JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 15, No. 2, March-April 1992

Fault Diagnosis for the Space Shuttle Main Engine

Ahmet Duyar*
Florida Atlantic University, Boca Raton, Florida 33431
and
Walter Merrill†
NASA Lewis Research Center, Cleveland, Ohio 44135

A conceptual design of a model-based fault detection and diagnosis system is developed for the Space Shuttle main engine. The design approach consists of process modeling, residual generation, and fault detection and diagnosis. The engine is modeled using a discrete time, quasilinear state-space representation. Model parameters are determined by identification. Residuals generated from the model are used by a neural network to detect and diagnose engine component faults. Fault diagnosis is accomplished by training the neural network to recognize the pattern of the respective fault signatures. Preliminary results for a failed valve, generated using a full, nonlinear simulation of the engine, are presented. These results indicate that the developed approach can be used for fault detection and diagnosis. The results also show that the developed model is an accurate and reliable predictor of the highly nonlinear and very complex engine.

Introduction

THIS paper describes a model-based fault diagnosis system based on a neural network classifier for the Space Shuttle main engine (SSME). The system may be used to monitor the life cycle of engine components and for the early detection, isolation, and the diagnosis of engine faults. As such, the proposed system will be one part of a larger, engine health monitoring system.[1] The health monitoring system will allow for accommodation of faults, reduce maintenance cost, increase engine availability, and be one part of an integrated, intelligent control system[2] for the SSME. A description of SSME dynamics and its modeling is given in a study by Duyar et al.[3] A summary of the major failures of the SSME that have occurred are outlined by Cikanek.[4] Several authors[5-8] survey the available methods and approaches for fault detection and diagnosis. In particular, the survey by Isermann[6] gives several examples of the use of identification techniques for process fault detection.

A fault is the abnormal behavior of a component due to physical change in the component. A fault event often impairs or deteriorates the system's ability to perform its specified tasks or mission. The detection task is defined as the act of identifying the presence of an unspecified fault. After a fault is detected, then the fault must be isolated to the component that has failed. During the process of isolation, the magnitude of the fault may be estimated. Fault diagnosis is the isolation and estimation of a fault mode. Once a fault is detected and diagnosed, the fault can be accommodated through reconfiguration of the system. Reconfiguration includes both hardware actions (e.g., activating backup systems) and software tasks (e.g., adjusting the feedback control gains). The detection and diagnostic tasks may be accomplished by an onboard processor, on line and in real time for fault accommodation, as well as by an off-line processor that analyzes recorded data for life-cycle analysis and preventive maintenance.

Received July 7, 1990; revision received Oct. 26, 1990; accepted for publication Feb. 21, 1991.

*Professor, Mechanical Engineering Department.
†Deputy Branch Chief, Advanced Controls Technology Branch, Mail Stop 77-1, 21000 Brookpark Road. Senior Member AIAA.

Initially, a brief description of the conceptual design of the model-based fault detection and diagnostic system (FDDS) is given. This is followed by a description of the process modeling, the residual generation method, and the detection and diagnostic system design. Finally, results of the application of the FDDS to the detection of a stuck valve fault using simulated data are presented.

Conceptual Design

Model-based fault detection methods rely on the determination of changes appearing in the system due to the existence of a fault, in comparison with the normal status of the system. For example, in aerospace applications control actuator faults may be represented as shifts in the parameters of the control gain matrix. Sensor faults may be represented as abrupt changes in the parameters of the output matrix or increases in measurement noise. These changes are determined by comparing the parameters of the observed process with the parameters obtained from the model of the normal process. The differences between these parameters are called residuals. The residuals and their patterns are analyzed for fault detection and diagnosis by comparing them with the known fault signatures of the process.

Fault signatures, which show the effect of a fault on the parameters, are generated by inducing faults in the nonlinear dynamic simulation of the process. Fault diagnosis is accomplished by training a neural network classifier to recognize the pattern of the respective fault signatures.

The design of the FDDS is accomplished in three stages: process modeling, residual generation and fault detection, and diagnostic classifier design. In the following sections the methods used in these stages are briefly explained.

Process Modeling

A complete nonlinear dynamic simulation of SSME performance was developed by Rocketdyne Division of Rockwell International Corporation.[9] In this study, this nonlinear model is considered as the unknown process. It is used for the generation of fault signatures by modifying the actuator models to include a fault model. The input-output data generated from this simulation are also used to identify the parameters of the engine. Due to its size and complexity (40 min of CPU time for 20 s of real-time operation with a VAX 8800), this nonlinear simulation cannot be used to generate data in real time to describe the normal mode of operation.

An off-line system identification algorithm developed by Eldem and Duyar[10] and the data generated from the nonlinear performance simulation are used to obtain linear point models of the SSME at 25 different operating points. The inputs of these models are the rotary motion of the valve actuator outputs of the oxidizer preburner oxidizer valve (OPOV), β_{OPOV}, and fuel preburner oxidizer valve (FPOV), β_{FPOV}. The point models have measurable state variables that simplify the model structure. The outputs, which are also the state variables, are the chamber inlet pressure P_C, mixture ratio MR, high-pressure fuel turbine speed S_{HPFT}, and high-pressure oxidizer turbine speed S_{HPOT}.

Two models of the SSME are used in the design of the FDDS: a linear state variable observer and a linear state-space model of the normal operation of the engine. Consider the discrete state-space representation of the engine, linearized about one operating point:

$$x(n+1) = Ax(n) + B(n) \tag{1}$$

$$y(n) = Cx(n) \tag{2}$$

here x, u, and y are the deviations of the state, the input, and the output vectors about an operating point. For mathematical simplicity, it is assumed that the system is not subject to disturbances and to sensor noise. Following the work of Eldem and Duyar,[10] it is assumed that the system is in α-canonical form such that the following is satisfied:

$$C = [0{:}H^{-1}]$$

$$A = A_0 + KHC$$

$$A_0^{\mu} = 0 \tag{3}$$

$$(HC)_{r_i} A_0^{\mu_i} = 0$$

$$(HC)_{r_i} A_0^{k} K_{c_j} = 0 \qquad \text{for} \qquad \mu_i > \mu_j \qquad \text{and} \qquad k < \mu_i - \mu_j$$

Here the subscript r_i and c_j denote the ith row and jth column, respectively. Superscripts indicate exponentiation. The structure matrix A_0 is lower left triangular and consists of zeros and ones only and is determined by the observability indices μ_k where k associates μ_k with the kth output and $\mu = \max\{\mu_i\}$. The matrix K is a deadbeat observer gain.

To estimate the states from the measured input and output data, a state variable filter

$$x_f(n+1) = Ax_f(n) + Bu(n) + K[y(n) - Cx_f(n)] \tag{4}$$

$$y(n) = Cx(n) \tag{5}$$

is used. Here the subscript f denotes the estimated values obtained from the filter. Equations (1–5) can be solved to give

$$x_f(n) = A_0^n x(0) + \sum_{i=1}^{\mu} A_0^{i-1} [KH{:}B] \begin{bmatrix} y(n-i) \\ u(n-i) \end{bmatrix} \tag{6}$$

Using the nillpotency of A_0, the previous equation yields the state variable filter equation

$$x_f(n) = \sum_{i=1}^{\mu} A_0^{i-1} [K{:}B] \begin{bmatrix} y(n-1) \\ u(n-1) \end{bmatrix} \qquad \text{for} \qquad n > \mu \tag{7}$$

$$y_f(n) = Cx_f(n) \tag{8}$$

which can be used to estimate the states from the measurements of the input and the output data.

Unlike the state variable filter, the state-space model of the engine estimates the state variables from the measurement of the input data only. The equation describing the state-space model of the engine is given as follows:

$$x_m(n) = \sum_{i=1}^{\mu} A_0^{i-1} [K{:}B] \begin{bmatrix} y_m(n-1) \\ u(n-1) \end{bmatrix} \qquad \text{for} \qquad n > \mu \tag{9}$$

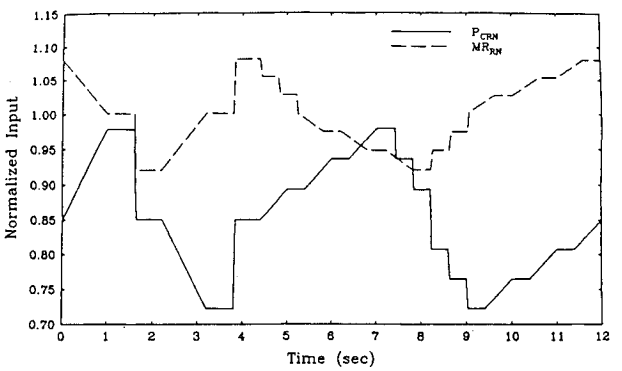
$$y_m(n) = Cx_m(n) \tag{10}$$

where the subscript m denotes the variables estimated by the model.

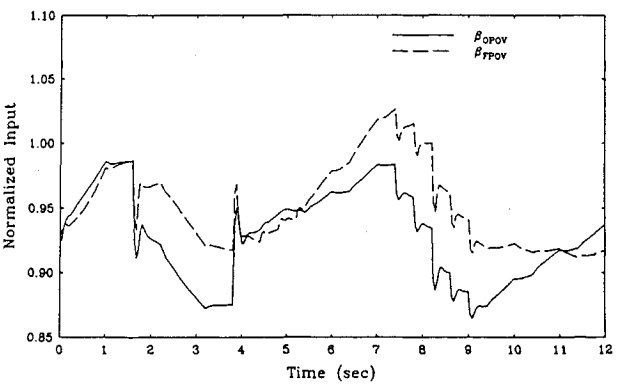
Both the state variable filter and the state-space model are tested by comparing their prediction with the actual output obtained from the nonlinear simulation. The point models can predict the output of the nonlinear simulation with very good accuracy.[11] These point models are linked to obtain a simplified quasilinear model of the SSME, valid within its full range of operation.[12] The parameters of the point models are regressed with the parameters determining the nominal operating conditions. The mixture ratio and the chamber pressure are considered as the parameters that determine a nominal operating condition. This simplified model is also tested by designing an input signal as shown in Fig. 1. The comparison of the responses of the linked model and the nonlinear simulation, again, indicated good agreement as shown in Figs. 2 and 3.

Residual Generation

As mentioned earlier, it is assumed that faults are indicated by changes in the parameters of the system as well as by internal, observable, but not necessarily measured, process state variables. The state variables can be estimated by a filter or a state-space model based on the known process parameters. The parameters of the process can be determined by using a system identification technique. Then residuals can be generated by taking the difference between the actual parameters and the observed parameters.

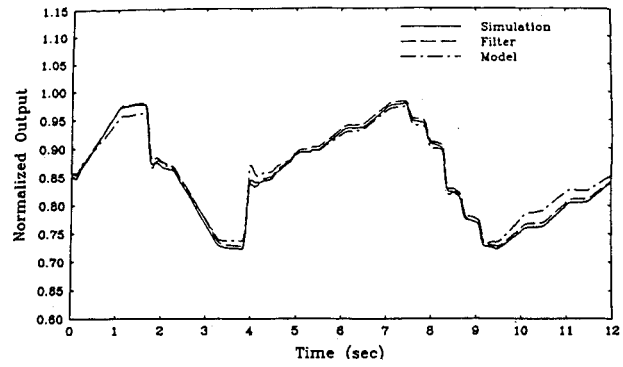


a) P_{CRN} and MR_{RN} requests

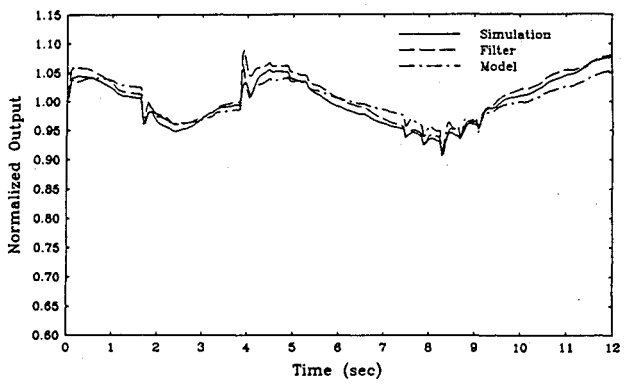


b) β_{OPOV} and β_{FPOV} signals

Fig. 1 Reference input signals and their corresponding valve inputs.

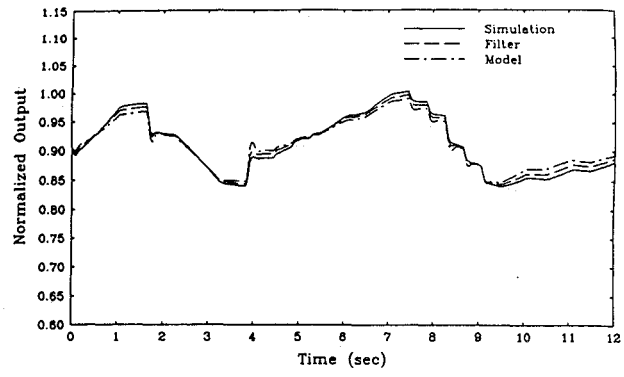


a) P_{CN} chamber inlet pressure

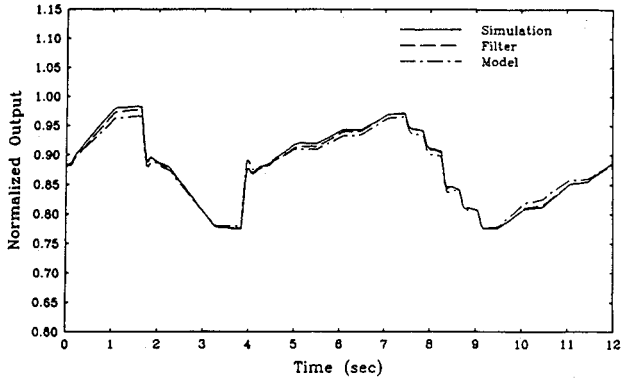


b) MR_N mixture ratio

Fig. 2 Comparison of the responses of the linked model and the filter with the nonlinear simulation.



a) S_{HPFTN} high-pressure fuel turbine speed



b) S_{HPOTN} high-pressure oxidizer turbine speed

Fig. 3 Comparison of the responses of the linked model and the filter with the nonlinear simulation.

Two kinds of residuals may be generated: 1) parameter, by comparing the identified parameters of the engine with the normal parameters, and 2) output, by taking the difference between the actual output and the output obtained from the estimated state variables. The simplest test to detect a fault is a comparison of residual magnitudes to a threshold value. Using the distribution of the variances of the residual under fault-free conditions, the threshold values can be determined to minimize false alarms and missed detections using the Neyman-Pearson criterion.[13] Other detection approaches that would statistically analyze the residuals are possible such as those based on likelihood ratios or sequential probabilities.[5]

In this study, the state variables of the system are used for the generation of output residuals. With the observed state variables, residuals are generated between the measured output and the output obtained using the observed state variables as

$$\delta y_1(n) = y(n) - Cx_f(n) \tag{11}$$

Measured output and the output obtained from the state-space model are also used to generate additional residuals as

$$\delta y_2(n) = y(n) - C_m x_m(n) \tag{12}$$

These residuals are generated by inducing stuck valve faults in the nonlinear dynamic simulation of the SSME. Both the OPOV and the FPOV are considered for this purpose. In this paper a stuck fault is defined such that a valve may not move above a certain angle, called the fault severity. However, the valve may move as commanded below this angle. Data are obtained at various stuck fault severity angles. Figure 4 shows the residuals obtained for two of the outputs, chamber inlet pressure and the mixture ratio, by using Eq. (12).

Fault Detection and Diagnosis

A neural network classifier is used for fault detection and diagnosis purposes. Following the work of Dietz et al.,[14] a two-layer network architecture combined with a back propagation algorithm is selected.

Neural Network Architecture

The neural network architecture employed in this study, as shown in Fig. 5, is a two-level architecture: 1) the classifier level where the faults are actually classified as belonging to a particular category (fault detection), and 2) the severity level where the severity (magnitude) of the fault that was identified in the classifier level is estimated.

Classifier Level

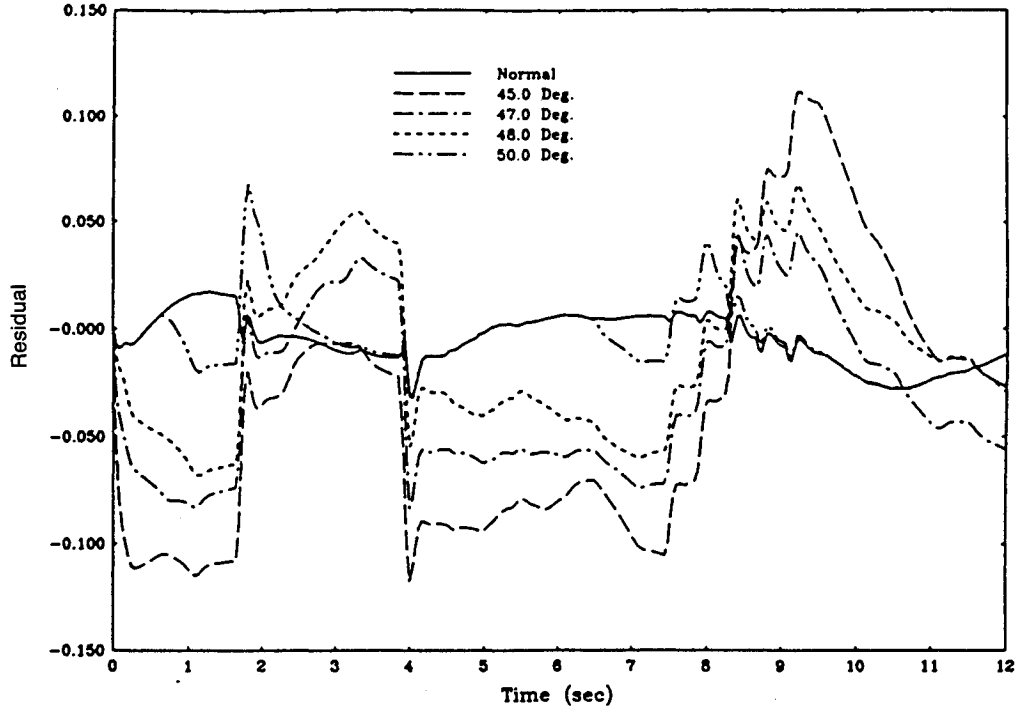
The classifier level consists of two networks, one associated with the chamber pressure residual and the other with the mixture ratio residual. Each of these two networks are three layers (including the input and output layers) feedforward networks with nonlinear hidden and output units. The weights in these networks are assigned using the generalized back propagation algorithm.[14]

Each of these networks consists of 200 input nodes, 20 hidden nodes, and 2 output nodes. The input to one network is a time sequence of the chamber pressure residual of length 200. The time step between residuals is 0.04 s with the total sequence time representing 8 s. Similarly, the other network receives the mixture ratio residual sequence as its input. For each of these two networks, one output node is associated with the OPOV stuck condition whereas the other output node corresponds to the FPOV stuck condition. In short, one output node is activated if an OPOV stuck condition is activated; the other is activated if an FPOV stuck condition is activated. The output activations are real numbers between 0 and 1.

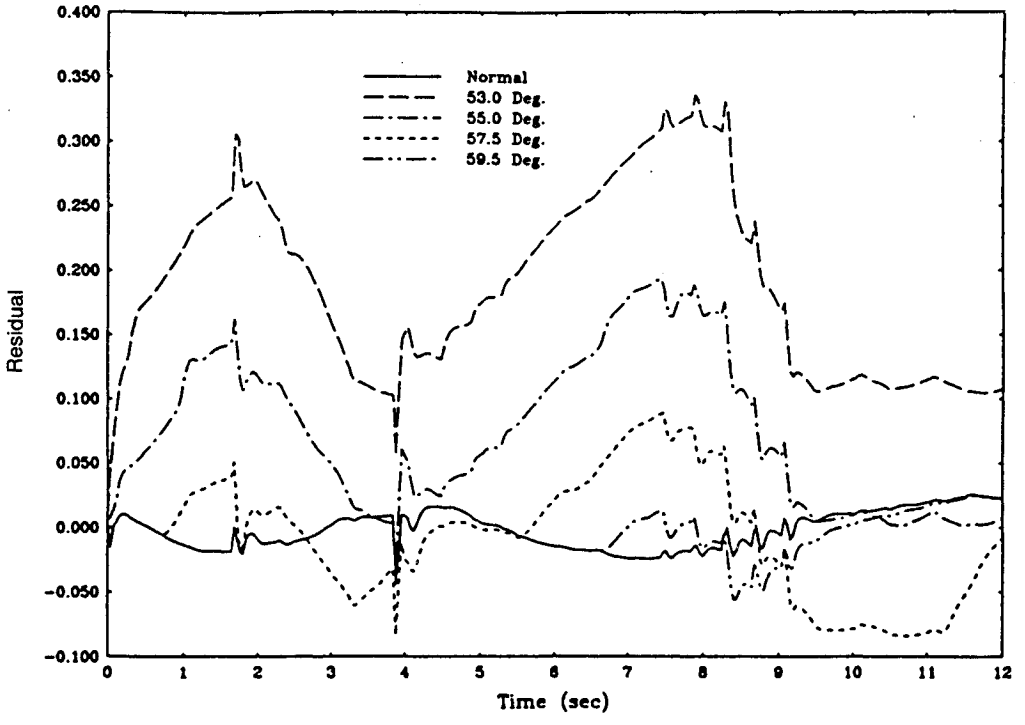
For network training, six fault scenarios were generated from the nonlinear dynamic simulation for the following fault conditions: 1) the OPOV valve stuck at 45, 47, and 50 deg,

respectively, and 2) the FPOV valve stuck at 53, 57, and 59.5 deg, respectively. Chamber pressure and mixture ratio residuals for these scenarios were generated as in Eq. (12) for a time span of 8 s in steps of 0.04 s (a 200 length sequence). That is, the residual data are generated as the difference of the actual output of the SSME nonlinear simulation and the output generated by the linked model.

During training, a residual pattern representing a fault condition is applied to the input level (200 nodes) and a 1 (indicating full activation) is applied to the corresponding output node. For instance, the chamber pressure residual corresponding to an OPOV stuck valve condition is applied to one of the classifier networks and an activation of 1 is imposed on the output node corresponding to the OPOV stuck condition. The network weights are then adjusted, invoking the back propagation algorithm, thus enabling the neural network to learn the imposed input-output pattern. Each of the classifier networks is trained using all six fault scenarios.



a) P_C residuals for OPOV stuck at various angles



b) *MR* residuals for FPOV stuck at various angles

Fig. 4 Output residuals for stuck valve failures.

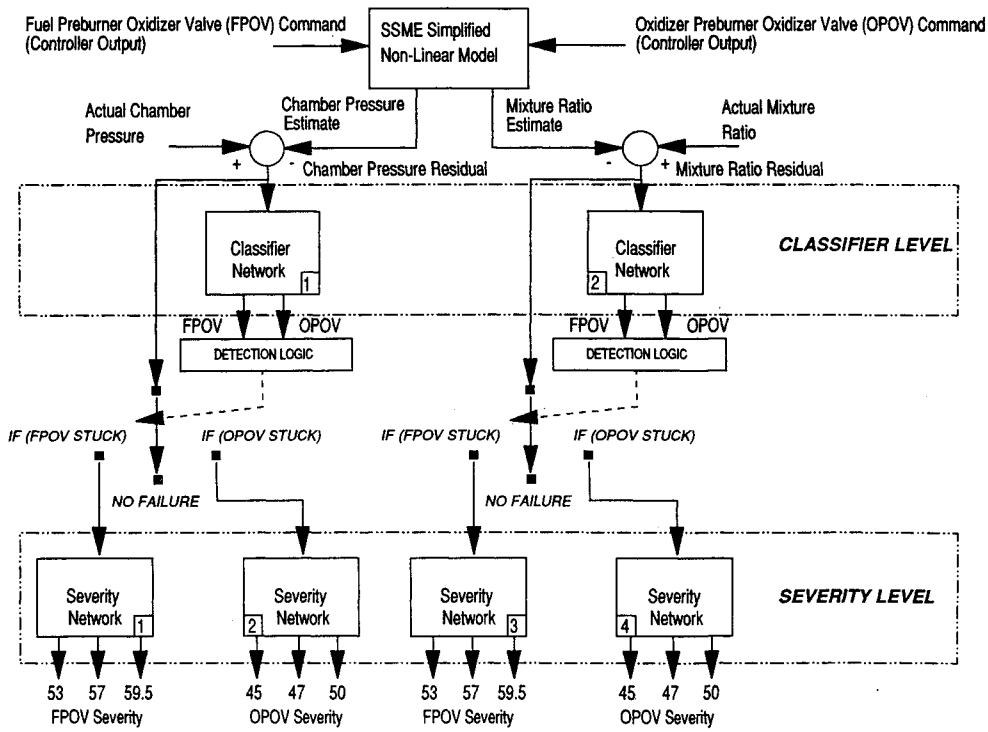
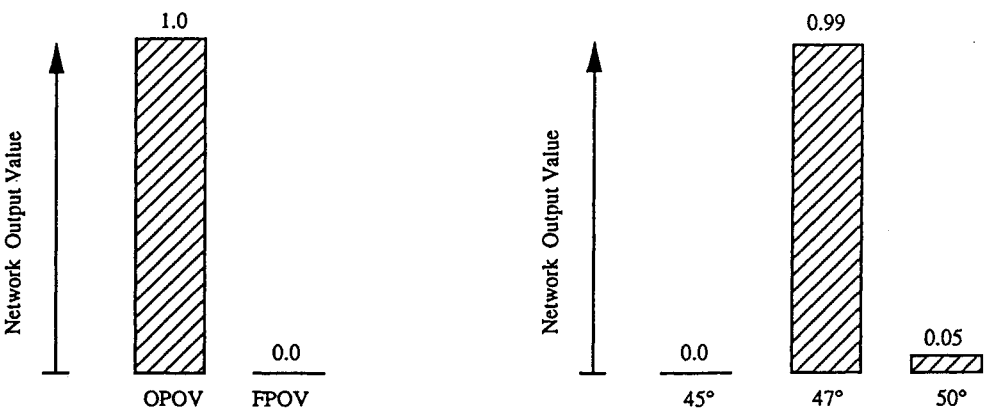


Fig. 5 Fault detection and diagnosis system architecture.



a) Fault type b) Severity

Fig. 6 Responses of the neural network to OPOV stuck at 47.5 deg.

Severity Level

The severity level consists of four networks, two associated with the chamber pressure residual (one for OPOV severity and the other for FPOV severity) and the other two with mixture ratio residual. Once again, each of these networks is a three-layer feedforward network where the weights are assigned using the back propagation algorithm.

Each of these networks consist of 200 input nodes, 20 hidden nodes, and 3 output nodes. Two of these networks receive the chamber pressure residual sequence as its input, whereas the other two networks receive the mixture ratio residual sequence as its input. The three output nodes correspond to the three severity levels (OPOV stuck at 45, 47, and 50 or FPOV stuck at 53, 57, and 59.5 deg).

The training is similar to that used in the classifier level. Once the input pattern is applied to a network, the node corresponding to the severity level of the input pattern is fully activated and the network weights learned through back propagation. For instance, the chamber pressure residual, corresponding to the 45-deg OPOV stuck valve fault scenario, is applied as the input and the output node corresponding to a 45-deg severity level is given an activation of 1. However, for the severity level networks, only those residual sequences that correspond to the appropriate network are used to train the network. For example, only mixture ratio residual sequences are used to train the two mixture ratio severity networks. As a result, each of these severity level networks is trained with three input representations rather than six as in the classifier level.

Results

Test data with severity levels not used in training were used to test both the classifier level and the severity level networks. The network architecture works as follows. Consider a fault scenario of the OPOV valve stuck at 47.5 deg. The two classifier level networks use their corresponding inputs (one uses the chamber pressure residual whereas the other uses the mixture ratio residual) to give an output activation corresponding to the fault (in this case the output node corresponding to the OPOV fault condition is activated in each of these two networks). Figure 6 illustrates the results obtained for this case.

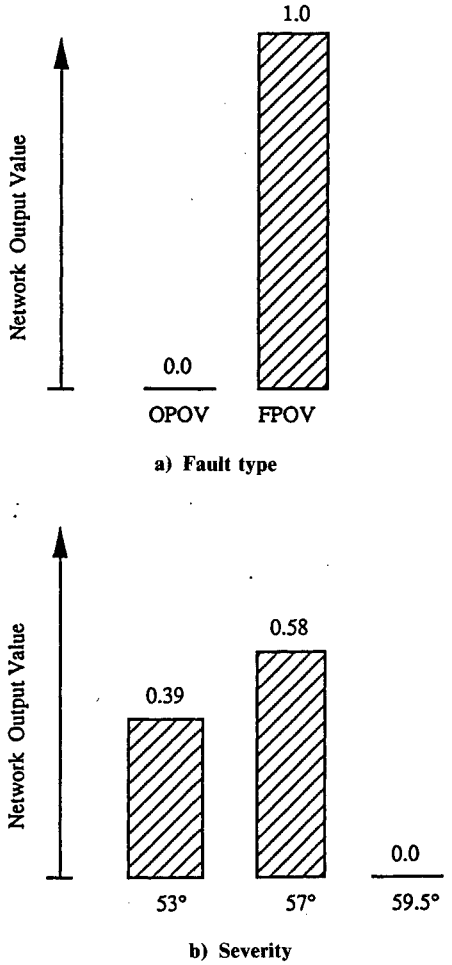


Fig. 7 Responses of the neural network to FPOV stuck at 55 deg.

Once the decision has been made as to what the fault is, the severity level networks are used to detect the severity of the fault. With the OPOV stuck condition, two severity level networks swing into action (each corresponding to one of the two residuals) to estimate the severity of the OPOV stuck valve. The other two severity level networks are dormant as they are trained to estimate the FPOV fault severity levels.

Consider another test case for the fault scenario of the FPOV valve stuck at 55 deg. Again the network architecture would work as in the previous case with the outputs of the classifier and severity level networks shown in Fig. 7. Note that in this case the network actually approximates the severity level since the weighted sum of the output activation magnitudes is 55.5. In both cases the networks correctly identify both the fault types and their severity.

Conclusions

A conceptual design of a model-based fault detection and diagnosis system has been presented for the Space Shuttle main engine. The design approach, which consists of process modeling, residual generation, and fault detection and diagnosis, is straightforward and resulted in a successful design. The engine is modeled using a discrete time, quasilinear state-space representation whose model parameters are determined by identification. Residuals generated from the model are used by a neural network to detect and diagnose engine component faults. Fault diagnosis is accomplished by training the neural network to recognize the pattern of the respective fault signatures. Preliminary results obtained from a nonlinear dynamic simulation of the Space Shuttle main engine for two failed oxidizer valve scenarios were presented and indicate that the developed approach can be used for fault detection and diagnosis. Unequivocal classifications of fault type were obtained along with accurate estimation of fault severity for scenarios not included in the training set.

Additionally, model accuracy results show that the developed model is an accurate and reliable predictor of the highly nonlinear and very complex engine. This ability of the model to generate accurate estimates combined with the sensitivity of the residual-based detection approach to engine faults results in two important features. First, it enables one to use relatively simple neural networks to successfully classify and estimate the severity of process faults. Simple networks trained for a specific fault pattern are more likely to be reliable fault indicators, an important feature in aerospace applications. Second, the amount of training required to learn the classification and severity patterns is minimal.

Finally, studies are needed, using the fault data generated with the nonlinear simulation and actual test data, to completely verify the validity of the fault detection concept presented in the paper. Also, additional research is required to incorporate more fault modes.

References

[1]Guo, T.-H., and Merrill, W. C., "A Framework for Real-Time Rocket Engine Diagnostics," NASA CP 3092, *Proceedings of the NASA Conference on Advanced Earth-to-Orbit Propulsion Technology,* May 1990, pp. 524–533.

[2]Merrill, W. C., and Lorenzo, C. F., "A Reusable Rocket Engine Intelligent Control," NASA TM 100963; see also AIAA Paper 88-3114, July 1988.

[3]Duyar, A., Guo, T.-H., and Merrill, W. C., "Identification of Space Shuttle Main Engine Dynamics," *IEEE Control Systems Magazine,* Vol. 10, No. 4, 1990, pp. 59–65.

[4]Cikanek, H. A., "Space Shuttle Main Engine Failure Detection," *IEEE Control Systems Magazine,* Vol. 6, No. 3, 1986, pp. 13–18.

[5]Willsky, A. S., "A Survey of Design Methods for Failure Detection in Dynamic Systems," *Automatica,* Vol. 12, No. 6, 1976, pp. 601–611.

[6]Isermann, R., "Process Fault Detection Based on Modelling and Estimation Methods," *Automatica,* Vol. 20, 1984, pp. 387–404.

[7]Gertler, J. J., "Survey of Model Based Failure Detection and Isolation in Complex Plants," *IEEE Control Systems Magazine,* Vol. 8, No. 6, 1988, pp. 3–11.

[8]Frank, P. M., "Fault Diagnosis in Dynamic Systems Using Analytical and Knowledge Based Redundancy—A Survey and Some New Results," *Automatica,* Vol. 26, No. 3, 1990, pp. 459–474.

[9]Anon., "Engine Balance and Dynamic Model," Rockwell International Corp., Rept. FSCM 02602, Spec. RL00001, Canoga Park, CA, Oct. 1981.

[10]Eldem, V., and Duyar, A. "Identification of Discrete Time Multivariable Systems; A Parameterization via α-Canonical Form," *Automatica* (submitted for publication).

[11]Duyar, A., Eldem, V., Merrill, W. C., and Guo, T.-H., "State Space Representation of the Open Loop Dynamics of the Space Shuttle Main Engine," *ASME Journal of Dynamic Systems Measurement and Control* (submitted for publication).

[12]Duyar, A., Eldem, V., Merrill, W. C., and Guo, T.-H., "A Simplified Dynamic Model of the Space Shuttle Main Engine," *ASME Journal of Dynamic Systems Measurement and Control* (to be published).

[13]Sage, A. P., and Melsa, J. L., *Estimation Theory with Applications to Communications and Control,* McGraw-Hill, New York, 1971, pp. 116–124.

[14]Dietz, W. E., Kiech, E. L., and Ali, M., "Jet and Rocket Engine Fault Diagnosis in Real Time," *Journal of Neural Network Computing,* Vol. 1, No. 1, 1989, pp. 5–18.