

**The Use of Neural Networks for
Sensor Failure Detection During
the Operation of a Control System**

By

**S.R. Naidu, E. Zafiriou
and
T.J. McAvoy**

The Use of Neural Networks for Sensor Failure Detection During the Operation of a Control System *

Sinnasamy R. Naidu, Evangelhos Zafriou
and Thomas J. Mc Avoy

Abstract

This paper discusses the use of the back propagation neural network for sensor failure detection in process control systems. The back propagation paradigm along with traditional fault detection algorithms such as the finite integral squared error method and the nearest neighbor method are discussed. The algorithm is applied to the Internal Model Control (IMC) structure for a first order linear time invariant plant subject to high model uncertainty. Compared to traditional methods, the back propagation technique is shown to be able to accurately discern the supercritical failures from their subcritical counterparts. The use of on-line adapted back propagation fault detection systems in nonlinear plants is also investigated.

INTRODUCTION

Significant interactions exist between the operation of the controller and that of the diagnostic module. The purpose of a diagnostic module is to detect failures of the actuators and sensors used in the control system. The main source of these interactions is the presence of model-plant mismatch. This mismatch can cause the performance of the control system to deteriorate and trigger false alarms by the detection system. On the other hand there is also the risk of missing serious failures when the threshold criteria for the detection system are relaxed in order to avoid false alarms.

The literature on Failure Detection is extensive and various approaches have been proposed. For a survey of the existing methods, the reader is referred to Isermann [1]

*Presented at the 1989 American Control Conference, Pittsburgh Pennsylvania, June 21-23, 1989. The authors are with the Chemical Engineering Program and the Systems Research Center of the University of Maryland, College Park, MD 20742. Correspondence should be addressed to E. Zafriou.

and Willsky [2]. In this paper, motivated by certain recent promising approaches to failure detection, we use Neural Networks to detect sensor failure in chemical process control systems. This Neural Networks approach appears to be better suited to meet the special needs of such systems. Kosut and Walker [3] attempted to quantify the problem of robust failure detection by defining a threshold measure that incorporates the effect of model uncertainty. Recently, Nett *et al* [4] propose an integrated approach to designing control and failure detection systems. Their approach holds great promise as it allows one to compensate for the effect of the control system actions on the signals that are monitored by the failure detection system for a multi-input, multi-output process. This method can be coupled to different kinds of failure detection criteria or measures. In addition to developing this new “four-parameter controller” framework, Nett *et al* [4] also develop analytic expressions for computing the type of threshold measure defined in [3]. This measure is a truncated H_2 -norm of the monitored signals (finite integral squared error-FISE) and a failure is detected only if its effect on the norm is larger than the maximum that can possibly be caused by model-plant mismatch and noise. The measure is inherently conservative in declaring a failure and this poses questions on its usefulness in process control applications. In such applications the model uncertainty is usually very large, while small sensor errors can have a significant economic effect. This seems to be quite different from the situation in flight control, where the model uncertainty is usually small while the type of failures that need to be detected are relatively large. In flight control applications, where large life-threatening failures are the ones that are important and where false alarms should be avoided to the greatest possible extent, the FISE criterion seems to be quite appropriate. Note that, in [4] as well as in [5], expressions are provided for computing the smallest detectable failures when using FISE criteria. The size of these failures can be used as a measure of the conservativeness of the method for a particular application.

Our approach to the problem of robust failure detection for process control systems, is to take into account explicitly the effect of the control system in designing the detection system, in the lines of [4]. However, instead of using the FISE criterion, we use a Neural Network to distinguish between patterns caused by sensor failures and those caused by

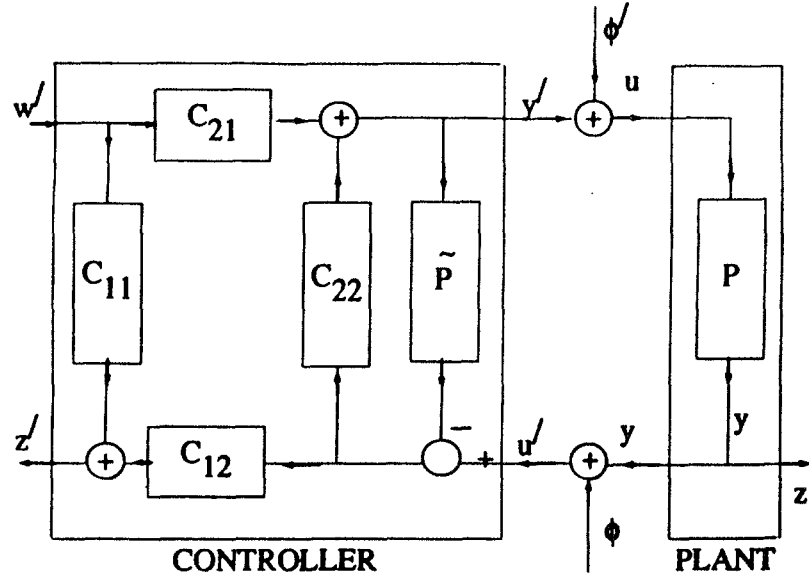


Figure 1: Four-Parameter controller-diagnostic module

SYSTEM DESCRIPTION

Consider the multi-parameter controller/diagnostic module studied in [4] as shown in Figure 1. Let z' , ϕ , ϕ' and w' correspond to diagnostic signal, sensor failure, actuator failure and set point change respectively. Further, P and \tilde{P} refer to the plant and its model.

A simple, yet nontrivial example is analyzed with a view to quantitatively demonstrate the efficacy of the proposed approach. Consider a standard IMC (Internal Model Control, see, e.g., Morari and Zafriou [6]) controller in conjunction with a stable, first order, linear, time invariant plant beset with relatively high model error, embedded in the process gain, for purposes of simplicity. For this system, the parameters of Figure 1 become as shown below, where Q is the transfer function of the IMC controller.

$$\begin{aligned} C_{12} &= I \\ C_{11} &= 0 \\ C_{21} &= Q = -C_{22} \end{aligned} \tag{1}$$

We will attempt the design of a sensor failure classifier based on neural networks with

back propagation. The diagnostic signal is

$$z' = [I + (P - \tilde{P}) Q]^{-1} [\phi + (P - \tilde{P}) Q w'] \quad (2)$$

In addition, we restrict fluctuations in setpoint as well as sensor failures to step functions of varying magnitude as well as onset times. An appropriate representation is,

$$\begin{aligned} P &= K/(\tau s + 1), \\ \tilde{P} &= 1/(\tau s + 1), \\ Q &= (\tau s + 1)/(\lambda s + 1), \\ w' &= m \exp(-\tau_w s)/s \\ \text{and } \phi &= \Omega \exp(-\tau_\phi s)/s \end{aligned} \quad (3)$$

These functions are subject to the constraints

$$\begin{aligned} |m| &\leq m_u \\ |K - 1| &\leq l \\ \tau_\phi, \tau_w &\in [0, T/2] \end{aligned} \quad (4)$$

Let the maximum steady state error allowed in the product quality be $\delta\%$ and the reference value of the output be $y_0 = 1$. The largest steady state fractional error caused by a sensor failure in the IMC control system for the operating region defined by m_u , is $\Omega/(1 - m_u)$. Therefore, supercritical sensor failures that must be detected by the diagnostic module must subscribe to the following

$$\begin{aligned} |\Omega| &> \phi_{cr} \\ \text{where } \phi_{cr} &= (1 - m_u)\delta/100. \end{aligned} \quad (5)$$

Subcritical failures, the complements of the above are permitted to persist in the dynamical system under consideration. The parameters governing the system: $\delta, m_u, l, T, \lambda$ and τ are set at 5.0, 0.4, 0.4, 0.025, 0.025 and 0.25 which lead to the small value of 0.03 for ϕ_{cr} .

The above values correspond to a 40% uncertainty in the steady state gain of the model, while no more than 5% steady state offset is permitted. Such values are quite common in chemical and oil industries (see, e. g., the distillation system in Prett and Garcia [7]). The IMC controller is designed to correspond to a nominal closed-loop time constant equal to one-tenth of the open-loop. Note that it is equivalent to a standard proportional-integral (PI) controller [6]. The length of the observation is taken equal to the nominal closed-loop time constant.

Let S denote the five dimensional subspace to which the parameters determining the diagnostic failure signal belong. The subspace S contains the five parameters Ω, K, m, τ_w and τ_ϕ such that

$$\begin{aligned} -2\phi_{cr} &\leq \Omega \leq 2\phi_{cr} \\ 1-l &\leq K \leq 1+l \\ -m_u &\leq m \leq +m_u \\ 0 &\leq \tau_w, \tau_\phi \leq 0.5T \end{aligned} \tag{6}$$

Hence we limit the signals used in the tests to those created by sensor failures up to twice the magnitude of the critical limit and to setpoint changes or failures that have lasted at least for 50 % of the time window span. Further let p_{ij} denote the probability that a signal with actual fault status index i be classified as possessing j . It must be understood that the fault status index of a subcritical failure is defined to be zero while the corresponding index for a supercritical failure is set at unity. Clearly p_{00} and p_{11} refer to the probability of not raising the alarm for a subcritical failure and the probability of accurately predicting a supercritical failure. On the other hand, p_{01} and p_{10} refer to the probability of false alarms and misses respectively. The sum of the latter pair is the measure of diagnostic error.

INTEGRAL MEASURE

Excluding the effect of noise, the miss threshold [3] is defined as the maximum of the integral of the variable z' over all values of w' and Δ where Δ refers to model error.

$$J_{\text{miss}} = \sup_{\Delta, w'} ||z'|| \tag{7}$$

In accordance with the FISE criterion, a failure must necessarily generate a diagnostic signal with norm exceeding the miss threshold to be detected in the presence of setpoint fluctuations. Further, diagnostics are to be performed over a finite temporal window $[0, T]$.

FISE diagnostics performed on 2048 samples, drawn randomly from the space of parameters S , are illustrated in Figure 2. The boxes and adjoining rectangles depicted in Figure 2, represent the fault cognition statistics due to FISE. In particular, p_{00} , the probability of correct identification of subcritical faults is that fraction of the total number

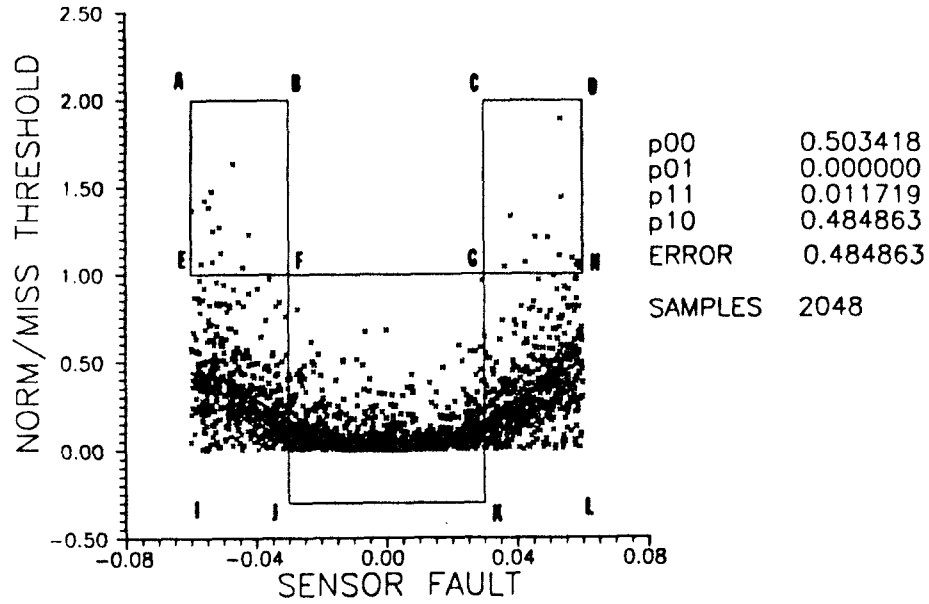


Figure 2: FISE (Calibration Set of Back Propagation)

of samples which is enclosed in FGKJ. Likewise, p_{11} , the probability of exact identification of supercritical failures is the fraction residing in the union of boxes ABFE and CDHG. Furthermore, p_{01} , the probability of false alarm and p_{10} , the probability of missing a supercritical failure are the fractions of the samples residing in BCGF and the union of EFJI and GHLK respectively. The parabola-shaped curve lying amid the samples describes the average of the fault status for each sensor fault size. The definitions stated here are valid for all the figures exhibiting fault detection statistics and therefore, are not repeated in the sections that follow. Since the samples were drawn randomly, the total number of supercritical samples differ from the population of subcritical counterparts, although not significantly. Judging from Figure 2, it is clear that FISE does not give rise to any false alarms. However, it misses nearly 97 % of supercritical failures. This is not surprising, given the nature of the criterion and the large model error, coupled with relatively small critical failure threshold.

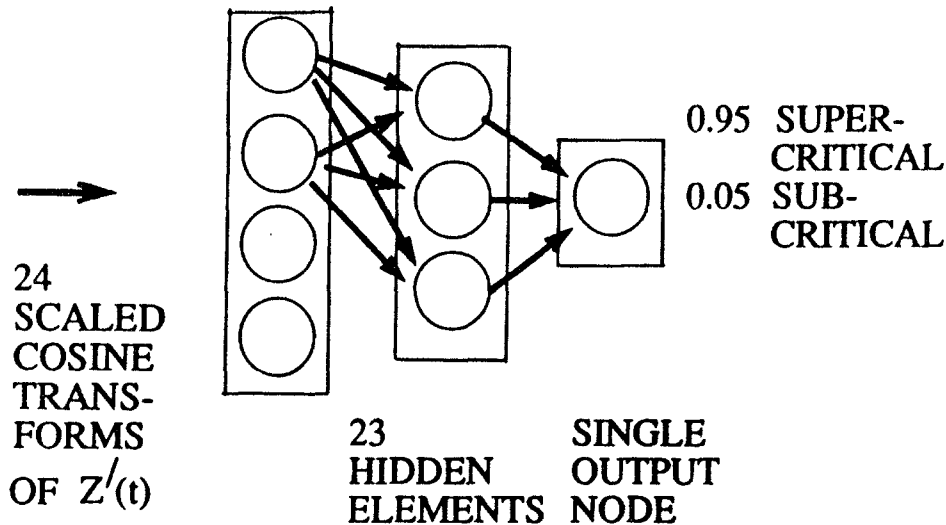


Figure 3: Back Propagation based diagnostics

DIAGNOSTICS WITH BACK PROPAGATION

Traditional applications of connectionist systems, reported in Rumelhart and McClelland's [8] compendium suggest that back propagation can be viewed as an accurate pattern recognition device. Recently, Gorman and Sejnowski [9] reported the application of the back propagation paradigm to the classification of sonar returns from similarly shaped undersea targets. Superior pattern classification properties with overall accuracy as high as 90 % were noted in their work. An accurate design of a biosensor based on the interpretation of fluorescence spectra is described in the work of Mc Avoy *et al* [10]. Hoskins and Himmelblau [11] reported a back propagation based design of process fault detection system with creditable performance characteristics. Recently, Passino *et al* [12] used a certain type of neural networks called the multilayer perceptron, as a numeric-to-symbolic converter in a failure diagnosis application on an aircraft example.

The objective here is to design a back propagation based detection system capable of detecting sensor failures that are smaller than those detected by FISE, by looking at a finite collection of Fourier coefficients of z' as can be seen in Figure 3. It is beyond the scope of this paper to discuss the governing equations of back propagation and the

various strategies of training and prediction. In essence, the unknown coefficients in the back propagation based approximation of fault status, referred to as weights, are altered in the direction along which the error function (the sum of squares of the deviations between the predictions and the exact values) is minimized. The interested reader is referred to [8] and Jacobs [13].

Input–Output Pair

While training the Neural Network, for a given diagnostic signal $z'(t)$, the input is a scaled vector of cosine transforms given by the integral

$$\left[(10/T) \int_0^T z'(t) \cos(n\pi t/T) dt \right]_{n=0, \dots, 23}$$

The output is set at 0.95 for supercritical failures and 0.05 if otherwise. During testing, net predictions greater than 0.50 are interpreted as supercritical.

The calibration data base is composed of continuous, random samples drawn from S while the test data base is a fixed discrete set of 2048 samples, also drawn from S . The parameters governing the test set were simply the end points of seven equally spaced subdivisions of the Ω -interval taken together with three equally spaced partitions of the intervals corresponding to the remaining parameters.

Initially, weights were randomized between -0.5 and $+0.5$, and the heuristic momentum method described in [13] was implemented. The learning process was periodically interrupted between 250 presentations of the calibration patterns, and diagnostics were generated on the fixed test set. Test as well as calibration diagnostics were monitored throughout the usage of the neuro-computer. Amplitude of fluctuations in the weight vector and diagnostic errors were found to decrease gradually with respect to the duration of training. After nearly 5,000 presentations of calibration patterns, the momentum factor [13] was increased to promote stability. For all practical purposes, convergent solutions were obtained after nearly two hours of training and testing which involved roughly 50,000 presentations of calibration patterns. The amount of time required to simulate the neural network in question is strictly real time. Since the neuro-computer was set to process the problem of interest exclusively, the time reported is a close approximation of

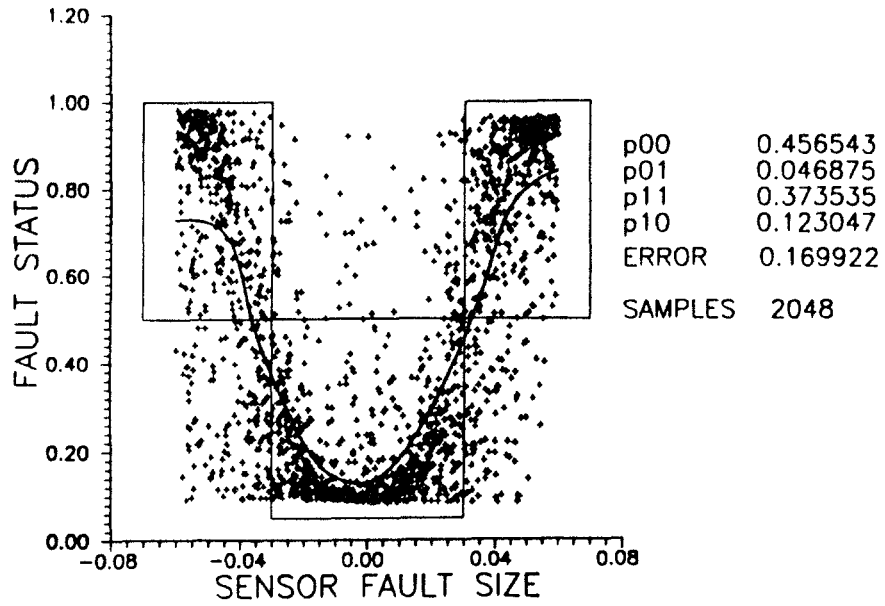


Figure 4: Back Propagation (Calibration Set)

CPU time. The networks were simulated on an IBM AT compatible Zenith 386 equipped with the C callable, ANZA Plus Neuro-computing system.

If different (random) initial weights are selected, the weight state vector at the termination of neurocomputation is usually different. However, the diagnostics of the neural net on a given database remains essentially invariant, thus supporting the well known conjecture [14] that the error surface of the back propagation network is composed of multiple global minima rather than local minima.

The governing equations of back propagation imply that the norm of the weight vector at the solution decreases with increasing value of the norm of the mean input vector. In typical applications [8], at the beginning of simulation, weights are randomized between moderate or small limits of opposite signs. In this work, tenfold amplification of each input vector generated convergent results within the reasonable time frame of 2 hours while unit amplification gave rise to a weight state of high predictive error which is for all practical purposes a non-solution. This is not surprising because of the relation of the back propagation learning law to the Steepest descent method, whose convergence properties can be significantly affected by variable scaling.

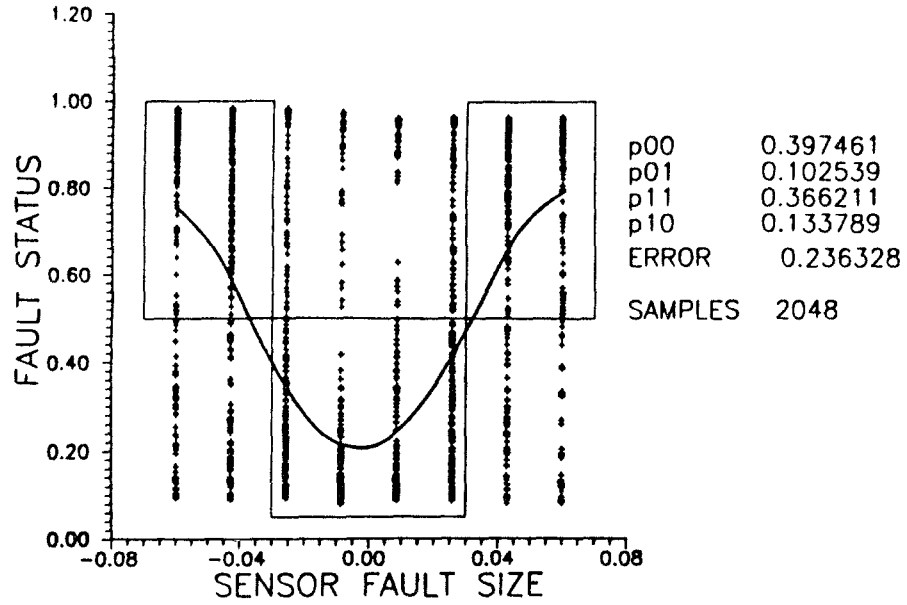


Figure 5: Back Propagation (Test Set)

Regarding the number of hidden nodes, note that, for purposes of simplicity, the dimension of the hidden layer was chosen strictly on the basis of the neural network's ability to correctly identify failures from a linear, uncertain plant subject to internal model control. The size of the hidden layer was typically determined by requiring the neural network under consideration to minimize the total diagnostic errors $[p_{01} + p_{10}]$ of calibration and test databases, between two and three hours of neurocomputation in real time. When the total number of hidden nodes were fewer than 23, diagnostic errors of calibration and test databases were very high, thus rendering the sum unacceptable in design consideration. As the number of hidden elements increase toward 23, both calibration and test diagnostic errors continue to decrease. Further increase of number of hidden elements beyond 23 caused the calibration error to fall while the test error grew, which led to the choice of 23 hidden computational neurons.

Diagnostics performed by the trained network on 2048 random samples appear in Figure 4 while the corresponding result for the test set is shown in Figure 5. Back propagation based diagnostic errors of test and calibration sets are significantly less than their FISE counterparts. In particular, although the former gives rise to false alarms, it

misses only 25 % of the supercritical failures while the latter misses nearly 97 %.

NONLINEAR SYSTEMS

We now consider the application of the trained back propagation to a plant with a nonlinear gain, whose values fall within the limits of the *uncertain* gain of the linear model used in training. Let U_{min} and U_{max} correspond to the minimum and maximum values of U , the manipulated variable, at steady state for the linear uncertain model. For the operating region defined by m_u and the model uncertainty bounded by l , these values are

$$\begin{aligned} U_{max} &= U_0 + m_u/(1-l) = 1.67 \\ U_{min} &= U_0 - m_u/(1-l) = 0.33 \end{aligned} \quad (8)$$

The value U_0 is the reference value for U , taken to be unity. Let the plant be described by the following, where $G(U)$ is a nonlinear function of U and y the true plant output (not the deviation from y_0).

$$\begin{aligned} dx/dt &= -x/\bar{\tau} + G(U) \\ y &= x/\tau \end{aligned} \quad (9)$$

Then in order for the uncertain linear model of the system under study to be a reasonable description of the nonlinear plant, $G(U)$ must satisfy the following, where (10) must be satisfied for all U between U_{min} and U_{max} .

$$1-l \leq dG/dU \leq 1+l \quad (10)$$

$$\lim_{U=U_0} dG/dU = 1 \quad (11)$$

$$G(U_0) = y_0 \quad (12)$$

Equation (10) describes the steady state gain uncertainty of the linear model, (11) the nominal steady state gain and (12) the relationship between the reference values for U_0 and y_0 which are assumed to correspond to an equilibrium state.

Second Order Non-linearity

$$G = 0.3U^2 + 0.4U + 0.3 \quad (13)$$

Figure 6 describes the variation of dG/dU with respect to the manipulated variable.

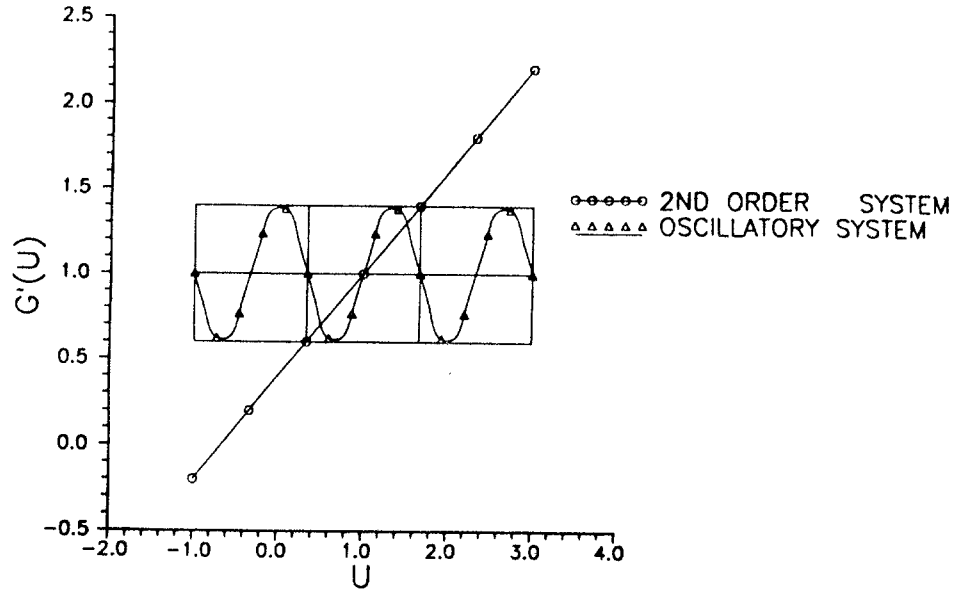


Figure 6: Nonlinear Systems

Note that during transients, the non-linear response can be nearly twice as large that of the linear reference model.

Oscillatory Non-linearity

$$G = U + (2l/3\pi) [1 - \cos [(3\pi/2)(U - 1)]] \quad (14)$$

Judging from Figure 6, it is clear that dG/dU is bounded by the limiting values of the linear system's process gain, not only during steady state but also throughout the transient case.

On-line Training Issues

Let us first focus on the second order nonlinearity. The test data base was chosen to be a fixed, discrete set of 512 samples drawn from S . Diagnostics performed by the back propagation trained exclusively on the linear plant, are illustrated in Figure 7. In comparison with the diagnostics performed by the above net on the test set pertaining to the linear plant, shown in Figure 5, one finds

$$\begin{aligned} p_{01,2nd \text{ order}} &> p_{01,linear} \\ p_{10,2nd \text{ order}} &< p_{10,linear} \end{aligned}$$

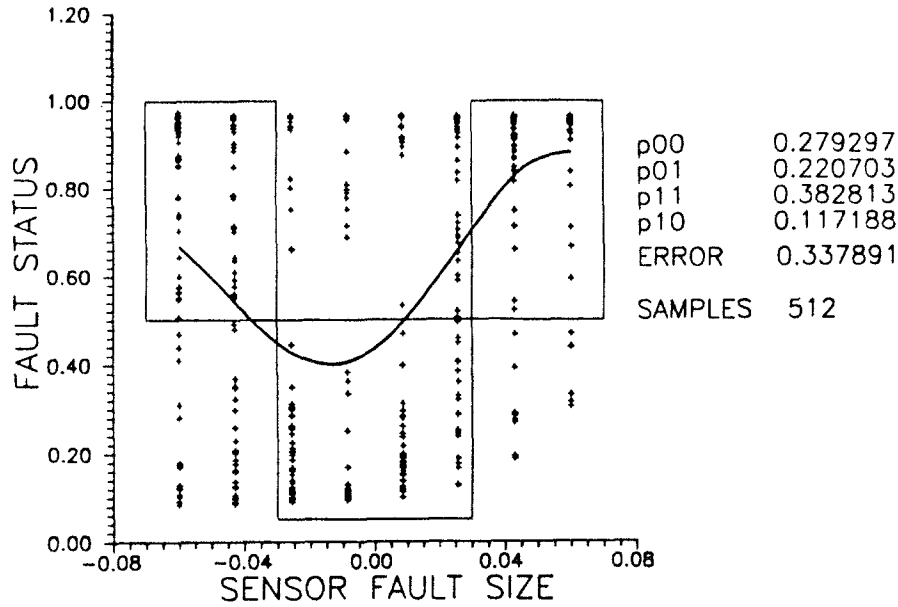


Figure 7: Back Propagation (2nd Order System before On-line Training)

This observation supports the fact that the diagnostic signal of the 2nd order plant can be significantly larger than its linear counterpart, due to the fact that the uncertain linear model is not a good description of the nonlinearity during the transient.

In order to reduce the false alarms associated with the non-linear plant, a hybrid calibration data base composed of subcritical signals from the plant and supercritical states from the model (linear system) was desired. Consider the operation of the neural network in parallel to the IMC controlled nonlinear plant whose linear model is assumed known. The neural network is to be trained on-line on plant data which can be safely assumed to be essentially failure free. Therefore training exclusively upon plant data will cause the neural net to only suppress false alarms. In order that the neural network does not miss supercritical sensor faults, fault rich database must necessarily be included in the knowledge base. This is generated from the linear model and uncertainty bounds known to the engineer. During on-line training, the neural net is linked back and forth between diagnostic signal outlets from the nonlinear plant and its model based simulation program. During the linkage with real plant, the network learns to suppress false alarms and while being connected to the computer model, it learns to not to miss supercritical

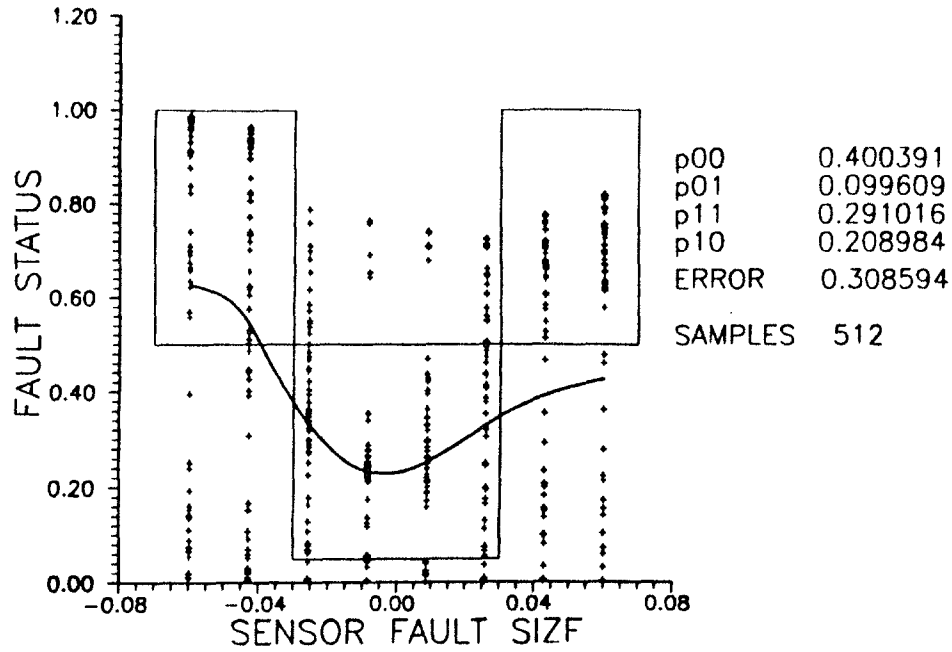


Figure 8: Back Propagation (2nd Order System after On-line Training)

faults. The scenario detailed thus far is clearly emulated by alternately feeding random patterns from the linear data base and its nonlinear counterpart.

Subcritical components were chosen from a collection of 512, random samples obtained by the methodology already described, while the supercritical were generated by the neuro-computer as and when needed, in a manner analogous to the previous simulations of the linear plant. During training, each presentation of a random pattern from the linear data base was followed by a random entry from the non-linear data base. By adopting an overall strategy of training followed by periodic testing, analogous to the previous simulation studies, convergent weights along with consistent predictions were noted after roughly two hours, the essence of which is plotted in Figure 8. On-line training reduces false alarms from 44 % to 20 % at the expense of a smaller, net increase in the percentage of misses of supercritical failures.

On-line training studies involving the oscillatory nonlinearity are only summarized, as the specifics on the design of hybrid calibration data base and the training strategy are identical to those of the 2nd order plant. The diagnostics performed on the test set prior to on-line training are shown in Figure 9. In comparison with the diagnostics performed

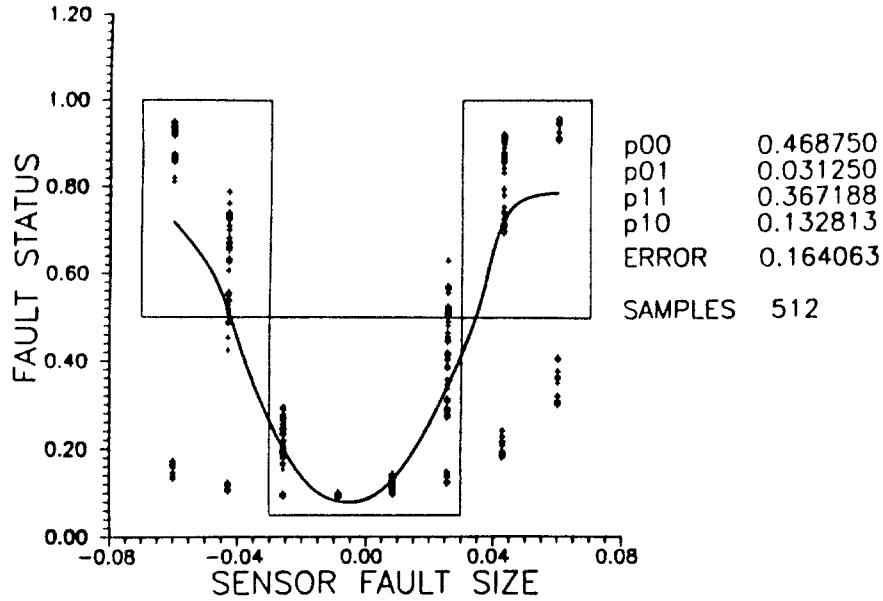


Figure 9: Back Propagation (Oscillatory System before On-line Training)

by the above net on the test set pertaining to the linear plant, shown in Figure 5, one finds

$$\begin{aligned} p_{01,\text{oscillatory}} &< p_{01,\text{linear}} \\ p_{10,\text{oscillatory}} &> p_{10,\text{linear}} \end{aligned}$$

Simulations analogous to those of the 2nd order plant were carried out and the diagnostics after on-line training are plotted in Figure 10. Upon on-line training, the probability of missing supercritical failures was reduced from 0.1328 to 0.1250 while the probability of a false alarm also decreased from 0.031 to 0.014.

NEAREST NEIGHBOR METHOD

In order to rigorously validate the proposed pattern recognition approach to sensor failure detection, we need to consider classification algorithms that are not restricted by the integral formulation. The Nearest Neighbor classifier, a well known statistical method employed in [9] in sonar studies is considered here. In Table 1, back propagation based diagnostics and the statistics from the current methodology are documented.

The knowledge base for the statistical method is a fixed discrete set of 648 samples

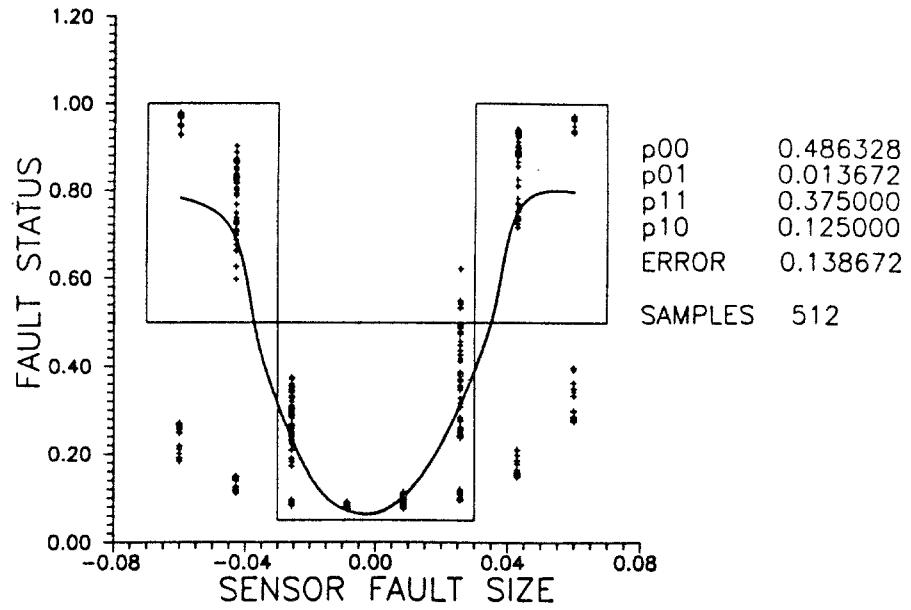


Figure 10: Back Propagation (Oscillatory System after On-line Training)

System	Method	p_{00}	p_{01}	p_{11}	p_{10}	Error
Linear	BPN	.398	.102	.366	.134	.236
	Nearest	.467	.033	.231	.269	.272
	Neighbor					

Table 1: Comparison of Back Propagation and Nearest Neighbor Diagnostics

drawn from S . The parameters governing the same were the end points of seven equally spaced subdivisions of the Ω -interval taken together with two equally spaced partitions of the intervals corresponding to the rest. In the evaluation of the linear plant, the test is composed of 648, random samples drawn from S . Comparison reveals that for the linear plant case, back propagation yields very similar diagnostics to the Nearest Neighbor, which is not surprising. Comparative studies involving 2nd order or oscillatory plant gain reveal that the statistical method classifies each test signal into the category of supercritical failures. The advantages of the back propagation diagnostics over the Nearest Neighbor classifier lie in the ability of the former to capture nonlinear characteristics as well as in its speed. For example, applying the Nearest Neighbor method to the above test set took over 12 hours on the Zenith 386; this was the reason why the knowledge base was limited to 648 samples. It is also important to notice that back propagation is less expensive in terms of storage.

CONCLUSION

A back propagation based sensor failure detection system has been proposed and compared to FISE diagnostics as well as the Nearest Neighbor classifier, for an IMC controlled system involving an uncertain linear time invariant first order model and linear or nonlinear plants that lie within the model uncertainty bounds. Detailed studies reveal that the proposed pattern recognition approach to failure detection using back propagation holds significant promise.

The main advantages of the proposed approach over existing methods are the method's ability to capture nonlinear characteristics, the possibility for on-line training and its speed during on-line implementation. There is no reason however to assume that back propagation is necessarily the best paradigm for sensor failure detection. Other Neural Network paradigms such as the Counter Propagation Network (Hecht-Nielsen [15]) and the Neocognitron (Fukushima, [16]) should also be investigated. Future work should also include the effects of actuator failure and study multi-input multi-output systems, with industrially significant formulations of plants and models.

References

- [1] R. Isermann, "Process Fault Detection Based on Modeling and Estimation Methods – A Survey", *Automatica*, **20**, pp 387–404, 1984.
- [2] A. S. Willsky, "A Survey of Design Methods for Failure Detection in Dynamic Systems", *Automatica*, **12**, pp 601–611, 1976.
- [3] R. L. Kosut and R. A. Walker, "Robust Fault Detection: The Effect of Model Error", *Proc. of 1984 Amer. Control Conf.*, pp 1094–1096, San Diego, CA, 1984.
- [4] C. N. Nett, C. A. Jacobson and A. T. Miller, "An Integrated Approach to Controls and Diagnostics: The 4-Parameter Controller", *Proc. of 1988 Amer. Control Conf.*, pp 824–835, Atlanta, Georgia, 1988.
- [5] A. Emami-Naeini, M. M. Akhter and S. M. Rock, "Effect of Model Uncertainty on Failure Detection: The threshold selector", *IEEE Trans. on Autom. Control*, **33**, pp 1106–1115, 1988.
- [6] M. Morari and E. Zafiriou, *Robust Process Control*, Prentice-Hall, Englewood-Cliffs, NJ, 1989.
- [7] D. M. Prett and C. E. Garcia, *Fundamental Process Control*, Butterworth Publishers, Stoneham, MA, 1988.
- [8] D. E. Rumelhart and J. L. McClelland, *Parallel distributed processing: Explorations in the microstructure of cognition*, MIT Press, Cambridge, MA, 1986.
- [9] R. P. Gorman and T. J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets", *Neural Networks*, **1**, pp 75–89, 1988.
- [10] T. J. Mc Avoy, N. S. Wang, S. Naidu, N. V. Bhat, J. Gunter and M. Simmons, "Interpreting Biosensor Data via Back Propagation", *Proc. of Joint Conference on Neural Networks*, **1**, pp 227–233, Washington, DC, 1989.

- [11] J. C. Hoskins and D. M. Himmelblau , “Artificial Neural Network Models of Knowledge Representation in Process Engineering ”, *Computers and Chemical Engineering*, **12**, pp 881–915, 1988.
- [12] K. M. Passino, M. A. Sartori and P. J. Antsaklis, “Neural Computing for Numeric-to-Symbolic Conversion in Control Systems ”, *IEE Control Systems Magazine*, **9**, pp 44–52, 1989.
- [13] R. A. Jacobs , “Increased Rates of Convergence Through Learning Rate Adaptation”, *Neural Networks*, **1**, pp 295–307, 1988.
- [14] R. Hecht-Nielsen, “Theory of the Backpropagation Neural Network”, *Proc. of Joint Conference on Neural Networks*, **1**, pp 593–605, Washington, DC, 1989.
- [15] R. Hecht-Nielsen , “Applications of Counter Propagation Networks ”, *Neural Networks*, **1**, pp 131–139, 1988.
- [16] K. Fukushima , “Neocognitron: A Hierarchical Neural Network Capable of Visual Pattern Recognition ”, *Neural Networks*, **1**, pp 119–130, 1988.