Available online at www.sciencedirect.com
SCIENCE DIRECT
Control Engineering Practice 14 (2006) 547–561
CONTROL ENGINEERING PRACTICE
www.elsevier.com/locate/conengprac

Fuzzy multi-objective design for a lateral missile autopilot

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

Department of Aerospace, Power & Sensors, Cranfield University, RMCS Shrivenham, Swindon SN6 8LA, UK

Received 2 June 2003; accepted 18 January 2005
Available online 2 March 2005

Abstract

Gain scheduled control is one very useful control technique for linear parameter-varying (LPV) and nonlinear systems. A disadvantage of gain-scheduled control is that it is not easy to design a controller that guarantees the global stability of the closed-loop system over the entire operating range from the theoretical point of view. Another disadvantage is that the interpolation increases in complexity as number of scheduling parameters increases. As an improvement, this paper presents a gain-scheduling control technique, in which fuzzy logic is used to construct a model representing a quasi-LPV or a nonlinear missile and to perform a control law. The fuzzy inference system is generated using a multi-objective evolutionary algorithm to optimize the performance characteristics of the plant.

Keywords: Gain scheduled control; Fuzzy multi-objective design

1. Introduction

One of the most popular methods for applying linear time-invariant (LTI) control theory to time-varying and/or nonlinear systems is gain scheduling (Rugh, 1993; Leith and Leithead, 2000). This strategy involves obtaining Taylor linearized models for the plant at finitely many equilibria (“set points”), designing an LTI control law (“point design”) to satisfy local performance objectives for each point, and then adjusting (“scheduling”) the controller gains in real time as the operating conditions vary. This approach has been applied successfully for many years, particularly for aircraft and process control problems. Relatively recent examples (some of which involve modern control design methods) include jet engines (Lin and Lee, 1996), active suspensions (Tran and Hrovat, 1993), high-speed drives (Beaven et al., 1995), missile autopilots (Nichols et al., 1993), and VSTOL aircraft (Hydg & Glover, 1993a,b, 1995).

Despite past success of gain scheduling in practice, until recently little has been known about it theoretically as a time-varying and/or nonlinear control technique. Also, determining the actual scheduling routine is more of an art than a science. While ad hoc approaches such as linear interpolation and curve fitting may be sufficient for simple static-gain controllers, doing the same for dynamic multivariable controllers can be a rather tedious process.

An early theoretical investigation into the performance of parameter-varying systems can be found in (Kamen and Khargonekar, 1984). During the 1980s, Rugh and his colleagues developed an analytical framework for gain scheduling using extended linearization (Baumann and Rugh, 1986; Rugh, 1993; Wang and Rugh, 1987). Also, Shamma and Athans (Shamma and Athans, 1990, 1991, 1992) introduced linear parameter-varying (LPV) systems as a tool for quantifying such heuristic design rules as “the resulting parameter must vary slowly” and “the scheduling parameter must

*Corresponding author. Tel.: +44 (0)1793 785260; fax: +44 (0)1793 785206.
E-mail addresses: A.Tsourdos@cranfield.ac.uk (A. Tsourdos), ejhughes@iee.org (E.J. Hughes), B.A.White@cranfield.ac.uk (B.A. White).

0967-0661/$ - see front matter © 2005 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2005.01.009

capture the nonlinearities of the plant". Shahruz and Behtash (Shahruz and Behtash, 1992) suggested using LPV systems for synthesizing gain-scheduled controllers, and Shamma and Cloutier (Shamma and Cloutier, 1993) have used LPV plant models with μ-synthesis (Balas et al., 1994; Packard and Doyle, 1993; Packard et al., 1993) for designing missile autopilots.

Attention has since turned to performance and design of parameter-dependent controllers for LPV systems. Various design methods which have been proposed share several common features, e.g., the current methods are based on extended state-space approaches to H_∞ optimal control for LTI systems (Doyle et al., 1989; Gahinet and Apkarian, 1994; Scherer, 1992; Stoorvogel, 1992), and LTV systems (Ravi et al., 1991). Performance is usually measured in terms of the induced L_2-norm, and controllers are designed for certain classes of parameter variations, e.g., real or complex values, arbitrarily fast or bounded rates of variation, shape of the parameter envelope etc. The resulting parameter-dependent controllers are scheduled automatically, so that the often arduous task of scheduling a complex multivariable controller a posteriori is avoided.

The existing methods also rely on linear matrix inequalities (LMI) for computing controllers, characterizing performance, and/or determining solubility of design problems. Many problems involving LMI have been found to arise in control theory (Boyd et al., 1994; Packard et al., 1991). Moreover, efficient algorithms have been developed for using convex programming to solve feasibility and optimisation problems involving LMI (Boyd et al., 1994; Nemirovskii and Gahinet, 1994); some have been incorporated into commercial software for control system design (Balas et al., 1994; Gahinet et al., 1995).

One approach involves LPV systems and controllers whose parameter dependence can be expressed as linear-fractional transformations (LFTs) (Doyle et al., 1991). The approach relies on scaled small-gain methods and scaled H_∞ optimisation to design parameter-dependent controllers that also resemble LFTs and provide closed-loop stability (Apkarian and Gahinet, 1995; Boyd et al., 1994; Packard, 1994). Initial design examples (Spillman et al., 1996) seem to indicate that the method is robust, but conservative (as small gain methods tend to be); the parameters are assumed to be complex and vary arbitrarily quickly.

Another approach uses Lyapunov methods (Vidyasagar, 1993) instead of the small-gain theorem to account for parameter variation. Becker (Becker, 1993; Becker and Packard, 1994) applied the notion of quadratic stability (Barmish, 1995; Khargonekar et al., 1990; Rotea et al., 1993) (well established for arbitrarily quickly varying LTV systems) to LPV systems, and extended it to quadratic performance by adding sufficient conditions for bounding the induced L_2-norm. These conditions are characterized by a single quadratic Lyapunov function (SQLF) that satisfies certain LMIs. Becker's doctoral thesis presents an LMIs method for providing quadratic performance using linear parameter-dependent feedback.

Since the parameters are expected to vary arbitrarily quickly, this SQLF method has also been found to be conservative (Apkarian et al., 1995; Apkarian et al., 1995; Wu et al., 1996). However, Wu and others (Becker, 1996; Wood, 1995; Wu, 1995) have reduced the conservatism of this natural extension to LMI-based H_∞ control (Gahinet and Apkarian, 1994; Iwasaki and Skelton, 1994) (though at a greater computational cost) by using parameter-dependent quadratic Lyapunov functions (PDLFs) that allow for a priori bounds on the parameters' rates of variation. Parameter-dependent quadratic Lyapunov functions have also been used by Haddad and Bernstein (Haddad and Bernstein, 1993; Haddad and Bernstein, 1995), and by Apkarian, Gahinet, and their colleagues (Feron et al., 1996; Gahinet et al., 1996) to study robustness of LTI systems to constant (or slowly varying) real, parametric uncertainty.

In this paper a sideslip velocity autopilot is designed for a realistic model of a tactical missile using fuzzy gain scheduling. The tail-controlled missile in the cruciform fin configuration (Horton, 1992) is modelled as a second-order quasi-linear parameter-varying (QLPV) system. This missile model is obtained from the Taylor linearized model of the horizontal motion by including explicit dependence of the aerodynamic derivatives on a state (sideslip velocity) and external parameters (longitudinal velocity and roll angle). The first contribution is to consider this detailed QLPV (and thus nonlinear) model.

The autopilot design is based fuzzy pole-placement control. The performance objectives related with the transient, i.e. settling time, rising time, peak overshoot are achieved with the fuzzy pole-placement. However, since our problem is one of tracking, an additional performance objective, that of zero steady-state error should be taken into account. This can be achieved with an integral term in forward loop. In this scheme, unknown parameters are estimated and based on these estimates, control parameters are updated. Computer simulations show that this approach is very promising for the motion control design for missiles, which are highly quasi-linear in dynamics. The optimisation of the fuzzy system is performed using a multi-objective evolutionary algorithm.

Evolutionary Algorithms are iterative optimization procedures inspired by Darwin's hypothesis that animals in the natural world adapt to fit the environment better through a process of survival of the fittest (Zalzala and Flemming, 1997). Unlike most classical optimisation methods that use the local gradient of the quality, or objective function, to perform a local search, evolutionary algorithms utilize a parallel search of the problem space. Where conventional algorithms are prone to becoming stuck in local optima, evolutionary algorithms are capable of finding global optima too.

In problems with multiple objectives, there may not be one single solution. There is more likely to be a set of solutions (called the Pareto set) where objectives conflict, leading to no one single solution which is better in all objectives.

In order to use a conventional optimization process with multiple objectives, the objectives must be combined into a single value to optimize. In order to construct the Pareto set, multiple runs of the optimizer must be performed with different combinations of the objectives (often weights are used to express the relative significance of each objective).

Because the evolutionary algorithm employs a parallel optimization search, it is possible to generate an approximation of the entire Pareto set in a single optimization run (Deb, 2001). The primary aim of this paper is the demonstration of the concept of applying multi-objective evolutionary algorithms to this form of control problem. The multi-objective algorithm used is considered a 'lower benchmark' on what is achievable through evolutionary methods. The basic non-dominated ranking is known to be weak at spreading out solutions across the Pareto set, and the convergence rate is not as fast as some more recent algorithms such as NSGA2 (Deb et al., 2000), SPEA2 (Zitzler et al., 2001) and ε-MOEA (Deb et al., 2003).

Section 2 details the missile model and coefficients, Section 3 describes the design of the controller and the structure of the fuzzy inference system. The multi-objective evolutionary algorithm is detailed in Section 4. Section 5.2.1 shows typical results from the optimization process and Section 6 concludes.

2. Quasi-linear parameter varying missile model

Missile autopilots are usually designed using linear models of non-linear equations of motion and aerodynamic forces and moments (Horton, 1995; Wise, 1992). The objective of this paper is the 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 (Fig. 1). The aerodynamic parameters in this model are derived from wind-tunnel measurements (Horton, 1992).

The starting point for mathematical description of the missile is the following non-linear model (Tsourdos et al., 1998; Horton, 1992) of the horizontal motion (on the xy plane in Fig. 1):

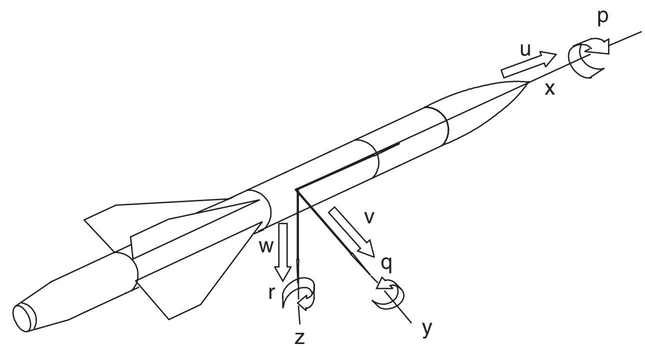

Fig. 1. Airframe axes.

$$\dot{v} = y_v(M,\lambda,\sigma)v - Ur + y_\zeta(M,\lambda,\sigma)\zeta = \tfrac{1}{2}m^{-1}\rho V_o S(C_{y_v}v + V_o C_{y_\zeta}\zeta) - Ur,$$

$$\dot{r} = n_v(M,\lambda,\sigma)v + n_r(M,\lambda,\sigma)r + n_\zeta(M,\lambda,\sigma)\zeta = \tfrac{1}{2}I_z^{-1}\rho V_o Sd(\tfrac{1}{2}dC_{n_r}r + C_{n_v}v + V_o C_{n_\zeta}\zeta). \quad (1)$$

where the variables are defined in Fig. 1.

Here v 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, $m = 125\,\text{kg}$ is the missile mass, $\rho = \rho_0 - 0.094h$ air density ($\rho_0 = 1.23\,\text{kg}\,\text{m}^{-3}$ is the sea level air density and h the missile altitude in km), V_o the total velocity in $\text{m}\,\text{s}^{-1}$, $S = \pi d^2/4 = 0.0314\,\text{m}^2$ the reference area ($d = 0.2\,\text{m}$ is the reference diameter) and $I_z = 67.5\,\text{kg}\,\text{m}^2$ 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 $\lambda \in [4.5^\circ, 45^\circ]$ and total incidence $\sigma \in [3^\circ, 30^\circ]$, has been calculated with the results summarized in Table 1.

The total velocity vector $\vec{V}_o$ is the sum of the longitudinal velocity vector $\vec{U}$ and the sideslip velocity vector $\vec{v}$, i.e. $\vec{V}_o = \vec{U} + \vec{v}$, with all three vectors lying on the xy plane (see Fig. 1). We assume that $U \gg v$, so that the total incidence σ, or the angle between $\vec{U}$ and $\vec{V}_o$, 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 non-linear 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

Table 1
Coefficients in non-linear 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: $s_f = d^{-1}[2.6-(1.3+m/500)]$

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.

2.1. Quasi-linear parameter-varying representation

The missile model introduced in Eq. (1) is non-linear with explicit state dependence, v, directly proportional to σ. The technique presented in this paper, gain scheduling (Rugh and Shamma, 2000; Leith and Leithead, 2000), starts from the quasi-linear parameter-varying form of this model.

Assume the non-linear model of the form,

$$\dot{x} = f(x, u, q),$$
$$y = h(x, u, q), \tag{2}$$

where x is the state vector, u is the control input of the system, y is the output and q an exogenous parameter.

The set, $\mathscr{E}$, of operating points (equilibrium points) of Eq. (2) depends on parameter $q \in \mathscr{R}$ and is denoted as

$$\mathscr{E} = \{(x_0, u_0, q) \in \mathscr{X} \times \mathscr{U} \times \mathscr{R} | \quad f(x_0, u_0, q) = 0\}. \tag{3}$$

The set of equilibrium for the missile is defined as $\mathscr{E} = \{(v_0, r_0, \zeta_0, U_0, \lambda_0) \in \mathscr{X} \times \mathscr{U} \times \mathscr{R} | \quad f(v_0, r_0, \zeta_0, U_0, \lambda_0) = 0\}$ where $f(v, r, \zeta, U, \lambda)$ represents the non-linear differential equation (1). The parameter $p = (v, U, \lambda)$ is introduced and it uniquely determines an operating point (equilibrium point) of the system (a point of $\mathscr{E}$). This parameter p depends on the state variable v and on the external parameters U and λ.

Note that the control of the lateral acceleration (latax), a_v, can be approximately performed via the control of the lateral velocity, $a_v = \dot{v} + Ur \approx y_v v$, since the term $y_\zeta \zeta$ will be small, due to the small lateral force generated by the tail fins compared to the tail fin moment.

Assuming incremental state variable, $\delta x = x - x_0(p)$, control input, $\delta u = u - u_0(p)$, and output, $\delta y = y - y_0(p)$, Taylor linearization of the nonlinear system (2) at an operating point of $\mathscr{E}$, uniquely defined by parameter p, leads to

$$\delta \dot{x} = \frac{\partial f}{\partial x_{|p}} \delta x + \frac{\partial f}{\partial u_{|p}} \delta u,$$
$$\delta y = \frac{\partial h}{\partial x_{|p}} \delta x. \tag{4}$$

For an equilibrium (v_0, r_0, ζ_0) uniquely defined by p, the Taylor linearization of (1) gives a linear model in incremental variables, $\delta v \doteq v - v_0$, $\delta r \doteq r - r_0$ and $\delta \zeta \doteq \zeta - \zeta_0$. In fact, dependence on p states a quasi-linear parameter-varying (QLPV) form (5), with p comprises both a state variable, v and external parameters, U and λ.

$$\delta \dot{v} = \left(\frac{\partial y_v}{\partial v} v + y_v + \frac{\partial y_\zeta}{\partial v} \zeta \right)_{|p} \delta v - U_{|p} \delta r + y_{\zeta|p} \delta \zeta,$$
$$\delta \dot{r} = \left(\frac{\partial n_v}{\partial v} v + n_v + \frac{\partial n_r}{\partial v} r + \frac{\partial n_\zeta}{\partial v} \zeta \right)_{|p} \delta v + n_{r|p} \delta r + n_{\zeta|p} \delta \zeta,$$
$$\delta y = \delta v. \tag{5}$$

In particular, for the straight level flight, we have $(v_0, r_0, \zeta_0) = (0, 0, 0)$, so that the incremental and absolute system forms are identical even if conceptually different, see Eq. (6).

$$\begin{bmatrix} \delta \dot{v} \\ \delta \dot{r} \end{bmatrix} = \begin{bmatrix} y_v(p) & -p_2 \\ n_v(p) & n_r(p) \end{bmatrix} \begin{bmatrix} \delta v \\ \delta r \end{bmatrix} + \begin{bmatrix} y_\zeta(p) \\ n_\zeta(p) \end{bmatrix} \delta \zeta,$$
$$\delta y = [1 \;\; 0] \begin{bmatrix} \delta v \\ \delta r \end{bmatrix}. \tag{6}$$

3. Fuzzy gain scheduling

3.1. An overview

Fuzzy logic has been a very useful tool for missile autopilot design. It has been successfully applied to the same missile model in the past (White et al., 2000; White et al., 2001a; White et al., 2001b) to improve the robustness of the autopilot based on non-linear dynamic inversion. Here instead of a global linearized autopilot we would use the fuzzy logic for smooth and stable gain scheduling. Fuzzy gain scheduling has become a very attractive approach for many aerospace control design application (Fleming et al., 2002; Hwang and Lin, 2003; Wu et al., 1999; Nam and Hong, 2003). The tuning of the fuzzy controller can be implemented either with the use of linear matrix inequalities or using genetic algorithms. It has been shown in (Molina-Cristobal et al., 2004) that using multi-objective genetic algorithms one can obtain a controller with better performance than when linear matrix inequalities are used, while maintaining the stability properties of the LMI optimization approach.

Fig. 2 shows a block diagram description of gain scheduling. The linear parameter-varying system constitutes a family of systems. For each member of this family of systems a LTI design is carried on. This discrete family of gain feedback controllers is first computed to guarantee performance at some operating points (related to p) and valid in its neighborhood. The linear interpolation of these controllers carries over the performance of the closed-loop system on the whole flight envelope. The controller provides an incremental control input $\delta\zeta$ and its total control to the plant is recovered as $\zeta(p) = \zeta_0(p) + \delta\zeta$.

3.2. Design of the sideslip velocity autopilot

Section 2.1 described the Horton missile in a linearized form and from Eq. (5) the incremental closed-loop system is derived for state feedback as follows:

$$\delta\dot{x} = A(p)\delta x + B(p)\delta u = [A(p) + B(p)K(p)]\delta x, \tag{7}$$

where $K(p) = [K_1(p)\ \ K_2(p)]$ is the gain scheduled controller.

A suitable pole placement for this closed-loop system (7) will guarantee stability and performance accordingly. A polytopic approach is chosen here to capture the system nonlinearities. The polytope has to be sufficiently fine and representative of the closed-loop system on the full flight envelope. However, the convex hull on the full range of flight envelope gives only very limited insight of the nonlinearities involved and a family of convex hulls representative of the system on the flight envelope is preferred.

Using evolutionary algorithms, the flight envelope is split as many times as necessary to satisfy the performance criteria. This approach leads to a discrete family of controllers each of them valid in its direct neighborhood i.e. its corresponding convex polytope. The family of polytopes, noted $\mathscr{P}$ has to cover the whole

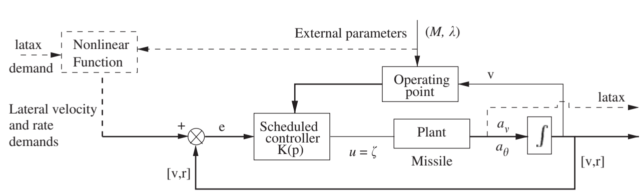

Fig. 2. Block diagram description of the gain scheduling controller.

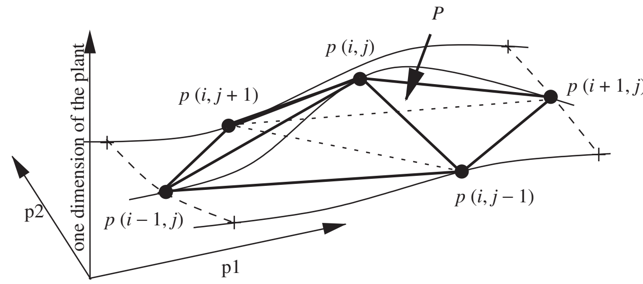

Fig. 3. A simple case of polytope P for a plant of dimension one and a parameter p of dimension two.

flight envelope, $\mathscr{F}$,

$$\bigcup_{(i,j,k)\in N^l\times N^m\times N^n} P(i,j,k) = \mathscr{P} \subset \mathscr{F}, \tag{8}$$

where (l, m, n) is the total number of controllers on each dimension. From the design results a family of controllers given by

$$\begin{aligned}\mathscr{K} = \{K(i,j,k)|\quad &\forall(i,j,k) \in N_l \times N_m \times N_n\\ &K(i,j,k) \text{ valid on polytope } P(i,j,k)\}.\end{aligned} \tag{9}$$

So far, the performance has been guaranteed only on each individual convex polytope. To extend this result to the whole flight envelope further constraints on the design have been imposed. The linear interpolation of the gains of successive controllers has been considered in the following. Each controller $K(i_0, j_0, k_0)$ has to satisfy the design objectives at $p(i_0, j_0, k_0)$ and in its direct neighborhood in a common domain with other neighboring controllers. In this work the direct neighbor of $K(i_0, j_0, k_0)$ considered is as far as the $p(i,j,k)\quad \forall(i,j,k) \in \{i_0 - 1, i_0 + 1\} \times \{j_0 - 1, j_0 + 1\} \times \{k_0 - 1, k_0 + 1\}$. Finally, any linear combination of these controllers in a direct neighborhood satisfies the performance. Fig. 3 is an attempt to visualize such polytopes for low dimensions.

The linear interpolation between these controllers carries over the performance properties of the closed-loop system to the whole flight envelope.

4. Multi-objective evolutionary algorithm

4.1. Overview to multi-objective approach

Evolutionary Algorithms are optimization 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 (Deb, 2001). Their use in optimization of control systems is growing (Fleming and Purshouse, 2002). A *population* of M independent individuals is maintained by the algorithm, each individual representing a potential solution to the problem. Each individual has one *chromosome*. This is the genetic description of the solution and may be broken into n sections called *genes*. Each gene represents a single parameter in the problem, therefore a problem that has eight unknowns for example, would require a chromosome with eight genes to describe it.

The three simple operations found in nature, natural selection, mating and mutation are used to generate new chromosomes and therefore new potential solutions. In this paper, an evolutionary strategy was used where new chromosomes were generated by a combination of mating (otherwise known as *crossover*) and applying Gaussian noise to each gene in each chromosome, with a standard deviation that evolved along with each gene. Each chromosome is evaluated at every generation using an *objective function* that is able to distinguish good solutions from bad ones and to score their performance. With each new generation, some of the old individuals die to make room for the new, improved offspring. Despite being very simple to code, requiring no directional or derivative information from the objective function and being capable of handling large numbers of parameters simultaneously, evolutionary algorithms can achieve excellent results.

4.2. 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 4.3) and then sorted (Section 4.4). 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 (10), and for the remaining 30% of the time, a copy of chromosome a is made. In (10), 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.

$$c_k = a_k + (b_k - a_k)(1.5U - 0.25). \tag{10}$$

The evolutionary strategy updates the standard deviation of the mutation and the value of each gene for every gene in each new chromosome, using (11). In (11), $\sigma'_k(x)$ 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.

$$\begin{aligned}
&\sigma'_k(x) = \sigma_k(x)\exp(\tau_0 N(0,1) + \tau_1 N_k(0,1)),\\
&\omega'_k(x) = \omega_k(x) + \sigma'_k(x) N_k(0,1),\\
&\tau_0 = \frac{1}{\sqrt{2\sqrt{n}}},\\
&\tau_1 = \frac{1}{\sqrt{2n}}.
\end{aligned} \tag{11}$$

4.3. Chromosome structure and objectives

4.3.1. 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 Fig. 4. Other membership function shapes could be used that will give a smoother output map, such as Gaussian, but it is much more difficult to maintain the sum of the functions at unity, and also guarantee that there will be no gaps. It is also likely that more membership functions will be required, unless the spread of each side of the Gaussian is controlled independently.

For the two inputs, the input 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. For example, for four member functions on an input, three genes are required to describe the relative positions of the peaks of the member functions as shown in Fig. 4. This process gives a total of four genes to represent the membership functions for three member functions per input, six genes for four member functions per input, and eight for five member functions. 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

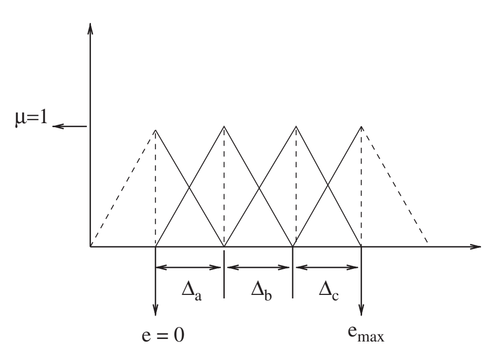

Fig. 4. Membership function structure.

outputs. Therefore with say four input member functions on each input, there are 16 possible rules, at 16 possible combinations of Mach and incidence. 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.

4.3.2. 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 summarize 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.

4.4. 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 optimization, 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 *maximization* strategy and have been adjusted to allow both of the objectives to be minimized.

5. Closed-loop performance requirements

The autopilot is required to track lateral acceleration commands a_{y_d} over the whole flight envelope. This should guide us to set a number of design objectives and use the evolutionary algorithm approach describe in previous section to obtain the fuzzy (PDC) controller for our missile lateral autopilot.

Settling time based cost function: For the maximum settling time case,

$$J_{1_i} = \frac{Re(np_i) - Re(pp_i)}{Re(np_i) - \sigma_{r_i}}, \tag{12}$$

where σ_{ri} is the location of the corresponding right boundary. Similarly for the minimum settling time case J_{2_i}.

Damping ratio based cost function: For the maximum damping ratio case,

$$J_{3_i} = \frac{\theta_{np_i} - \theta_{pp_i}}{\theta_{np_i} - \theta_{\zeta_{\max}}}, \tag{13}$$

where α_{np_i} and α_{pp_i} are the angles between the negative real axis and np_i and pp_i, respectively. $\alpha_{\zeta_{\max}} = \cos^{-1}\zeta_{\max}$ is the lower bound required for α_{pi}. Similarly for the minimum damping ratio case J_{4_i}.

Composite cost function:

$$J_i = \max\{J_{1_i}, J_{2_i}, J_{3_i}, J_{4_i}\}. \tag{14}$$

To find a best solution that results a desired controller (i.e $J_i \leqslant 1$), a global minima search method, otherwise the evolutionary algorithm approach described in previous section will be used.

5.1. Pole colouring

The controller synthesis can initially be done by assigning the dominant poles, pp_i to the specific locations which satisfy certain settling time and damping ratio constraints. These poles are paired with nominal poles, np_i, which are attached to a D-stability region (Söylemez, 1999).

5.1.1. Experimental results

Figs. 5–8 show a set of typical non-dominated surfaces after 100 generations for each of the membership function configurations. Neither of the three objectives can be minimized simultaneously, so the

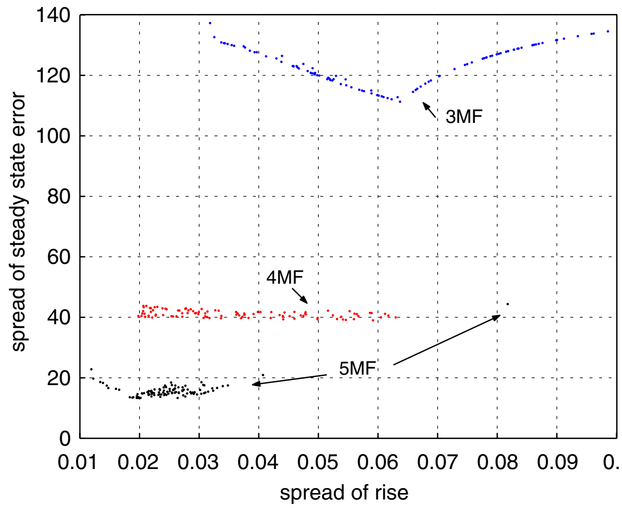

Fig. 5. Non-dominated set for 5 membership functions (MF) each, 4 MF each and 3 MF each (spread of steady-state error against spread of rise).

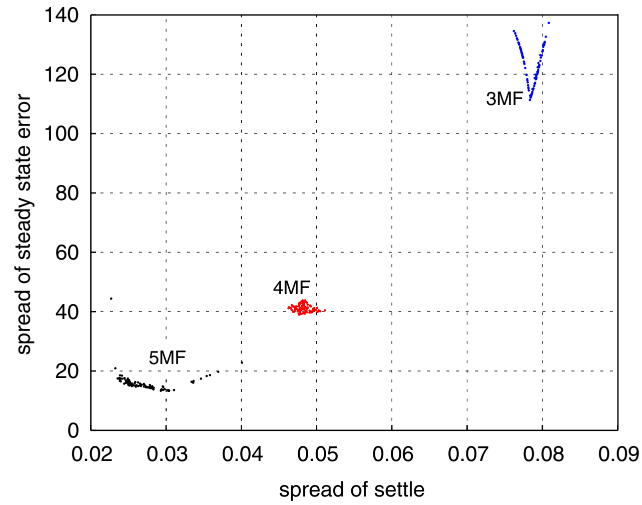

Fig. 7. Non-dominated set for 5 membership functions (MF) each, 4 MF each and 3 MF each (spread of steady-state error against spread of settle).

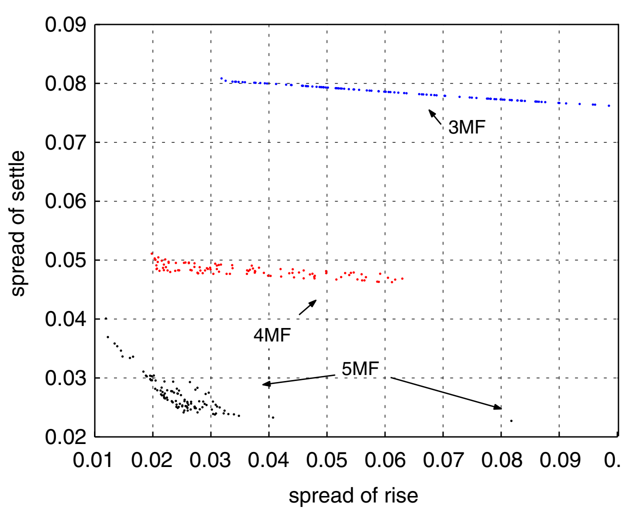

Fig. 6. Non-dominated set for 5 membership functions (MF) each, 4 MF each and 3 MF each (spread of settle against spread of rise).

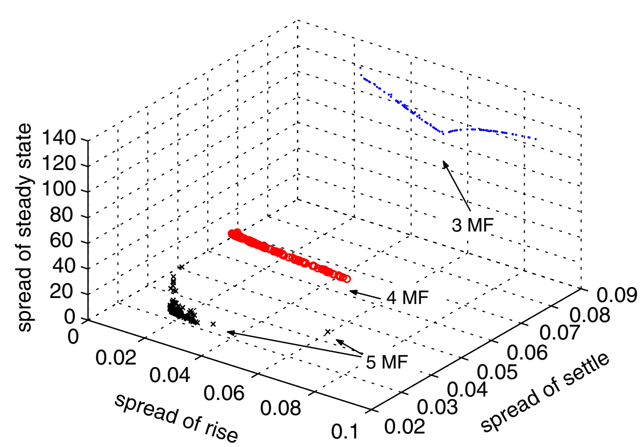

Fig. 8. Non-dominated set for 5 membership functions (MF) each, 4 MF each and 3 MF each, 3D plot of all three objectives.

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.

Figs. 9 and 10 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 with the lowest spread of rise time. Figs. 11–14 show the corresponding plots for the trials with 4 and 5 membership functions per input.

It is apparent from Figs. 9, 11 and 13 that the Mach input is dominant when shaping the control surfaces, also the spread of rise times plays a significant role in the evolutionary process. The lines on the plots progress relatively smoothly with respect to the objective surface, whereas Figs. 10, 12 and 14 have little correlation with the progression of the objectives. This effect suggests that fewer membership functions are required for the incidence input.

Figs. 15 and 16 show the surfaces generated by the fuzzy inference systems for the two control gains for the solution with 5 membership functions in both inputs that minimizes the error in the pole locations.

Simulation response for the normalized acceleration is shown in Fig. 17. It can been easily observed that the controller meets all the objectives and the steady state error in within the 10% bounds.

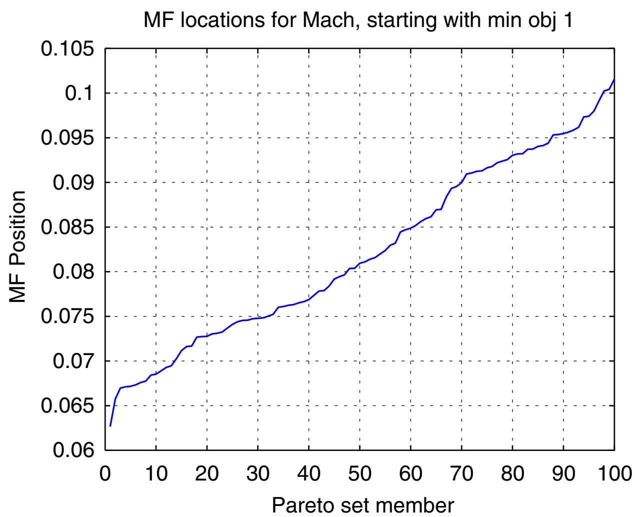

Fig. 9. Membership function locations for Mach and 3 member functions.

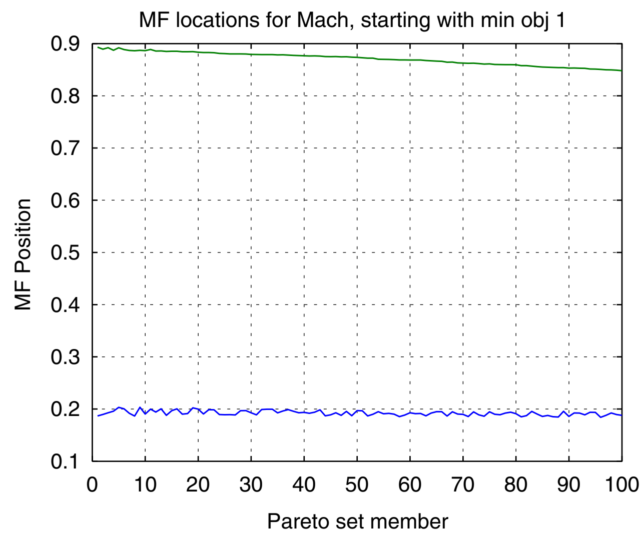

Fig. 11. Membership function locations for Mach and 4 member functions.

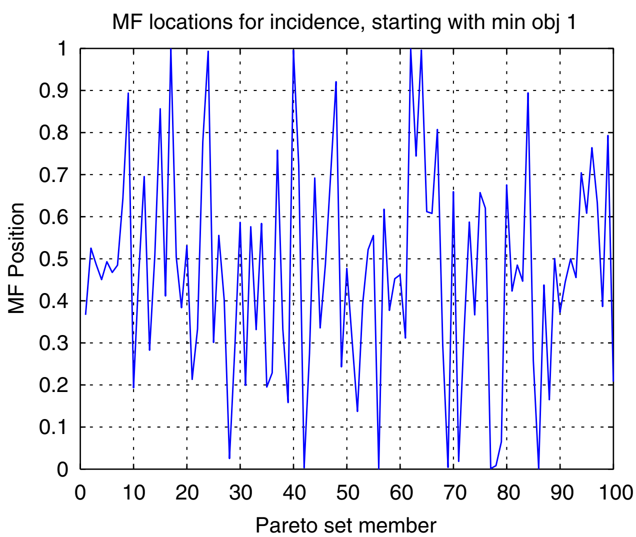

Fig. 10. Membership function locations for Incidence and 3 member functions.

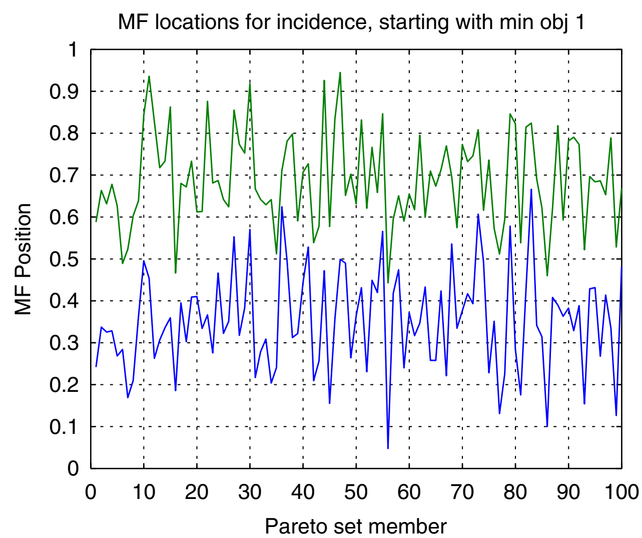

Fig. 12. Membership function locations for Incidence and 4 member functions.

5.2. *Tracking control design*

The controller described in previous section would result only to a desired transient of all local models by placing the poles of all the local systems within a specified area. Since however the objective of a missile autopilot, apart of good transient is minimum steady state error we have to include that in the other design specifications. Zero steady state error can be achieved with an integral term in the forward path.

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

$$x_i = \int_{t_0}^{t} \mathbf{e}\,\mathrm{d}t = \int_{t_0}^{t} (y - r)\,\mathrm{d}t. \tag{15}$$

Therefore,

$$\dot{x}_i = [y_d - r]. \tag{16}$$

The state space now is described by

$$\begin{bmatrix} \dot{x} \\ - \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A(p) & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ x_i \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \zeta + \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix} r. \tag{17}$$

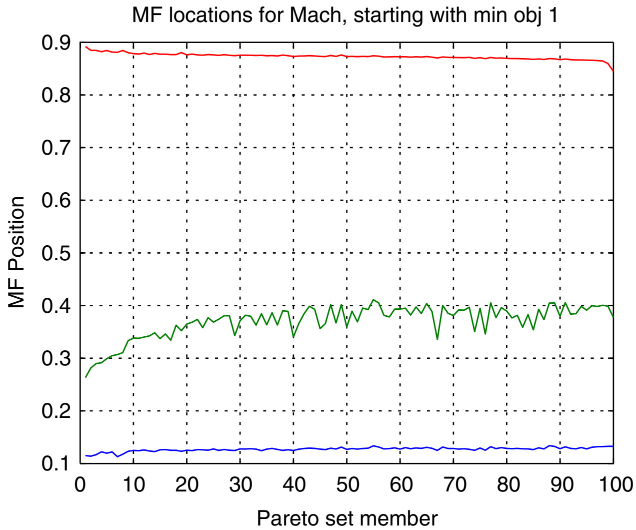

Fig. 13. Membership function locations for Mach and 5 member functions.

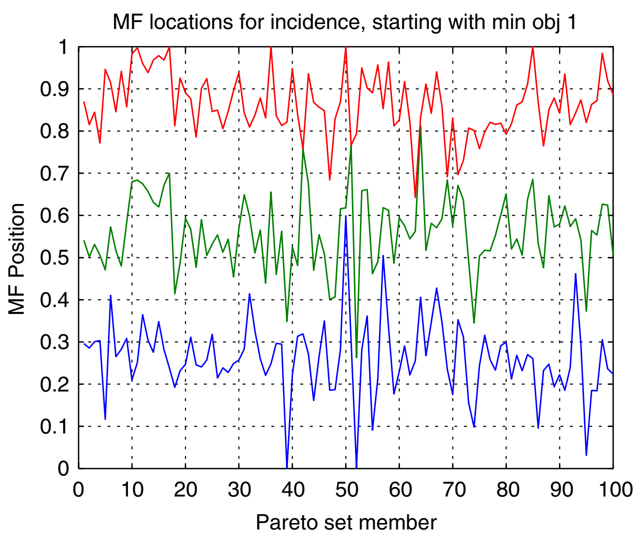

Fig. 14. Membership function locations for Incidence and 5 member functions.

The compensated system therefore becomes:

$$\begin{bmatrix} \dot{x} \\ - \\ \dot{x}_i \end{bmatrix} = \begin{bmatrix} A(p) - BK(p) & -BK_i \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ - \\ x_i \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ - \\ \mathbf{I} \end{bmatrix} r. \quad (18)$$

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.

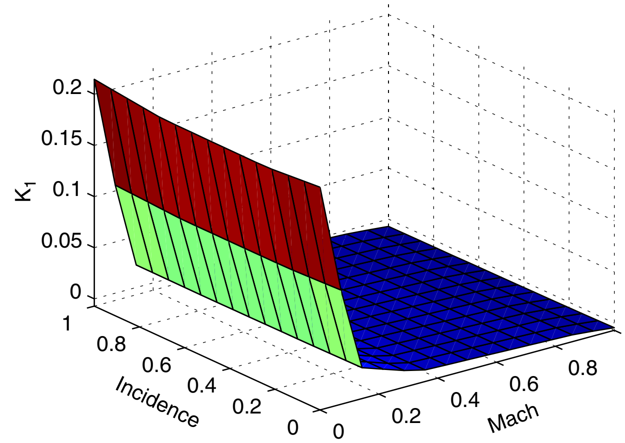

Fig. 15. Example gain surface for K_1 and 5 MF per input.

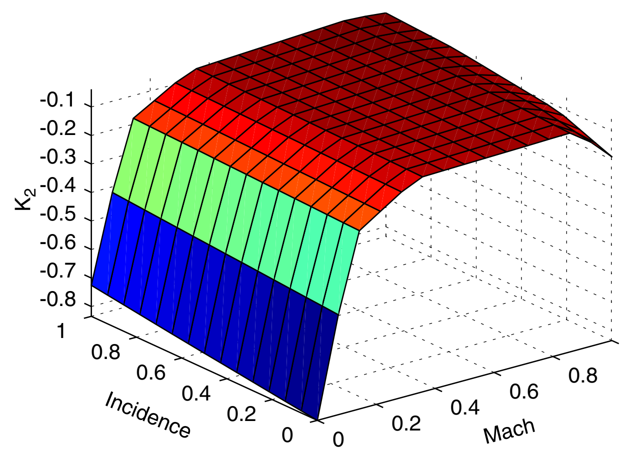

Fig. 16. Example gain surface for K_2 and 5 MF per input.

This adds an extra requirement to the selection of gains for pole placement.

5.2.1. Experimental results

Fig. 18 shows a set of typical non-dominated surfaces after 100 generations for each of the membership function configurations. Both of the objectives cannot be minimized simultaneously, so the Pareto front forms curves.

Figs. 19 and 20 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. Figs. 21, 22, 24 and 19 show the corresponding plots for the trials with 4 and 5 membership functions per input.

It is clear from Figs. 19, 21 and 23 that the Mach input is dominant when shaping the control surfaces.

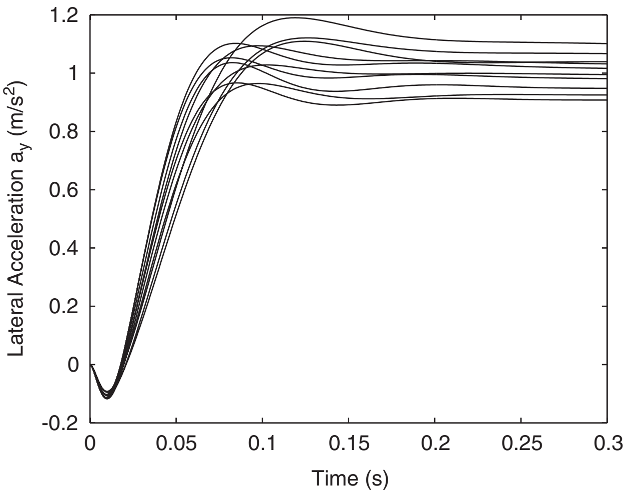

Fig. 17. Normalized acceleration using the pole colouring approach.

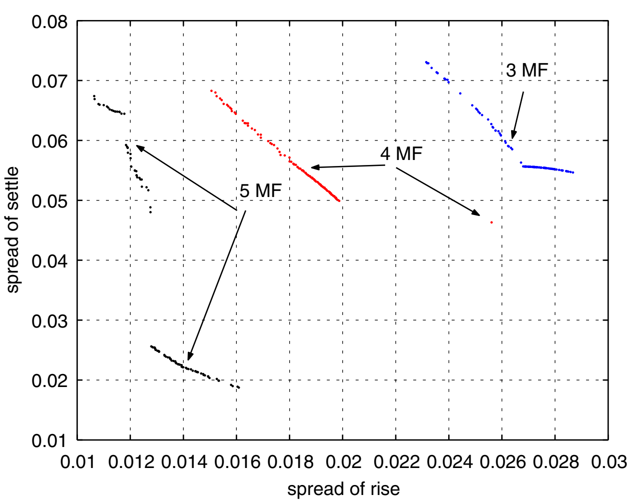

Fig. 18. Pareto set for 5 Membership Functions (MF) each, 4 MF each and 3 MF each.

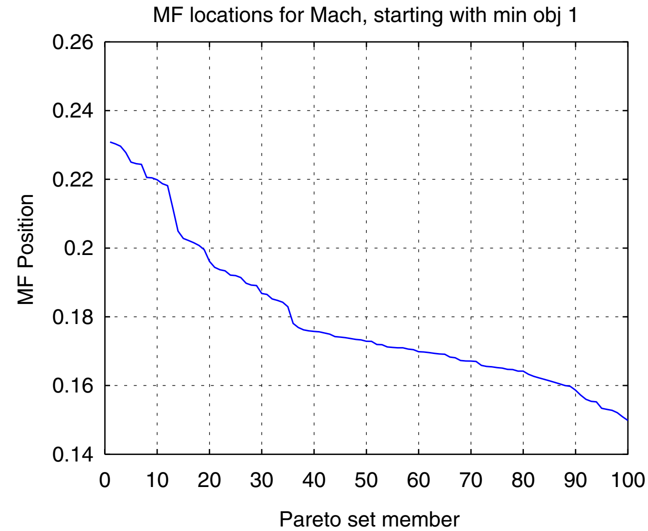

Fig. 19. Membership function locations for Mach and 3 member functions.

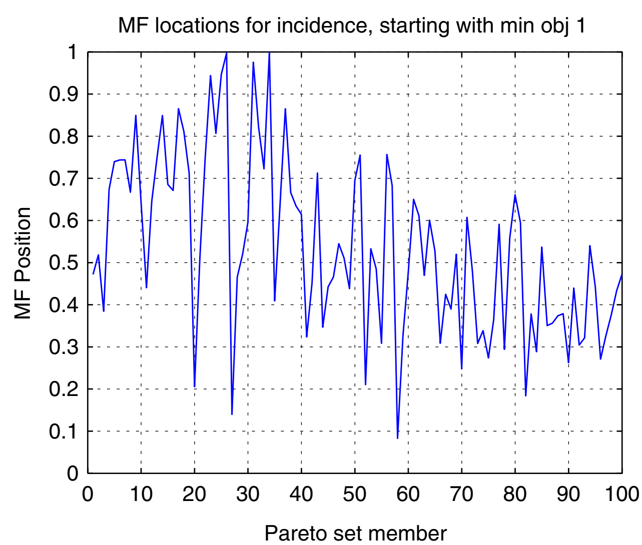

Fig. 20. Membership function locations for Incidence and 3 member functions.

The lines on the plots progress smoothly with respect to the objective surface, whereas Figs. 20, 22 and 24 have little correlation with the progression of the objectives. This effect suggests that fewer membership functions are required for the incidence input.

Figs. 25–27 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 minimizes the error in the spread of rise times.

Fig. 28 shows the controller gains obtained by fuzzy gain scheduling using 5 membership functions and the controller gains obtained using LPV shelf-scheduling controller using 100 local models. Since the values of the controller gains obtained from the two different gain scheduling design approaches are very close we can conclude that our 5 MF fuzzy gains scheduling can achieve similar performance as the one using LPV shelf-scheduling controller using 100 local models. That is achieved without the use of much higher gain values. Therefore, one could claim that the use of evolutionary algorithms for fuzzy gain scheduling design allows us to reduce the number of local models while achieving similar properties as LPV shelf scheduling controller using LMI type approaches. This approach removes the usual ad-hoc choice of the necessary number of local models that have to be selected in the gain scheduling.

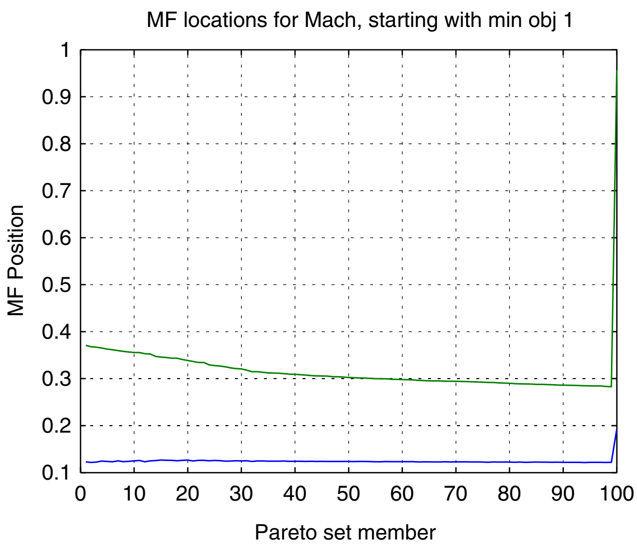

Fig. 21. Membership function locations for Mach and 4 member functions.

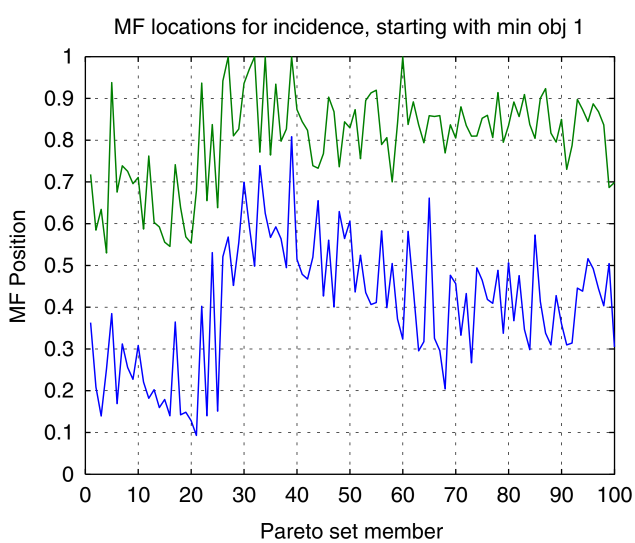

Fig. 22. Membership function locations for Incidence and 4 member functions.

Simulation response for the normalized acceleration shown in Fig. 29. It can be observed that the controller meets all the objectives including zero steady state error.

6. 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 to optimism the fuzzy inference system removes the requirement of expert knowledge to design the fuzzy landscape as the multi-objective algorithm is capable of

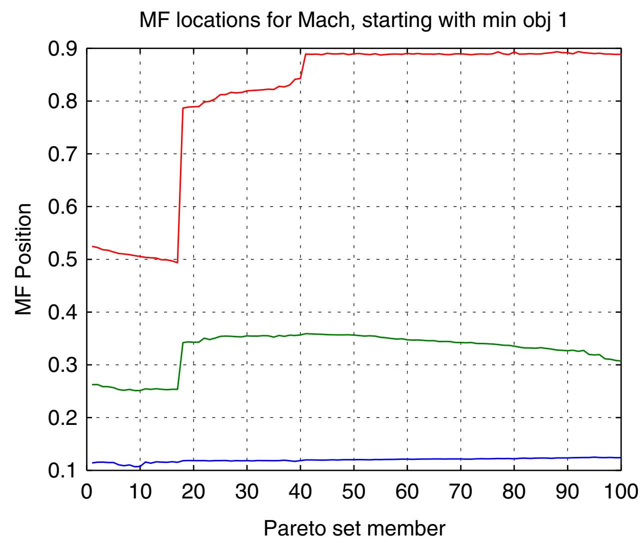

Fig. 23. Membership function locations for Mach and 5 member functions.

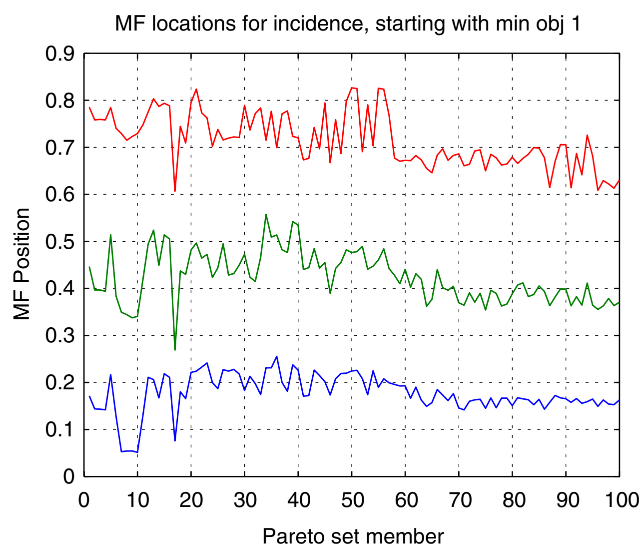

Fig. 24. Membership function locations for Incidence and 5 member functions.

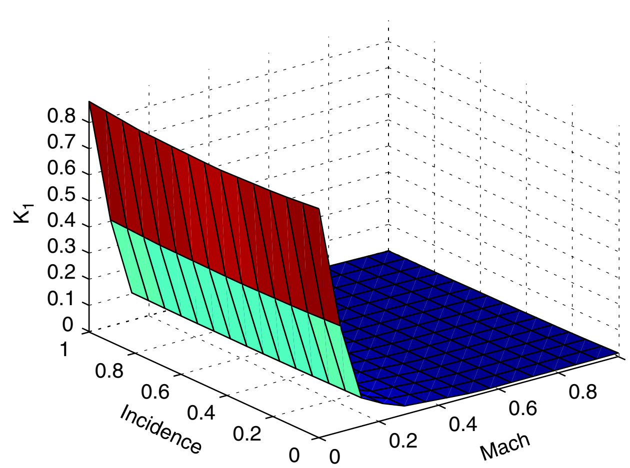

Fig. 25. Example gain surface for K_1 and 5 MF per input.

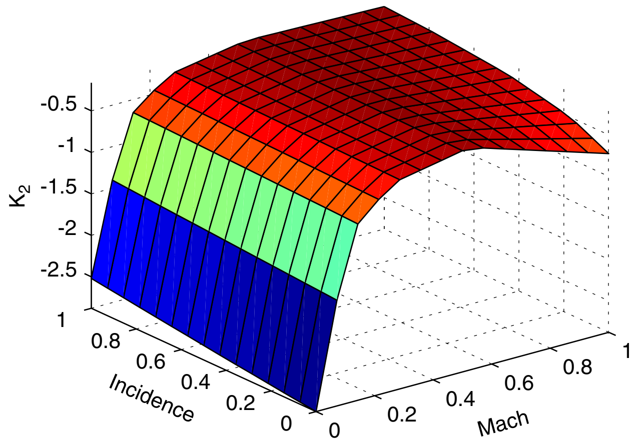

Fig. 26. Example gain surface for K_2 and 5 MF per input.

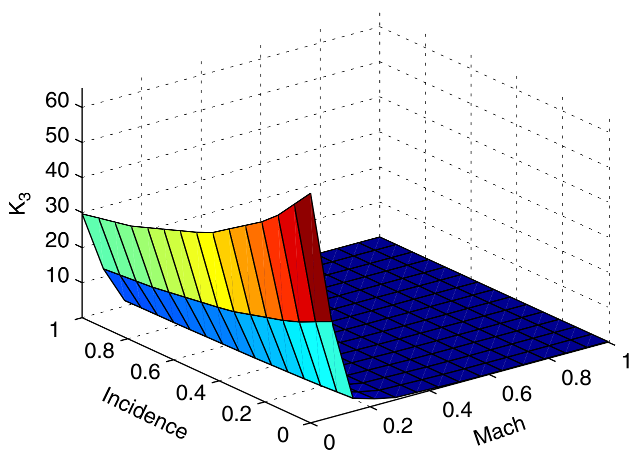

Fig. 27. Example gain surface for K_3 and 5 MF per input.

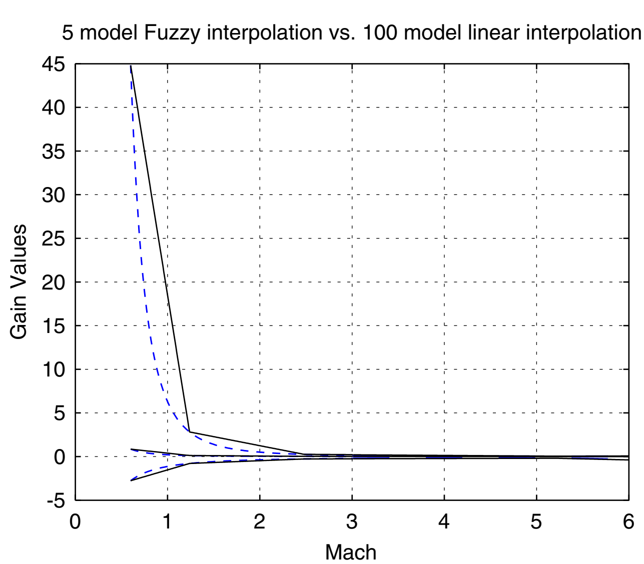

Fig. 28. 5 model fuzzy interpolation vs. 100 model linear interpolation.

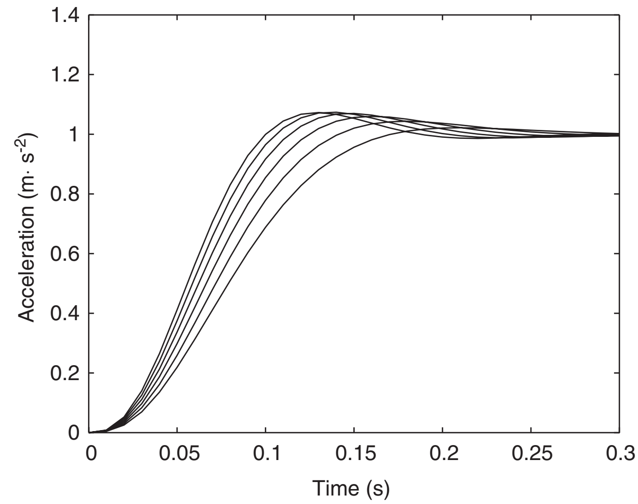

Fig. 29. Normalized acceleration using the pole colouring approach.

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.

References

Apkarian, P., Biannic, J. M., & Gahinet, P. (1995). Self-scheduled H_∞ control of missile via linear matrix inequalities. *Journal of Guidance Control and Dynamics*, *18*(3), 532–538.

Apkarian, P., & Gahinet, P. (1995). A convex characterization of the gain-scheduled H_∞ controllers. *IEEE Transactions on Automatic Control*, *40*(10), 853–864.

Apkarian, P., Gahinet, P., & Becker, G. (1995). Self-scheduled H_∞ control of linear parameter varying systems: A design example. *Automatica*, *31*(9), 1251–1261.

Balas, G. J., Doyle, J. C., Glover, K., Packard, A., & Smith, R. (1994). *The μ-analysis and synthesis toolbox*. Cambridge, MA: The MathWorks Inc.

Barmish, B. R. (1995). Necessary and sufficient conditions for quadratic stability of an uncertain system. *Journal of Optimization Theory and Applications*, *46*, 299–408.

Baumann, W. T., & Rugh, W. J. (1986). Feedback control of nonlinear systems by extended linearization. *IEEE Transactions on Automatic Control*, *31*(1), 40–46.

Beaven, R. W., Wright, M. T., Garvey, S. D., & Friswell, M. I. (1995). The application of setpoint gain scheduling to high-speed independent drives. *Control Engineering Practice*, *3*(11), 1581–1585.

Becker, G. (1993). *Quadratic stability and performance of linear parametrically dependent systems*. Ph.D. Thesis, Department of Mechanical Engineering, University of California, Berkeley.

Becker, G. (1996). Additional results on parameter-dependent controllers for LPV systems. In *13th IFAC world congress* (vol. G, pp. 351–356).

Becker, G., & Packard, A. (1994). Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems and Control Letters*, *23*(3), 205–215.

Boyd, S., El Ghaoui, L., Feron, E., & Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory*. Philadelphia, PA: SIAM.

Deb, K. (2001). *Multi-objective optimization using evolutionary algorithms*. New York: Wiley.

Deb, K., Agrawal, S., Pratap, A., Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *Parallel problem solving from nature VI conference* (pp. 849–858), Paris, France, September.

Deb, K., Mohan, M., Mishra, M. (2003). A fast multi-objective evolutionary algorithm for finding well-spread Pareto optimal solutions. Technical Report 2003002, KanGAL, http://www.iitk.ac.in/kangal/papers/k2003002.pdf, February.

Doyle, J. C., Glover, K., Khargonekar, P. P., & Francis, B. (1989). State-space solutions to standard H_2 and H_∞ control problems. *IEEE Transactions on Automatic Control*, *34*(8), 831–847.

Doyle, J., Packard, J., Zhou, K. (1991). Review of LFTs, LMIs and μ. In *30th IEEE conference on decision and control* (vol. 2, pp. 1227–1232).

Feron, E., Apkarian, P., & Gahinet, P. (1996). Analysis and synthesis of robust control systems via parameter-dependent Lyapunov functions. *IEEE Transactions on Automatic Control*, *41*(7), 1041–1046.

Fleming, P. J., Chipperfield, A. J., & Bica, B. (2002). Fuzzy scheduling control of a gas turbine aero-engine: A multi-objective genetic algorithm. *IEEE Transactions Industrial Electronics*, *49*(3), 526–548.

Fleming, P. J., & Purshouse, R. C. (2002). Evolutionary algorithms in control systems engineering: a survey. *Control Engineering Practice*, *10*(11), 1223–1241.

Gahinet, P., & Apkarian, P. (1994). A linear matrix inequality approach to H_∞ control. *International Journal of Robust and Nonlinear Control*, *4*(4), 421–448.

Gahinet, P., Apkarian, P., & Chilali, M. (1996). Affine parameter-dependent Lyapunov functions for real parametric uncertainty. *IEEE Transactions on Automatic Control*, *41*(3), 436–442.

Gahinet, P., Nemirovski, A., Laub, A. J., & Chilali, M. (1995). *LMI control toolbox*. Cambridge, MA: The Mathworks Inc.

Haddad, W., & Bernstein, D. (1993). Explicit construction of quadratic Lyapunov functions for the small gain, positivity, circle, and Popov theorems and their application to robust stability. *International Journal of Robust and Nonlinear Control*, *3*(4), 313–339.

Haddad, W. M., & Bernstein, D. (1995). Parameter-dependent Lyapunov functions and the Popov criterion in robust analysis and synthesis. *IEEE Transactions on Automatic Control*, *40*(3), 536–543.

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.

Hwang, C. L., & Lin, C. L. (2003). A dynamically fuzzy gain—scheduled design for missile autopilot. *Aeronautical Journal*, *107*, 599–606.

Hyde, R. A., & Glover, K. (1993a). The application of scheduled H_∞ controllers to a VSTOL aircraft. *IEEE Transactions on Automatic Control*, *38*(7), 1021–1039.

Hyde, R.A., Glover, K. (1993b). Taking H_∞ control into flight. In *32nd IEEE Conference on Decision and Control* (vol. 2, pp. 1458–1463).

Hyde, R. A., & Glover, K. (1995). VSTOL first flight on an h_∞ control law. *Computing and Control Engineering Journal*, *6*(1), 11–16.

Iwasaki, T., & Skelton, R. E. (1994). All controllers for the general H_∞ control problem: LMI existence conditions and state space formulas. *Automatica*, *30*, 1307–1317.

Kamen, E. W., & Khargonekar, P. P. (1984). On the control of linear systems whose coefficients are functions of parameters. *IEEE Transactions on Automatic Control*, *29*(1), 25–33.

Khargonekar, P. P., Petersen, I. R., & Zhou, K. (1990). Robust stabilization of uncertain linear systems: Quadratic stability and H_∞ control theory. *IEEE Transactions on Automatic Control*, *35*(3), 356–361.

Leith, D. J., & Leithead, W. E. (2000). Survey of gain-scheduling analysis and design. *International Journal of Control*, *73*(11), 1001–1025.

Lin, S. T., & Lee, C. M. (1996). Multivariable control of the J-85 turbojet engine for full flight envelope operation. *Journal of Guidance, Control and Dynamics*, *19*(4), 913–920.

Molina-Cristobal, A., Griffin, I. A., Fleming, P. J., Owens, D. H. (2004). Multiobjective control: A comparative study of a multi-objective genetic algorithm and linear matrix inequalities. In *Conference of the North American fuzzy information processing society* (pp. 417–422).

Nam, Y., & Hong, S. K. (2003). Affine parameter-dependent ts fuzzy controller with application to aircraft roll control. *Journal of Intelligent and Fuzzy Systems*, *13*(1), 25–33.

Nemirovskii, A., Gahinet, P. (1994). The projective method for solving linear matrix inequalities. In *1994 American Control Conference* (vol. 1, pp. 840–844).

Nichols, R. A., Reichert, R. T., & Rugh, W. J. (1993). Gain scheduling for H_∞ controllers: A flight control example. *IEEE Transactions on Control Systems Technology*, *1*, 69–79.

Packard, A. (1994). Gain scheduling via linear fractional transformations. *Systems and Control Letters*, *22*(2), 79–92.

Packard, A., & Doyle, J. C. (1993). The complex structured singular value. *Automatica*, *29*(1), 71–109.

Packard, A., Doyle, J. C., & Balas, G. (1993). Linear, multivariable robust control with a perspective. *Journal of Dynamic Systems, Measurement and Control*, *115*(2B), 426–438.

Packard, A., Zhou, K., Pandey, P., Becker, G. (1991). A collection of robust control problems leading to LMIs. In *34th IEEE conference on decision and control* (vol. 2, pp. 1245–1250).

Ravi, R., Nagpal, K. M., & Khargonekar, P. P. (1991). H_∞ control of linear time-varying systems: A state space approach. *SIAM Journal on Control and Optimization*, *29*, 1394–1413.

Rotea, M. A., Corless, M., Da, D., & Petersen, I. R. (1993). Systems with structured uncertainty: Relations between quadratic and robust stability. *IEEE Transactions on Automatic Control*, *38*(5), 799–803.

Rugh, W. J. (1993). Analytical framework for gain scheduling. *IEEE Control Systems Magazine*, *11*, 799–803.

Rugh, W. J., & Shamma, J. S. (2000). Survey paper: Research on gain scheduling. *Automatica*, *36*, 1401–1425.

Scherer, C. (1992). H_∞ control state by state feedback for plants with zeros on the imaginary axis. *SIAM Journal on Control and Optimization*, *30*(1), 123–142.

Shahruz, S. M., & Behtash, S. (1992). Design of controllers for linear parameter varying systems by the gain scheduling technique. *Journal of Mathematical Analysis and Applications*, *168*(1), 195–217.

Shamma, J. S., & Athans, M. (1990). Analysis of gain scheduled control for nonlinear plants. *IEEE Transactions on Automatic Control*, *35*(8), 898–907.

Shamma, J. S., & Athans, M. (1991). Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*, *27*(3), 559–564.

Shamma, J. S., & Athans, M. (1992). Gain scheduling: Potential hazards and possible remedies. *IEEE Control Systems Magazine*, *13*, 101–107.

Shamma, J. S., & Cloutier, J. R. (1993). Gain-scheduled missile autopilot design using LPV transformations. *Journal of Guidance, Control and Dynamics*, *16*(2), 256–263.

Söylemez, M.T. (1999). *Pole assignment for uncertain systems*. UMIST Control Systems Centre Series. Hertfordshire: Research Studies Press.

Spillman, M., Blue, P., Banda, S., Lee, L. (1996). A robust gain scheduling example using linear parameter-varying feedback. In *13th IFAC World congress* (vol. H, pp. 221–226).

Stoorvogel, A. A. (1992). *The H_∞ Control Problem: A State Space Approach*. Englewood Cliffs, NJ: Prentice-Hall.

Tran, M.N., Hrovat, D. (1993). Application of gain scheduling to design of active suspensions. In *32nd IEEE conference on decision and control* (vol. 2, pp. 1030–1035).

Tsourdos, A., Blumel, A., White, B. A. (1998). Trajectory control of a nonlinear homing missile. In *14th IFAC symposium on automatic control in aerospace* (pp. 118–123).

Vidyasagar, M. (1993). *Nonlinear system analysis* (2nd ed.). Englewood Cliffs, NJ: Prentice Hall International Editions.

Wang, J., & Rugh, W. J. (1987). Parameterized linear systems and linearization families for nonlinear systems. *IEEE Transactions on Circuits and Systems*, *34*(6), 650–657.

White, B. A., Blumel, A. L., Hughes, E. J. (2000). Fuzzy autopilot design using a multiobjective evolutionary algorithm. In *IEEE congress on evolutionary computation* (pp. 54–61).

White, B. A., Blumel, A. L., Hughes, E. J. (2001a). Multiobjective optimization of fuzzy logic scheduled controllers for missile autopilot design. In *Conference of the North American Fuzzy Information Processing Society* (pp. 1758–1763).

White, B. A., Blumel, A. L., Hughes, E. J. (2001b). Multi-objective evolutionary design of fuzzy autopilot controller. In *International Conference Evolutionary Multi-Criterion Optimization* (pp. 668–680).

Wise, K. A. (1992). Comparison of 6 robustness tests evaluating missile autopilot robustness to uncertain aerodynamics. *Journal of Guidance, Control and Dynamics*, *15*(4), 861–870.

Wood, G. (1995). *Parameter-dependent control of mechanical systems*. Ph.D. Thesis, St. John's College, Cambridge University.

Wu, F. (1995). *Control of linear parameter varying systems*. Ph.D. Thesis, Department of Mechanical Engineering, University of California, Berkeley.

Wu, F., Yang, X., Packard, A., & Becker, G. (1996). Induced L_2-norm control for LPV system with bounded parameter variation rates. *International Journal of Robust and Nonlinear Control*, *6*(9–10), 983–998.

Wu, Z. Y., Fujimori, A., & Tsunetomo, H. (1999). Gain-scheduled control using fuzzy logic and its application to flight control. *Journal of Guidance Control and Dynamics*, *22*(1), 175–178.

Zalzala, A.M.S., Flemming, P.J. (eds.). (1997). *Genetic algorithms in engineering systems*. The Institution of Electrical Engineers.

Zitzler, E., Laumanns, M., Thiele, L. (2001). *SPEA2: Improving the strength Pareto evolutionary algorithm*. Technical Report 103, Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, Switzerland, May.