MULTI-OBJECTIVE FUZZY DESIGN OF A LATERAL AUTOPILOT FOR A QUASI-LINEAR PARAMETER VARYING MISSILE

Evan J. Hughes, Antonios Tsourdos & Brian A. White *

* *Department of Aerospace, Power and Sensors,*
Cranfield University,
RMCS, Shrivenham, Swindon, UK, SN6 8LA.
ejhughes@iee.org
a.tsourdos@rmcs.cranfield.ac.uk
b.a.white@rmcs.cranfield.ac.uk

Abstract: Modern missiles are required to operate over an expanded flight envelope to meet the challenge of highly manoeuvrable targets. In such a scenario, an autopilot derived from linearisation about a single flight condition will be unable to achieve suitable performance for the whole envelope.
Gain Scheduling allows LTI control theory to be applied to time-varying and systems by obtaining linearised models at many set-points and designing a control law to satisfy local performance objectives for each point. The controller gains are then adjusted in real time as the operating conditions vary.
In this paper a fuzzy pole-placement control design technique is applied to the autopilot design for the missile, with the fuzzy control surfaces being designed using a multi-objective evolutionary algorithm. The missile is modelled to be quasi-linear with unknown parameters. The performance objectives related with the transient, i.e. settling time and rising time are achieved with the fuzzy pole-placement.

Keywords: Evolutionary Algorithms, Guidance Systems, Fuzzy Logic, Missiles, Multi-objective Optimisation

1. QUASI-LINEAR PARAMETER VARYING MISSILE MODEL

Missile autopilots are usually designed using linear models of nonlinear equations of motion and aerodynamic forces and moments (Horton, 1995). The objective of this paper is robust design of a lateral acceleration autopilot for a quasi-linear parameter varying missile model. This model describes a reasonably realistic airframe of a tail-controlled tactical missile in the cruciform fin configuration (Figure 1). The aerodynamic parameters in this model are derived from wind-tunnel measurements (Horton, 1992). Previous work has looked at designing an autopilot using feedback linearisation with optimised fuzzy trajectory control (A.L.Blumel *et al.*, 2001; Blumel A.L. and B.A., 1998; Hughes E.J. and B.A., 2000).

The starting point for mathematical description of the missile is the following nonlinear model (Tsourdos *et al.*, 1998), (Horton, 1992) of the horizontal motion (on the *xy* plane in Figure 1):

$$\begin{aligned}\dot{v} &= y_v(M,\lambda,\sigma)v - Ur + y_\zeta(M,\lambda,\sigma)\zeta \\ &= \frac{1}{2}m^{-1}\rho V_o S(C_{y_v}v + V_o C_{y_\zeta}\zeta) - Ur \\ \dot{r} &= n_v(M,\lambda,\sigma)v + n_r(M,\lambda,\sigma)r + n_\zeta(M,\lambda,\sigma)\zeta \\ &= \frac{1}{2}I_z^{-1}\rho V_o S d\left(\frac{1}{2}dC_{n_r}r + C_{n_v}v + V_o C_{n_\zeta}\zeta\right). \end{aligned} \quad (1)$$

where the variables are defined in Figure 1. Here v

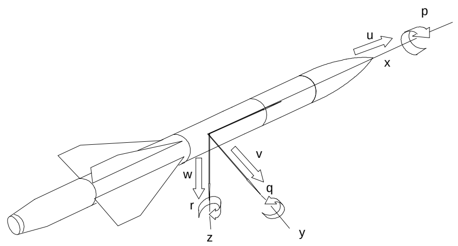

Fig. 1. Airframe axes.

is the sideslip velocity, r is the body rate, ζ the rudder fin deflections, y_v, y_ζ semi-non-dimensional force derivatives due to lateral and fin angle, n_v, n_ζ, n_r semi-non-dimensional moment derivatives due to sideslip velocity, fin angle and body rate. Finally, U is the longitudinal velocity. Furthermore, $m = 125$ kg is the missile mass, $\rho = \rho_0 - 0.094h$ air density ($\rho_0 = 1.23$ kgm^{-3} is the sea level air density and h the missile altitude in km), V_o the total velocity in ms^{-1}, $S = \pi d^2/4 = 0.0314$ m^2 the reference area ($d = 0.2$ m is the reference diameter) and $I_z = 67.5$ kgm^2 is the lateral inertia. For the coefficients $C_{y_v}, C_{y_\zeta}, C_{n_r}, C_{n_v}, C_{n_\zeta}$ only discrete data points are available, obtained from wind tunnel experiments. Hence, an interpolation formula, involving the Mach number $M \in [0.6, 6.0]$, roll angle $\lambda \in [4.5°, 45°]$ and total incidence $\sigma \in [3°, 30°]$, has been calculated with the results summarised in Table 1.

Table 1. Coefficients in nonlinear model (1)

	Interpolated formula
C_{y_v}	$0.5[(-25+M-60\|\sigma\|)(1+\cos 4\lambda)+$ $(-26+1.5M-30\|\sigma\|)(1-\cos 4\lambda)]$
C_{y_ζ}	$10+0.5[(-1.6M+2\|\sigma\|)(1+\cos 4\lambda)+$ $(-1.4M+1.5\|\sigma\|)(1-\cos 4\lambda)]$
C_{n_r}	$-500-30M+200\|\sigma\|$
C_{n_v}	$s_m C_{y_v}$, where: $s_m = d^{-1}[1.3+0.1M+0.2(1+\cos 4\lambda)\|\sigma\|+$ $0.3(1-\cos 4\lambda)\|\sigma\|-(1.3+m/500)]$
C_{n_ζ}	$s_f C_{y_\zeta}$, where: $s_f = d^{-1}[2.6-(1.3+m/500)]$

The total velocity vector V_o is the sum of the longitudinal velocity vector U and the sideslip velocity vector v, i.e. $V_o = U + v$, with all three vectors lying on the xy plane (see Figure 1). We assume that $U \gg v$, so that the total incidence σ, or the angle between U and V_o, can be taken as $\sigma = v/V_o$, as $\sin\sigma \approx \sigma$ for small σ. Thus, we have $\sigma = v/V_o = v/\sqrt{v^2+U^2}$, so that the total incidence is a nonlinear function of the sideslip velocity and longitudinal velocity, $\sigma = \sigma(v, U)$.

The Mach number is obviously defined as $M = V_o/a$, where a is the speed of sound. Since $V_o = \sqrt{v^2+U^2}$, the Mach number is also a nonlinear function of the sideslip velocity and longitudinal velocity, $M = M(v, U)$.

It follows from the above discussion that all coefficients in Table 1 can be interpreted as nonlinear functions of three variables: sideslip velocity v, longitudinal velocity U and roll angle λ.

For an equilibrium (v_0, r_0, ζ_0) it is possible to derive from (1) a linear model in incremental variables, $\bar{v} \doteq v - v_0$, $\bar{r} \doteq r - r_0$ and $\bar{\zeta} \doteq \zeta - \zeta_0$. In particular, for the straight level flight (with gravity influence neglected), we have $(v_0, r_0, \zeta_0) = (0, 0, 0)$, so that the incremental and absolute variables are numerically identical, although conceptually different.

2. DESIGN OF LATERAL MISSILE AUTOPILOT

2.1 Control design via fuzzy pole-placement

The general structure of the feedback control law is given in (2) where $\mathbf{x}$ is the state variable vector to be determined in terms of $\mathbf{x}$ and the reference signal.

$$\mathbf{u_a} = -K(p)^T \mathbf{x} \tag{2}$$

It should be noted that $K(p)$ and in particular the values of the longitudinal and lateral controllers $\mathbf{K_1(p)}$ and $\mathbf{K_2(p)}$ are obtained using the pole-placement technique.

Substituting the control law in the state equation yields $\dot{x} = A(p)^* x$ with the augmented matrix A^* to be given by $A^* = A(p) - BK(p)^T$. The characteristic equation of the augmented system can now be determined from $|\lambda I - A(p)^*|$

Equating the above mathematical expression of the characteristic polynomial of the augmented system with the one of the desired (obtained using the desired performance characteristics) the coefficients of the pole-placement controller for each of the local models are easily obtained.

2.2 Fuzzy Inference System

A Takagi-Sugeno (T-S) fuzzy controller (Tanaka and Sugeno, 1985) is used to determine the control gains required for any given Mach and incidence angle in order to generate a system with a given performance characteristic. The system has two inputs, Mach and incidence, and generates three outputs which are the three control gains required for the PI controller.

The Takagi-Sugeno (T-S) fuzzy controller is composed of r rules that can be represented as:

Plant rule i: **If** e_i is M_j and e_i is I_k
Then $\delta K_{n_i} = K_{n_i}$,
$i = l, 2, \ldots r$, and $n = 1, \ldots 3$,

Where M_j and I_k are individual membership functions of the two inputs and K_{n_i} is the required set of gains for the rule.

The T-S fuzzy model infers the gains $K_{n_i}(t)$ as the output of the fuzzy model, given all the rules, as follows, where ν_i is the total degree of membership for rule i.

$$K_{n_i} = \frac{\sum_{i=1}^{r} \nu_i [\delta K_{n_i}]}{\sum_{i=1}^{r} \nu_i} \tag{3}$$

2.3 Tracking control design

This controller would produce a desired transient response of all the local models by placing the poles of all the local systems within a specified area. However since our aim is good tracking performance for the missile, we should include into to the design specification peak overshoot, settling time and zero steady-state error. Zero steady state error can be achieved with an integral term in the forward path.

The new augmented model would contain one more state variable to account for this integral term. This new state variable is defined as:

$$x_i = \int_{t_0}^{t} \mathbf{e} dt = \int_{t_0}^{t} (y - r)\, dt \tag{4}$$

Therefore,

$$\dot{x}_i = [y_d - r] \tag{5}$$

The state space now is described by:

$$\begin{bmatrix} \dot{x} \\ - \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A(p) & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \zeta + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} r \tag{6}$$

The compensated system therefore becomes:

$$\begin{bmatrix} \dot{x} \\ - \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A(p) - BK(p) & -BK_i \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ - \\ x_i \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ - \\ \mathbf{I} \end{bmatrix} r \tag{7}$$

The characteristic polynomial of the compensated system is then equated with the desired polynomial at each step to adapt the controller gains.

The compensated system is of one order higher than the nominal one. This is because of the integral term, added for tracking purposes. The third pole has to be placed however in such location that the third order compensated system to behave similar to a second order. This add an extra requirement to the selection of gains for pole placement.

3. MULTI-OBJECTIVE EVOLUTIONARY ALGORITHM

3.1 Introduction

Evolutionary Algorithms are optimisation procedures which operate over a number of cycles (generations) and are designed to mimic the natural selection process through evolution and survival of the fittest. In this paper, a multi-objective evolutionary strategy (Deb, 2001) has been applied for optimising the fuzzy control surfaces.

3.2 Algorithm structure

The evolutionary strategy begins by generating an initial population of 50 chromosomes at random with the standard deviations of the mutations all set initially as one eighth of the total range of each gene. The initial population is evaluated and objective values generated (see section 3.3) and then sorted (section3.4). Crossover and mutation are then applied to the chromosomes to generate another 50 chromosomes. These new chromosomes are then evaluated and the best 50 from all 100 chromosomes are chosen for the next generation. The process is repeated for 100 generations.

The crossover operation takes each chromosome in turn (chromosome a), and for each chooses a second chromosome at random (with replacement) to cross with (chromosome b). A new chromosome (c) is generated 70% of the time using (8), and for the remaining 30% of the time, a copy of chromosome a is made. In (8), a_k, b_k & c_k are gene k of chromosomes a, b & c and U_k is a uniform random number in the range [0,1] chosen anew for each gene and each chromosome a.

$$c_k = a_k + (b_k - a_k)(1.5U - 0.25) \tag{8}$$

The evolutionary strategy updates the standard deviation of the mutation and the value of each gene for every gene in each new chromosome, using (9). In (9), $\sigma'_k(x)$ is the standard deviation of gene k of chromosome x, $\omega'_k(x)$ is the value of gene k of chromosome x, $N(0,1)$ is a random number with zero mean and unity variance Gaussian distribution and is chosen once per chromosome, $N_k(0,1)$ is a random number with zero mean and unity variance Gaussian distribution and is chosen afresh for every gene, and n is the number of genes in each chromosome.

$$\begin{aligned} \sigma'_k(x) &= \sigma_k(x) \exp(\tau_0 N(0,1) + \tau_1 N_k(0,1)) \\ \omega'_k(x) &= \omega_k(x) + \sigma'_k(x) N_k(0,1) \\ \tau_0 &= \frac{1}{\sqrt{2\sqrt{n}}} \\ \tau_1 &= \frac{1}{\sqrt{2n}} \end{aligned} \tag{9}$$

3.3 Chromosome structure and objectives

Chromosome The chromosome structure needs to represent both the membership functions for the two inputs, and the output values for every possible rule. Three, four and five membership functions have been used for each of the two inputs. The member functions are triangular and overlapping to always give a unity sum as shown in figure 2 For the two inputs, the input

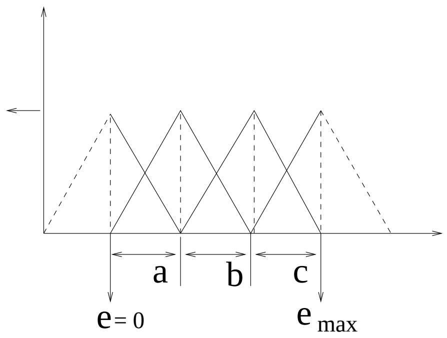

Fig. 2. Membership function structure

ranges are $e_0 = 0.6$ to $e_m = 6$ for the Mach number, and $e_0 = 0^\circ$ to $e_m = 30^\circ$ for the incidence. Each of the genes must lie in the range (0,1].

With n member functions per input, there will be n^2 possible rules. The output value for each the rules is simply a triplet of constants, one for each of the three outputs. At each Mach–incidence combination, the three control gains are calculated by evaluating a local model of the system. The gains calculated by the local model are then associated with the corresponding rule and used to create the fuzzy control surface.

Objectives The performance is tested by generating the step response of the system for 100 uniformly spaced points in the Mach / incidence domain (10 per input). The rise time and settling time of the system are recorded at each point. Two objectives are then generated that summarise the performance of each chromosome.

The first objective is taken as the difference between the slowest and fastest rise times of the 100 trials for each chromosome. The second objective is the difference between the slowest and fastest settling times.

3.4 Non-dominated Ranking

With multiple objectives, a Pareto-optimal set of results (Deb, 2001) may be formed where no single solution is better than any other in all objectives. These solutions are said to be *non-dominated* as no solution can be chosen in preference to the others based on the all objectives alone. There exists a single Pareto-optimal set of solutions to the problem. At any intermediate stage of optimisation, a set of non-dominated results will have been identified. This set may or may not be the Pareto optimal set.

A non-dominated ranking method (Deb, 2001) is used in the evolutionary algorithm to generate and maintain a non-dominated set of results. Conventional evolutionary algorithms often use a ranking method where the calculated objective values are sorted and assigned a rank that is dependent only upon their position in the list, rather than their objective value. The ranking operation helps to prevent premature convergence of the evolutionary algorithm.

The non-dominated ranking system operates by first identifying the non-dominated solutions in the population and assigning them a rank of one. A dummy value (1 in this implementation) is assigned to these solutions and a sharing process is applied. With the sharing, the dummy values of the individuals' are reduced if they have near neighbours (in the objective space). The sharing process ensures that a spread of solutions is obtained across the non-dominated front. The minimum value assigned to the level-one solutions is identified and then reduced slightly (by 1%) and used as a dummy value for the next level of processing. The level-one individuals are removed from the population and the identification–sharing process repeated on the remaining set, using the reduced dummy value for the sharing operation. The ranking process is continued until all of the individuals have been accounted for. The resulting objectives are intended to be used with a *maximisation* strategy and have been adjusted to allow both of the objectives to be minimised.

4. SIMULATION RESULTS

Figure 3 shows a set of typical non-dominated surfaces after 100 generations for each of the membership function configurations. Both of the objectives cannot be minimised simultaneously, so the Pareto front forms curves. All of the solutions on the non-dominated front are valid solutions to the problem and it is down to the system designer to choose a single solution for use in the control system.

Figures 4 and 5 show the locations of the membership functions for the 3 membership function system (MF 1 and 3 are at zero and 1 respectively). The positions are sorted to correspond to the order of the points on the Pareto set, with point 1 corresponding to the solution in the top left hand corner of the Pareto set. Figures 6, 7, 8 & 9 show the corresponding plots for the trials with 4 and 5 membership functions per input.

It is clear from figures 4, 6 & 8 that the Mach input is dominant when shaping the control surfaces. The lines on the plots progress smoothly with respect to the objective surface, whereas figures 5, 7 & 9 have little correlation with the progression of the objectives. This effect suggests that fewer membership functions are required for the incidence input.

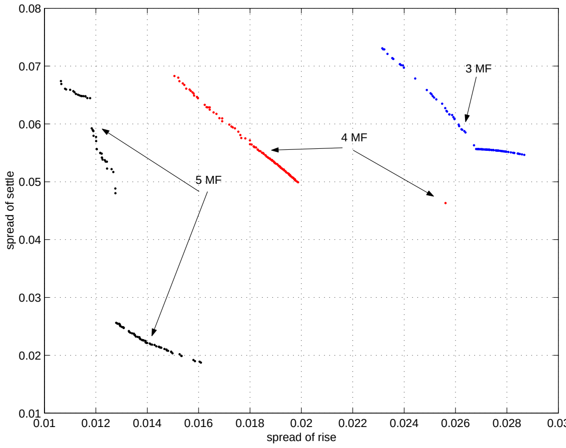

Fig. 3. Pareto set for 5 Membership Functions (MF) each, 4 MF each and 3 MF each

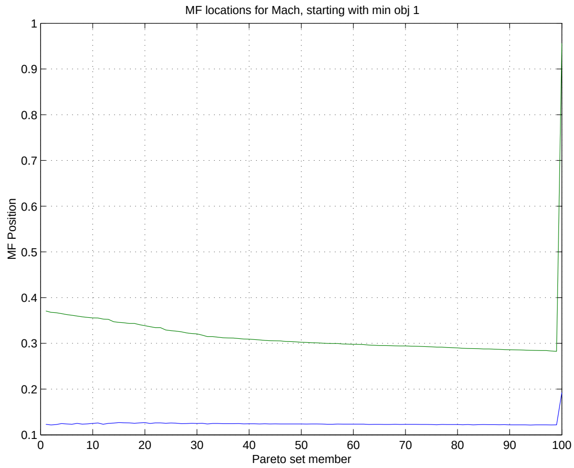

Fig. 6. Membership function locations for Mach and 4 member functions

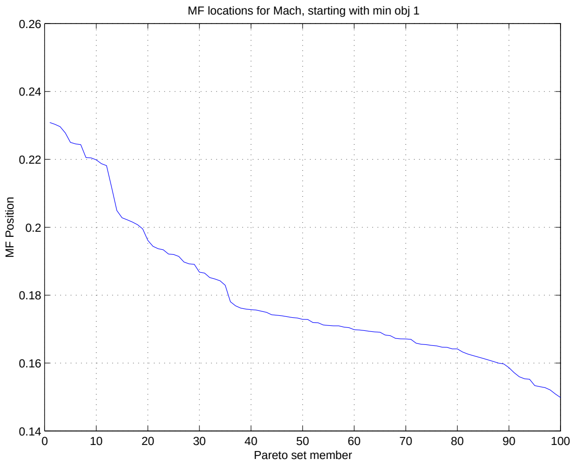

Fig. 4. Membership function locations for Mach and 3 member functions

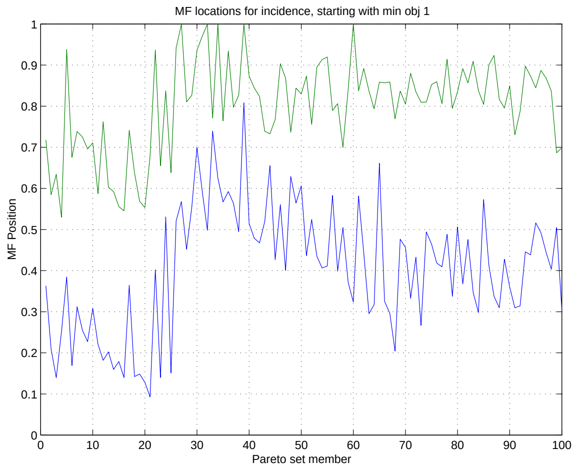

Fig. 7. Membership function locations for Incidence and 4 member functions

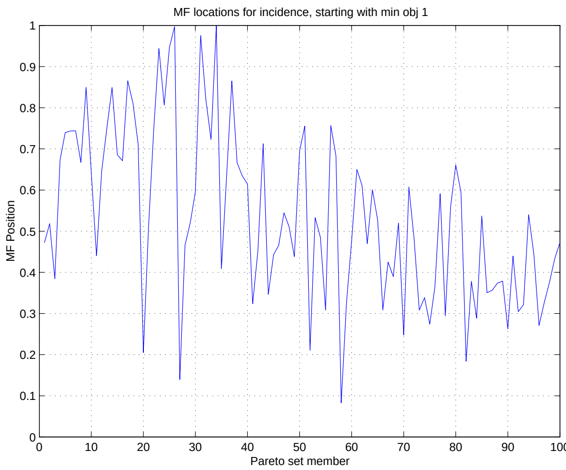

Fig. 5. Membership function locations for Incidence and 3 member functions

Figures 10, 11 & 12 show the surfaces generated by the fuzzy inference systems for the three control gains for the solution with 5 membership functions in both inputs that minimises the error in the spread of rise times.

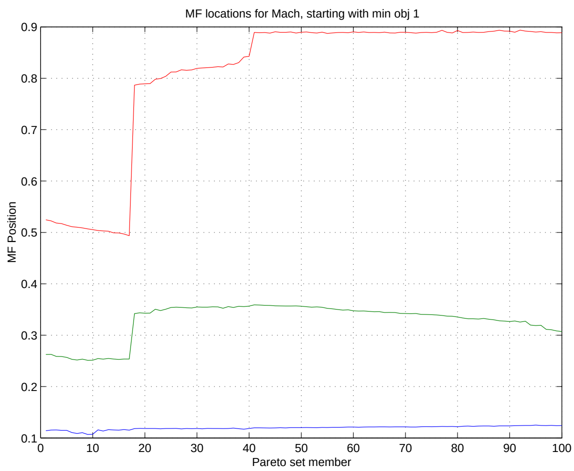

Fig. 8. Membership function locations for Mach and 5 member functions

5. CONCLUSIONS

This paper has shown that a fuzzy pole-placement controller can be designed for complex non-linear systems to produce given performance over a range of plant conditions. The use of evolutionary algorithms

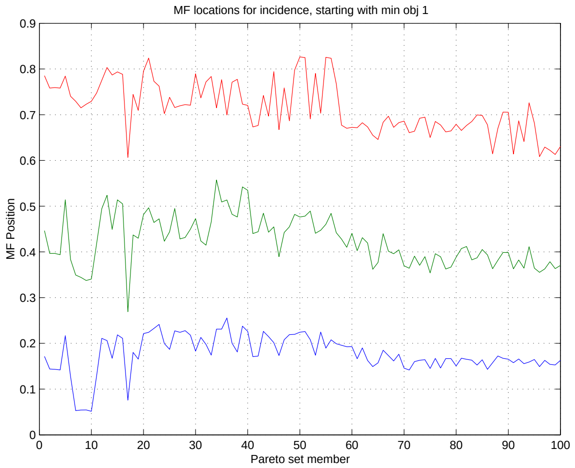

Fig. 9. Membership function locations for Incidence and 5 member functions

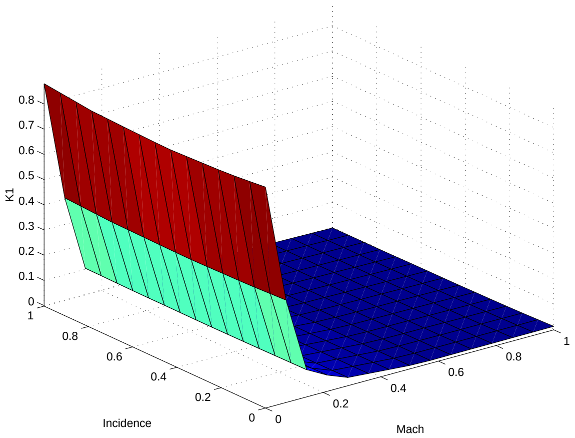

Fig. 10. Example gain surface for K_1 and 5 MF per input

to optimise the fuzzy inference system removes the requirement of expert knowledge to design the fuzzy landscape as the multi-objective algorithm is capable of discovering a range of solutions with little designer intervention. The multi-objective formulation allows many potential solutions to be generated simultaneously. The designer can then choose a candidate solution whilst being informed of what other solutions to the problem may exist.

The work shows that for the chosen plant, increasing the number of membership functions in Mach will improve performance, while three or four membership functions for the incidence input would suffice. This allows the total number of membership functions and rules used in the system to be minimised, allowing processing speed to be maximised.

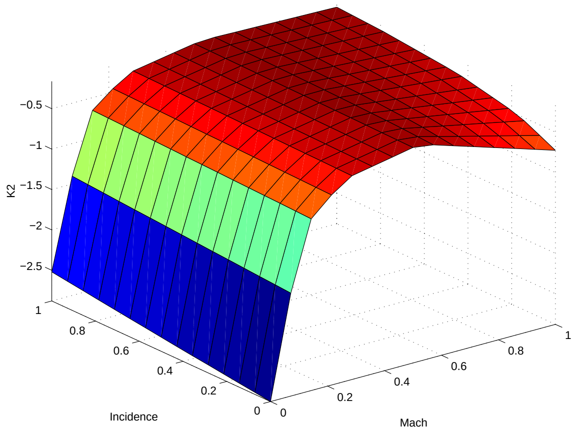

Fig. 11. Example gain surface for K_2 and 5 MF per input

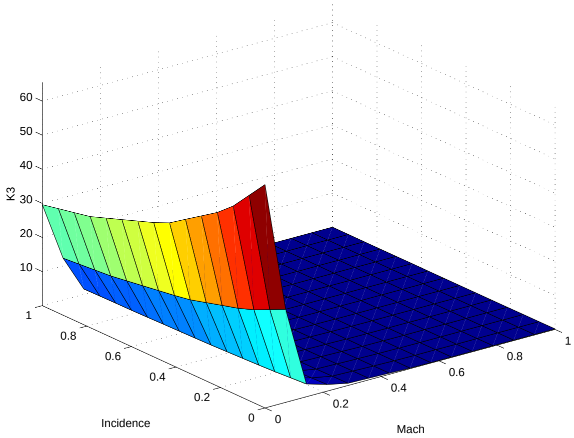

Fig. 12. Example gain surface for K_3 and 5 MF per input

REFERENCES

A.L.Blumel, E.J.Hughes and B.A.White (2001). Multi-objective evolutionary design of fuzzy autopilot controller. In: *EMO 2001*. Springer LNCS 1993. pp. 668–680.

Blumel A.L., Hughes E.J. and White B.A. (1998). Design of robust fuzzy controllers for aerospace applications. In: *Proceedings of the 18 NAFIPS Conference*. Vol. 1. pp. 438–442.

Deb, Kalyanmoy (2001). *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons.

Horton, M. P. (1992). A study of autopilots for the adaptive control of tactical guided missiles. Master's thesis. University of Bath.

Horton, M. P. (1995). Autopilots for tactical missiles: An overview. *IMechE: Journal of Systems and Control Engineering* **209**, 127–139.

Hughes E.J., Blumel A.L. and White B.A. (2000). Fuzzy autopilot design using a multiobjective evolutionary algorithm. In: *Congress on evolutionary computations*. Vol. 1. pp. 4–5.

Tanaka, T. and M. Sugeno (1985). Fuzzy indentification of systems and its applications to modeling and control. *IEEE Transactions on Systems, Man and Cybernetics* **15**(1), 116–132.

Tsourdos, A., A. Blumel and B. A. White (1998). Trajectory control of a nonlinear homing missile. In: *14th IFAC Symposium on Automatic Control in Aerospace*. pp. 118–123.