Multiobjective Design of a Fuzzy Controller for a Nonlinear Missile Autopilot

Evan J. Hughes, Antonios Tsourdos and Brian A. White
Dept. Aerospace, Power and Sensors, Cranfield University,
Royal Military College of Science, Shrivenham,
Swindon, Wiltshire, England. SN6 8LA
e.j.hughes@rmcs.cranfield.ac.uk
a.tsourdos@rmcs.cranfield.ac.uk
b.a.white@rmcs.cranfield.ac.uk

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 optimise the performance characteristics of the plant.

1 Introduction

The performance of an air vehicle is highly dependent on the capabilities of the guidance, navigation and control systems. To achieve improved performance in such aerospace systems, it is important that more sophisticated control systems be developed and implemented. In particular, as the performance envelope is expanded, the control schemes must become adaptive and quasi-linear, to provide performance over a greater range, in the face of changing operating conditions.

The tracking performance of a missile is also dependent on the location within the flight envelope and varies with factors such as Mach number and incidence. Several approaches, including adaptive control [1], [2], nonlinear control [3], and gain scheduling [4] have been used to alleviate these tracking problems.

One of the most popular methods for applying linear time-invariant (LTI) control theory to time-varying and/or quasi-linear systems is gain scheduling [5]. This strategy involves obtaining Taylor linearised 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.

Despite past success of gain scheduling in practice, until recently little has been known about it theoretically as a time-varying and/or quasi-linear 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 multi-variable controllers can be a rather tedious process.

An early theoretical investigation into the performance of parameter-varying systems can be found in [6]. During the 1980's, Rugh and his colleagues developed an analytical framework for gain scheduling using extended linearisation [5]. Also, Shamma and Athans [7] 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 capture the nonlinearities of the plant". Shahruz and Behtash [8] suggested using LPV systems for synthesising gain-scheduled controllers.

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 [9], [10], and LTV systems [11]. 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.

In this paper fuzzy pole-placement control design technique is applied to the autopilot design for the missile. The missile motion is modelled to be quasi-linear with unknown parameters. Based on the quasi-linear model, we adopt for design procedure the fuzzy pole-placement method. 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 to apply the motion control design for missiles, which are highly quasi-linear in dynamics. The optimisation of the fuzzy system is performed using a multiobjective evolutionary algorithm [12].

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 multiobjective evolutionary algorithm is detailed in section 4. Section 5 shows typical results from the optimisation process and section 6 concludes.

2 Missile model

Missile autopilots are usually designed using linear models of nonlinear equations of motion and aerodynamic forces and moments [13], [14]. The objective of this paper is robust design of a sideslip (yaw) velocity autopilot for a nonlinear 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 [15].

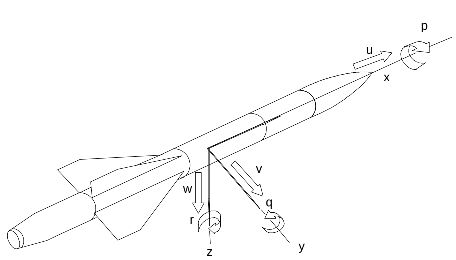

Figure 1: Airframe axes.

The starting point for mathematical description of the missile is the following nonlinear model [16], [15] of the horizontal motion (on the xy plane in Figure 1):

Table 1: Coefficients in nonlinear model (1).

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

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

where the variables are defined in Figure 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, $m = 125$ 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.

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 Figure 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 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.

3 Design of Lateral Missile Autopilot

3.1 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 $K(p)$ is determined recursively but its structure and in particular the values of the longitudinal and lateral controllers $\mathbf{K_1}(\mathbf{p})$ and $\mathbf{K_2}(\mathbf{p})$ are obtained using the pole-placement technique.

Substituting the control law in the state equation yields:

$$\dot{x} = A^* x \tag{3}$$

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 $|\lambda I - A^*|$

The coefficients $A_{i,j}$ are parameter dependent. 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.

3.2 Fuzzy Inference System

A Takagi-Sugeno (T-S) fuzzy controller [17] is used to determine the natural frequency and damping ratios 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 two outputs, natural frequency and damping.

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 $\delta\omega_{n_i} = \omega_{n_i}$,
$i = l, 2, \ldots, r$,

Where M_j and I_k are individual membership functions of the two inputs and ω_{n_i} is the required natural frequency for the rule.

The T-S fuzzy model infers $\omega_{n_i}(t)$ 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.

$$\omega_{n_i} = \frac{\sum_{i=1}^{r} \nu_i [\delta\omega_{n_i}]}{\sum_{i=1}^{r} \nu_i} \tag{4}$$

The second output for the damping ratio is calculated in a similar manner.

3.3 Tracking control design

This controller would result only to a desired transient of all local models by placing the poles of all the local systems within a specified area. However since our aim is good tracking for the missile we should include to the design specification except peak overshoot and settling time, zero steady-state error. This 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:

$$x_i = \int_{t_0}^{t} \mathbf{e} dt = \int_{t_0}^{t} (y - r)\, dt \tag{5}$$

Therefore,

$$\dot{x}_i = [y_d - r] \tag{6}$$

The state space now is described by:

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

The compensated system therefore becomes:

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ - \\ \dot{x}_i \end{bmatrix} &= \begin{bmatrix} A(p) - BK(p) & -BK_i \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ - \\ x_i \end{bmatrix} \\ &+ \begin{bmatrix} \mathbf{0} \\ - \\ \mathbf{I} \end{bmatrix} r \end{aligned} \tag{8}$$

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

4 Evolutionary Algorithm

4.1 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 [12]. 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 (section4.4). 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 (9), and for the remaining 30% of the time, a copy of chromosome a is made. In (9), 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.

$$c_k = a_k + (b_k - a_k)(1.5U - 0.25) \quad (9)$$

The evolutionary strategy updates the standard deviation of the mutation and the value of each gene for every gene in each new chromosome, using (10). In (10), $\sigma'_k(x)$ 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.

$$\begin{aligned} \sigma'_k(x) &= \sigma_k(x)\exp(\tau_0 N(0,1) + \tau_1 N_k(0,1)) \\ \omega'_k(x) &= \omega_k(x) + \sigma'_k(x) N_k(0,1) \\ \tau_0 &= \frac{1}{\sqrt{2\sqrt{n}}} \\ \tau_1 &= \frac{1}{\sqrt{2n}} \end{aligned} \quad (10)$$

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. Four membership functions are 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

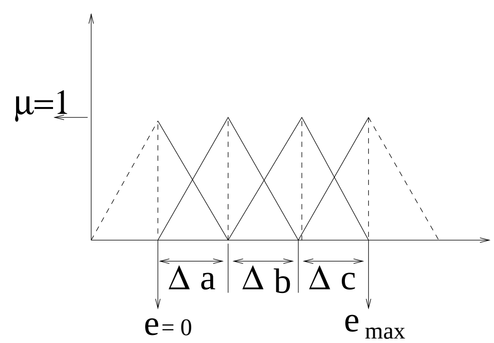

Figure 2: Membership function structure

For the two inputs, the input ranges are $e_0 = 0.6$ to $e_m = 6$ for the Mach number, and $e_0 = 0°$ to $e_m = 30°$ for the incidence. Three genes are used for each input to describe the relative positions of the peaks of the member functions as shown in figure 2. This process gives a total of 6 genes to represent the membership functions. Each of the 6 genes must lie in the range (0,1].

The output value for each the rule is simply a pair of constants, one for each of the two outputs. Therefore with four input member functions on each input, there are 16 possible rules, giving a total of 32 genes to represent the output functions. The 16 gene values for the natural frequency output must lie in the range [15 50] and the 16 gene values for the damping ratio output must lie in the range [0.6 0.99]. Thus the chromosome length is a total of 38 genes.

4.3.2 Objectives:

The performance is tested by generating the step response of the system for 100 uniformly spaced points in the natural frequency/ damping ratio domain. The rise time, overshoot and final error is recorded at each point. Two objectives are then generated that summarise the performance of the chromosome.

The first objective is a combination of the rise time and the error. The objective is to maximise the worst of: the longest rise time; and ten times the absolute error of the step response after 0.3 seconds.

The second objective is to minimise the worst overshoot of the response. In this paper, the two objectives are intended to give a flat response from the plant. By making the objectives dependent on the state of Mach and incidence, any required non-linear response pattern could be generated.

4.4 Non-dominated Ranking

With multiple objectives, a Pareto-optimal set of results [12] 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 [12] 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.

5 Experimental Results

Figure 3 shows a typical non-dominated surface after 100 generations. Both of the objectives cannot be minimised simultaneously, so the lower leading edge of the points indicates the set of best possible solutions (marked with crosses in circles).

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.

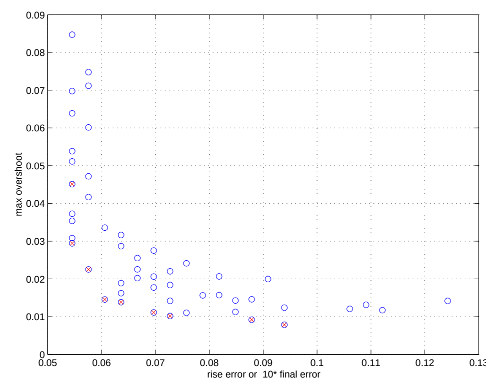

Figure 3: Non-dominated optimisation surface

Figures 4 and 5 show the surfaces generated by the fuzzy inference systems for both the natural frequency ω_n, and the damping ratio δ for the solution that minimises the error in the rise time and final value. As only 100 generations were used, the surfaces shown are unlikely to be part of the true Pareto set, however they are likely to be quite close. With more generations, the surfaces will become smoother as the membership functions are refined further.

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 optimise the fuzzy inference system removes the requirement of expert knowledge to design the fuzzy landscape as the multiobjective algorithm is capable of discovering a range of solutions with little designer intervention.

The multiobjective formulation allows many potential solu-

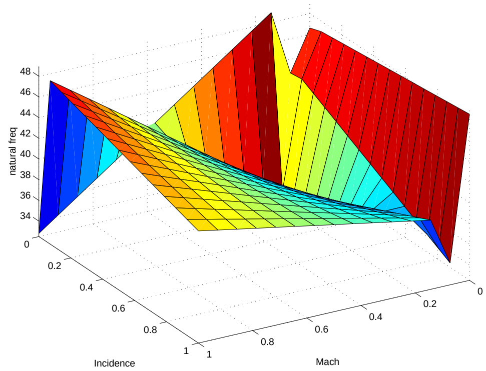

Figure 4: Control surface for ω_n

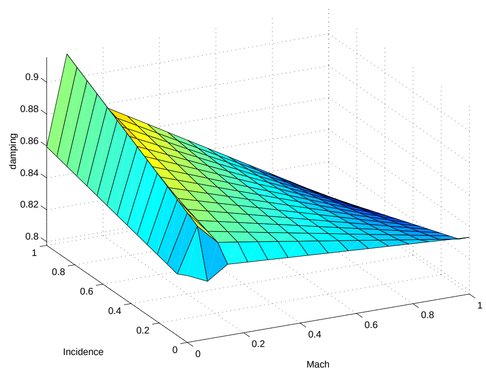

Figure 5: Control surface for δ

tions to be generated simultaneously. The designer can then choose a candidate solution whilst being informed of what other solutions to the problem may exist.

Acknowledgements

The authors would like to acknowledge the use of the Department of Aerospace, Power, and Sensors DEC Alpha Beowulf cluster for this research.

References

[1] E. Kamen, T. Bullock, and C. Song, "Adaptive control applied to missile autopilots," in *American Control Conference*, pp. 555–560, 1988.

[2] C. Lin and J. Cloutier, "High performance, adaptive, robust bank-to-turn missile autopilot," in *AIAA Guidance, Navigation, Control Conference*, pp. 123–137, 1991.

[3] A. Isidori, *Nonlinear Control Systems: An Introduction*. New York: Springer-Verlag, Second ed., 1989.

[4] D. A. Lawrence and W. J. Rugh, "Gain scheduling dynamic linear controllers for a linear plant," *Automatica*, vol. 31, no. 3, pp. 381–390, 1995.

[5] W. J. Rugh, "Analytical framework for gain scheduling," *IEEE Control Systems Magazine*, vol. 11, pp. 799–803, 1993.

[6] E. W. Kamen and P. P. Khargonekar, "On the control of linear systems whose coefficients are functions of parameters," *IEEE Transactions on Automatic Control*, vol. 29, no. 1, pp. 25–33, 1984.

[7] J. S. Shamma and M. Athans, "Guaranteed properties of gain scheduled control for linear parameter-varying plants," *Automatica*, vol. 27, no. 3, pp. 559–564, 1991.

[8] S. M. Shahruz and S. Behtash, "Design of controllers for linear parameter varying systems by the gain scheduling technique," *Journal of Mathematical Analysis and Applications*, vol. 168, no. 1, pp. 195–217, 1992.

[9] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, 1989.

[10] P. Gahinet and P. Apkarian, "A linear matrix inequality approach to H_∞ control," *International Journal of Robust and Nonlinear Control*, vol. 4, no. 4, pp. 421–448, 1994.

[11] R. Ravi, K. M. Nagpal, and P. P. Khargonekar, "H_∞ control of linear time-varying systems: A state space approach," *SIAM Journal on Control and Optimization*, vol. 29, pp. 1394–1413, 1991.

[12] K. Deb, *Multi-objective optimization using evolutionary algorithms*. John Wiley & Sons, 2001.

[13] M. P. Horton, "Autopilots for tactical missiles: An overview," *IMechE: Journal of Systems and Control Engineering*, vol. 209, pp. 127–139, 1995.

[14] K. A. Wise, "Comparison of 6 robustness tests evaluating missile autopilot robustness to uncertain aerodynamics," *Journal of Guidance, Control and Dynamics*, vol. 15, no. 4, pp. 861–870, 1992.

[15] M. P. Horton, "A study of autopilots for the adaptive control of tactical guided missiles," Master's thesis, University of Bath, 1992.

[16] A. Tsourdos, A. Blumel, and B. A. White, "Trajectory control of a nonlinear homing missile," in *14th IFAC Symposium on Automatic Control in Aerospace*, pp. 118–123, 1998.

[17] T. Tanaka and M. Sugeno, "Fuzzy indentification of systems and its applications to modeling and control," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116–132, 1985.