Fuzzy Scheduling Control of a Gas Turbine Aero-Engine: A Multiobjective Approach

Andrew J. Chipperfield, Beatrice Bica, and Peter J. Fleming

***Abstract*—This paper investigates the use of a nonconventional approach to control a gas turbine aero-engine. The rationale behind this study is the need to develop advanced tools and techniques that can assist in improving the performance of the system and simultaneously enhance the flexibility of the control strategy. Modern techniques are required for many complex systems where increasingly strict performance and regulatory requirements must be achieved. This is particularly true of aerospace systems where consideration of safety, reliability, maintainability, and environmental impact are all necessary as part of the control requirements. This paper investigates a combination of two such potential techniques: fuzzy logic and evolutionary algorithms. Emerging from new requirements for gas turbine aero-engine control, a flexible gain scheduler is developed and analyzed. A hierarchical multiobjective genetic algorithm is employed to search and optimize the potential solutions for a wide envelope controller covering idle, cruise, and full-power conditions. The overall strategy is demonstrated to be a straightforward and feasible method of refining the control system performance and increasing its flexibility.**

***Index Terms*—Decision making, fuzzy scheduling, gas turbine engine control, multiobjective genetic algorithms.**

I. Introduction

FUZZY LOGIC is an attractive technique for the control of poorly understood, unmodeled, or complex systems where the experience of human operators is available to provide qualitative “rules of thumb” [1]. Despite many successes, however, fuzzy control has not been regarded as a rigorous discipline due to the lack of formal synthesis techniques and guarantees of stability, performance, and robustness. This is particularly true for flight control applications where certification standards require a very stringent assessment of the system qualities and performance. The perceived advantages of fuzzy control, such as reduced development time and simplicity of implementation, are expected to outweigh the disadvantages, and research has been active in this field. Indeed, this has been recognized by industry, and within Europe efforts have increased to define a standard, based on ISO-9000 general system development guidelines, for the development methodology of fuzzy logic systems [2].

Manuscript received February 11, 1999; revised November 21, 2001. Abstract published on the Internet March 7, 2002. The work of B. Bica was supported by the CVCP and an Overseas Research Studentship from The University of Sheffield.

A. J. Chipperfield and P. J. Fleming are with the Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, S1 3JD, U.K. (e-mail: A.Chipperfield@Sheffield.ac.uk).

B. Bica was with the Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, S1 3JD, U.K. She is now with Unilever Research, Bebington Wirral, CH63 3JW, U.K.

Publisher Item Identifier S 0278-0046(02)04925-0.

Gas turbine engine control is a demanding task that requires the satisfaction of many, often competing, performance measures. While the design of a control system for a conventional propulsion system poses few hard problems for the control engineer, there may be many candidate solutions available and the choice of the “correct” system is paramount. Furthermore, new concepts in aircraft engines, such as the variable-cycle engine, are likely to include more controllable elements and will almost certainly require the application of advanced control techniques if they are to realize their full potential benefits. A particular problem with gas turbine engine control, and nonlinear systems in general, is that the gains of the controller have to be scheduled over the operating envelope of the plant. At each design point, the controller is required to meet a set of performance objectives. These include, but are not limited to, steady-state accuracy, transient accuracy, disturbance rejection, stability, stall margin, structural integrity, and engine degradation. The control of the engine over its operating range must then ensure that there is a smooth transition between these design points such that actuator demands are consistent and that small- and large-scale transient accuracy is achieved. Additionally, the overall control strategy will usually be chosen to provide some compromise between optimising the thrust specific fuel consumption, speed of response and minimizing the rate of thermal changes (to reduce thermal fatigue).

Conventionally, fuzzy rules are established by a combination of knowledge, experience and observation and may thus not be optimal. Additionally, in spite of efforts to formalise a development standard for fuzzy controllers, fine tuning its performance is still a matter of trial and error. Many studies have shown that evolutionary algorithms can be successfully employed in the design and tuning of fuzzy controllers close to the optimal solution and that these may be made to implement effective self-tuning and adaptive schemes (see, for example, [3] and [4]). Few though, have employed true Pareto-optimization techniques when tuning the controller. Recognising that gas turbine engine controllers are required to meet a number of design criteria, this paper investigates a multiobjective-genetic-algorithm (MOGA)-based approach to the design and tuning of fuzzy scheduling controllers for such systems.

In this paper, after a short introduction to gas turbine engine control and MOGAs, a structured approach to the design of a satisfactory fuzzy scheduling replacement for the original controller is considered. Having determined the structure and sets of suitable parameters for the new controller, a set of low-level MOGAs is employed to tune them directly on a nonlinear model of the system. It is demonstrated that the proposed approach allows a number of alternative fuzzy controllers to be found that

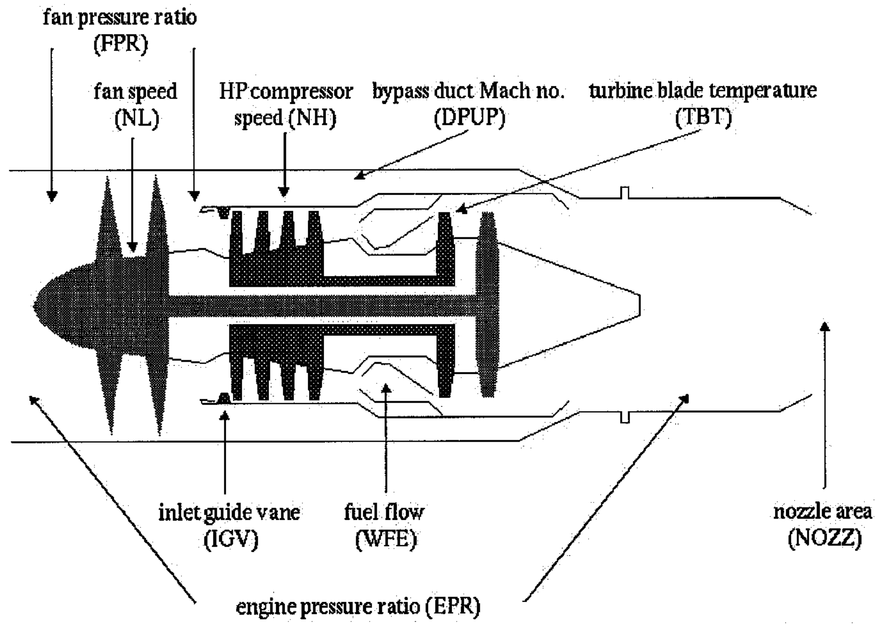


Fig. 1. Gas turbine engine.

offer different control tradeoffs while still having a known structure and, hence, relationship with the rules determined from observations of the plants characteristics.

II. Gas Turbine Engine Control

The mechanical layout of a typical twin-spool gas turbine engine is shown in Fig. 1. Each spool comprises a number of compressor and turbine stages aero-thermodynamically coupled to each other. Air is drawn into the fan (or low-pressure (LP) compressor) through the inlet guide vanes (*IGV*), which are used to match the airflow to the fan characteristics, and compressed. The air is then further compressed by the high-pressure (HP) compressor before being mixed with fuel, combusted, and expelled through the HP and LP turbines. A portion of the air from the fan exit may bypass the HP compressor and turbines. This air is mixed with the combusted air/fuel mixture from the turbine exit before being ejected through the jet pipe and nozzle (*NOZZ*) to produce thrust.

The characteristics of operation of a fixed-cycle gas turbine engine, such as specific thrust and specific fuel consumption, are fundamental to the engine design. The design thus becomes a compromise between meeting the conflicting requirements for performance at different points in the flight envelope and the achievement of low life-cycle costs, while maintaining structural integrity. However, variable geometry components, such as the inlet guide vanes and nozzle area, may be used to optimize the engine cycle over a range of flight conditions with regard to thrust, specific fuel consumption, and engine life, assisting in the reduction of life-cycle costs. In future engine designs, such as the variable-cycle engine, it is likely that the number of these variable geometry devices will be increased allowing the engine to efficiently assume a number of different operating cycles (e.g., turbofan, turbojet, etc.). Such engines will require precise control of these devices if the potential benefits are to be realized. Fuzzy systems offer the possibility of implementing flexible nonlinear schedulers that may be of significant advantage in controlling such engines effectively [5].

Dry-engine control of a conventional engine is normally based on a single closed-loop control of fuel flow for thrust rating, engine idle and maximum limiting, and acceleration control. The closed-loop concept provides accuracy and repeatability of control of defined engine parameters under all operating conditions, and automatically compensates for the effects of engine and fuel system aging. The gas turbine engine considered in this study is the twin-spool Rolls-Royce Spey, which has been used in both civil and military applications. Fig. 2 shows the baseline configuration for a typical controller for this engine. A nonlinear thermodynamic model of the Spey engine, developed by the U.K. Defence Evaluation and Research Agency (DERA), with inputs for fuel flow (*WFE*), HP *IGV*, and exhaust nozzle area (*NOZZ*), is used to simulate dynamic behavior. Further inputs for flight conditions (altitude, Mach number, and temperature) allow engine operation to be simulated over the full flight envelope. Sensors provided from the engine outputs are high- and low-pressure spool speeds (*NH* and *NL*), bypass duct Mach number (*DPUP*), and turbine and jet pipe temperatures (*TBT* and *JPT*). Other outputs, such as the (fan) low-pressure surge margin (*LPSM*) and gross thrust (*XGN*), are calculated directly from internal engine parameters.

A single-input, *NHDem*, derived from the pilot's lever angle, is used to determine the thrust setting. The digital proportional plus integral (PI) fuel-flow controller, shown in Fig. 3(a), uses this and the measured HP compressor speed, *NH*, to determine the required fuel-flow demand. To protect the engine from over-acceleration, *NHDem* is rate limited. A third input, air data, is required to correct the *NH* value for changes in flight conditions, i.e., temperature and pressure, so that the controller can operate over the full-flight envelope. As the controller is also required to operate over a range of engine conditions, such as idle, cruise, and full power, the fuel-flow controller uses gain

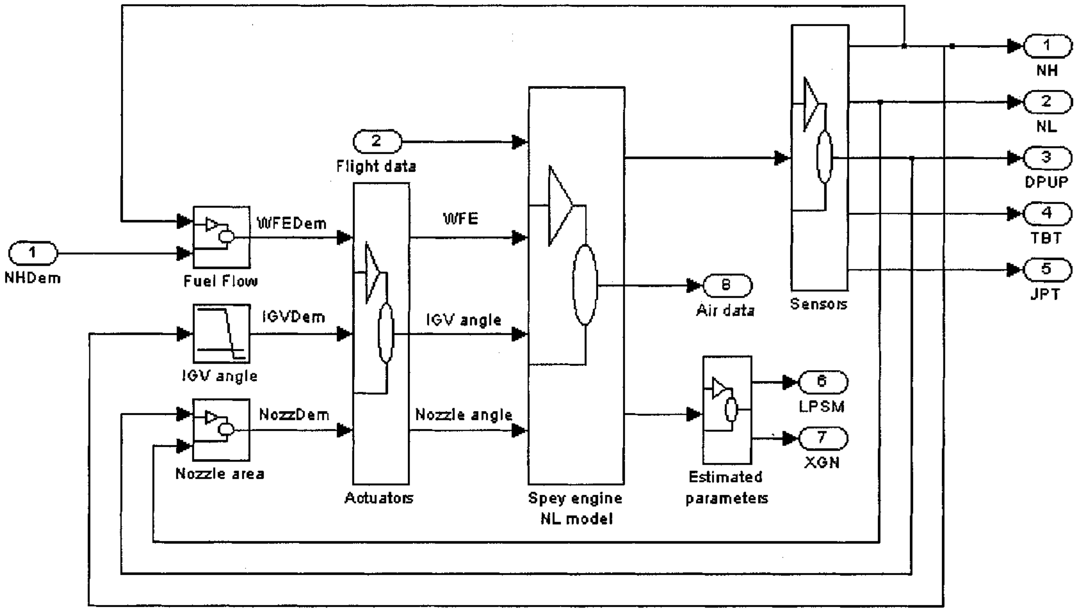


Fig. 2. Conventional control scheme.

schedules to accommodate these nonlinearities in the system dynamics, blocks *GP_1* and *GI_1* in Fig. 3(a). The gain schedules for sea-level-static conditions are shown in Fig. 3(b) where these are obtained by determining the P and I gain at each of a number of design points and linearly interpolating at off-design points. However, a problem with this approach is that linearly interpolating between design points may not adequately capture the engine dynamics and therefore may result in suboptimal control [6].

An important characteristic of gas turbine engine control is the ability to follow a "working line" in order to reduce the risk of stall or surge occurring. These are undesirable as they can lead to loss of thrust and, in severe circumstances, destruction of the engine. *LPSM* is a measure of how close the fan is to stall and can indirectly be controlled against bypass duct Mach number, the ratio of static to dynamic pressure in the bypass duct, and *NL* as shown in Fig. 2. Fig. 4(a) shows the relationship between *DPUP* and *NL* for minimum and maximum nozzle areas. Thus, as *LPSM* is related to *DPUP*, the controller can be scheduled against *NL* and air data to maintain a minimum *LPSM* over the flight envelope in the same manner as fuel flow is controlled in Fig. 3.

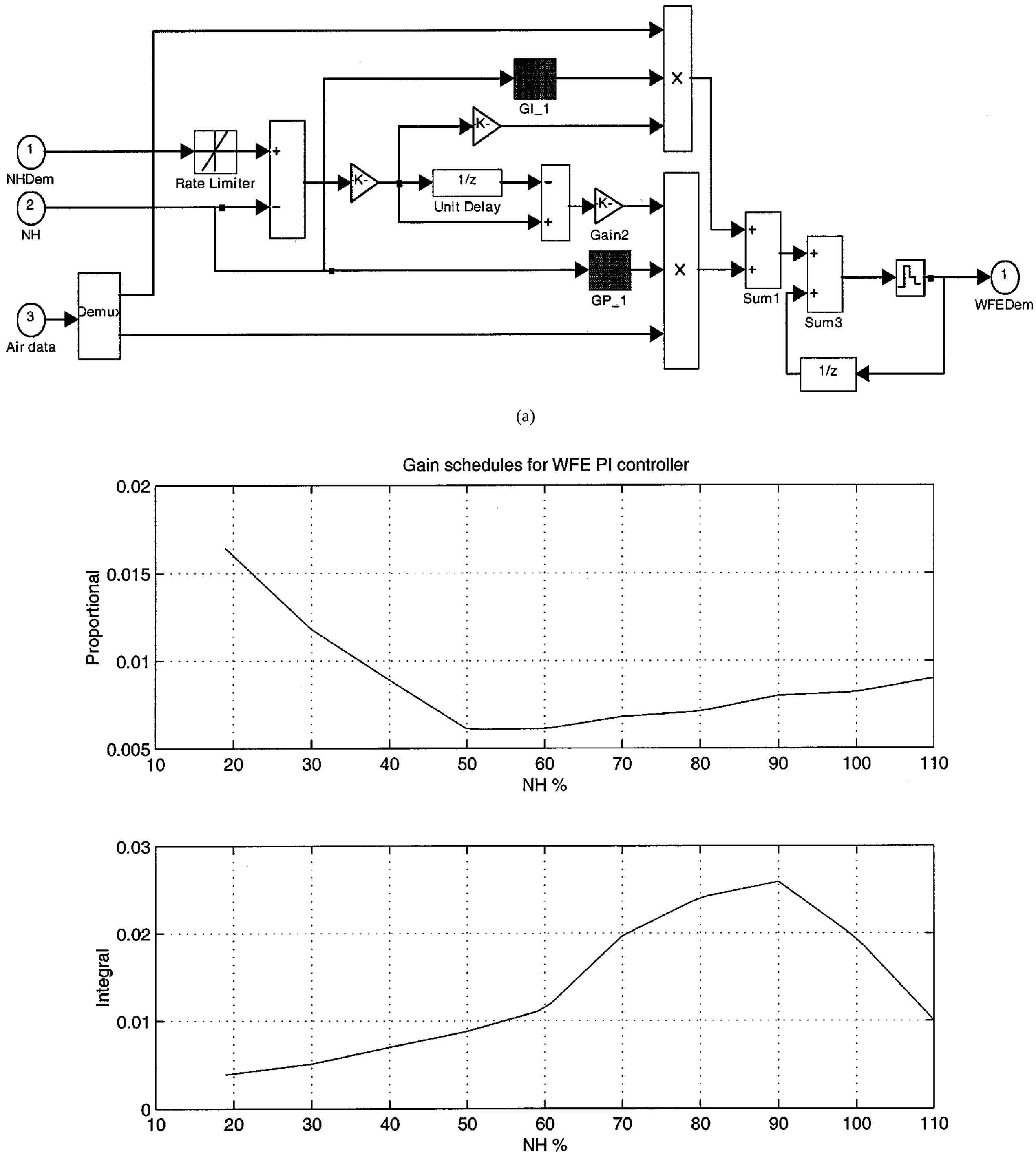
Fig. 4(b) shows the relationship between *NH* and *NL* for different *IGV* angles and nozzle areas. Clearly, the *IGV* angle has the greatest effect on the *NL*/*NH* mapping and either *NH* or *NL* may be used to control the *IGV* angle. As *IGV* positioning cannot usually be achieved with a high degree of accuracy, it is typical to position the *IGV*s against an open-loop mapping. In the baseline Spey controller, the *IGV*s are held at one end-stop (32°) below a corrected 78% *NH* and at the other end-stop (-8°) above 91% *NH*. The *IGV*s are moved proportionally between these values. This ensures reasonable control between *NH* and *NL* and helps maintain the working line. As with the fuel and nozzle schedules, this linear schedule may not be optimal.

The engine should satisfy the following large-signal performance criteria when subject to a 65%–100% step change in *NH* demand:

1) 70% *NH* rise time $\leq$ 4.63 s;
2) 10% *NH* settling time $\leq$ 5.85 s;
3) *XGN* $\geq$ 56.38 *KN*;
4) *XGN* rise time $\leq$ 5.0 s;
5) *LPSM* $\geq$ 6.6%;
6) *JPT* $\leq$ 833 °C;
7) *FPR* $\leq$ 3.18;
8) *TBT* $\leq$ 1540 °C;
9) $dTBT/dt \leq 1320$ °/s.

Objectives 1), 2), and (3) are typical dynamic performance requirements for a military engine. *XGN* is the engine thrust and is used as a measure of the accuracy between nominal and controlled engine performance in objective 3). In objective 4), the limit on *XGN* rise time is required as the nozzle area can be used to trim thrust by affecting changes in the engine pressure ratios. This, in turn, impacts on *LPSM*, objective 5), and the combination of objectives 4) and 5) helps ensure good dynamic performance and stability. Objectives 6) and 8) are the maximum nozzle and turbine temperatures, respectively, lower values indicating lower thermal loading. Objective 7) measures fan pressure ratio (*FPR*) and helps maintain structural integrity as well as thrust mapping and limiting. Finally, objective 9) is a measure of thermodynamic stress on the turbine. A lower value for this objective is an indicator of a longer engine life.

The controlled engine is also required to satisfy a similar set of small-signal dynamic performance criteria at each of five representative operating points over the *NH* range. These are 55% (idle), 65%, 75%, 85%, and 95% *NH*. The goals for objectives 1)–9) are selected to represent the desired performance at each of these operating points using the original PI controller as a baseline.

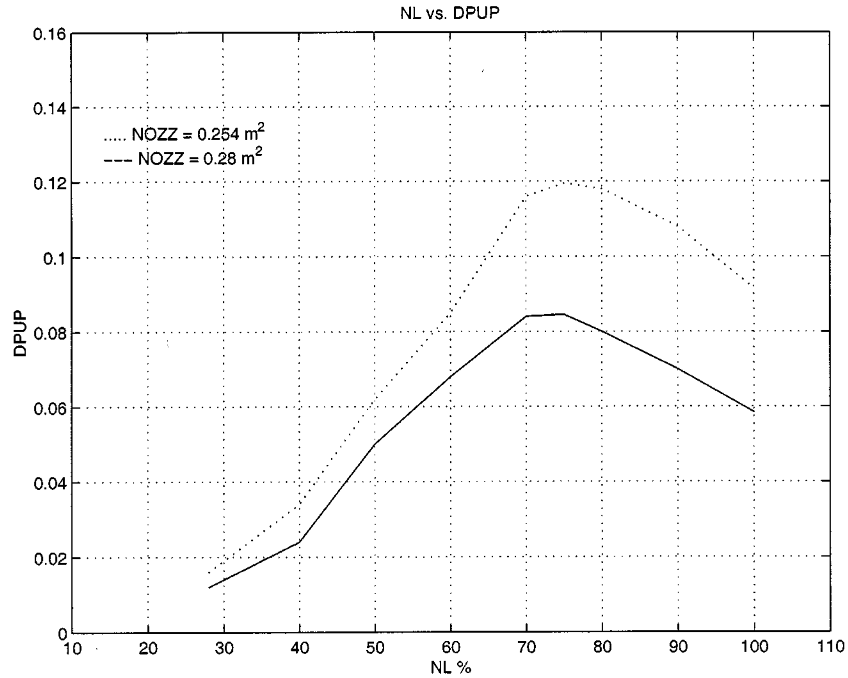


(a)

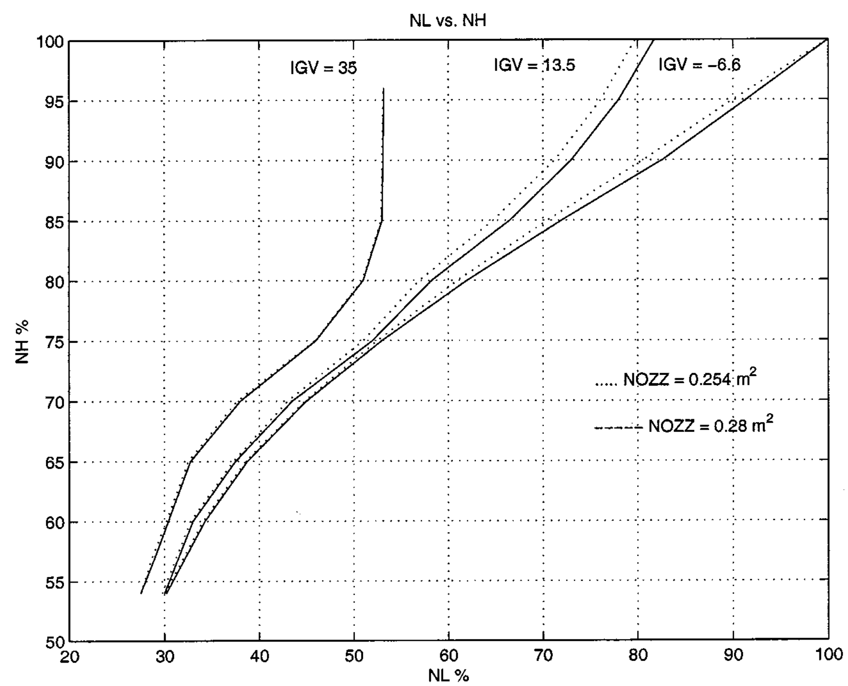
(b)

Fig. 3. (a) Digital PI fuel flow controller. (b) Gain schedules.

The aim of the controller design is, therefore, to provide stabilizing control of the engine while meeting a number of operating and performance constraints. The controller is not necessarily intended to improve the performance of the engine, rather to design and tune a fuzzy scheduling controller that has a known structure and, hence, relationship, with the conventional controller of Figs. 2 and 3. This is achieved by designing the structure of the fuzzy schedulers from observations of the plants characteristics and operating conditions and using multiobjective search to refine the controller parameters. The reason for the choice of fuzzy schedulers is that in previous work it has proved difficult to identify polynomial schemes that perform acceptably [7]. For different parametric values, these fuzzy scheduling controllers will offer different control and performance tradeoffs and it is the role of the engineer to select those most appropriate to the requirements. The attraction of using fuzzy schedulers is that no assumptions have to be made *a priori* concerning the degree of nonlinearity or cross coupling in the plant and that the controller may also be suitable for post-design tuning and/or adaptation. These properties are desirable for the control of con-



(a)



(b)

Fig. 4. (a) Effect of nozzle area on *DPUP* for different *NL* speeds. (b) Relationship between *NL* and *NH* for different *IGV* angles and nozzle areas.

ventional engines and will be of even more importance in the cost-effective operation of future gas turbines.

III. MOGA

The use of multiobjective optimization in control, and engineering design in general, recognizes that most practical problems require a number of design criteria to be satisfied simultaneously, viz

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

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

Clearly, for this set of functions, $\boldsymbol{F}(\boldsymbol{x})$, 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 noninferior or nondominated solutions to the multiobjective optimization problem and are conventionally sought through the solution of an appropriately formulated nonlinear programming problem. However, this approach requires the precise expression of a, usually not well understood, set of weights and goals. If the tradeoff surface between the design objectives is to be better understood, repeated application of such methods will be necessary.

If goal and/or priority information is available for the design objectives, such as those given in Section II, then it may be possible to differentiate between some of the nondominated solutions. In the example presented here, a MOGA is employed where goal information is derived from the performance of a standard controller and priorities are assigned according to the relative engineering importance of these performance criteria. These considerations have been formalized in terms of a transitive relational operator, *preferability*, based on Pareto dominance, which selectively excludes objectives according to their priority and whether or not the corresponding goals are met [8]. A full description of the implementation of the MOGA is described in [9] and its application to designing a conventional controller for a short-take-off/vertical-anding gas turbine engine given in [10].

As the population of the MOGA evolves, tradeoff information will be acquired. In response to the optimization so far, the control engineer may wish to investigate a smaller region of the search space or even move on to a totally new region. This can be achieved by resetting the goals supplied to the MOGA, which, in turn, affects the ranking of the population and modifies the fitness landscape concentrating the population on a different area of the search space. The priorities of design objectives may also be changed interactively using this scheme, thus altering the order in which design constraints are satisfied.

IV. Fuzzy Scheduling Controller Design

Fuzzy logic is an increasingly popular method of handling systems associated with uncertainty, unmodeled dynamics, or simply where human experience is required. Its ability to deal with imprecise data can often offer an immediate benefit over conventional mathematical reasoning. It has been widely employed in control problems, particularly due to its ability to mimic the behavior of nonlinear plants. By ensuring that a properly formulated rule base is found, a fuzzy system can provide smooth transitions between operating regimes. It posses good interpolator capabilities which can be employed to implement effective gain schedulers [11], [12].

The most common applications of fuzzy gain scheduling to date are the tuning of proportional–integral–derivative (PID) controller gains. When operating conditions change, the fuzzy scheduling controller adjusts the gains through a collection of "IF-THEN" rules. Zhao *et al.* [13] were among the first to implement an effective fuzzy scheduling PID controller. Their tuning process produced online gains over the entire operating range of the plant with controller parameters determined from error signals and their first derivatives. However, difficulties arise in analysing the stability properties of the compensated system. A potential solution, identified in [13], is the use of a supervisory level of control in the fuzzy scheduler that identifies the onset of instability and adjusts the controller gains accordingly. The performance of the resulting control scheme compared favorably with Zeigler–Nichols and Kitamori controllers. An alternative approach based on a hierarchical structure is described by Pedrciz and Peters [14]. Here, a number of local controllers are employed to determine the global system behavior and a supervisory controller determines the relative contribution of two locally valid PID controllers at any operating condition. At any given operating point, the controller output is some interpolation of the locally valid controllers.

Another popular fuzzy scheduling mechanism, that also posses good interpolator characteristics, is the Tagaki–Sugeno–Kang (TSK) system [15]. The combined quantitative–qualitative character of such system is achieved by a rule structure where the output is an analytical combination of the fuzzy expressions. Each rule usually represents a linear description of the system at a specific operating point to ensure coverage of the entire operating range by the rule base. The TSK system is able to accurately approximate nonlinear systems and thus the interpolation is undertaken between the locally valid linear approximations of the nonlinear system rather than interpolating between the controller gains.

For the controller considered here, the gain scheduling control can be achieved by using a nonlinear Mamdani mapping. The object of the design is therefore to produce a set of five nonlinear gain schedules for the P and I maps of the fuel flow and nozzle controllers and the *IGV* position. The procedure for designing the fuzzy schedulers is structured in two stages. First, fuzzy schedules are hand designed to approximate the original mappings between the scheduled variables. Having designed suitable fuzzy structures, their performance is assessed through simulation. The second phase in the design process is to encode the parameters of the fuzzy schedulers such that they may be tuned with a multiobjective genetic algorithm to meet the performance criteria described in Section II. Although the representation used by the MOGA is capable of automating the first phase to find suitable fuzzy structures, the use of predefined structures allows the effects of structural changes in the scheduler to be assessed.

Following a conventional fuzzy design approach [16], a set of fuzzy schedulers was designed. For the fuel flow controller of Fig. 3, the P and I schedules were replaced with fuzzy systems comprising seven membership functions and four rules each. The nozzle controller required eight membership functions for the P schedule and seven for the I schedule each employing four rules. A satisfactory *IGV* schedule was built with four membership functions and two rules. The set of fuzzy schedulers thus requires a total of 33 membership functions, 18 rules, and ten scaling factors.

Fig. 5 compares the performance of this fuzzy controller with the original baseline controller of Section II to a 65%–100% step change in *NH* demand for the performance measures described in Section II. The continuous line shows the response of the engine with conventional schedulers and the dashed line with the fuzzy schedulers. The relative lack of discrepancy between the two system responses demonstrates the ability of the fuzzy

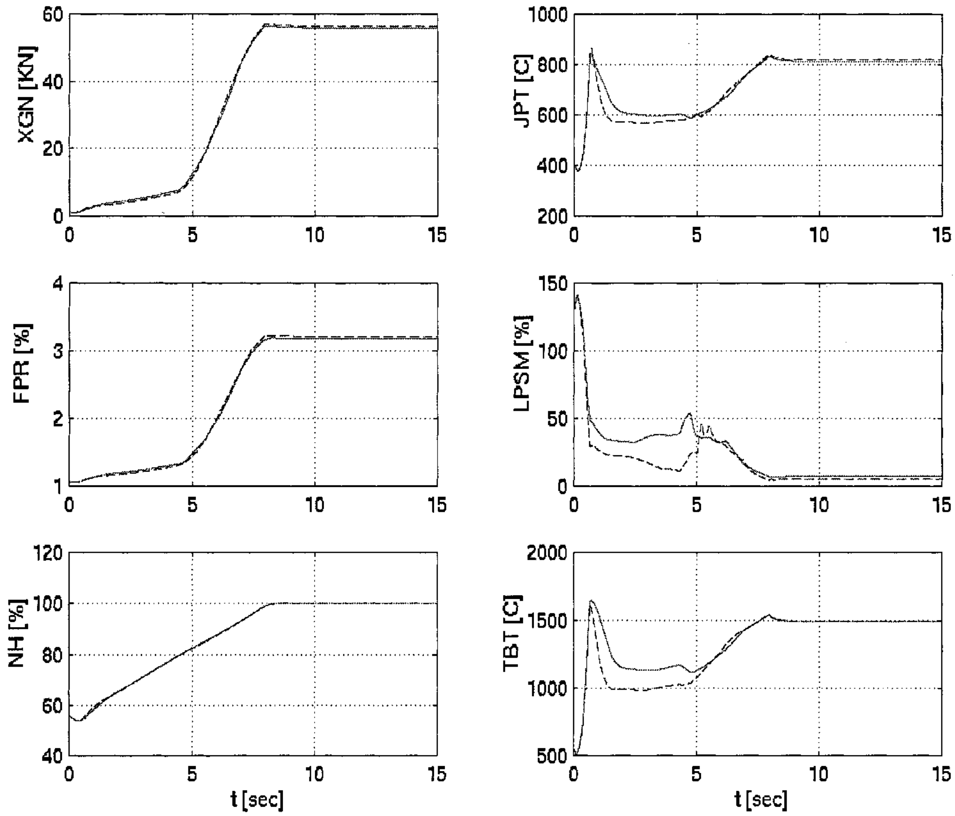


Fig. 5. Comparison of original and fuzzy schedulers.

schedulers to replicate the behavior of the original gain schedulers—by including the fuzzy logic mapping within the controller, the performance of the original system is matched. The primary aim of developing these fuzzy mappings was not necessarily to improve the performance of the original controller, rather it was intended to show that fuzzy schedulers with a known structure and relationship to the original controller could perform the desired control adequately. However, it is possible to make some observations from the response plots of Fig. 5. For example, it can be seen that a change in *LPSM* is related to other system parameters, specifically an increase in the nozzle and the turbine temperature leads to a reduced *LPSM*. This relationship is most apparent during the initiation of the set point change and it appears that *LPSM* is related to the rate of change of the temperatures. The fuzzy scheduling system is also able to marginally improve the turbine blade temperature and the surge margin while maintaining the thrust mapping and *FPR*. These observations make the prospect of optimising the fuzzy schedules, to offer improved performance and/or different tradeoffs between the design objectives, an attractive proposition.

V. MOGA Controller Tuning

Having designed a suitable fuzzy scheduling controller that satisfactorily approximates the response of the original PI controllers, the rule base, membership functions, and scaling factors may now be tuned. Depending on the particular problem, the number and type of encoded parameters can considerably influence the overall performance of the optimization process [17]. The selection of a chromosome structure can affect not only the quality of the solutions, but also the computational effort required to find them. A difficulty typically encountered when formulating such optimization problems is the choice of the most appropriate representation to provide a satisfactory compromise between the accuracy of the solutions and the computational burden. A comprehensive encoding of the fuzzy system parameters, scaling factors, membership functions, and rule base means that no area of the solution space is infeasible, but may mean that because of the computational complexity and nature of the solution space that satisfactory solutions will not be found in a reasonable time [18]. Similarly, a simpler representation may reduce the computational complexity at the expense of the solution optimality. Thus, the choice of representation for the fuzzy scheduler will be a compromise between completeness and feasibility.

The most general chromosome would have the following structure for each schedule to be encoded:

$$R_1, R_2, \ldots, R_m, M_1, M_2, \ldots, M_n, \mathrm{SF}_I, \mathrm{SF}_O \tag{1}$$

where $R_{1,\ldots,m}$ are the m possible rules and $M_{1,\ldots,n}$ are the n associated membership functions. Both the rules and the membership functions would each be a set of parameters that describe either the rule or shape and location of the membership function. SF_I and SF_O are the input and output scaling factors, respectively.

Employing observations of the system from the previous section and knowledge of the design points, the number of membership functions for each schedule can be upper bounded while ensuring that satisfactory control, in terms of that of the baseline controller, may be achieved. Similarly, the number of rules and

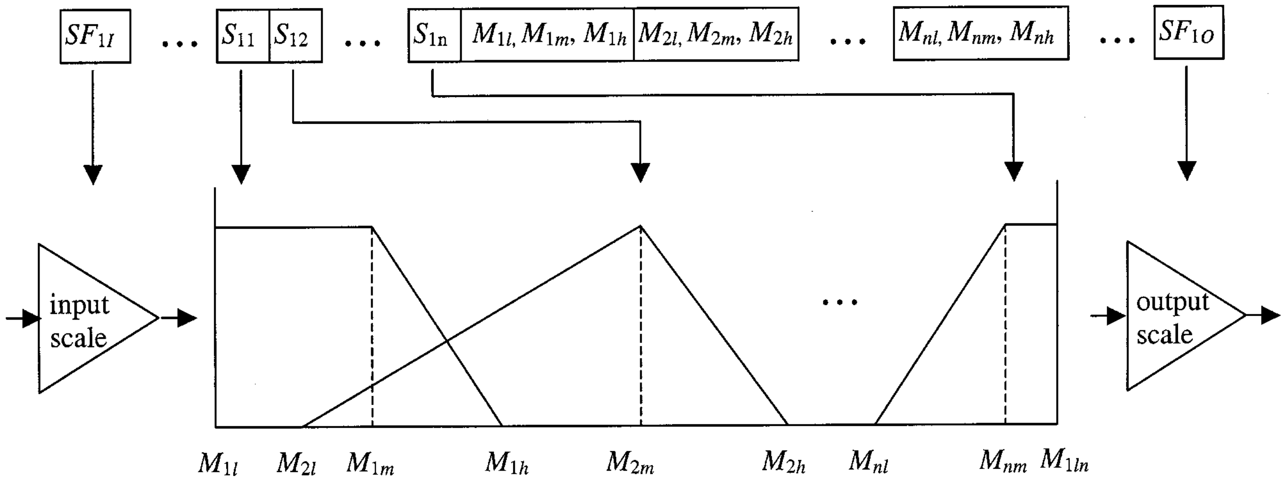


Fig. 6. Chromosome for representing a fuzzy schedule.

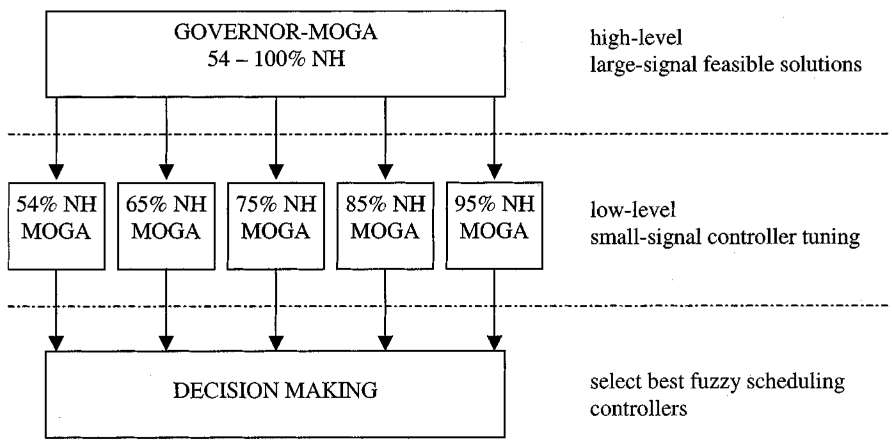


Fig. 7. Hierarchical MOGA structure.

bounds on the scaling factors may also be upper bounded. However, because of the nonlinearity of the engine operating characteristics and the high degree of cross coupling between inputs and outputs, radically changing the shape of the schedulers can put the engine into unfeasible or unstable operating regions. An appropriate strategy for optimising the schedules was found to be one where the rules were fixed and the membership functions and scaling factors were free to be adjusted.

The chromosome for representing each schedule is based on the structured encoding of Tang *et al.* [19] and is shown in Fig. 6. It consists of the catenation of the input scaling factor SF_{1I}, a set of n control genes $S_{1,\ldots,n}$, the set of coordinates for each of the n membership functions $M_i = [M_{il}, M_{im}, M_{ih}]\ i = 1, \ldots, n]$, and the output scaling factor SF_{1O}. The control genes determine whether an individual membership function is activated or not. For the examples considered here, all of these control genes are active so that every solution found will have the same structure. The full chromosome is made up of the sets from the hand designed fuzzy controller for the five schedules. If the control genes are manipulated by the MOGA then a search may be made for all fuzzy schedules up to the maximum number of membership functions specified. In this case, the MOGA could be used to find the fuzzy structure and tune its parameters. However, this would have the disadvantages of a much increased computation time and may result in overly complex schedulers offering only marginally improve performance or reduced stability.

As previously described, the fuzzy schedulers are required to satisfy both small- and large-signal responses corresponding to local and global performance criteria. Therefore, tuning of the schedulers requires the search to be performed over the entire range of parameters simultaneously and not just locally at predefined points. In order to accommodate this, the MOGA is divided into two hierarchical levels as shown in Fig. 7. A high-level MOGA, the governor MOGA, is employed to search for controller structures that satisfy the large-signal objectives described in Section II. Satisfactory solutions found by this high-level MOGA will then need to tested at individual design points and refined if necessary.

At the lower level, a further set of MOGAs is employed at the normal on-design points of 54% (idle), 65%, 75%, 85%, and 95% *NH*. This additional level is introduced to perform fine tuning at the design points and to assist in decision making through tradeoff analysis. These low-level MOGAs are populated entirely by satisfactory solutions found by the governor-MOGA and may, therefore, be applied after a general set of solutions has been found.

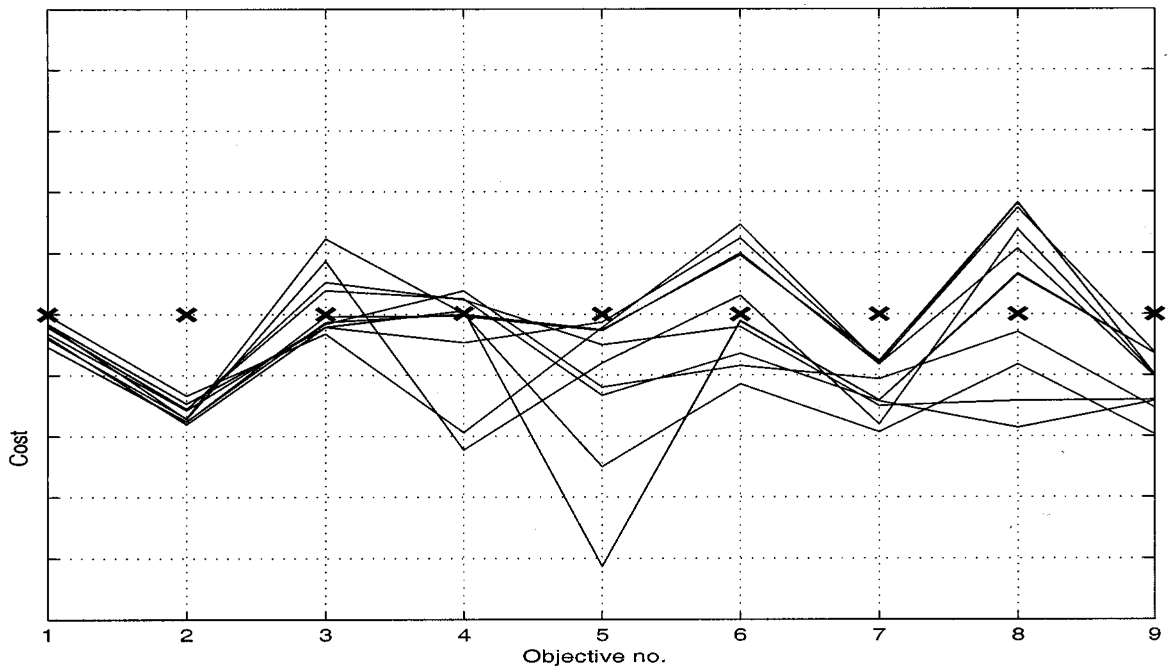


Fig. 8. Family of solutions produced by the high-level MOGA.

Each of the low-level MOGAs generates a matrix of locally tuned fuzzy schedulers, each line of the matrix representing a particular solution. If an exact solution is found to be present in all of the low-level MOGAs, then that solution is deemed superior as it will be both locally and globally Pareto optimal. However, it is unrealistic to expect identical chromosomes from independent MOGAs. A decision making process is, thus, required to identify those chromosomes that are sufficiently close to one another, and that exist in all the low-level MOGAs, to be deemed acceptable solutions. To briefly illustrate how this boundary matrix analysis process works, consider the two potential MOGA solutions, Sol_1 and Sol_2, generated at different operating points

$$\mathrm{Sol}_1 = \lfloor f_{11}, f_{12}, \ldots, f_{1n} \rfloor \text{ and } \mathrm{Sol}_2 = \lfloor f_{21}, f_{22}, \ldots, f_{2n} \rfloor \tag{2}$$

where f_{1i} and f_{2i}, $i = 1, \ldots, n$, are the n objective function values for Sol_1 and Sol_2, respectively. Defining a tolerance vector

$$\mathrm{Tol} = \lfloor \Delta_1, \Delta_2, \ldots, \Delta_n \rfloor \tag{3}$$

where Δ_i is the maximum permissible deviation between f_{1i} and f_{2i}, an acceptable solution is given by

$$|\mathrm{Sol}_1 - \mathrm{Sol}_2| \leq \mathrm{Tol}. \tag{4}$$

That is, both solutions Sol_1 and Sol_2 are within the predefined tolerances for all of the objective values. The overall solution of the decision making process is a single solution, either an individual solution or an average value, thus

$$\mathrm{Sol}_f = \left[\mathrm{Sol}_1 | \mathrm{Sol}_2 | \frac{\mathrm{Sol}_1 + \mathrm{Sol}_2}{2} \right]. \tag{5}$$

When two, or more, solutions from different low-level MOGAs are sufficiently close in chromosome and objective function space, they can be considered as a satisfactory controller. That is, both the large- and small-signal performance criteria are satisfied and the small-signal controller tuning performed by the low-level MOGAs can be assumed to have not adversely affected the large-signal performance.

Percentage Trade-offs

1	(2)	0	6	1	3	0	0	1	4
2	0	(1)	4	0	1	1	0	1	4
3	6	4	(7)	7	12	6	14	9	9
4	1	0	7	(2)	2	0	0	1	3
5	3	1	12	2	(3)	1	1	4	5
6	0	1	6	0	1	(1)	0	0	2
7	0	0	14	0	1	0	(2)	1	2
8	1	1	9	1	4	0	0	(2)	6
9	4	4	9	3	5	2	2	6	(4)
	1	2	3	4	5	6	7	8	9

Fig. 9. The tradeoff matrix.

VI. Results

Fig. 8 shows a typical tradeoff graph for the fuzzy scheduling controllers found by the MOGA for the large-signal controller design criteria. The x axis shows the design objectives and the y axis the performance of controllers in each objective domain. In the tradeoff graph, each line represents a nondominated solution for the design objectives of Section II. The cross marks in Fig. 7 show the design goals and where the lines appear below these marks then that objective has been satisfied. Tradeoffs between adjacent objectives result in the crossing of the lines between them, whereas concurrent lines indicate that the objectives do not compete with one another. For example, reducing the thrust

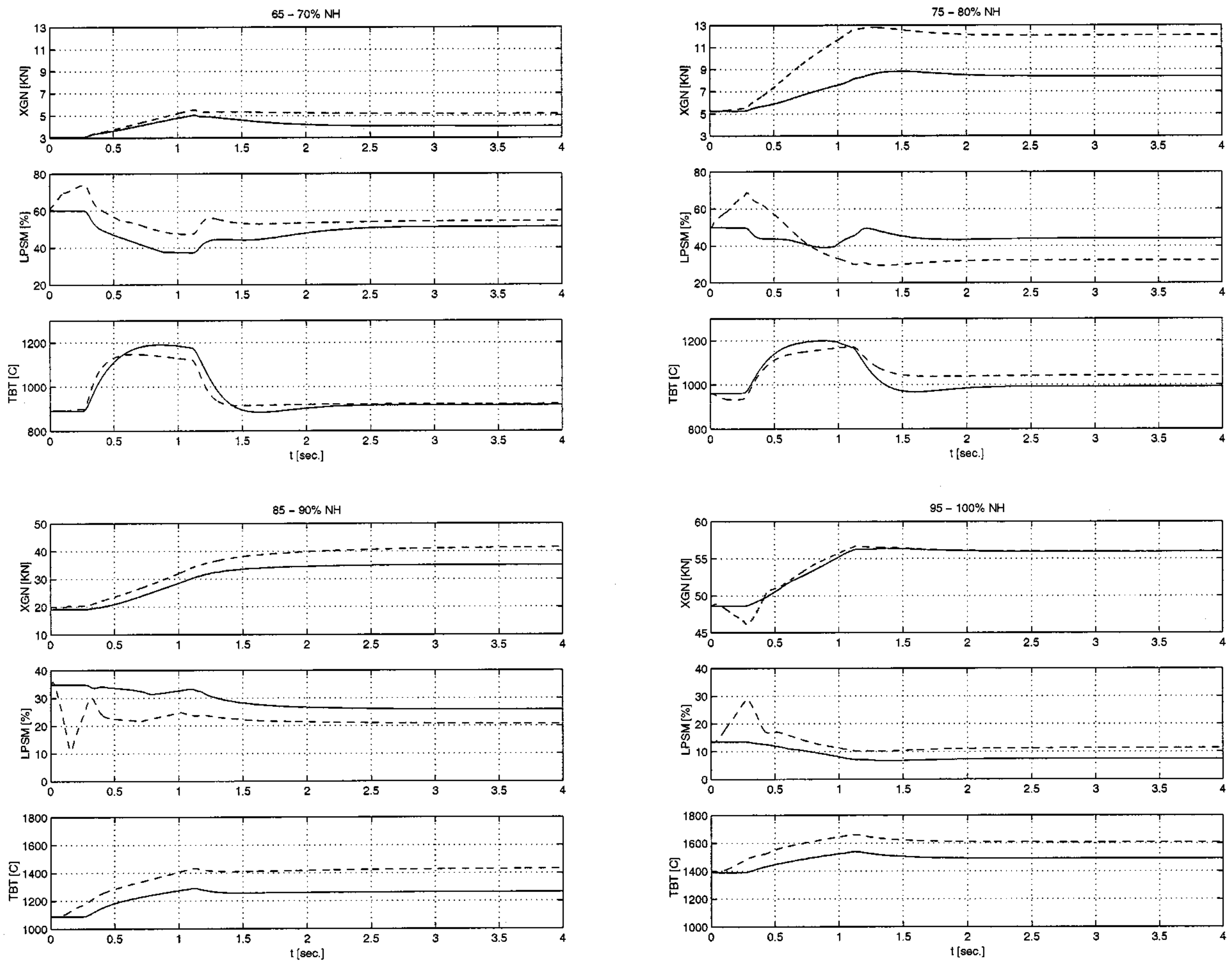


Fig. 10. Small-signal engine response with original and fuzzy schedules at design points.

rise time, objective 4), will usually result in a deterioration in *LPSM* performance, objective 5), and vice-versa. Similarly, *NH* rise and settling times, objectives 1) and 2), are clearly related to one another and do not exhibit the high level of conflict apparent between *LPSM* and *XGN* rise time.

Examining the tradeoff graph shown in Fig. 8 reveals that none of the potential solutions satisfies all of the design objectives simultaneously. In particular, solutions that violate the thrust settings, objectives 3) and 4), and *JPT* and *TBT*, objectives 6) and 8), respectively, can be found that offer improved performance in the other objectives. The goals are derived from the performance of the engine with the original baseline controller and the objectives do not therefore represent physical limits on the system. For example, the maximum permissible *TBT* is 1713 °C. The original control system achieves a maximum *TBT* of 879 °C, and this is the value employed as the goal for this objective. A candidate solution that achieves a *TBT* of 950 °C is considered valid since it is well within safe operating limits. Satisfactory control solutions may, therefore, be found that trade off improved dynamic performance against increased temperatures.

Unfortunately, Fig. 8 only provides information about the achievement of the desired goals and is unable to offer information about the strength of the conflict between objectives, i.e., a quantitative description of the tradeoffs. In many problems, a quantitative analysis of the conflicts may have an important role in the decision making process. In this example, supplementary information, derived from a tradeoff matrix, is employed in the low-level MOGAs to target search effort where it is most needed. This is achieved by concentrating the search in areas where the conflicts between objectives is greatest. Instead of analyzing all of the nine objectives, tests are only performed on the objectives that are more prone to conflict. If the tradeoffs between these objectives are acceptable, then the remaining objectives are generally likely to comply.

An example of a tradeoff matrix is shown in Fig. 9. This is a measure of competition and maps the conflict between objectives onto three-dimensional space. Each objective is related to each of the others with respect to the degree of competition between them and a relative percentage measure of conflict determined. The values given in Fig. 10 therefore relate not only the number of other objectives in conflict but also how much they are in conflict. A full description of the algorithm used to build the tradeoff matrix is given in [20]. For the solutions of Fig. 8, it can be seen that thrust is the cause of the majority of the tradeoffs—it is involved in 67% of conflicts. Thrust is the main controlled, but unmeasured, parameter of the engine that needs to be maximized while simultaneously achieving low costs from the other performance measures and constraints. By implication, attempts to maximize the thrust will lead to degradation in one or more of the other objectives. Therefore, when assessing

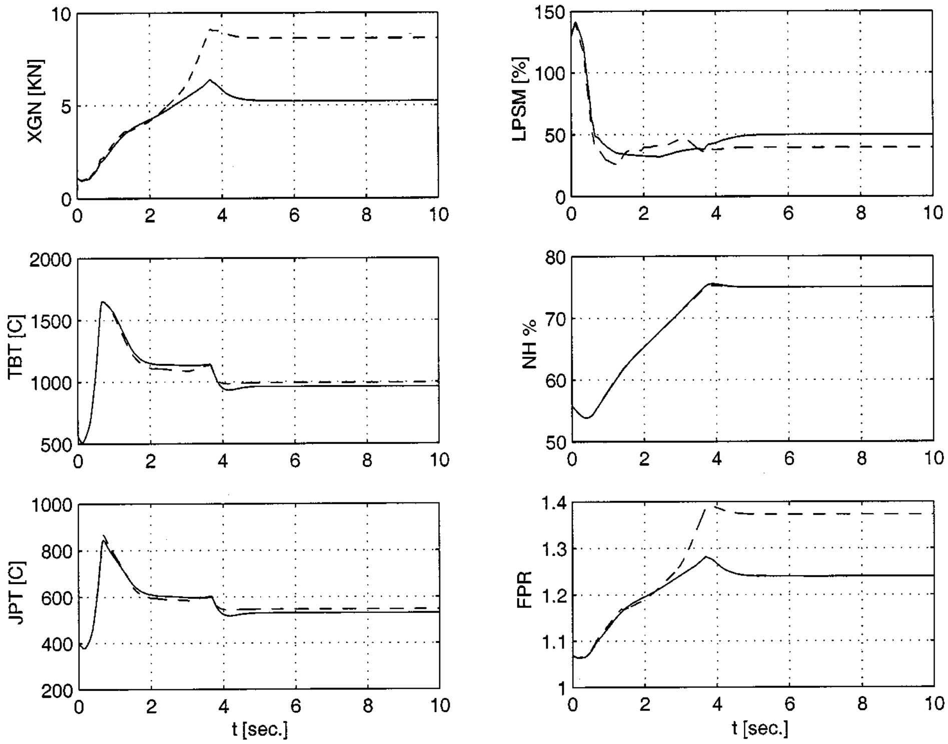


Fig. 11. Small-signal responses at idle/operational transition.

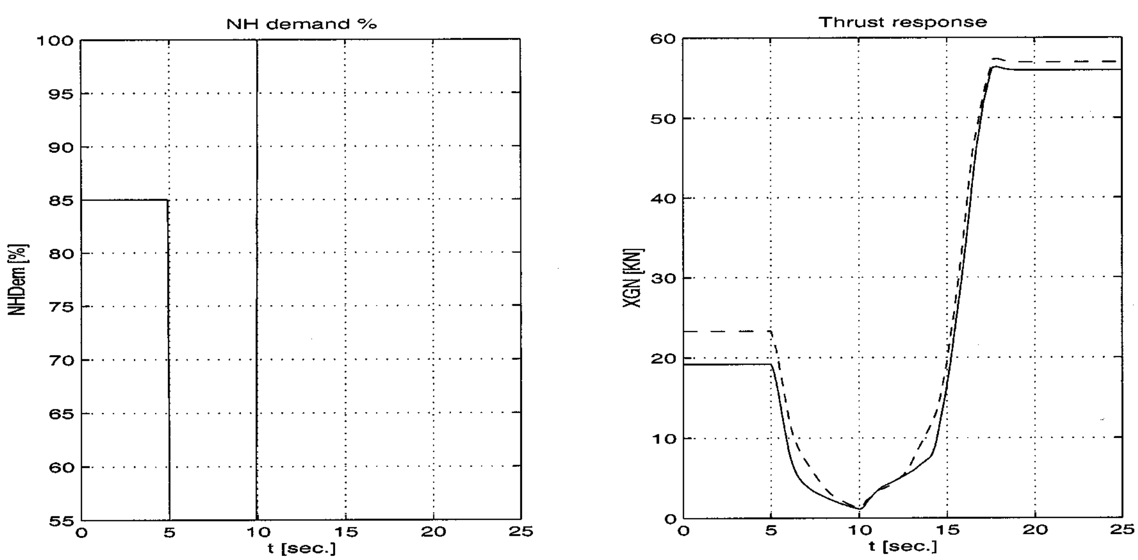


Fig. 12. Large-signal *NH* maneuver and corresponding thrust response.

the performance of potential solutions, most of the effort can be focused on *XGN* and those objectives that trade off against it.

For the tradeoffs of Fig. 8, tradeoff matrix analysis indicates that *LPSM* (12%), *FPR* (14%), and *TBT* (9%) are most in conflict with *XGN*. For another set of Pareto solutions, a different set of conflicts would exist. These conflicts are logical if the thermodynamic properties of the engine are taken into account but may not be known *a priori* for an arbitrary system. An increase in the thrust demand is achieved by increasing the compressor pressure, which is itself directly dependent upon the gas temperature. Thus, maximizing thrust and minimizing pressure and temperature parameters are usually contradictory requirements.

The solutions from the governor MOGA provide the initial populations for the five low-level MOGAs of Fig. 7 that attempt to fine tune the small-signal responses. Initially, the idle point is excluded from this tuning and only the operational design points considered. The objective values derived from these optimizations are stored in matrices and subsequently examined with the boundary matrix procedure described previously. After the local fine-tuning and decision-making process, a family of potential solutions valid over the range 65%–100% *NH* was found. These solutions meet the required level of performance at each design point and over the wide flight envelope—excluding the idle range. Responses of a typical candidate solution for a

5% step change in set point (65%–70%, 75%–80%, 85%–90%, and 95%–100% *NH*) at the different design points are shown in Fig. 10. Here, the continuous line shows the engines' response with the baseline controller and the dashed line shows the response with the fuzzy schedulers. The MOGA-tuned fuzzy scheduler is able to improve the thrust performance over the wide functional range of *NH* while maintaining the structural integrity and the aerodynamic stability of the engine.

The transition between idle and operational conditions requires an abrupt change in control gains as shown in Fig. 3(b). Tuning the idle part of the fuzzy schedules after the operational range has the advantage of reducing the number of free parameters and, thus, the size of the search space. The rules and gains affecting the idle regime can be tuned once satisfactory schemes have been identified for the operational range and the resulting controllers tested over the entire operating range. Fig. 11 shows the performance of the fuzzy scheduler of Fig. 10 tuned around the idle/operational transition to a step change in set point of 55%–75% *NH*. The fuzzy offers a significant thrust improvement over this transition at the cost of a marginally reduced *LPSM* and *TBT*. The response also shows a smooth transition between operating regimes.

Finally, Fig. 12 shows the response of the final fuzzy scheduling controller to an input demand of 85%–55%–100% *NH*. This is a demanding maneuver taking the engine from cruise to idle and then to full power in quick succession. The fuzzy scheduling controller always improves the thrust output but also reduces overshoot and settling time compared with the original controller. The thrust increase is achieved without violating the physical constraints on the engine.

VII. Concluding Remarks

This paper has investigated the use of a nonconventional approach to the control of a conventional gas turbine aero-engine. The rationale behind this study was the need to develop new techniques that may be able to improve the performance, and simultaneously enhance the flexibility, of the control strategy for future concepts in aero-engines. As engines become more complex and have more controllable and measurable parameters, the need for such techniques will increase if they are to achieve their full potential.

The approach described was based around the tuning of a known fuzzy structure and differs from conventional methods in that the controllers are assessed at a set of design points simultaneously and over the entire operating range. This attempts to tune the controller at both the operating points and across operating point and regimes. The resulting controllers where shown to offer improved thrust responses while still satisfying stability and structural design constraints. The parameterization of the fuzzy system is sufficiently flexible to allow the search for different fuzzy structures, with different sets of rules and membership functions, or allow post design adaptation of the fuzzy controllers. Finally, while this paper has considered the control of gas turbine engines, the approach should be applicable to the control of other complex systems such as intelligent buildings, process and transportation systems.

Acknowledgment

The authors gratefully acknowledge the useful comments and suggestions made during this work by S. MacKenzie of DERA (Pyestock) and H. Betteridge of Rolls-Royce plc. The nonlinear model of the Spey engine was developed by DERA (Pyestock).

References

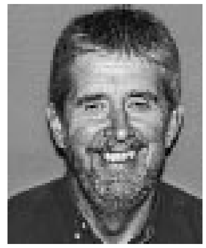
[1] C. C. Lee, "Fuzzy logic in control systems: Fuzzy logic controller—Parts I & II," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 404–435, Mar./Apr. 1990.
[2] G. Schram, U. Kaymak, and H. B. Verbruggen, "Fuzzy logic control," in *In Robust Flight Control: A Design Challenge*, J. F. Magni, S. Bennani, and J. Terlouw, Eds. London, U.K.: Springer-Verlag, 1997, pp. 135–147.
[3] D. A. Linkens and H. O. Nyongesa, "Genetic algorithms and fuzzy control. Part 1: Offline system development and application," *Proc. IEE—Control Theory Applicat.*, vol. 142, no. 3, pp. 161–176, 1995.
[4] ——, "Genetic algorithms and fuzzy control. Part 2: Online system development and application," *Proc. IEE—Control Theory Applicat.*, vol. 142, no. 3, pp. 177–185, 1995.
[5] K. R. Garwood and D. R. Baldwin, "Emerging requirements for dual and variable cycle engines," in *Proc. 10th Int. Symp. on Air Breathing Engines*, 1992, pp. 104–112.
[6] A. G. Shutler, "Engine control law implementation: Effects of scheduled variable geometry on closed-loop performance," Rolls Royce, Bristol, U.K., Rep. CSAN 2840, 1991.
[7] A. J. Chipperfield, C. M. Fonseca, P. J. Fleming, and H. C. Betteridge, "Design of a wide envelope controller for a STOVL gas turbine engine," in *Proc. 14th IFAC World Congr.*, vol. C, 1999, pp. 479–484.
[8] C. M. Fonseca and P. J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," *Evolutionary Comput.*, vol. 3, no. 1, pp. 1–16, 1995.
[9] ——, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 26–37, Jan. 1998.
[10] ——, "Multiobjective optimization and multiple constraint handling with evolutionary algorithms. II. Application example," *IEEE Trans. Syst., Man, Cybern. A*, vol. 28, pp. 38–47, Jan. 1998.
[11] C. D. Yang, T. M. Kuo, and H. C. Tai, "H_∞ gain scheduling using fuzzy rules," in *Proc. IEEE Conf. Decision and Control*, vol. 4, 1996, pp. 3794–3799.
[12] B. H. Cho and H. C. No, "Design, stability and performance of a robust fuzzy logic gain scheduler for nuclear steam generators," *IEEE Trans. Nucl. Sci.*, vol. 44, pp. 1431–1441, Aug. 1997.
[13] Z. Y. Zhao, M. Tomizuka, and S. Isaka, "Fuzzy gain scheduling and PID controllers," *IEEE Trans. Syst., Man, Cybern.*, vol. 23, pp. 1392–1398, Aug. 1993.
[14] W. Pedricz and J. F. Peters, "Hierarchical fuzzy controllers: Fuzzy gain scheduling," in *Proc. IEEE Int. Conf. Systems, Man, and Cybernetics*, vol. 2, 1997, pp. 1139–1143.
[15] K. Tanaka and M. Sana, "Fuzzy stability criterion of a class of nonlinear systems," *Inform. Sci.*, vol. 71, no. 1–2, pp. 3–29, 1993.
[16] E. Cox, "Fuzzy fundamentals," *IEEE Spectrum*, vol. 25, pp. 58–61, Oct. 1992.
[17] K. P. Dahal and J. R. McDonald, "Generator maintenance scheduling of electric power systems using genetic algorithms with integer representation," in *Proc. Int. Conf. GALESIA'97*, 1997, pp. 2456–561.
[18] D. Driankov, M. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control*. London, U.K.: Springer-Verlag, 1993.
[19] K. S. Tang, C. Y. Chan, and K. F. Man, "A simultaneous method for fuzzy memberships and rules optimization," in *Proc. IEEE Int. Conf. Industrial Technology*, Shanghai, China, 1996, pp. 279–283.
[20] P. Schroder, "Multivariable control of active magnetic bearings," Ph.D. dissertation, Dept. Automat. Control Syst. Eng., Univ. Sheffield, Sheffield, U.K., 1998.

Andrew J. Chipperfield received the B.Sc. degree in computer systems engineering from the University of Wales, Bangor, U.K., and the Ph.D. degree in parallel processing in computer-aided control systems design from The University of Sheffield, Sheffield, U.K., in 1989 and 1995, respectively.

He is currently a Lecturer in Aerospace Engineering in the Department of Automatic Control and Systems Engineering, The University of Sheffield. His current research interests include optimization and decision making in control systems design, reconfigurable flight control systems, gas turbine engine control, and systems integration for aerospace applications. He also has a number of years experience of computer control in industrial organizations.

Beatrice Bica received the B.Sc. degree in control engineering from the Politechnica University of Bucharest, Bucharest, Romania, and the Ph.D. degree in fuzzy control and modeling of gas turbine aero-engines from The University of Sheffield, Sheffield, U.K., in 1997 and 2001, respectively.

She is currently a Researcher with Unilever Research, Port Sunlight, U.K.



Peter J. Fleming is Professor of Industrial Systems Engineering in the Department of Automatic Control and Systems Engineering, The University of Sheffield, Sheffield, U.K., and is also the Director of the Rolls-Royce University Technology Centre for Control and Systems Engineering. He has previously held the positions of Professor of Computer Systems Engineering at the University of Wales, Bangor, U.K., Associate Professor at Syracuse University, Syracuse, NY, and Research Scientist at NASA, Langley, VA. His research interests include industrial applications of control, computer-aided control system design, and control applications of optimization and distributed systems. He has published extensively in these fields. He is currently Editor of the *International Journal of Systems Science*, Chair of the International Federation for Automatic Control (IFAC) Publications Committee, and a member of IFAC's Executive Board

Prof. Fleming is a member of the European Union Control Association.