Multi-objective Evolutionary Design of Fuzzy Autopilot Controller

Anna L. Blumel, Evan J. Hughes, and Brian A. White

Department of Aerospace, Power, and Sensors,
Cranfield University, Royal Military College of Science,
Shrivenham, Swindon, SN6 8LA, UK,
ejhughes@iee.org,
http://www.rmcs.cranfield.ac.uk/departments/daps/

Abstract. A multi-objective evolutionary algorithm is used to determine the membership function distribution within the outer loop control system of a non-linear missile autopilot using lateral acceleration control. This produces a design that meets objectives related to closed loop performance such as: steady state error, overshoot, settling and rising time. The evolutionary algorithm uses non-dominated sorting for forming a Pareto front of possible solutions. This paper shows that fuzzy controllers can be produced for engineering problems, with the multi-objective algorithm allowing the designer the freedom to choose solutions and investigate the properties of very complex systems.

1 Introduction

One of the main requirements for an autopilot design is to yield as fast as possible response with the minimum of overshoot so that any command is attained quickly and is of the required magnitude. For low g demands only a slight overshoot of short duration is usually acceptable, since it can compensate for loss of acceleration during the initial transient. For high g demands, overshoot is usually unacceptable since the airframe structural load limit may be exceeded.

In order that the autopilot yields an accurate and instantaneous reaction, it is very important to assess the quality of the lateral acceleration response which is quantified in terms of rise time, settling time, maximum percentage overshoot with almost no steady state error. This means that while tuning the trajectory control parameters, the optimisation process should consider those four criteria simultaneously. Hence the single optimisation problem has become multi-objective, being able to provide the designer with multiple solutions. The four criteria are conflicting in their nature and a compromise solution may well be taken.

The aim of this paper is to track the missile lateral acceleration demand in the presence of uncertainties introduced through the aerodynamic coefficients. The g demands are considered for both pitch and yaw planes, using the missile rudder and elevator as control surfaces hence yielding a system with 2 inputs and 2 controlled outputs.

Feedback linearisation of the autopilot is combined with a fuzzy logic trajectory controller to provide control of the missile lateral acceleration. A multi-objective evolutionary algorithm based on non-dominated sorting is used to evolve the fuzzy controllers. In previous research [1, 2] a combination of an input/output linearisation technique (nonlinear control law) and a fuzzy logic trajectory controller have been considered for velocity control. This paper extends the techniques to direct control of acceleration.

2 Fuzzy Trajectory Controller

The autopilot design system shown in Fig. 1 consists of the missile model and autopilot simulation feeding the performance analysis and objective generation functions, allowing the evolutionary algorithm to optimise the fuzzy control surfaces.

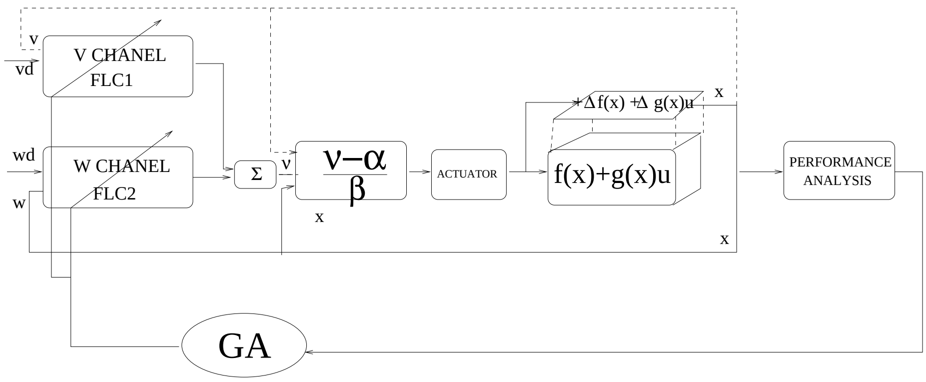

Fig. 1. Trajectory control design

The missile model is surrounded by an inner feed-back linearisation loop, with outer loops containing the fuzzy trajectory controllers for the pitch and yaw channels. The fuzzy controllers for pitch and yaw are identical as the system is symmetrical, therefore only one fuzzy surface needs to be evolved.

The missile dynamics are presented by

$$\begin{aligned} \dot{x} &= f(x) + \triangle f(x) + (g(x) + \triangle g(x))u \\ y &= h(x) \end{aligned} \tag{1}$$

which presents the multi-modelling frame. A fast $250[rads/sec]$ second order linear actuator is included within the missile dynamics. Fin angles and fin rates are states of the system. The non-linear control law is $\frac{\nu-\alpha}{\beta}$, derived by the feedback linearisation technique, which decouples the system.

The trajectory controller has been designed, based on a fuzzy inference engine, as a two input - one output system with five membership functions for each variable. The membership functions' positions and the rules are generated using an evolutionary algorithm.

3 Missile Model

The missile model used in this study derives from a non-linear model produced by Horton of Matra-British Aerospace [3]. It describes a 5 DOF model in parametric format with severe cross-coupling and non-linear behaviour. This study has considered the reduced problem of a 4 DOF controller for the pitch and yaw planes without roll coupling. The angular and translational equations of motion of the missile airframe are given by:

$$
\begin{aligned}
\dot{r} &= \frac{1}{2} I_{yz}^{-1} \rho V_o S d \left(\frac{1}{2} d C_{nr} r + C_{nv} v + V_o C_{n\zeta} \zeta\right) \\
\dot{v} &= \frac{1}{2m} \rho V_o S \left(C_{yv} v + V_o C_{y\zeta} \zeta\right) - U r \qquad (2) \\
\dot{q} &= \frac{1}{2} I_{yz}^{-1} \rho V_o S d \left(\frac{1}{2} d C_{mq} q + C_{mw} w + V_o C_{m\eta} \eta\right) \\
\dot{w} &= \frac{1}{2m} \rho V_o S \left(C_{zw} w + V_o C_{z\eta} \eta\right) + U q \qquad (3)
\end{aligned}
$$

where the axes(x, y, z), rates(r, q) and velocities (v, w) are defined in Fig. 2.

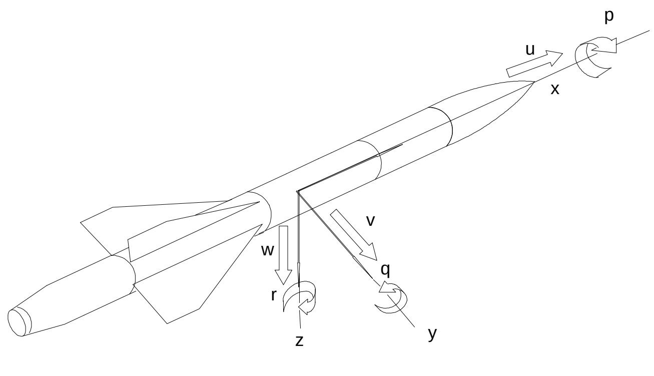

Fig. 2. Airframe axes

Equations 2 & 3 describe the dynamics of the body rates and velocities under the influence of external forces (e.g. C_{zw}) and moments (e.g. C_{mq}), acting on the frame. These forces and moments, derived from wind tunnel measurements, are non-linear functions of Mach number longitudinal and lateral velocities, control surface deflection, aerodynamic roll angle and body rates.

The description of the model is obtained from data supplied by Matra-BAE and detailed in the Horton report [4]. As both horizontal and vertical lateral motion is symmetric in format, both will be dealt with together, taking into account the appropriate sign changes in derivatives for each lateral direction.

It has been shown [5] that the desired tracking performance for lateral acceleration can be obtained by assuming an exact knowledge of aerodynamic coefficients and missile configuration parameters (i.e., reference area, Mach number, mass, moment of inertia). In practice however, this assumption is not valid and also, if there are parameter variations or external disturbances, feedback-linearisation can no longer guarantee the desired performance or neither is robustness guaranteed. For these reasons, a combination of an input/output linearisation technique (nonlinear control law) and a fuzzy logic controller(trajectory controller) have been chosen to be considered here.

4 Evolutionary Algorithm Structure

The proposed framework maintains a population of fuzzy rule sets with their membership functions and uses the evolutionary algorithm to automatically derive the resulting fuzzy knowledge base.

A hybrid real valued/binary chromosome has been used to define each individual fuzzy system. Figure 3 shows the chromosome structure for a five membership function system. The real valued parameters are defined as being the $[\delta a \, \delta b \, \delta c \, \delta d]$ and lie in the range $(0, 1]$. The binary component encodes the set of rules used in the system. Each rule is either on or off (0/1) and corresponds to the form:

$$\text{if } A_i \text{ AND } B_j \text{ then } O_k \tag{4}$$

where A_i denotes membership function i of input A, B_j denotes membership function j of input b, and O_k denotes membership function k of the output O. This process allows a full set of rules to be developed for the fuzzy system, but maintains a fixed length chromosome. The five membership function structure leads to a chromosome with 12 real valued genes and 125 binary genes. The fuzzy system used product for the member function 'AND'. The 'OR' function was not required as the rules were all expressed as 'AND' terms. The implication method was to choose the minimum value and crop the output member functions. The aggregation method was to choose the maximum values of the set of member functions. A centroid approach was used to defuzzify the output.

The evolutionary algorithm[6] follows the usual format of ranking, selection, crossover, mutation and evaluation but with the real and binary parts of the chromosomes being processed separately. A multi-objective approach was used to identify good solutions. A method known as non-dominated ranking was used in the evolutionary algorithm to allow the multi-objective problem to be handled

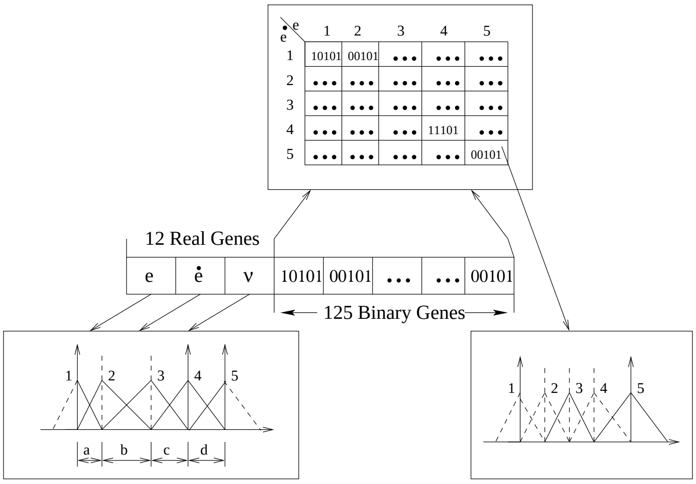

Fig. 3. FLC chromosome structure

easily. A detailed description of the NSGA non-dominated ranking process may be found in [7], and is based on several layers of classifications of the individuals.

To classify the individuals, the population is ranked on the basis of non-domination. With a single objective it is simple to distinguish between two solutions as one will have a better objective value than the other. If we have multiple objectives, solution A may be better on objective 1 but worse on all other objectives. We can no longer say which solution is best and so we say they are *non-diminated*.

In the NSGA algorithm, all non dominated individuals are classified into one category (with a dummy fitness value, which is proportional to the population size, to provide an equal reproductive potential for these individuals). To maintain the diversity of the population, these classified individuals are shared using their dummy fitness values. Then this group of classified individuals is ignored and another layer of non dominated individuals is considered. The process continues until all individuals in the population are classified.

The NSGA algorithm has been used instead of other algorithms such as MOGA [8] as when the work was started, there were no indications from the problem as to which algorithms may be better. Work has since shown that in this context, the performance of NSGA and MOGA is very similar.

A population of 100 individuals was maintained by the algorithm. Each generation, 20 individuals were selected using stochastic universal sampling for breeding. The choice of using 20 individuals per generation is a compromise between processing costs and adequate sampling of the search space. By maintaining a larger population of 100, some of the benefits of a larger population are maintained. Crossover was performed at a rate of 0.9, with intermediate crossover being used for the real values and uniform multi-point crossover for the binary

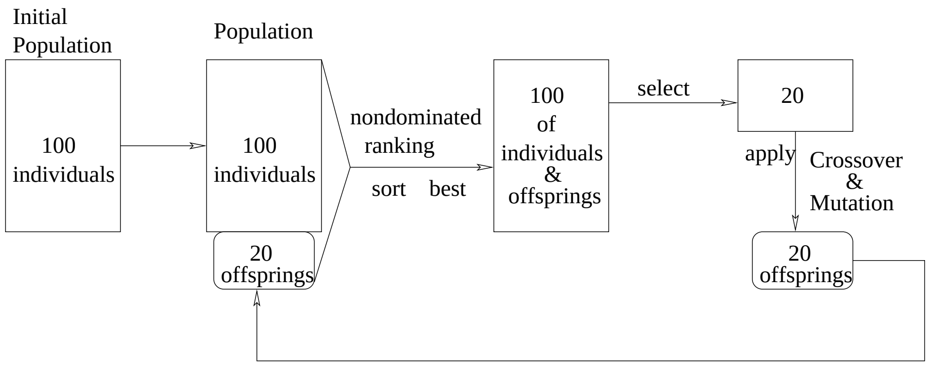

Fig. 4. Genetic algorithm structure

part. A mutation rate of 2/137 was used to give on average two mutations per chromosome. Selective pressure (SP) of 1.7. The high crossover rate and low selective pressure is to slow convergence to help prevent local optimum being exploited. The twenty new individuals were evaluated and then concatenated to the old population, forming a set of 120 individuals. Non-dominated ranking was then applied to this set and the best 100 were taken for the next generation.

In this application much of the feasible space of the controller is little used (see the results section). The genes responsible for these areas will settle to some semi-random state. That is why sometimes having a very similar used control surfaces may have very different chromosomes. This feature upsets the sharing process. A token value of $\sigma_{share} = 0.5$ was used, because in this problem, varying σ_{share} has little effect.

5 Objectives

The closed loop performance criteria are chosen as follows:

1. Side-slip velocity steady state error:

$$\eta_1(x) = \frac{Er^\star - Er(\mathbf{x})}{Er_{max} - Er_{min}} \tag{5}$$

2. Overshoot:

$$\eta_2(x) = \frac{Os^\star - Os(\mathbf{x})}{Os_{max} - Os_{min}} \tag{6}$$

3. Rise time:

$$\eta_3(x) = \frac{Tr^\star - Tr(\mathbf{x})}{Tr_{max} - Tr_{min}} \tag{7}$$

4. Settling time:

$$\eta_4(x) = \frac{Ts^\star - Ts(\mathbf{x})}{Ts_{max} - Ts_{min}} \ . \tag{8}$$

Table 1 shows the reference points used in the objective calculations.

Table 1. Closed loop performance criteria

Steady State Error(%)	Settling time(sec)	Rising time(sec)	Overshoot(%)
$Er^\star = 0.0$	$Ts^\star = 0.15$	$Tr^\star = 0.08$	$Os^\star = 4.5$
$Er_{max} = 2.0$	Ts_{max}=0.25	$Tr_{max} = 0.14$	$Os_{max} = 25.0$
$Er_{min} = 0.0$	$Ts_{min} = 0.1$	$Tr_{min} = 0.07$	$Os_{min} = 2.0$

6 Results

6.1 Lateral acceleration control

In the previous work [2] velocity was controlled. In this paper, the technique has been applied to lateral acceleration control. As a result of the multi-objective optimisation multiple solutions were obtained from which the designer can choose one which satisfies the requirements and is preferred.

In Fig. 5 we have shown a set of lateral acceleration responses from different fuzzy controllers. Some are bad with high overshoot values, very slow on rise time or settling time, but some are very good with almost no steady state error and no overshoot.

In Figs. 6 to 11 we have shown three of the fuzzy gain surfaces and their corresponding acceleration responses. Figure 7 which is best on steady state error, Fig. 9 which is within 6% error from the demand and probably not acceptable, although is with no overshoot and with satisfactory rise time and finally Fig. 11 which has been too slow on rise time and settling time but within 3% on steady state error and may not be considered as an acceptable one by the designer. The dashed line is for the augmented acceleration which possesses almost identical closed loop performance criteria with the lateral acceleration. The only difference is in the non-minimum phase effect which can be seen in the solid line for the lateral acceleration. Just to remind, here the augmented acceleration has been used to design the nonlinear control law, but the actual lateral acceleration has been used as the controlled output.

The fuzzy surface has been developed with the model exercising the nominal aerodynamic coefficients. Figure 6 shows the fuzzy surface of a trajectory controller generated by the evolutionary algorithm with the paths taken for the nominal (circles) case. This is the controller that delivers the smallest steady state

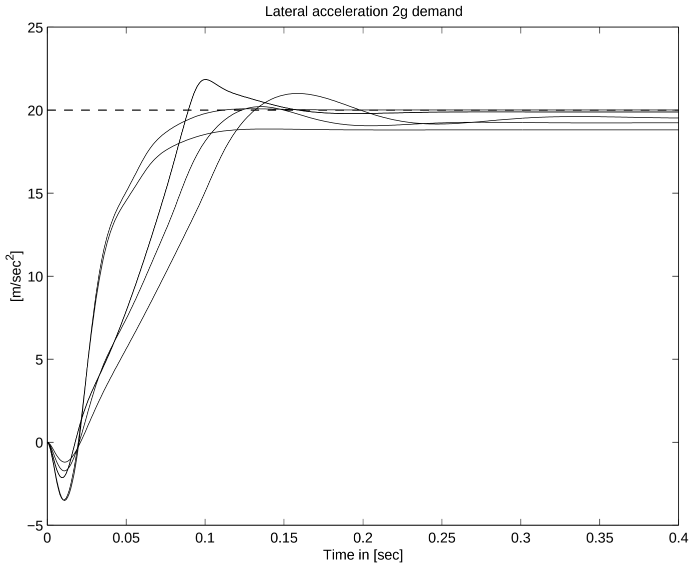

Fig. 5. A set of lateral acceleration responses

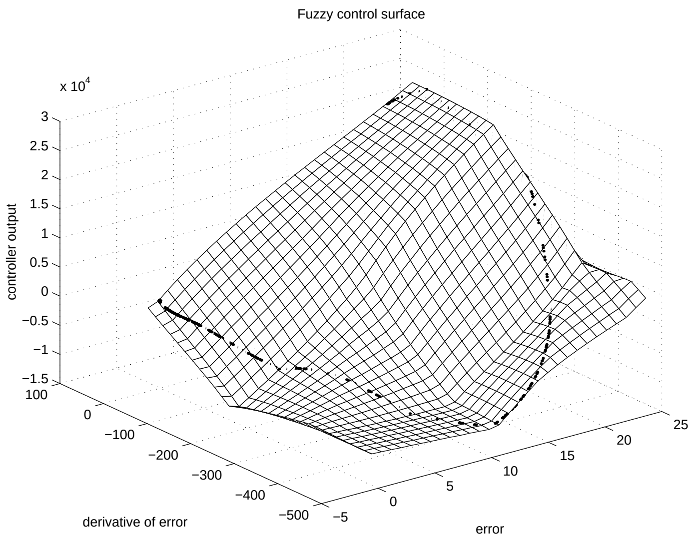

Fig. 6. Fuzzy control surface (a)

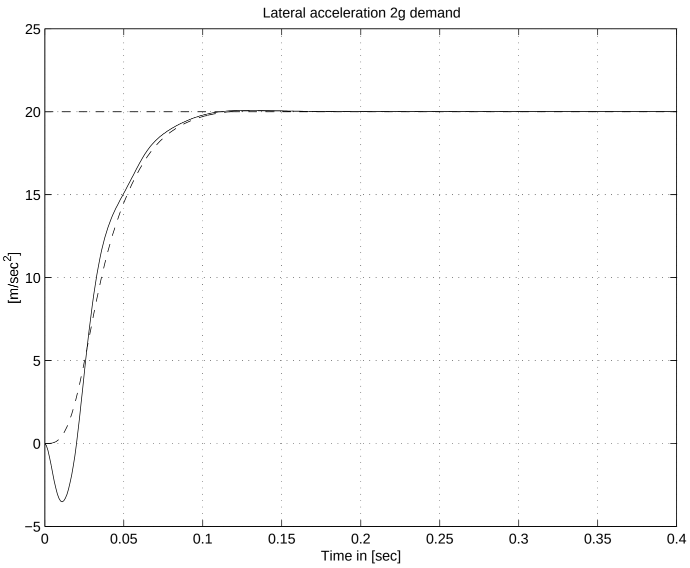

Fig. 7. Acceleration Response (a), best on steady state error

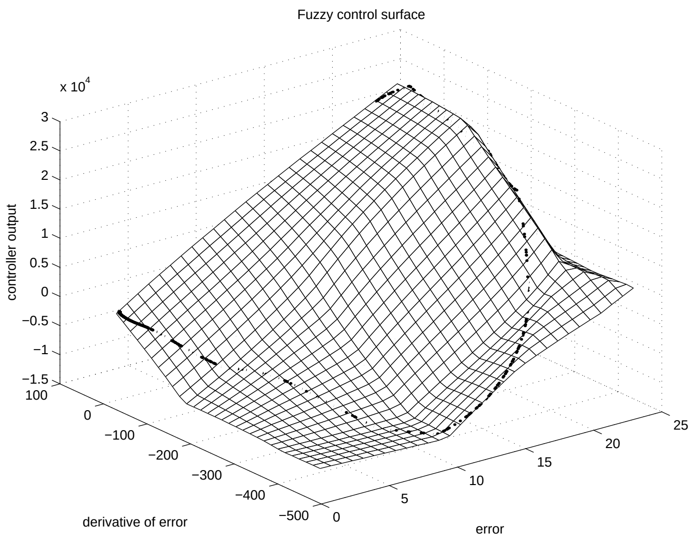

Fig. 8. Fuzzy control surface (b)

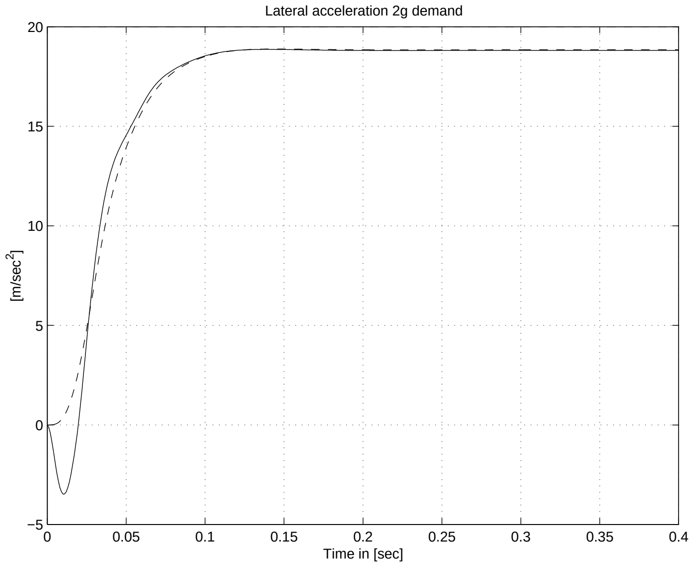

Fig. 9. Acceleration Response (b), < 6% steady state error

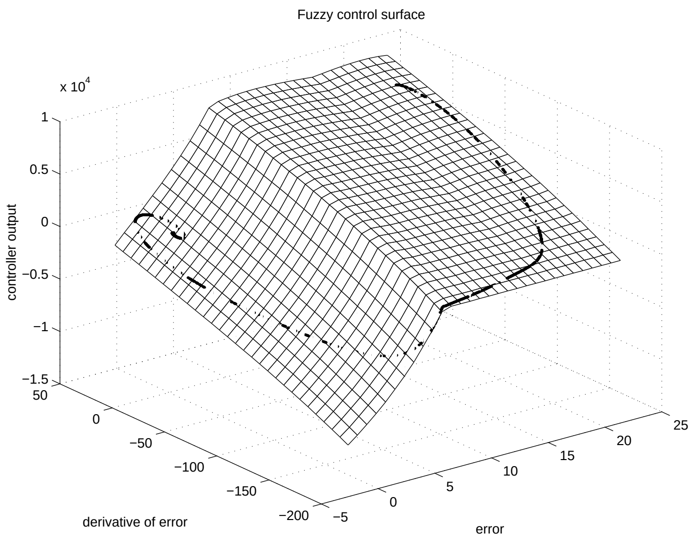

Fig. 10. Fuzzy control surface (c)

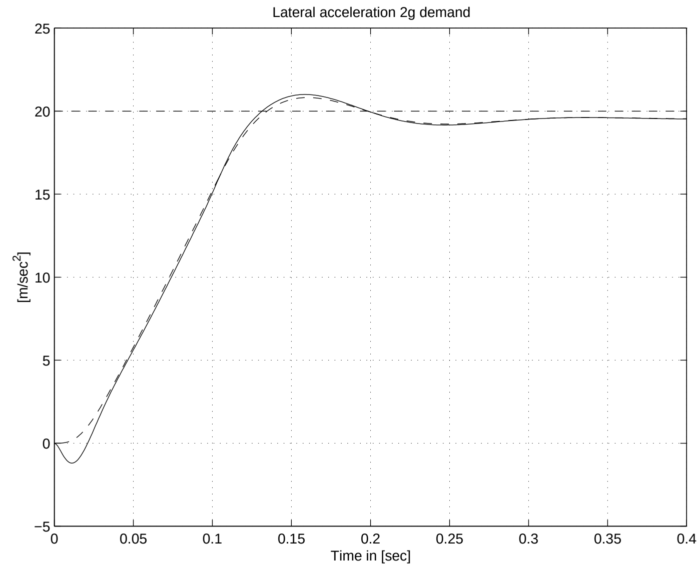

Fig. 11. Acceleration Response (c), slow rise time

error. The fuzzy logic controller has been tuned for a demand of $20[m/sec^2]$ corresponding to 2g pull lateral acceleration and is plotted on the graphs. In a typical run, about 80% of the solutions in the final population are non-dominated.

A detailed trade off surface for the individuals in a population has been illustrated in Fig. 12 and 13. The intention here is to show how solutions evolve within generations, so trade off dynamics can also be seen. Each point in each plot is a non-dominated solution. As there are four objectives, there are six possible combinations plotted. Before the convergence is achieved there are some bad and unwanted solutions clustered along the high scale of each objective - shown in Fig. 12 that die out as the evolution progresses. After convergence (approximately after 50 generations) most of the objective values are within the specified required range, which is an indication that solutions have converged towards the desired feasible area - see the scale in Fig. 13. There are few solutions strongly dominated on one objective which obviously will not been considered further by the decision maker. The final choice of solutions by the decision maker will be by applying subjective preferences for some of the objectives over others. For example, the steady state error is the primary objective to be minimised. Up to ten percent overshoot could be acceptable etc. This allows the designer to narrow down the set of solutions that need to be further analysed in detail.

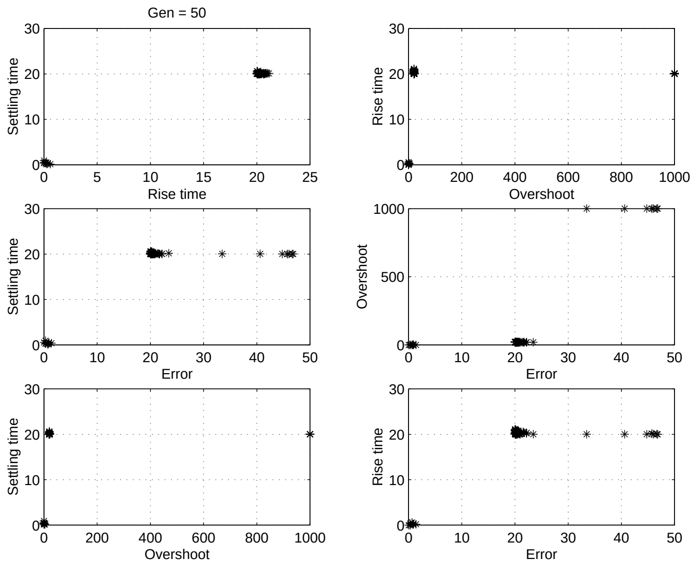

Fig. 12. Trade-off between each objective at generation 50

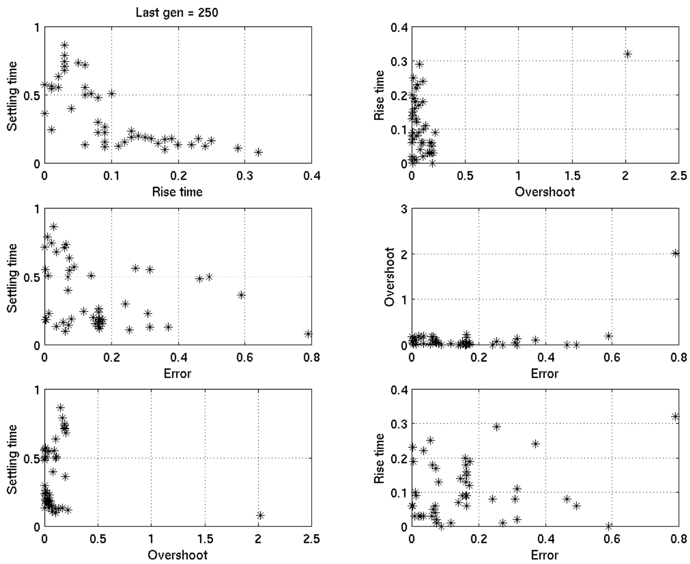

Fig. 13. Trade-off between each objectives at last generation 250

7 Conclusions

For the control of a non-linear missile's lateral acceleration, we have shown that the evolutionary algorithms can produce a good set of results that populate the Pareto solution set, allowing the system designer the flexibility of trading one solution against others to achieve a desired performance.

As Fig. 7 demonstrates, by monitoring the usage of the controller surface, combined with the fuzzy control approach, an insight can be gained into the operation of robust controllers that are created using the evolutionary process.

8 Acknowledgements

The authors would like to thank Matra-BAE for providing the data for the missile model.

References

1. H. E. Blumel A.L. and W. B.A., "Design of robust fuzzy controllers for aerospace applications," in *Proceedings of the 18 NAFIPS Conference*, vol. 1, pp. 438–442, 1998.
2. B. A. Hughes E.J. and W. B.A., "Fuzzy autopilot design using a multiobjective evolutionary algorithm," in *Congress on evolutionary computations*, vol. 1, pp. 4–5, 2000.
3. M. Horton, "A study of autopilots for the adaptive control of tactical guided missiles," Master's thesis, Univ. of Bath, 1992.
4. T. A. Blumel A.L. and B. White, "Non-linear horton missile model," tech. rep., The Royal Military College of Science,Cranfield University, D.A.P.S., R.M.C.S.,C.U., Shrivenham, Wilts, SN6 8LA, September 16 1997.
5. T. A. White, B.A. and B. A.L., "Lateral acceleration control design of a nonlinear homing missile," in *Proc. of the* 4^{th} *IFAC Nonlinear Control Systems Design Symposium*, 708-713 1998. The Netherlands.
6. A. Zalzala and P. J. Fleming, eds., *Genetic algorithms in engineering systems.* The Institution of Electrical Engineers, 1997.
7. N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1995.
8. C. M. Fonseca and P. J. Flemming, "Multiobjective genetic algorithms made easy: Selection, sharin g and mating restriction," in *GALESIA 95*, pp. 45–52, 12-14 September 1995. IEE Conf. Pub. No. 414.