Defence Science Journal, Vol. 52, No. 4, October 2002, pp. 409–416

Secondary Injection Thrust Vector Control Power Plant Linearisation using Fuzzy Logic for a Launch Vehicle

S. Subha Rani and S. Sibi

PSG College of Technology, Coimbatore – 610 004

and

P.P. Mohanlal

Vikram Sarabhai Space Centre, Thiruvananthapuram – 695 022

ABSTRACT

Control forces are required for steering a launch vehicle to guide it to follow an optimal trajectory. Launch vehicle control involves two control loops, the inner loop deals with short-period dynamics, stability and the outer loop, known as the guidance loop, optimises the trajectory. The general nonlinear plant model is first approximated as a linear time-varying plant over a nominal trajectory and then segmented as linear, time-invariant plant models at different time intervals. A major part of the plant model is the control power plant, which for a secondary injection thrust vector control system used for the solid booster stage of a launch vehicle is nonlinear due to various reasons. The controllers designed for different time regimes assume the control power plant as linear and are adapted smoothly by a technique called gain scheduling to cope with the plant model changes wrt time. In this paper, a fuzzy logic-based pre-compensator is developed to linearise the control power plant so that the controller design becomes valid. Simulation results are presented to validate the design and a novel preprocessing technique is developed to reduce the size of the fuzzy inference system.

Keywords: Secondary injection thrust vector control system, launch vehicle, fuzzy logic, nonlinear plant model, fuzzy inference system, control power plant, control loop, fuzzy inference system, pre-compensator

NOMENCLATURE

$d1$ Physical opening of pintle valve in the pitch axis

$d2$ Physical opening of pintle valve in the yaw axis

$d1_{eff}$ Effective port opening in the pitch axis

P Injectant tank pressure

Fs Side force developed for single quadrant operation

F Side force developed when control forces are required in the mutually perpendicular plane.

1. INTRODUCTION

Two kinds of control power plants are used mainly for launch vehicle control, the engine gimbol control and the secondary injection thrust

Revised 02 November 2001

vector control (SITVC). The former is used for liquid engines and the latter for large solid boosters at lower stages. In a SITVC system, the main nozzle is surrounded by a torroidal tube near the convergent end of the nozzle and many electro-mechanically controlled port openings are provided uniformly on the torroid. Through these ports, a nonflammable liquid at high pressure is injected into the nozzle jet. The resulting deflection of the jet gives rise to side forces, which can be used for steering the main vehicle.

Apart from actuator nonlinearities, there are three major nonlinearities in a SITVC system. In this paper, a fuzzy logic-based pre-compensator is designed and developed to linearise all the nonlinearities except the actuator nonlinearity.

The following three schemes were developed for the above design:

(a) Adaptive neuro-fuzzy inference scheme (ANFIS) using TSK consequents

(b) Adaptive standard additive model (ASAM)-based design

(c) A reduced ASAM to represent the same pre-compensator by a novel preprocessing technique.

All the three methods are found to be effective in linearising the control power plant. To evaluate the performance of the plant with and without pre-compensator, simulations were done using standard signals. The simulation study was carried out in the Matlab/Simulink system-simulation environment. The response of the compensated plant and the reference model were compared to justify the effectiveness of the compensator designed.

2. PLANT DESCRIPTION

The control power plant for the SITVC system[1] has three types of nonlinearities apart from the actuator nonlinearity. These are

(a) Nonlinear relationship between the side force developed and the port opening

(b) Nonlinearity due to the injectant tank pressure variation

$$d1_{eff} = d1 \times \sqrt{\frac{P}{7.85}}$$

For single quadrant operation, control forces are required in either pitch or yaw plane only. For that, the port opening of injectors in the required plane alone should be adjusted. When control force is required in a particular plane, say, pitch plane, the side force developed, is given by $Fs = f\,(d1_{eff})$, where $f\,(d1_{eff})$ is calculated using curve fitting technique from the actual test data.

(c) Nonlinearity due to simultaneous port opening of the adjacent quadrant ports (yaw-axis control).

When control forces are required in the mutually perpendicular plane, simultaneous port opening of injectors on the adjacent quadrants become necessary, and then the side force developed is given by

$$F = Fs * (1 - 0.025 * d)$$

where d is the minimum of $d1$ and $d2$.

Hence, the plant under consideration has three inputs ($d1$, $d2$ and P) and one output (F). A pre-compensator is developed to take care of all the above nonlinearities except the actuator nonlinearity, which thus makes the compensated plant to behave as a linear system. Thus, the assumption for the controller design becomes valid and its stability and performance analysis by linear techniques become justified, and hence, the whole design and analysis become robust.

Using the test data for (a) and other nonlinear equations for (b) and (c) mentioned above, 2890 samples of training vectors were generated for the plant. The pre-compensator can be treated as the inverse of the plant but for the scale factor, K whose value is taken as 20 for convenience. The input to the pre-compensator are command voltage (F/K), $d2$ and P (where $d2$ and P are the same as that of the plant) and its output is u.

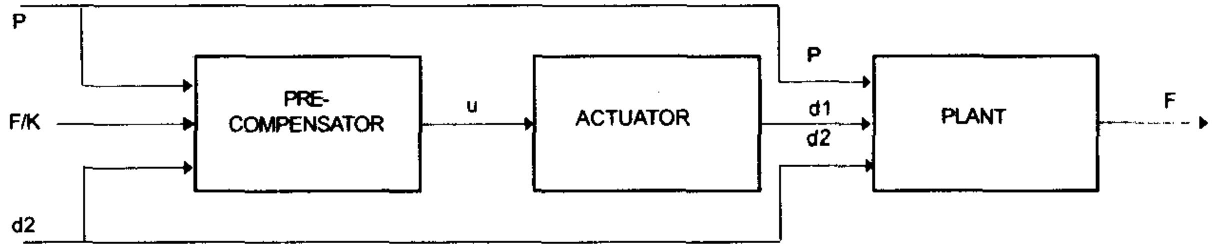

Figure 1. Block diagram of the overall system

which, in turn, is given as input to the plant as shown in fig 1.

Figure 1 shows the block diagram of the overall system with pre-compensator, actuator and the nonlinear plant. In simulation experiments, a linear second-order model of damping ratio 0.7 and bandwidth 5Hz is taken as the actuator.

3. PRE-COMPENSATOR DESIGN USING ANFIS WITH TSK CONSEQUENTS

The pre-compensator is represented using ANFIS[2] subject to the following constraints:

(a) Sugeno-type fuzzy inference system used for modelling

(b) Weighted average method used for de-fuzzification.

(c) Unity weight assigned for each rule.

The TS-type fuzzy model having rule consequents as linear functions of input variables makes the system notation very compact and efficient. Since the training data is uniformly distributed, the initial set of fuzzy rules are constructed using grid partitioning technique. Four Gaussian-type membership functions are provided for each of the three input variables, and thus, 64 inference rules are provided in total. Figures 2(a) and 2(b) show the overall linear relation-ship between the pre-compensator input, F/K, and the plant output, F for different values of $d2$ and P.

4. ADAPTIVE STANDARD ADDITIVE MODEL-BASED PRE-COMPENSATOR DESIGN

4.1 Standard Additive Model

The standard additive model (SAM) includes almost all fuzzy systems found in practice, and

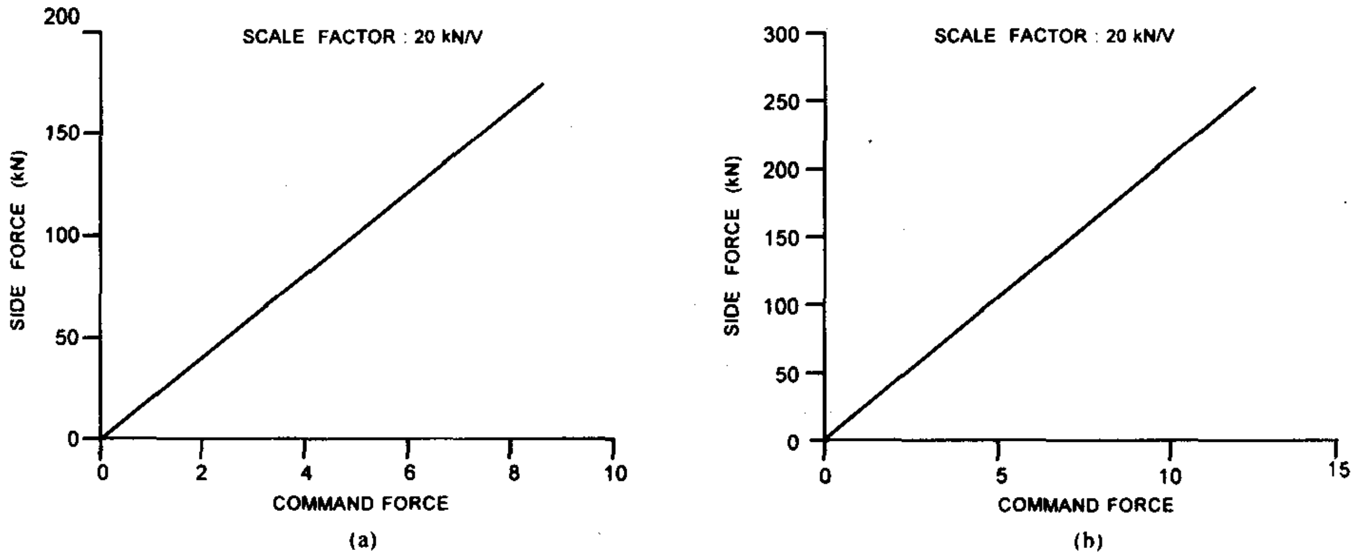

Figure 2. Overall linear relationship between the pre-compensator input, *F/K* and the plant output, *F* for command force *vs* side force: (a) for *d2* = 8 and *P* = 3.5 and (b) for *d2* = 0 and *P* = 8.0

hence, can be viewed as the most generalised form. The SAM has the simple form of a convex sum and can be extended to more complex fuzzy models where the then-part fuzzy sets are linear operators (TSK) or nonlinear operators. SAMs can combine any number of weighted fuzzy systems into a common fuzzy system. Figure 3 shows the architecture of a SAM[3,4].

In SAM, the fired then-part set

$$B_j^1 = a_j(x) B_j \tag{1}$$

where $a_j(x)$ is the degree to which the input $x = (x_1, x_2, \ldots, x_n)$ fires or activates the j^{th} rule and is found using the product combiner,

$$a_j(x) = \prod_{i=1}^{n} a_j^i(x_i).$$

The sum of fired then-part sets

$$B = \sum_{j=1}^{m} w_j B_j^1 \tag{2}$$

$$= \sum_{j=1}^{m} w_j a_j(x) B_j \tag{3}$$

The SAM output, $F(x)$ = Centroid $[B(x)]$

$$= \frac{\sum_{j=1}^{m} w_j * a_j(x) * V_j * C_j}{\sum_{j=1}^{m} w_j * a_j(x) * V_j} \tag{4}$$

$$= \sum_{j=1}^{m} p_j(x) C_j \tag{5}$$

where

$$p_j(x) = \frac{w_j * a_j(x) * V_j}{\sum_{j=1}^{m} w_j * a_j(x) * V_j} \tag{6}$$

C_j is the then-part set centroids and

V_j is the then-part set volume or area.

The m coefficients $p_1(x), p_2(x), \ldots, p_m(x)$ are convex in that each term is non-negative $p_j(x) \geq 0$ and they sum to unity.

$$\sum_{j=1}^{m} p_j(x) = 1 \tag{7}$$

Thus, $F(x)$ is a convex sum of the m then-part set centroids. In general, V_j s can be different for different j and the same is true for w_j as well.

4.2 Pre-compensator Design

Since the training data is uniformly distributed in the product space, grid partitioning technique

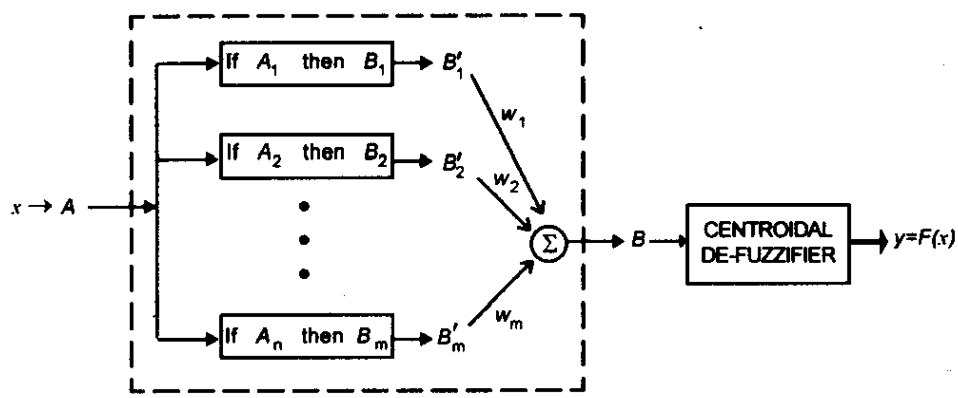

Figure 3. Architecture of standard additive model

is chosen for generating the initial fuzzy inference system, and later, for fine tuning, supervised gradient descent algorithm is used. After trial and error, the size of the system is fixed as (9,5,5), i.e., 9 membership functions for the first input which is the command voltage and 5 membership functions each for the second and third input $d2$ and P, respectively. Thus, a total of 225 fuzzy rules are generated. Here, Gaussian-type membership functions of antecedent part and real number for consequent parts are taken.

4.2.1 *Adaptive Algorithm for* SAM

The supervised gradient descent method[5] is used for learning the SAM model of a pre-compensator. Unlike other fuzzy systems, SAM systems have the flexibility to tune rule weights w_j and then-part volumes V_j in addition to the tuning of antecedent and consequent membership function parameters. This makes SAM systems to learn faster than any other system. Here, the volumes are normalised to unity. The SAM output is given as

$$F(x)=\frac{\sum_{j=1}^{m} w_j * a_j(x) * C_j}{\sum_{j=1}^{m} w_j * a_j(x)} \tag{8}$$

where m = 225

$$= \sum_{j=1}^{m} p_j(x) * C_j \tag{9}$$

where

$$p_j(x) = \frac{w_j * a_j(x)}{\sum_{j=1}^{m} w_j * a_j(x)} \tag{10}$$

The tuning process is as given below:

(a) For each input vector, calculate the output $F(x)$ using Eqns (8) to (10).

(b) Compute the squared error $E(x)$ = ½* $[d(x)-F(x)]^2$, where $d(x)$ and $F(x)$ are the desired and the actual output of the fuzzy system for input x. In this method, the inference rules are tuned so as to minimise the objective function E.

(c) Update the rule weights as follows:

$$w_j(t+1)=w_j(t)-\mu(t)\frac{\partial E}{\partial w_j}$$
$$=w_j(t)+\mu(t)\varepsilon(x)\frac{p_j(x) * [C_j - F(x)]}{w_j(t)} \tag{11}$$

where $\mu(t)$ = Learning constant at time t, and $\varepsilon(x) = d(x) - F(x)$.

- Update the consequent parameters such that the centroids in the consequent parts are adjusted as below.

$$C_j(t+1) = C_j(t) - \mu(t)\frac{\partial E}{\partial C_j}$$
$$= C_j(t) + \mu(t)\ \varepsilon(x)\ p_j(x) \tag{12}$$

- Update the change in adjustable parameters of the antecedent part where the centre (m_j) and width (σ_j) of the gaussian-type membership functions are adjusted as below

$$m_j(t+1)=m_j(t)-\mu(t)\frac{\partial E}{\partial m_j}$$
$$=m_j(t)+\mu(t)\varepsilon(x)p_j(x)$$
$$[C_j-F(x)]\left(\frac{x-m_j}{\sigma_j^2}\right) \tag{13}$$

$$\sigma_j(t+1)=\sigma_j(t)-\mu(t)\frac{\partial E}{\partial \sigma_j}$$
$$=\sigma_j(t)+\mu(t)\varepsilon(x)p_j(x)$$
$$[C_j-F(x)]\frac{(x-m_j)^2}{(\sigma_j)^3} \tag{14}$$

The learning rules of Eqns (11) to (14) are to adaptively change the tuning parameters in a direction to minimise the objective function E. Thus using these learning rules, the tuning parameters

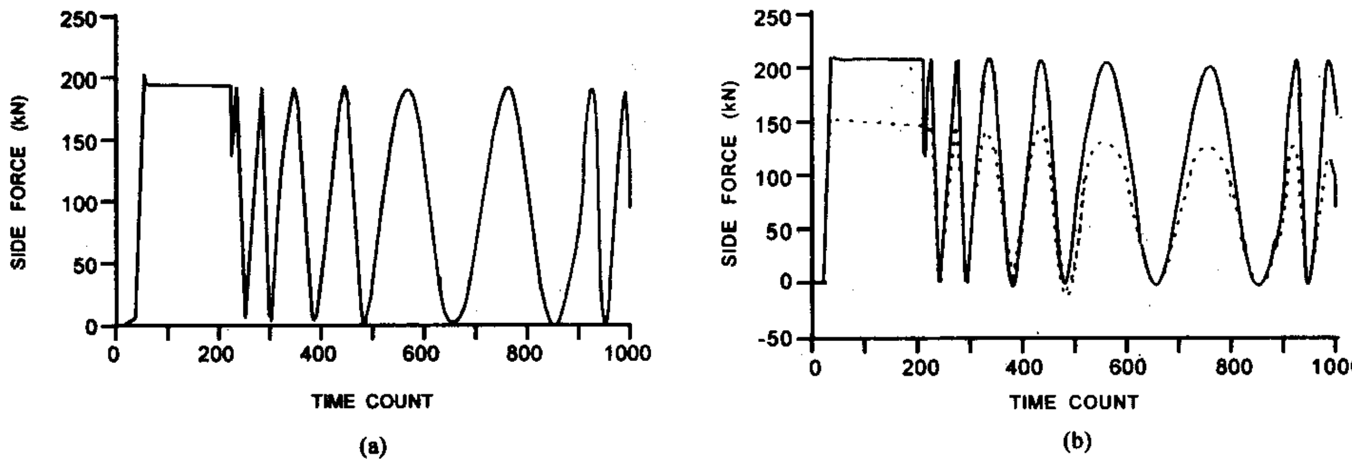

Figure 4. (a) Reference output (b) comparison of compensated and uncompensated output

of inference rules are optimised to minimise the inference error between the desired output and the output of the fuzzy reasoning. Here, batch learning is used for updating the parameters.

Figure 4 (a) shows the response of the reference model for an untrained input sequence of 1000 samples. Figure 4(b) shows the response of the plant with and without pre-compensator for the same 1000 samples to verify the effectiveness of the linearisation scheme.

5. NOVEL PREPROCESSING TECHNIQUE FOR PRE-COMPENSATOR DESIGN

The fuzzy inference system required to represent the pre-compensator is reduced in size by preprocessing its command input. This is a novel approach but is specific for a system depending on its features. Based on the training data, the actual profile of the command input is as shown in Fig. 5(a). The preprocessed command profile is given in Fig. 5(b). The smoothening in the pre-processed command profile makes possible the fuzzy inference system size reduction. The three input of pre-compensator are command (F/K), $d2$, P and its output is u. For the preprocessing of command input, single quadrant operation ($d2 = 0$) is assumed. The preprocessing accounts for the pressure-dependent nonlinearity of the output at $d2 = 0$.

For single quadrant operation, the side force developed is given by

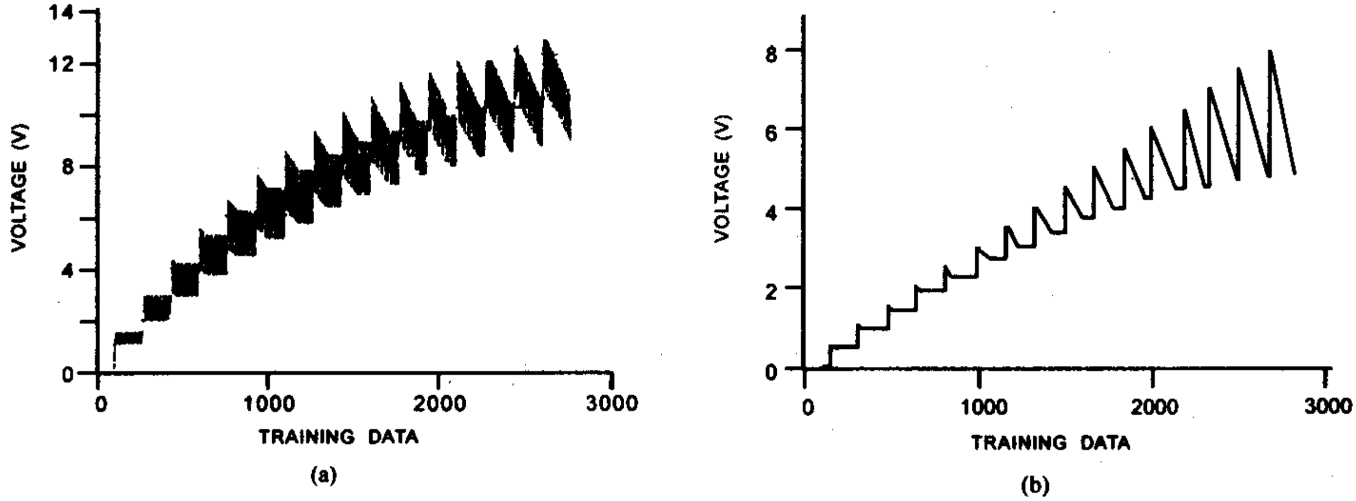

Figure 5. (a) Command input (b) preprocessed command input

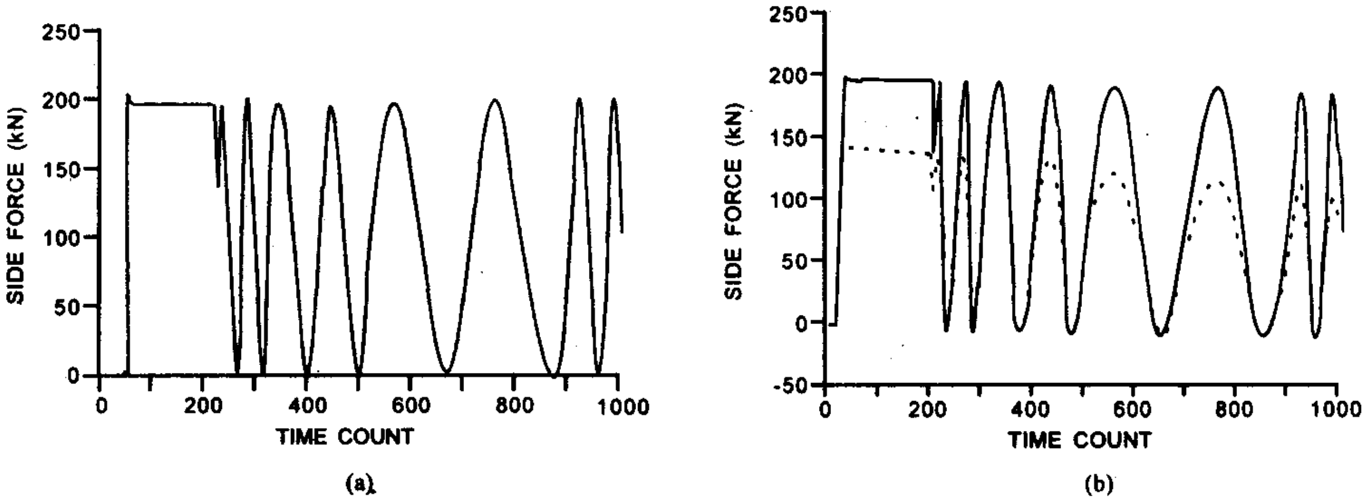

Figure 6. (a) Reference output and (b) comparison of compensated and uncompensated output preprocessed case

$$Fs = f(d1_{eff}) = 0.1919*d1_{eff}^3 - 5.6558*d1_{eff}^2 + 65.07*d1_{eff} \quad (15)$$

where

$$d1_{eff} = d1 * \left(\frac{P}{7.85}\right)^{1/2} \quad (16)$$

$$Fs = 0.1919*\left[d1*\left(\frac{P}{7.85}\right)^{1/2}\right]^3 - 5.6558*\left[d1*\left(\frac{P}{7.85}\right)^{1/2}\right]^2 + 65.07*d1*\left(\frac{P}{7.85}\right) \quad (17)$$

Let $z = d1 * P^{1/2}$

then

$$Fs = \frac{0.1919}{(7.85)^{3/2}} * z^3 - \frac{5.6558}{7.85} * z^2 + \frac{65.07}{(7.85)^{1/2}} * z \quad (18)$$

$$= k1 * z^3 + k2 * z^2 + k3 * z = g(z) \quad (19)$$

where

$$k1 = \frac{0.1919}{(7.85)^{3/2}}, k2 = -\frac{5.6558}{7.85}, \text{and } k3 = \frac{65.07}{(7.85)^{1/2}} \quad (20)$$

The inverse of the polynomial in Eqn (19) is obtained using curve-fitting technique as

$$z = g^{-1}(Fs)$$

The actual command input of pre-compensator is *F*/*K* from which the equivalent values of side force, *F* are computed knowing the scale factor *K* whose value is 20. Using the fitted inverse polynomial,

$$z^1 = g^{-1}(F)$$

which now is the combination of preprocessed command and pressure ($P^{1/2}$). The preprocessed command so obtained is used for modelling the pre-compensator by replacing the command input of original training data.

The initial fuzzy inference system for the pre-compensator is generated using grid partitioning technique. Thus, the centres of the initial membership functions are spaced equally along the domain of each input variable. After trial and error, it was found sufficient to use a system of size (9,4,2), i.e., 9 membership functions for command input, 4 membership functions for *d2* and 2 membership functions for pressure *P* in place of (9,5,5) for the original SAM system, reducing the size of rule base from 225 to 72 without loosing accuracy. Supervised gradient descent algorithm is used for fine tuning.

Figure 6(a) shows the response of the reference model for an untrained input sequence of 1000 samples. Figure 6(b) shows the response of the plant with and without pre-compensator for the same 1000 samples to verify the effectiveness of the linearisation scheme.

6. CONCLUSION

The development of pre-compensator is carried out successfully using three different schemes: (i) ANFIS with TSK consequents, (ii) ASAM-based design, and (iii) a novel preprocessing technique by which the pre-compensator is represented using a reduced ASAM. The adaptive algorithms have been developed for the second and third methods and also the preprocessing scheme developed has resulted in substantial reduction in the size of the inference system. The design of pre-compensator has enabled the overall compensated system to behave as the desired linear system.

REFERENCES

1. Twardy, H. Methods of thrust vector control. Technical Report-ESRO-TT-143.

2. Jang, J.S.R. ANFIS: Adaptive network-based fuzzy inference systems. *IEEE Trans. Syst. Man Cybernetics*, May 1993, **23**(3), 665–85.

3. Bart, Kosko. Fuzzy engineering. Prentice Hall Inc., New Jersey, 1997.

4. Bart, Kosko. Neural networks and fuzzy systems: A dynamical systems approach to machine intelligence. Prentice Hall of India, 1992.

5. Nomura, H.; Hayashi, I. & Wakami, N. A learning method of fuzzy inference rules by descent method. *Proc. IEEE Inter. Conf. fuzzy Syst.*, 1992, 203–10.

Contributors

Dr (Ms) S Subha Rani obtained her PhD from the PSG College of Technology, Coimbatore. Presently, she is Assistant Professor in the Dept of Electronics and Communication Engineering, PSG College of Technology, Coimbatore. Her areas of interest include: Optimal control, systems identification, soft computing, and embedded systems.

Ms S Sibi obtained her BTech (Electrical & Electronics Engg) from the LBS College of Engineering, Calicut University, Kerala, in 1998 and ME (Control System) from the PSG College of Technology, Coimbatore. Her research interests include: Optimal control, neural networks, and fuzzy systems.