

A Function Approximation Approach to Anomaly Detection in Propulsion System Test Data

Bruce A. Whitehead[†]

University of Tennessee Space Institute

W. Andes Hoyt[‡]

ERC, Inc.

October 24, 1995

Nomenclature

f = function to be approximated by the neural network

g = Gaussian bar basis function

M = number of basis functions per input variable

N = number of inputs to the neural network

\mathbf{x} = vector of inputs to the neural network

μ = basis function center

σ = basis function width

Subscripts

j = which basis function

k = component of input vector

*Presented in greater detail to the 29th AIAA/SAE/ASME/ASEE Joint Propulsion Conference, Monterey, CA, June 1993.

[†]Associate Professor of Computer Science, University of Tennessee Space Institute, Tullahoma TN 37388-8897

[‡]Project Manager, ERC, Inc., P.O. Box 417, Tullahoma TN 37388

Introduction

A fully-instrumented ground test of a propulsion system typically generates a very large quantity of data. We propose a screening system to identify the relatively small volume of data which is unusual or anomalous in some way. Limited and expensive human analysis can then focus on this small volume of unusual data.

A liquid-fueled rocket engine, such as the SSME, is a system with physical and informational interfaces to other systems. A relatively clean interface is defined by a boundary drawn around the engine controller and the system it controls (valves, turbopumps, etc.). The external influences which affect the state of the system include the informational interface of commands given to the controller, as well as physical interfaces such as the fuel inlet, the oxidizer inlet, and the venting and repressurization interfaces to both the fuel and oxidizer tanks. The following 14 measurements at these interfaces, together with 4 turbopump configuration variables, form the 18 inputs to our neural network model:

- Parameter 287, Commanded Main Combustion Chamber Pressure, PSI
- Parameter 819, Engine Fuel Inlet Pressure #2, PSI
- Parameter 821, Engine Fuel Inlet Pressure #1, PSI
- Parameter 827, Engine Fuel Inlet Pressure #3, PSI
- Parameter 830, Fuel Bleed Interface Pressure, PSI
- Parameter 835, Fuel Pressurization Interface Pressure, PSI
- Parameter 836, Fuel Pressurization Venturi Inlet Pressure, PSI
- Parameter 837, Fuel Pressurization Venturi Delta Pressure, PSI
- Parameter 858, Engine Oxidizer Inlet Pressure #2, PSI
- Parameter 859, Engine Oxidizer Inlet Pressure #1, PSI
- Parameter 860, Engine Oxidizer Inlet Pressure #3, PSI
- Parameter 878, Heat Exchanger Interface Pressure, PSI
- Parameter 881, Heat Exchanger Venturi Inlet Pressure, PSI

- Parameter 883, Heat Exchanger Venturi Delta Pressure, PSI
- Installed Low-Pressure Fuel Turbopump (2109R6 or 2218R2)
- Installed High-Pressure Fuel Turbopump (4604, 4406R1, or 6108)
- Installed Low-Pressure Oxidizer Turbopump (2106R2 or 2118)
- Installed High-Pressure Oxidizer Turbopump (0810, 2030, 2315R1, 4108, or 9409)

Under nominal steady-state operating conditions the behavior of the engine is, in principle, determined by what transpires at these interfaces. To the extent that the SSME is a deterministic system, there would exist some function f which predicts the nominal steady-state value of any desired engine parameter from measurements at all interfaces at the system boundary, e.g., from the measurements listed above. A neural network can be trained to approximate f . Since this study attempts to predict only steady-state parameter values, transients due to SSME power-level changes and to fuel and oxidizer repressurization were automatically removed from the time series inputs to the neural network.

Our approach to identifying anomalous behavior is to train the neural network not to classify anomalies, but to predict *nominal* values of engine parameters. Representative anomalous data is not needed for training because the neural network is not trying to predict anything about anomalies. Rather, the neural network is trying to predict, as accurately as possible, the nominal steady-state value of each engine parameter under the given interface conditions at each point in time. The detection of anomalies depends upon surrounding the neural network's predictions with a nominal confidence interval. To achieve an acceptably low false alarm rate, the confidence interval is set to five standard deviations above and below the predicted value. Measured values falling outside this confidence interval are flagged as anomalous.

Method

Three function-approximation techniques were investigated: (i) a novel neural network architecture based on Gaussian bar basis functions (ii) a standard backpropagation neural network;¹

and (iii) standard linear regression. (i) and (ii) are nonlinear function approximation techniques, in contrast to (iii). The Gaussian bar basis function architecture was investigated because of two difficulties in the application of backpropagation to real problems: the slowness of the gradient descent optimization when multiple layers are involved, and the possibility of getting stuck on a local minimum.

To model a function f with N input variables, we propose a set of semi-local Gaussian bar basis functions^{2,3} spaced equally along each dimension of the input data. The set of possible values of each input variable x_k is covered by a set of M basis functions

$$g_{jk}(x_k) = \exp \left[-\frac{(x_k - \mu_{jk})^2}{2\sigma_k^2} \right], j = 1, \dots, M \quad (1)$$

where the M centers $\mu_{1k}, \dots, \mu_{Mk}$ are equally spaced along the interval of x_k covered by the training data. Altogether, there are accordingly MN basis functions g_{jk} , $j = 1, \dots, M$, and $k = 1, \dots, N$. This proposed Gaussian bar basis function network approximates a real-valued function $f(x)$ as a linear combination of these basis functions using a gradient descent learning rule.⁴

NASA engineers selected two tasks for our study: (a) to detect anomalies in engine parameter 42 for SSME Tests 901-671, 901-672, and 901-673; and (b) to detect anomalies in engine parameter 221 for SSME Tests 902-548, 902-549, and 902-550. The Gaussian bar basis function network was implemented with $M = 60$ and $N = 18$; backpropagation with 1-8, 10, 12, and 16 middle-layer nodes.

Results and Discussion

As expected, the Gaussian bar basis function technique showed more rapid training than backpropagation in both tasks. This result was expected because only a small fraction of the weights are adjusted for each training example.

Figures 1 and 2 show the nominal predictions made by the Gaussian bar basis function

neural network for these two SSME tests.

In SSME test 901-672 all three function-approximation techniques correctly identified the anomaly at the right time. This lends support to our hypothesis that nominal behavior of the SSME can be predicted from the external influences crossing the system boundary. The prediction accuracy allows the fluctuation due to the anomaly to be clearly distinguished from the noise fluctuations that nominally arise inside the SSME and due to measurement errors. The anomaly is detected at approximately 5 times the RMS noise level.

In SSME test 902-549, the Gaussian bar basis function network constructed a much tighter confidence interval than the other two techniques, and hence detected the anomaly at 20 times the RMS noise level; linear regression at 4-6 times the RMS noise level; and backpropagation at only 2-3 times the RMS noise level.

The chief appeal of the function-approximation approach to anomaly detection is that it is not limited to detecting specific, foreseen classes of anomalies, in contrast with most other types of neural network and expert system approaches. Instead, training requires nominal data only, and anomaly detection is based on nominal confidence intervals. Based on the results we have presented, it appears that the function-approximation approach, and its implementation using basis functions, merits more extensive investigation of its practicality for screening large amounts of propulsion system test data. The approach needs to be tested on a larger scale, and it needs to be extended to cover transient as well as steady-state data. Both types of further work are in progress and will be reported at a later date.

Acknowledgements

A portion of this work was supported by NASA Contract NAS8-39184. We thank Timothy Choate, JoAnne Malone, Michael Whitley, Catherine McLeod, Gary Lyles, Marc Neely, Eric Sander, and Jeff Cornelius for their assistance.

References

¹Rumelhart, D. E., Hinton, G. E., and Williams, R. J., “Learning internal representations by error propagation,” in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition Volume 1: Foundations* The MIT Press, Cambridge, MA, 1986, pp. 318–364.

²Hartman, E. and Keeler, J. D., “Semi-local units for prediction,” in *IJCNN-91-Seattle: International Joint Conference on Neural Networks*, IEEE, 1991, Vol. II, pp. 561–566, IEEE.

³Hartman, E. and Keeler, J. D., “Predicting the future: Advantages of semilocal units,” *Neural Computation*, Vol. 3, 1991, pp. 566–578.

⁴Widrow, B. and Hoff, M. E., “Adaptive switching circuits,” in *1960 WESCON Convention*, 1960, pp. 96–104.

Figure Captions

Fig. 1 Neural network screening of data from SSME test 901-672, which was determined by NASA to have an anomaly at 96 seconds indicated in the Fuel Preburner Oxidizer Valve Actuator Position. The dotted lines show a nominal confidence interval of plus or minus 5 times the nominal RMS prediction error. The middle dotted line is the neural network’s prediction. (Sloping segments in the accompanying power-level plot represent transient regions in which no predictions were made. Different steady-state segments are connected by straight lines for visual continuity.) The anomaly is indicated when the measured value of this valve position (solid line) moves outside the nominal confidence interval (bounded by dotted lines).

Fig. 2 Neural network screening of data from SSME test 902-549, which was determined by NASA to have an anomaly at 120 seconds indicated in the Pressure Change in the POGO Surge Suppression System. Confidence intervals as in Figure 1.