

NASA/TM—1998-208824

ARL-TR-1817  
AIAA-98-3605



# Neural Network-Based Sensor Validation for Turboshaft Engines

James C. Moller  
Miami University, Oxford, Ohio

Jonathan S. Litt and Ten-Huei Guo  
Lewis Research Center, Cleveland, Ohio

Prepared for the  
34th Joint Propulsion Conference and Exhibit  
cosponsored by the AIAA, ASME, SAE, and ASEE  
Cleveland, Ohio, July 13-15, 1998

National Aeronautics and  
Space Administration

Lewis Research Center

---

November 1998

## Acknowledgments

The authors would like to thank the U.S. Army Strategic and Advanced Computing Center, formerly the U.S. Army Artificial Intelligence Center for supporting this work.

Trade names or manufacturers' names are used in this report for identification only. This usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Available from

NASA Center for Aerospace Information  
7121 Standard Drive  
Hanover, MD 21076  
Price Code: A03

National Technical Information Service  
5285 Port Royal Road  
Springfield, VA 22100  
Price Code: A03

NEURAL NETWORK-BASED SENSOR VALIDATION FOR TURBOSHAFT ENGINES

James C. Moller  
 Department of Manufacturing Engineering  
 Miami University  
 Oxford, OH 45056

Jonathan S. Litt  
 U.S. Army Research Laboratory  
 Vehicle Technology Center  
 Lewis Research Center  
 Cleveland, Ohio 44135

Ten-Huei Guo  
 Member AIAA  
 Lewis Research Center  
 National Aeronautics and Space Administration  
 Cleveland, OH 44135

**ABSTRACT**

Sensor failure detection, isolation, and accommodation using a neural network approach is described. An autoassociative neural network is configured to perform dimensionality reduction on the sensor measurement vector and provide estimated sensor values. The sensor validation scheme is applied in a simulation of the T700 turboshaft engine in closed loop operation. Performance is evaluated based on the ability to detect faults correctly and maintain stable and responsive engine operation. The set of sensor outputs used for engine control forms the network input vector. Analytical redundancy is verified by training networks of successively smaller bottleneck layer sizes. Training data generation and strategy are discussed. The engine maintained stable behavior in the presence of sensor hard failures. With proper selection of fault determination thresholds, stability was maintained in the presence of sensor soft failures.

system unavailability may well have more dire consequences than the loss of some information. Sensor malfunctions are particularly pernicious in that they may lead to mission termination when all systems are in fact functioning properly.

Variables in complex systems are often correlated and this information can be used to detect, isolate and recover incorrect sensor readings. Dunia, et al.<sup>1</sup> have categorized sensor faults into four classes: total failure, drift, loss of precision, and fixed bias. An early technique used to perform detection, isolation, and recovery was Kalman filtering.<sup>2,3,4</sup> The success of this approach is dependent upon the fidelity of the engine model embedded in the filter.<sup>5</sup> Another successful approach uses linear fault models.<sup>6</sup> Additionally, Guo and Nurre<sup>7</sup> have demonstrated how lost or incorrect sensor values can be recovered using the remaining valid measurements through a neural network, a precursor to this work.

**NOMENCLATURE**

Variable	Description
$N_g$	% of gas generator design speed
$N_g'$	% of gas generator design speed indicated in cockpit
$N_p$	% of power turbine design speed
$P_{s3}$	Compressor exit pressure
$Q_s$	Shaft torque
$T_2$	Compressor inlet temperature
$T_{45}$	Interturbine gas temperature
$w_f$	Fuel flow rate

The fault detection, isolation, and accommodation (FDIA) method selected here was to integrate an autoassociative neural network into the path of information passed from the sensors back to the controlling entities. Responses to total failure and drift faults have been investigated here. Autoassociative neural networks have the general feature of being able to perform functional mappings. Kramer<sup>8</sup> and Saund<sup>9</sup> have introduced and applied a specific network architecture that is effective for filtering and fault isolation. Among the principal advantages of this approach over competing methods is that identification, isolation, and accommodation can be done with a single passage of information through the network.

**INTRODUCTION**

On-line fault detection and diagnosis is an area with great potential. The ability to detect and isolate a fault as it happens allows immediate decisions to be made about system availability and likelihood of mission completion. In a battlefield environment, for example,

**T700 SYSTEM OPERATION**

The T700 is a turboshaft engine. It is used in the Blackhawk and Apache helicopter airframes. The principal input from the cockpit is the percent of collective stick. Engine operation is controlled by the

---

This paper is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

QUANTITY	RECEIVING UNIT / PRINCIPAL PURPOSE	TYPE
% power turbine speed ( $N_p$ )	ECU / Trim	Proximity
Compressor inlet temperature ( $T_2$ )	HMU / Fuel schedule	Thermo-mechanical (bellows)
% gas generator speed ( $N_g$ )	HMU / Fuel schedule	Mechanical (fly weights)
% gas generator speed ( $N_g'$ )	Cockpit / Display	Inductive
Shaft torque ( $Q_s$ )	ECU / Load sharing	Proximity
Turbine inlet temperature ( $T_{45}$ )	ECU / Overtemperature protection	T/C
Compressor exit pressure ( $P_{s3}$ )	HMU / Fuel schedule	Pneumo-mechanical (bellows)
Trim signal	HMU / $N_p$ governing	

Table 1. Sensors involved in control for the T700 engine.

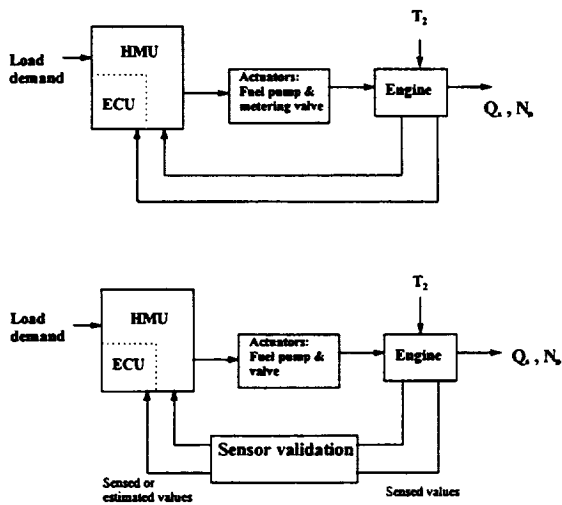


Figure 1. Schematic of the T700 closed loop system shown with and without sensor validation. Sensor validation is inserted in the feedback path to pass sensed values and estimates of faulted sensed values.

Electrical Control Unit (ECU) and Hydromechanical Unit (HMU). The principal ECU control functions are to guard against power turbine overspeed and overtemperature as well as to maintain the power turbine at 100% of design speed by sending a trim signal to the HMU. A schematic of the closed loop system is shown in Figure 1. Detailed descriptions of the system operation are given by Duyar, et al.,<sup>10</sup> Curran and Levine,<sup>11</sup> and Prescott and Morris.<sup>12</sup> The sensors which form the engine control system inputs are listed in Table 1.

Examples of the potentially catastrophic response of traditional controllers to sensor faults can be readily simulated. The simulated response of a T700 engine without FDIA to a sudden loss of sensed compressor exit pressure ( $P_{s3}$ ) is shown in Figure 2 in the form of

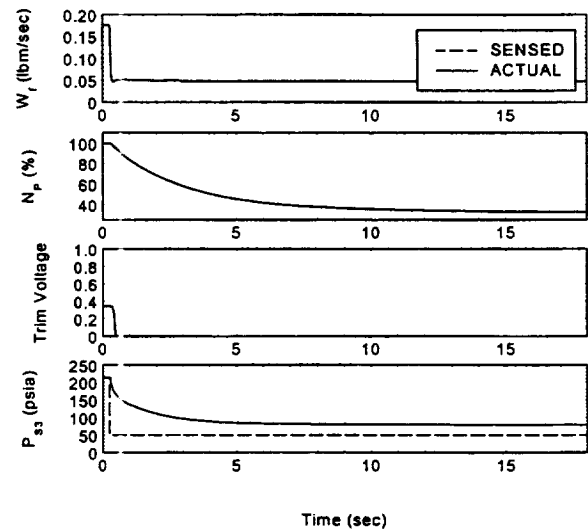
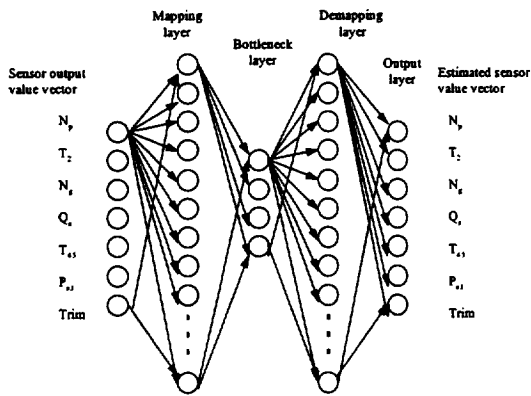


Figure 2. Response of a simulated T700 closed loop system having no sensor validation scheme to a hard fault in sensed compressor exit pressure ( $P_{s3}$ ) to 50 psia at 0.25 seconds.

histories of critical engine parameters. The immediate effect of a loss in sensed compressor exit pressure is a loss in fuel flow ( $w_f$ ) from the hydromechanical unit (HMU). Sensed  $P_{s3}$  drives a lever mechanism in the HMU. The mechanism has the function of creating the appropriate fuel flow past a metering valve by multiplying scheduled ( $w_f/P_{s3}$ ) by sensed  $P_{s3}$ . An abnormally low  $P_{s3}$  reduces the flow area in the metering valve. As the rate of enthalpy release to the turbines drops, the speeds of both power turbine and gas generator spools decrease. This leads to a decrease in the actual  $P_{s3}$ . The decrease in gas generator speed is sensed by the HMU. Shortly after the fault, it switches from scheduling fuel on a trim schedule to an acceleration schedule in response to the decreasing speed. Because the sensed  $P_{s3}$  does not rise as it would in normal engine operation, the fuel metering valve



**Figure 3. Schematic representation of the autoassociative neural network architecture applied to sensor validation. Nodes in the demapping layers are typically nonlinear. Output nodes are typically linear. Bottleneck layer nodes may be either linear or nonlinear.**

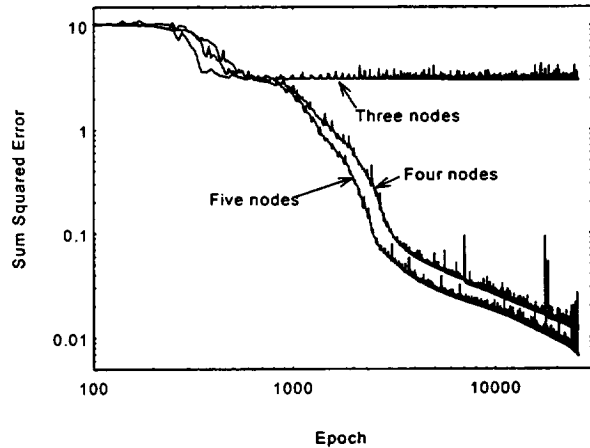
output is reduced to the engine minimum flow rate and power turbine speed drops below 40% of design speed. The corresponding shaft torque, which keeps the helicopter aloft, drops to near zero.

### NEURAL NETWORK

The autoassociative neural networks described by Kramer<sup>8</sup> and Saund<sup>9</sup> have three hidden layers. The first demaps the input data. The second layer, known as the bottleneck layer reduces the number of values passed to the system intrinsic degrees of freedom. The third layer maps this information to an output vector in the same space as the input vector. A schematic of the network architecture is shown in Figure 3. These networks have the property of mapping the input vector onto the nearest point on the functional surface generated by training. Errant elements of input vectors, then, become mapped to a vector with a smaller error. By using nonlinear processing elements in the mapping and demapping layers, nonlinear functionality can be captured during training. The bottleneck layer may be either linear or nonlinear. The output layer is typically linear.

#### Selection of bottleneck layer size

A critical value in applying the dimensionality reduction approach to a particular problem is the number of nodes in the bottleneck layer. If the number of nodes is greater than the system degrees of freedom, errant sensor information may be unnecessarily passed through the bottleneck layer. If the number is smaller than the degrees of freedom, the network outputs cannot adequately reconstruct the system behavior due to insufficient information having been passed through



**Figure 4. Error histories during training for different bottleneck layer sizes. The constancy for the three-noded case indicates loss of information through the network while the smaller error and steady decline in the four-noded case indicates a match between the system degrees of freedom and the bottleneck size.**

the bottleneck layer. One approach to finding the number of system degrees of freedom would be to use the equations governing the system to find the functional relations among them. Another is to examine and compare the performance of various network architectures. This can be particularly useful if (as is the case with the T700 engine system) more than one function is used to characterize these relationships depending upon the location within the operating envelope.

Networks having different numbers of bottleneck nodes were trained on the same data set for the same number of epochs. It is anticipated that, below a certain number of bottleneck nodes, the sum of squared errors will have a distinctly larger minimum on account of the loss of critical information passing through the bottleneck. A comparison of the sums of squared errors as a function of epoch number is shown in Figure 4. Several trainings of each network architecture were performed using random initial guesses for nodal weights and biases in order to avoid training to a local error minimum. Based on these results, the appropriate bottleneck size was determined to be four nodes. Although the five-noded case had smaller error after the selected number of epochs, the error trend suggests the four-noded case could have been trained further to achieve the same error. Additionally, because of the similarity in the results among the four- and five-noded cases and the marked difference from the three noded case, it appears likely that the four-noded case passes the minimum amount of information for construction of an input vector estimate.

### Training Approach

The selected network is both nonlinear and multilayered. As such, it will likely have local minima in its error surface. To ensure convergence to a global minimum, large momentum was included. Nyguen-Widrow<sup>13</sup> initial guesses for weights and biases and variable learning rate were also included to speed up convergence. Training data were generated from runs of a closed loop component-based real time nonlinear simulation.<sup>14</sup> Engine inlet temperature and torque demands ranged from  $-40^{\circ}$  to  $74^{\circ}$  F and 116 to 410 ft-lb, respectively. This covers most of the engine operating envelope. The data were normalized by their nominal full scale values. Target vectors of normalized sensed steady state engine outputs were selected to resolve the operating envelope finely enough that the incremental change in each parameter was smaller than the fault threshold planned for it. Each training vector contained the sensed values of the parameters listed in Table 1.

Training was performed in two steps. First, preliminary weights and biases were generated by training on a data set of 88 unfaulted engine output vectors for approximately 12,000 epochs. These weights and biases were used as initial guesses for a second training round in which vectors containing both faulted and unfaulted values were presented as input. In this set of 704 vectors, 88% contained a fault in one sensed value. With the exception of  $N_p$ , these faults were randomly biased by 10 to 100% of the unfaulted values. Because  $N_p$  is to remain nearly constant over the entire T700 operating envelope, the random perturbations in it ranged up to 10%. The second training required approximately 25,000 epochs. At the termination of training, the root-mean-squared error between input and output vector elements was approximately 4%.

### Integration into T700 Simulator

The test bed was adapted from the simulation used to generate training data.<sup>14</sup> Subroutines for fault injection were inserted at points where sensed values were calculated. Up to three faults of preset type can be injected at scheduled times. The vector of sensed values enters the network at each time step, every 0.006 seconds. If any estimated value differs from the corresponding input value by more than a preset threshold, the estimate rather than the sensed value is passed to the controller. For all subsequent time steps, the network takes as input the estimated values for any faulted sensors from the previous time step. Thus a sensor, once determined to be faulted, is removed from future calculations. A schematic of the fault creation and checking function integrated into the flow of information among the program modules representing the engine, HMU, and ECU is shown in Figure 5.

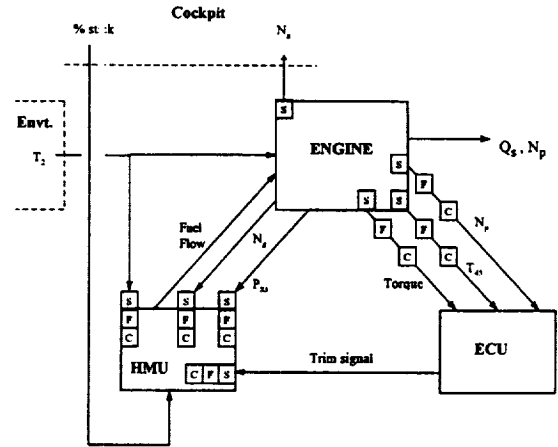


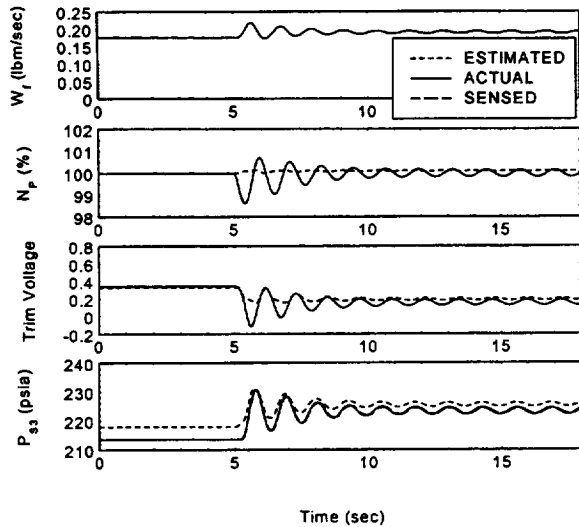
Figure 5. Schematic display of the flow of information among the T700 engine, electrical control unit, and hydromechanical unit. S, F, and C represent the sensor, fault injection, and fault checking and substitution, respectively.

### SIMULATION RESULTS

#### Network matching of engine dynamic response

The network training results were based upon steady state engine behavior. In order to test the network's ability to predict engine dynamic behavior, the network output based upon sensed values was compared with simulated engine quantities during and after a sudden change in load demand. The results are shown in Figure 6 in which the collective stick is ramped from 50 to 53 percent between 5 and 5.1 seconds. The discrepancy prior to 5 seconds is the result of modeling error in the network training. Most traces agree to approximately the same level following the stick movement. The principal exceptions are power turbine speed and compressor inlet temperature. The network was trained on constant power turbine speed. As such, the weights and biases pass a nearly constant power turbine speed estimate over a wide range of input values. Temperature data included a wide range, but information about the typical constancy of temperature was not part of the training data.

The response of an unfaulted engine to a change in demand is to begin hunting for a new equilibrium point at which the engine is kept in trim. To accomplish this, the trim signal goes through a fluctuation large enough to cause the power turbine speed to overshoot. The decaying oscillations are centered about the steady state values for the new demand. Also, several other parameters fluctuate at the fuel control hunting frequency. These fluctuations are not all in phase with each other, however. No information on phase relations is included in the training data set. The network, by training on steady state trim values, apparently does not



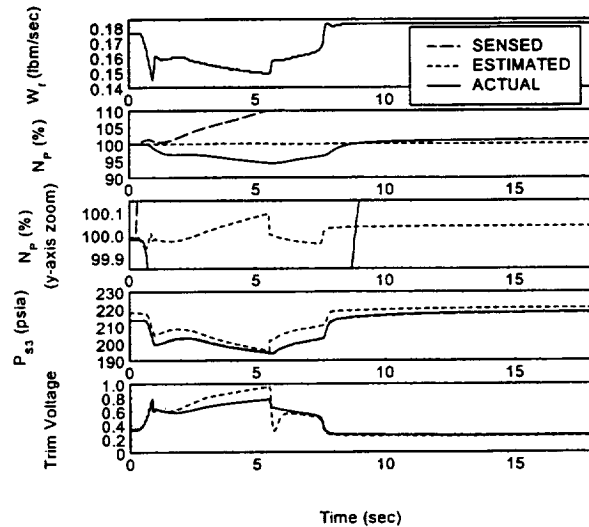
**Figure 6. Response of a healthy simulated T700 closed loop system and the trained network to a sudden change in load demand. The collective stick was ramped from 50 to 53% from 5.0 to 5.1 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values.**

include an essential feature of the dynamic character of the ECU. The network estimate maintains the phase of the trim signal oscillation. The amplitude of oscillation, however, is much smaller. This will be shown later to have important consequences when the trim signal is determined to be faulted.

#### Effect of threshold level

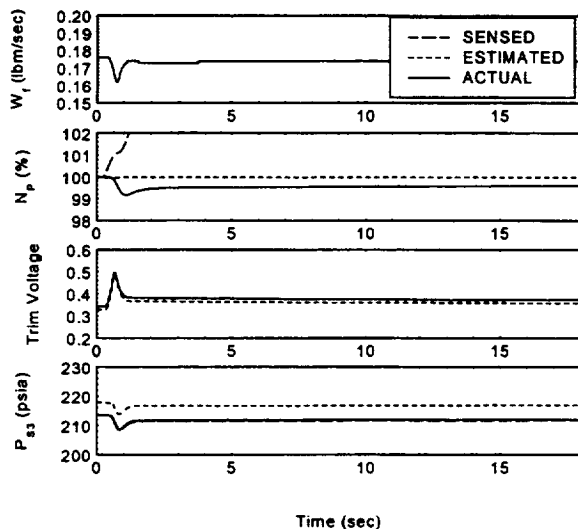
Thresholds were initially selected to be on the order of twice the modeling error. The thresholds selected have a large bearing on the accuracy of the fault detected. If a fault level is selected too narrowly for a given parameter, the result can be a misdiagnosis, unnecessary removal of valid information from the control loop, and degradation of the FDIA capability. Similarly, if a fault level is selected too large for a given parameter, the result can be a misdiagnosis in another parameter. Two cases were run to illustrate both the latter behavior and proper behavior.

When the fault injected is a ramp type, the gradual movement of the sensed value away from the actual value may lead to gradual changes in engine condition and in the corresponding correctly-sensed values. There are some combinations of fault thresholds for which the logic algorithm would correctly recognize the existence of a fault but incorrectly identify the sensor in which it was occurring. An example is shown in Figure 7. In this case, a ramp fault was injected to power turbine speed. The power turbine threshold was set to 10%. The ECU interprets the fault as an



**Figure 7. Response of the T700 system including sensor validation to a ramp fault in percent power turbine speed of 3%/second. Tolerance thresholds were 10% in power turbine speed and 0.1 volts in trim signal. The fault is initiated at 0.25 seconds. A spurious fault is detected in trim signal at 0.89 seconds. A correct fault is detected in power turbine speed at 5.52 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values. Following substitution, estimated and sensed trim signal values coincide.**

overspeed and reacts by changing the trim signal sent to the HMU. Fuel flow is correspondingly reduced and the actual power turbine speed decreases. The first fault detected is at 0.89 seconds in the trim signal. This is a spurious fault. The switching to the estimated trim value leads to a brief cessation of the power turbine deceleration. As the power turbine sensed speed continues to deviate further from the design point, the logic algorithm correctly detects the fault at 5.52 seconds and replaces the sensed value with the estimate. The value of power turbine speed passed to the ECU is now within fractions of a percent of the design speed. This substitution leads to a sudden shift in the estimated trim signal passed by the network to the HMU. Fuel flow is correspondingly increased and the power turbine accelerates. From 5.52 to 7.51 seconds, the sensed power turbine speed gradually declines from a value above the trim reference speed to a value below it. At 7.51 seconds, the ECU detects enough of an underspeed to act to retrim the engine. Again, the HMU continues to be passed the estimated trim signal. The result is increased acceleration of the power turbine. As the turbine accelerates, the estimated turbine speed rises above the trim reference speed and then remains constant. Although the ECU perceives an overspeed, no further corrective action is taken because the error is within the designed trim deadband.



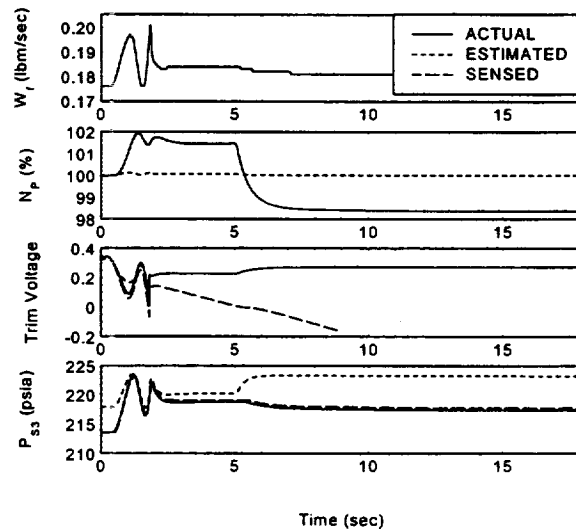
**Figure 8. Response of the T700 system including sensor validation to a ramp fault in percent power turbine speed of 3%/second. Tolerance thresholds were 1% in power turbine speed and 0.1 volts in trim signal. The fault is initiated at 0.25 seconds. A correct fault is detected in power turbine speed at 0.61 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values.**

A second case was run with a smaller fault tolerance (1%) for percent power turbine speed. A single correct power turbine speed fault was detected at 0.61 seconds. These results are shown in Figure 8. When the estimated power turbine speed value is substituted for the sensed value, a small underspeed is detected. The ECU re-trims the engine based on this and successive estimated values. At steady state, the actual and estimated power turbine speeds differ by 0.4%. This bias exists because the estimated and design power turbine speeds agree closely enough that the ECU concludes the engine is in trim.

**Engine response to demand changes in presence of a sensor fault**

Once a sensor fault has been detected and accommodated, the engine will likely be expected to appropriately respond to subsequent changes in load demand. This was examined for two fault cases: the first in the trim signal and the second in compressor exit pressure.

A negative ramp fault was injected for the trim signal case. These results are shown in Figure 9. Prior to fault detection, the HMU interprets this signal as a demand to increase fuel flow to compensate for a power turbine underspeed. As a result, the power turbine actually

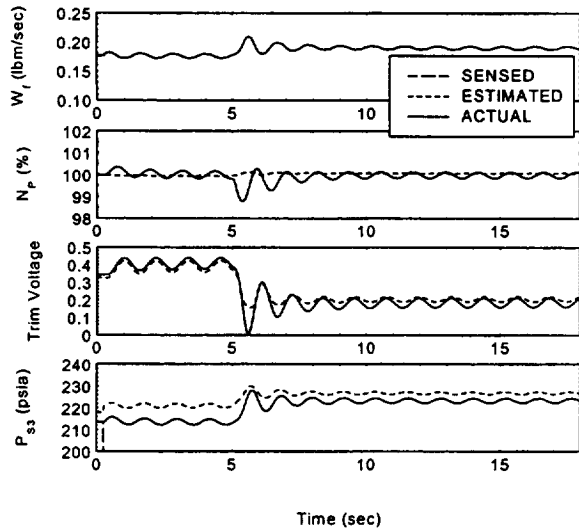


**Figure 9. Response of the T700 system including sensor validation to a ramp fault in trim signal of -0.1 volts/second followed by a change in load demand. The trim signal tolerance threshold was 0.1 volts. The fault is initiated at 0.25 seconds. A correct fault is detected in trim signal at 1.82 seconds. The collective stick was ramped from 50 to 53% from 5.0 to 5.1 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values. Following substitution, estimated and actual trim signal values coincide.**

accelerates. When the fault is detected in the trim signal, the substituted value is larger and constant. The result is that the power turbine equilibrates to a steady overspeed value. At 5 seconds, the torque load on the output shaft is increased. The power turbine decelerates to below the design speed. For the power turbine to return to trim, the dynamic character of the trim signal is very important. As mentioned earlier, the dynamic character of the trim signal during load demand changes is not captured by the steady state network training. The estimated trim signal does not increase enough to bring the engine back to trim. While a steady state is reached, the power delivered does not increase.

A hard fault in compressor exit pressure at 0.25 seconds is examined in the second case. The results are shown in Figure 10. The fault is immediately detected. The substituted pressure is larger than the actual value. This has an immediate effect of increasing fuel flow and power turbine speed shortly thereafter. The ECU then acts to bring the engine back to trim. Due to the positive bias between actual compressor pressure and estimated compressor pressure, however, the system goes into a limit cycling mode as the HMU overschedules fuel flow and the ECU acts in correction.





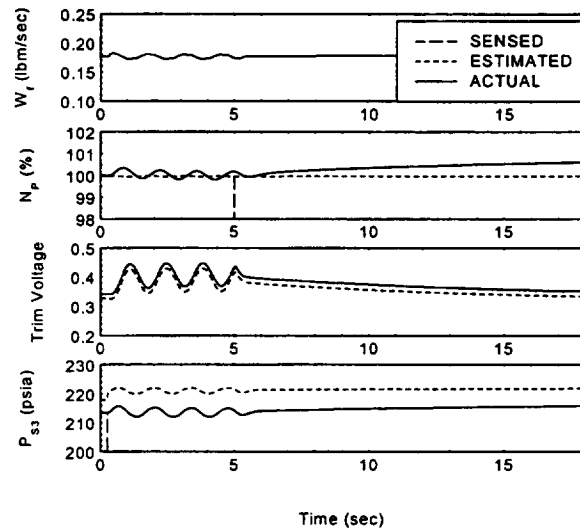
**Figure 10. Response of the T700 system including sensor validation to a hard fault in sensed compressor exit pressure ( $P_{33}$ ) to 50 psia at 0.25 seconds followed by a change in load demand. A correct fault is detected in  $P_{33}$  at 0.252 seconds. The collective stick was ramped from 50 to 53% from 5.0 to 5.1 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values.**

It should be noted here that the second case is identical to the one described to demonstrate the system response to a sensor fault without validation (Figure 2). The sensor validation scheme applied here was effective at maintaining stable engine behavior where the consequences would otherwise likely have been catastrophic.

When load demand is increased in the presence of a compressor exit pressure sensor fault, the result with sensor validation (Figure 10) is closer to healthy engine behavior (Figure 6) than is the trim signal sensor fault case (Figure 9). The power turbine briefly underspeeds and the system parameters oscillate as the ECU hunts for a new equilibrium. The principal feature distinguishing the post acceleration behavior in this case from the healthy engine is that the engine limit cycles whereas in the healthy case the oscillations decayed. The mean power delivered in the two cases, however, agree to within fractions of a percent.

#### Multiple faults

Multiple sensor faults are potentially more difficult to accommodate in that, with each successive fault, the amount of information available comes closer to the system's minimum number of degrees of freedom. The test case selected included two faulty sensors: compressor exit pressure and power turbine speed. The



**Figure 11. Response of the T700 system including sensor validation to a hard fault in sensed compressor exit pressure ( $P_{33}$ ) to 50 psia at 0.25 seconds followed by a hard fault in sensed power turbine speed ( $N_p$ ) at 5.0 seconds. A correct fault is detected in  $P_{33}$  at 0.252 seconds. Some sensor lags are small enough to make sensed values indistinguishable from actual values.**

signal from the former is used by the HMU while that of the latter is used by the ECU. The results are shown in Figure 11. Similar to previous cases, compressor exit pressure undergoes a hard fault at 0.25 seconds. The limit cycling response is as before. At 5.0 seconds, a hard power turbine speed sensor fault of 50% of actual speed is injected. It is immediately detected. Substitution with estimated speed distinctly reduces the oscillations in system parameters because the speed error inferred by the ECU is small and comparatively steady. The estimated power turbine speed is below the design speed. The ECU acts to accelerate the power turbine until the estimated value agrees with the design value. At steady state, the power turbine has in fact oversped by 0.8 %.

Other fault combinations have yet to be tested. It is expected that combinations which include the trim signal will be problematic due to the inability of the present FDIA to recreate the necessary dynamic trim signal behavior.

#### CONCLUDING REMARKS

An autoassociative neural network has been created that maps normal T700 engine behavior to within tolerance thresholds over a wide range of torque demands and engine inlet conditions. This network has been integrated into a component-based real-time simulation and is effective at detecting sensor faults and substituting appropriate estimated sensor values given

an appropriate selection of fault thresholds. Among the issues yet to be addressed is the selection of tolerances to adequately identify faults from each sensor over a variety of flight conditions. Generally, the integration of a neural network-based sensor validation scheme into the closed loop engine operation resulted in stable engine behavior in response to faults. Hard sensor faults in any one of the sensed quantities were correctly detected and accommodated. System response in ramped cases depended upon the comparative sizes of fault detection thresholds. Unaccommodated, these sensor faults would have been catastrophic to the engine and airframe. The behavior around the time of variable substitution of the system with the integrated neural network is as expected in terms of engine and controller response. While the network embedded in the control has been shown to accommodate multiple sensor faults, this is not necessarily true of all multiple fault combinations.

Future work might involve network training to include engine dynamic response and discrimination between sensor and system faults. Strategies for threshold selection might also be investigated, with respect to both minimization of misdiagnoses and engine-to-engine variation.

#### ACKNOWLEDGEMENTS

The authors would like to thank the U. S. Army Strategic and Advanced Computing Center, formerly the U. S. Army Artificial Intelligence Center, for supporting this work.

#### REFERENCES

1. Dunia, R., S.J. Qin, T.F. Edgar, and T.J. McAvoy, "Identification of Faulty Sensors Using Principal Component Analysis," *AIChE J*, vol. 42, no. 10, October 1996.
2. Deckert, J.C., N.N. Desai, J.J. Deysy, and A.S. Willsky, "F-8 DFBW Sensor Failure Identification Using Analytic Redundancy," *IEEE Trans. on Automatic Control*, AC-22, pp. 795-803, 1977.
3. DeLaat, J.C., and W.C. Merrill, "A Real-Time Implementation of an Advanced Sensor Failure Detection, Isolation, and Accommodation Algorithm," NASA TM 83553, 1984.
4. Merrill, W.C., J.C. DeLaat, and W.M. Bruton, "Advanced Detection, Isolation, and Accommodation of Sensor Failures: Real-Time Evaluation," *AIAA J. of Guidance, Control, and Dynamics*, vol. 11, no. 6, 1988.
5. Guo, T.-H., J. Musgrave, and C.S. Lin, "Neural Network Based Sensor Validation for Reusable Rocket Engines," 1995 American Control Conference, Seattle, WA, June 21-23, 1995.
6. Litt, J., M. Kurtkaya, and A. Duyar, "Sensor Fault Diagnosis for a T700 Turboshift Engine," *J. Guidance, Control, and Dynamics*, vol. 18, no. 3, pp. 640-642, 1994.
7. Guo, T.H. and J. Nurre, "Sensor Failure Detection and Recovery by Neural Networks," NASA TM 104484, 1991.
8. Kramer, M.A., "Autoassociative Neural Networks," *Computers in Chem Engrg.*, vol. 16, no. 4 pp. 313-328, 1992.
9. Saund, E., "Dimensionality-Reduction Using Connectionist Networks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, no. 3, March 1989.
10. Duyar A., Z. Gu, and J.S. Litt, "A Simplified Model of the T700 Turboshift Engine," *J. of the American Helicopter Society*, Oct. 1995, pp. 62-70.
11. Curran, J.J., and A.M. Levine, "T700-GE-700 Engine Design Report," vol. 7, R78AGE032, General Electric Company, 1978.
12. Prescott, W.E., and H.F. Morris, "T700 Training Guide." SEI-418, General Electric Company, 1987.
13. Nguyen, D., and B. Widrow, "Improving the Learning Speed of 2-Layer Neural Networks by Choosing Initial Values of the Adaptive Weights," *Proc. of the IJCNN*, vol. 3, pp. 21-26, 1990.
14. Ballin, M.G., "A High Fidelity Real-Time Simulation of a Small Turboshift Engine," NASA TM 100991, July, 1988.



REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE November 1998	3. REPORT TYPE AND DATES COVERED Technical Memorandum		
4. TITLE AND SUBTITLE Neural Network-Based Sensor Validation for Turboshaft Engines			5. FUNDING NUMBERS WU-519-30-53-00 1L161102AH45	
6. AUTHOR(S) James C. Moller, Jonathan S. Litt and Ten-Huei Guo				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) NASA Lewis Research Center Cleveland, Ohio 44135-3191 and U.S. Army Research Laboratory Cleveland, Ohio 44135-3191			8. PERFORMING ORGANIZATION REPORT NUMBER E-11432	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) National Aeronautics and Space Administration Washington, DC 20546-0001 and U.S. Army Research Laboratory Adelphi, Maryland 20783-1145			10. SPONSORING/MONITORING AGENCY REPORT NUMBER NASA TM-1998-208824 ARL-TR-1817 AIAA-98-3605	
11. SUPPLEMENTARY NOTES Prepared for the 34th Joint Propulsion Conference and Exhibit sponsored by the AIAA, ASME, SAE, and ASEE, Cleveland, Ohio, July 13-15, 1998. James C. Moller, Miami University, Oxford, Ohio 45056; Jonathan S. Litt and Ten-Huei Guo, NASA Lewis Research Center. Responsible person, Jonathan S. Litt, organization code 0300, (216) 433-3748.				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Unclassified - Unlimited Subject Category: 63  This publication is available from the NASA Center for AeroSpace Information, (301) 621-0390.			12b. DISTRIBUTION CODE Distribution: Nonstandard	
13. ABSTRACT (Maximum 200 words) Sensor failure detection, isolation, and accommodation using a neural network approach is described. An autoassociative neural network is configured to perform dimensionality reduction on the sensor measurement vector and provide estimated sensor values. The sensor validation scheme is applied in a simulation of the T700 turboshaft engine in closed loop operation. Performance is evaluated based on the ability to detect faults correctly and maintain stable and responsive engine operation. The set of sensor outputs used for engine control forms the network input vector. Analytical redundancy is verified by training networks of successively smaller bottleneck layer sizes. Training data generation and strategy are discussed. The engine maintained stable behavior in the presence of sensor hard failures. With proper selection of fault determination thresholds, stability was maintained in the presence of sensor soft failures.				
14. SUBJECT TERMS Sensor; FDIA; Neural network; Turboshafths			15. NUMBER OF PAGES 14	
			16. PRICE CODE A03	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT	