating the objectives, re-evaluation of individuals was only performed if they had been affected by the genetic operators [11]. This reduced the number function evaluations required by 20 to 30%.

RESULTS

Using a population size of 40, the GA was run for 100 generations in the first instance. A list of the best 50 individuals was continually maintained during the execution of the GA allowing the final selection of controller to be made from the best structures found by the GA over all generations.

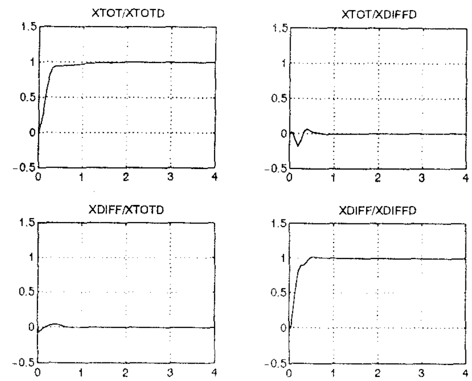
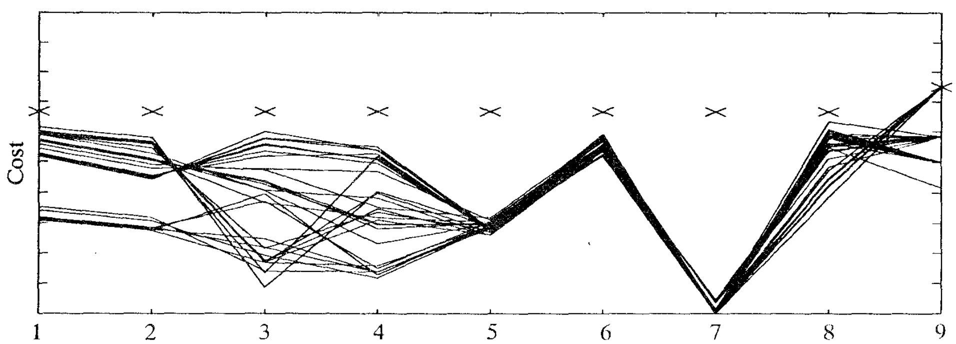


Figure 3: Typical optimized ASTOVL response

Fig. 3 shows a typical response for a controller found by the MOGA. It can be clearly seen that all of the design objective have been satisfied. However, from such responses it is difficult to determine the relative merits of one controller against another over the entire population. This is particularly true if on-line preference



0	0.525	0.35	Objective	XTOT rise-time
0	0.75	0.5	Objective	XTOT settling-time
0	15	10	Objective	XTOT overshoot
0	0.075	0.05	Objective	XDIFF/XTOTD
0	0.525	0.35	Objective	XDIFF rise-time
0	0.75	0.5	Objective	XDIFF settling-time
0	15	10	Objective	XDIFF overshoot
0	0.075	0.05	Objective	XTOT/XDIFFD
0	12	9	Objective	Controller complexity

Figure 4: Sample ASTOVL trade-off graph and design objectives

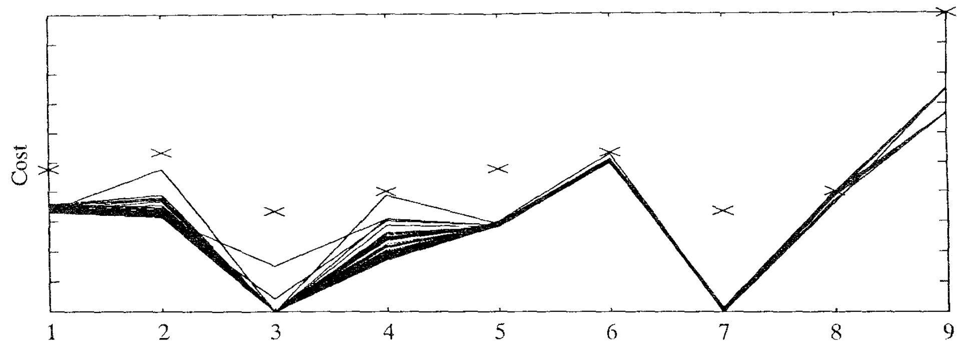


Figure 5: Trade-off graph for revised design objectives

articulation is to be used to guide the search during an optimization.

Fig. 4 illustrates a typical trade-off graph for the ASTOVL controller and a user interface for interactively setting design goals and examining non-dominated solutions in a population or database. In the plot, each line represents a non-dominated individual found by the MOGA. Objectives 1, 2 and 3 are the rise-time, settling-time and overshoot respectively for the XTOT channel and objective 4 is the cross-coupling between XTOTD and XDIFF. Objectives 5 to 8 correspond to the same design specifications on the XDIFF channel and objective 9 is the overall controller complexity. The *y*-axis shows the performance of individuals in each objective domain with cross-marks showing the design goals.

In Fig. 4, trade-offs between adjacent design objectives result in the crossing of lines between them whereas non-crossing lines indicate that objectives do not compete with one another. For example, the XTOT channel settling-time, overshoot and cross-coupling (objectives 2, 3 and 4) appear to compete quite heavily while the same trade-offs are not exhibited by the XDIFF channel. Only the preferred individuals, those that satisfy the design goals, are shown. When no individuals are preferred, the non-dominated individuals are displayed. An additional feature of the user interface is the ability to move the position of the objectives on the *x*-axis. This affords the control engineer a convenient mechanism for examining the trade-offs between non-adjacent design objectives.

Having satisfied the original design goals, the control engineer is now free to enhance the performance of the controllers. The relative degree of under-/over-attainment of the design goals is clearly visible in Fig. 4 and the designer may take advantage of this information when setting new design goals.

Fig. 5 shows the new trade-off graph produced when the goals are reset and the MOGA is allowed to continues for a further 25 generations. The cross-marks on the plot correspond to the new goal set of [0.25, 0.4, 5.0, 0.03, 0.25, 0.4, 5.0, 0.03, 12]. In this goal set, all of the performance goals have been tightened while the controller complexity has been relaxed. This allows more complex controllers to be considered in order to meet the stricter performance requirements. However, as many satisfactory structures already exist in this region, the most complex controllers have not had their parameters tuned sufficiently to meet these new design requirements.

By changing the values of the goals, the search is forced to examine a smaller area of the trade-off surface. Individuals that do not now satisfy the design goals are no longer preferred and the population is forced to evolve towards a new region of the design space. Thus, a more accurate picture of the trade-off surface in that region is constructed.

CONCLUDING REMARKS

This paper has shown how MOGAs may be applied to the design of gas turbine engine control systems. Using a single unified formulation, a number of competing design objectives may be simultaneously optimized through search in both controller structure and parameter space. The MOGA approach has a clear advantage over conventional multiobjective optimization methods in that it allows a number of non-dominated controller structures to be examined in a single design cycle. In addition, the software tools described in this paper have been built upon a standard and familiar CACSD software package, MATLAB. This allows the retention of existing modelling and simulation routines and facilitates the rapid development of objective functions for complex systems.

A simple user interface has been demonstrated that allows the control engineer to examine the trade-offs and interplay between design objectives. The control engineer may interact with the MOGA through the successive articulation of preferences, guiding the optimization on the basis of design requirements rather than the properties of the objective functions. Such a process allows a closer interaction between the control engineer and the primary design tools, hopefully leading to a more informed design procedure.

Whilst interactive use is desirable, the numerically intensive nature of evaluating objective functions may render such an approach infeasible. In such cases, parallel processing techniques could be employed to alleviate the computational burden. Similarly, the nicheing mechanisms which arise in some distributed population structures may prove beneficial to the MOGA. Finally, whilst this paper has considered the application of MOGAs to gas turbine engine design, the procedures and techniques discussed should prove useful in the wider field of CACSD and CAE in general.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of this research by a UK EPSRC grant on "Evolutionary Algorithms in Systems Integration and Performance Optimization" (GR/K 36591). The authors also wish to thank Carlos Fonseca for the multiobjective extensions to the GA Toolbox and the user interface.

REFERENCES

[1] H. Mukai, "Algorithms for Multicriterion Optimization", *IEEE Trans. Automat. Contr.*, Vol. AC-25, No. 2, pp. 177-186, 1980.

[2] C. -L. Hwang and A. S. M. Masud, *Multiple Objective Decision Making - Methods and Applications: A State of the Art Survey*, Springer-Verlag, Germany, 1979.

[3] C. M. Fonseca and P. J. Fleming, "Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalisation", *Proc. ICGA 5*, pp. 416-423, 1993.

[4] S. D. Hancock, *Gas Turbine Engine Controller Design Using Multi-Objective Optimization Techniques*, PhD Thesis, University of Wales, Bangor, 1992.

[5] D. Dasgupta and D. R. McGregor, "Nonstationary Function Optimization using the Structured Genetic Algorithm", *PPSN 2*, pp. 145-154, 1992.

[6] A. J. Chipperfield, P. J. Fleming and H. P. Pohlheim, "A Genetic Algorithm Toolbox for MATLAB", *Proc. Int. Conf. Systems Engineering*, Coventry, UK, 6-8 Sept., pp. 200-207, 1994.

[7] C. M. Fonseca and P. J. Fleming, "Multiobjective Optimal Controller Design With Genetic Algorithms", *Proc. IEE Control '94*, pp. 745-749, 1994.

[8] K. Deb and D. E. Goldberg, "An Investigation of Niche and Species Formation in Genetic Function Optimization", *Proc. ICGA 3*, pp. 42-50, 1989.

[9] C. M. Fonseca and P. J. Fleming, "Multiobjective Genetic Algorithms Made Easy: Selection, Sharing and Mating Restriction", submitted to *IEE/IEEE GALESIA* '95, 1995.

[10] H. Mühlenbein and D. Schlierkamp-Voosen, "Predictive Models for the Breeder Genetic Algorithm: I. Continuous Parameter Optimization", *Evolutionary Computation*, Vol. 1, No. 1, pp. 25-49, 1993.

[11] P. Oliveira, J. Sequeira and J. Sentieiro, "Selection of Controller Parameters using Genetic Algorithms", *Engineering Systems with Intelligence*, Kluwer Academic, pp. 431-438, 1991.