MULTIOBJECTIVE OPTIMAL CONTROLLER DESIGN WITH GENETIC ALGORITHMS

C M Fonseca and P J Fleming

University of Sheffield, United Kingdom

INTRODUCTION

Finding a controller for a given plant in order to achieve a number of design objectives is a common control design problem. As well as closed-loop plant stability, design objectives often include measures such as rise time, settling time, overshoot, asymptotic tracking, decoupling and regulation, gain and phase margins, small disturbance response and bounds on frequency response magnitudes.

Theoretical results (1) show that, in the linear time invariant case, all the objectives mentioned above are convex with respect to the closed-loop transfer function. They also show that, once a stabilizing controller is found, the design problem can be formulated as a convex multiobjective optimization problem in Q, a transfer function design parameter, which is a mapping of all stabilizing controllers. However, when addressed in the controller parameter domain, the same design problem is not necessarily convex, and generally difficult to solve.

Other objectives, such as open-loop controller stability and low controller complexity, may not result in convex optimization problems and, therefore, are not included in the results mentioned above. Non-linear systems pose similar difficulties, as their mathematical treatment is generally more complex.

Genetic algorithms (2, 3) have been shown to be useful in addressing ill-behaved optimization problems, being able to cope with discontinuities, multimodality and uncertain function evaluations, and their single objective formulation has been extended by the authors to include multiple objectives. The paper shows how genetic search can be interactively used to design controllers of given complexity, in a multiobjective sense, while learning about the trade-off between the design objectives.

MULTIOBJECTIVE OPTIMIZATION

The solution of a multiobjective optimization (MO) problem generally consists of a family of points, the Pareto-optimal set. Points in this set are such that improvement in any one objective can only be achieved at the expense of degradation in at least one of the remaining objectives. Pareto-optimal points are also called non-dominated, or non-inferior, solutions to the MO problem.

Methods such as the weighted sum approach, the ε-constraint method and goal programming (4) have been conventionally used to search for non-inferior solutions. Consisting of appropriate non-linear programming formulations of the multiobjective problem, all of them produce one solution at a time, and require a precise expression of usually not well known weights and/or priorities, prior to the optimization process. If the trade-off between objectives is to be better understood, repeated application of such methods is required. Also, discontinuities and multimodality are not satisfactorily handled by conventional gradient-based optimizers, which can only be expected to produce local solutions.

Genetic algorithms (GAs), on the other hand, can search for many non-inferior solutions in parallel, while being better able to cope with ill-behaved functions. The multiobjective formulation of the genetic algorithm (5) enables the designer, here the decision maker (DM), to progressively articulate their preferences as the optimization proceeds. The intermediate trade-off information produced by the approach, though generally sub-optimal, gives the DM useful insight into the problem before preferences are refined. Computational effort can, in this way, be utilized in the optimization of the final design rather than other non-dominated, but later discarded, solutions.

OVERVIEW OF MULTIOBJECTIVE GENETIC ALGORITHMS

Multiobjective genetic algorithms (MOGAs) differ from conventional GAs at the selection level. The concept of Pareto dominance is used in conjunction with the designer's preferences to assess individual performance, maintaining objectives separate throughout the optimization process.

Rank-based selection

Individuals are ranked on the basis of how many individuals in the current population strictly outperform them. In a simple Pareto GA, individuals are compared according to dominance. Combining dom-

CONTROL '94, 21–24 March 1994. Conference Publication No. 389, © IEE 1994.

inance with preferences expressed by an external DM in terms of goals and priorities, a new relational operator (preferable to) can be defined. The on-line change of preferences allows the search to be effectively guided towards particular regions of the Pareto optimal surface without constraining it in decision variable space.

Certain objectives, such as, for example, closed-loop stability, need to be satisfied before the optimization of other objectives can take place. This can be seen as them having priority over the remaining. Prioritization of objectives constitutes an extension to the MOGA formulation proposed in (5) and will be reported in full in a future paper.

Higher priority objectives are optimized in a Pareto fashion until all of them meet their goals, at which point the optimization of the conventional objectives takes place. In this way, any infeasible individual (e.g., a non-stabilizing controller) is always considered worse than any feasible one. Also, whenever only infeasible individuals exist in the population, the optimization of the high priority objectives provides the necessary basis for evolution towards feasible solutions.

Niche induction techniques

The preferred region of the Pareto-optimal surface is a region of flat fitness for the GA and, therefore, a phenomenon known as genetic drift may occur. Genetic drift consists of individuals clustering around certain optimal regions as opposed to others of equivalent fitness for no reason other than stochastic errors associated with the selection process and due to the population being finite.

The other important aspect of the MOGA is the use of niche induction techniques (6) to promote and maintain the uniform sampling of the region of the trade-off surface relevant to the DM. Fitness sharing, implemented in the objective domain, penalizes individuals in more populated regions of the trade-off surface, in favour of those more isolated. Mating restriction reduces the formation of low performers by promoting the mating of individuals similar to one another.

THE DESIGN PROBLEM

The problem considered here has been proposed earlier by Barrat and Boyd (7). It consists of designing a discrete-time regulator for the single-input single-output plant

$$P(s) = \frac{1}{s^2} \cdot \frac{4-s}{4+s} \tag{1}$$

Such a double integrator plant with excess phase provides a simple but realistic basis for illustrating de-

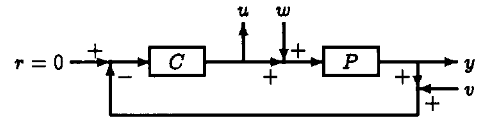

Fig. 1: Closed-loop system

sign trade-offs. The discretization of the plant using a zero-order hold at 10 Hz gives the transfer function

$$P(z) = \frac{-0.00379\,(z-1.492)\,(z+0.7679)}{(z-1)^2\,(z-0.6703)} \tag{2}$$

The complete regulator system is presented in Figure 1. The discrete-time outputs u and y are, respectively, the actuator and the plant outputs. The discrete-time inputs w and v represent actuator and sensor noise, and are considered to be driven by independent white noise.

Design objectives

A number of objectives including noise sensitivity and robustness measures is considered here. A controller of given complexity is sought which represents a compromise between some or all of the following objectives.

Closed-loop stability. This is probably the most basic objective to be satisfied. If no stabilizing controller is known for a particular plant, non-stabilizing controllers can still be ranked according to how far from being stabilizing they are, as indicated by, for example, the maximum of the absolute values of the poles of the corresponding closed-loop system.

Stabilizing controllers can be found by minimizing the degree of instability of the closed-loop system until stability is achieved, at which point the remaining objectives become active. Since closed-loop stability is an absolute design requirement, it is set up as a high priority objective.

Output and actuator variance. The trade-off between steady-state output and actuator variances due to the presence of process and measurement noise, if considered in isolation from other objectives, can be computed analytically from LQG theory. The approach is to find the minimum of the linear combination of the two variances

$$J = \lim_{k\to+\infty} E\{y_k^2 + \rho u_k^2\} \tag{3}$$

for various settings of the parameter ρ until suitable values of output and actuator variance are found.

The complexity of the regulators found analytically is directly related to that of the plant. The need for simpler controllers has prompted much interest and work in the area of model reduction. In fact, there is no analytical solution for controllers with arbitrarily fixed complexity, the design of which requires the use of optimization techniques.

Sensitivity to additive loop perturbations. The M-circle radius, defined as the minimum distance from the Nyquist plot of the loop gain $PC(\exp j\omega T)$ to the critical point -1, is a measure of robustness which combines both gain and phase margins. It relates to the maximum sensitivity of the system, defined as

$$\|S\|_\infty = \left\|\frac{1}{1+PC}\right\|_\infty \tag{4}$$

in the following way:

$$M = 1/\|S\|_\infty \tag{5}$$

Therefore, minimizing the maximum sensitivity, $\|S\|_\infty$, corresponds to maximizing the M-circle radius, and thus the stability margin of the system. Variations in the loop gain may appear as a consequence of variations in the parameters of the system.

Sensitivity to additive plant perturbations. This second measure of robustness expresses the ability of the regulator to maintain closed-loop stability in the presence of stable additive plant perturbations. Characterizing plant perturbations ΔP in terms of the maximum magnitude of their frequency response, the smallest stable perturbation D which will destabilize the closed-loop system is known to be inversely proportional to the maximum magnitude of the closed-loop transfer function from r to u,

$$1/D = \left\|\frac{C}{1+PC}\right\|_\infty \tag{6}$$

Additive plant perturbations may arise from inaccurate modelling of the plant, either due to tolerances in the parameters or ignored plant dynamics.

Open-loop controller stability. It is often required that controllers be open-loop stable. This constraint can be implemented simply by requiring the maximum absolute value of the controller poles to be less than one.

IMPLEMENTATION

The several objectives and all GA routines were written as MATLAB M and MEX files. The Genetic Algorithm Toolbox (8) was used to implement the GA, while the objective functions made extensive use of the relevant routines in the Control Systems Toolbox.

Parameter encoding

The genetic algorithm requires decision variables, here the controller parameters, to be encoded into a bit string. Controllers were parameterized in terms of their roots (poles and zeros) and their gain. The fact that an infinite number of different systems, represented by possibly very different pole-zero patterns, can exhibit the same input-output behaviour within arbitrary finite accuracy (9) makes the pole-zero domain very rich in approximate solutions for the GA to explore.

Zeros and poles were defined through pairs of Gray encoded real parameters, respectively the average α and the deviation from the average β of each pair of roots, as suggested by Kristinsson and Dumont (10). A positive deviation indicates real roots and negative deviation indicates complex conjugate roots.

A pair of zeros and a pair of poles were encoded as α and β pairs in the interval between -1 and 1. Most controllers defined in this way are open-loop stable and minimum phase. The gain was also Gray encoded, in the interval between 0 and 100. 16 bits were used for each parameter, which lead to 80-bit long chromosomes.

Genetic operators

The MOGA consisted of a standard generational GA with multiobjective ranking and sharing and mating restriction implemented in the objective domain. Linear ranking imposed a fixed selective pressure $\sigma = 2$ on the population, adaptively affected by the sharing mechanism. The population size was 100 individuals.

The recombination operator used was reduced-surrogate (11), shuffle (12) crossover, applied with probability 0.7. This crossover variant ignores the ordering of the bits in the chromosome while being just as disruptive as single or double point crossover. It also produces offspring different from their parents whenever possible.

Mutation was applied to all individuals after crossover. The bit mutation rate was set in terms of the probability of the individuals as a whole not undergoing mutation. If there was no crossover, this probability of survival should be at least the inverse of the selective pressure to enable selection to recover from the errors introduced by mutation. For length ℓ chromosomes,

$$P_m \leq 1 - \sigma^{-1/\ell} \tag{7}$$

where σ is the selective pressure. A probability of survival, P_s, 25% higher than the limit $1/\sigma$ was found

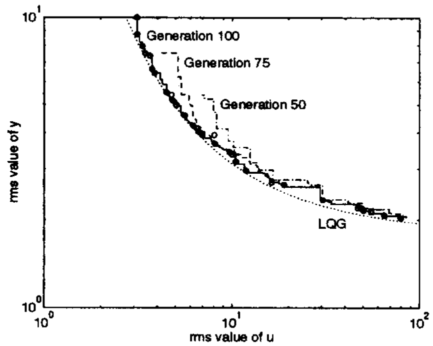

Fig. 2: Evolution of the noise sensitivity trade-off.

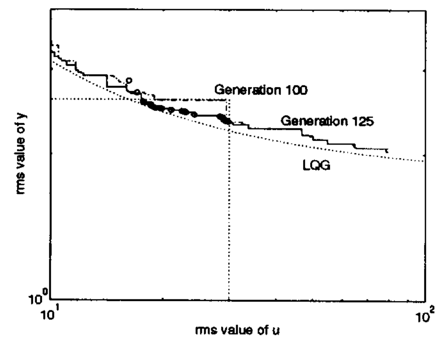

Fig. 3: Noise sensitivity trade-off after change of preferences.

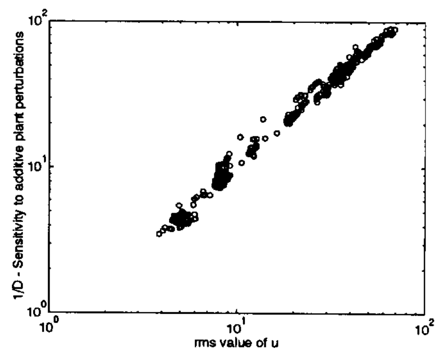

Fig. 4: Two largely non-competing objectives.

to perform well. No fine tuning of the operator probabilities was attempted.

Finally, only individuals affected by the genetic operators were re-evaluated, as proposed by Oliveira et al (13). This reduced the average number of actual function evaluations per generation by 20 to 30%.

RESULTS

Noise sensitivity trade-off

Figure 2 shows how the GA evolved a family of second order controllers for the bi-objective problem involving noise sensitivity. Closed-loop stability was set up as a higher priority objective as discussed above. The region of the trade-off curve to be evolved was delimited by setting the goal vector to $u_{rms} = 100$ and $y_{rms} = 10$.

A partial description of the desired trade-off could be found after 50 generations. As the search progressed, this description was improved and extended, covering most of the region of interest after 100 generations.

The figure also shows the non-dominated individuals in the hundredth generation (marked o). Note how the population is more or less uniformly distributed along the trade-off surface, which shows the effectiveness of the niche techniques used. It is also worth noting that, in this case, second order controllers comes very close to the best that can be achieved with any stabilizing controller. LQG regulators for this plant have order 3.

By changing the goal values, it is possible for the designer to zoom in on a portion of the trade-off curve. For example, setting $u_{rms} = 30$ and $y_{rms} = 2.6$ as the new goals and running the GA for a further 25 generations produced the trade-off curve shown in Figure 3. Individuals which do not achieve the new goals are no longer preferred, causing the whole population to evolve towards the new preferred region, and providing a more accurate description of the trade-off in that region.

Trade-offs involving noise sensitivity and robustness measures

Noise sensitivity can also be traded off against the sensitivity to additive loop and plant perturbations. After 50 generations, the graph in Figure 4 could be produced. The strong direct relationship shown between actuator variance and the sensitivity to additive plant perturbations confirms the intuition that less control action leads to greater robustness. Learning that these two objectives are, to a great extent, non-competing is important for the designer, as it conceptually reduces the complexity of the problem.

Suppose the designer could decide upon a maximum

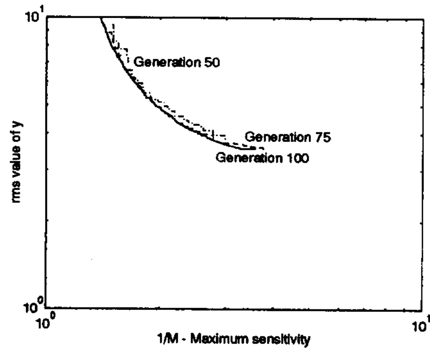

Fig. 5: Maximum sensitivity versus output variance ($u_{rms} \leq 10$ and $1/D \leq 10$).

sensitivity to plant perturbations of $1/D = 10$. Also, suppose that a corresponding actuator variance of 10, which can be expected from the graph in Figure 4, is acceptable but should not be exceeded. Increasing the priority of these two objectives converts the problem into a problem with two objectives, output variance and sensitivity to loop perturbations, and three constraints, closed-loop stability, maximum actuator variance and maximum sensitivity to plant perturbations. The corresponding trade-off curve is presented in Figure 5. All controllers found were open-loop stable.

CONCLUDING REMARKS

Multiobjective Genetic Algorithms were used to find families of reduced order regulators for a simple but realistic plant. Design trade-offs were produced which provide insight into the closed-loop specifications of the regulated plant achievable with a second order controller.

The effectiveness of the MOGA becomes apparent when one considers the number of function evaluations per preferred point found. In the first example, out of approximately 7000 actual function evaluations (generation 100), 189 (2.7%) were preferred points, in a search space as large as 1.2×10^{24}.

The ability to refine requirements on-line allows the designer to interact with the optimization algorithm, learning about design trade-offs and concentrating computational effort in the region of the trade-off most likely to produce a final design.

Acknowledgement. The first author gratefully acknowledges support by Programa CIENCIA, Junta Nacional de Investigação Científica e Tecnológica, Portugal.

References

1. Boyd, S. P., Balakrishnan, V., Barrat, C. H., Khraishi, N. M., Li, X., Meyer, D. G. and Norman, S. A., 1988, *IEEE Transactions on Automatic Control, 33*, 268–283
2. Goldberg, D. E., 1989, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley, Reading, Massachusetts, USA
3. Fleming, P. J., and Fonseca, C. M., 1993, in "Proceedings of the 12th IFAC World Congress", Sydney, Australia, 383–390
4. Hwang, C.-L., and Masud, A. S. M., 1979, "Multiple Objective Decision Making – Methods and Applications: A State of the Art Survey", Springer-Verlag, Berlin, Germany
5. Fonseca, C. M., and Fleming, P. J., 1993, in "Genetic Algorithms: Proceedings of the Fifth International Conference", Morgan Kaufmann, San Mateo, California, USA, 416–423
6. Deb, K., and Goldberg, D. E., 1989, in "Proceedings of the Third International Conference on Genetic Algorithms", Morgan Kaufmann, San Mateo, California, USA, 42–50
7. Barrat, C., and Boyd, S., 1989, *IEEE Control Systems Magazine, 9* 46–52
8. Chipperfield, A. J., Fonseca, C. M., and Fleming, P. J., 1992, in "IEE Colloquium on Genetic Algorithms for Control Systems Engineering, Digest 1992/106", The Institution of Electrical Engineers London, U.K., 3/1–3/6
9. Hippe, P., and Stahl, H., 1987, *International Journal of Control, 45*, 1469–1477
10. Kristinsson, K. and Dumont, G. A., 1992, *IEEE Transactions on Systems, Man, and Cybernetics, 22*, 1033–1046
11. Booker, L., 1987, in "Genetic Algorithms and Simulated Annealing", Pitman, London, U.K., 61–73
12. Caruana, R. A., Eshelman, L. J., and Schaffer, J. D., 1989, in "Proceedings of the Eleventh International Joint Conference on Artificial Intelligence", Morgan Kaufmann, San Mateo, California, USA, 750–755
13. Oliveira, P., Sequeira, J., and Sentieiro, J., 1991, in "Engineering Systems with Intelligence", Kluwer Academic, the Netherlands, 431–438