2nd IEEE MEDITERRANEAN SYMPOSIUM ON
NEW DIRECTIONS IN CONTROL & AUTOMATION
JUNE 19 - 22, 1994
MALEME-CHANIA, CRETE, GREECE

Neural Networks in Fault Identification in Large Communication Networks

Nikitas J. Dimopoulos, Stephen Neville, Andrew Watkins, Kin F. Li, Eric. G. Manning

Department of Electrical and Computer Engineering,
University of Victoria,
PO BOX 3055, Victoria, BC, V8W 3P6
CANADA

Abstract – In this work, we present our efforts in developing techniques for detecting the onset and diagnosis of a fault. The Diagnosis domain is that of large Cable Television Networks. We are using stable recurrent neural networks to model the dynamic behavior of some of the measured parameters both for normal operation and during a fault. Deviations from the model indicate the onset of a fault, while the properties of the behavior during a fault are indicative of the fault modality.

I. INTRODUCTION

A. *Structure of the Network*

A Cable Television Network incorporates a number of high frequency amplifiers forming (for conventional networks) a tree. In more advanced networks the structure incorporates a double ring from which subscriber drops emanate. In this work, we are focusing in conventionally structured tree networks. There are two categories of amplifiers, the ones belonging to the main trunk and the ones forming subscriber drops. Additionally, power supplies are located throughout the network, each one powering a limited number (typically three) of amplifiers. The majority of the main trunk amplifiers are equipped with a status monitor which uses a reverse channel to report the status of the amplifier to the head office. Subscriber drops and power supplies are not normally monitored.

Typical variables which are monitored include [2]

> Forward Pilot level, Reverse Pilot level, Raw DC voltage into the amplifier, Regulated voltage of amplifier power supply, DC current into forward and reverse sections of the amplifier, Temperature inside the trunk station, Reverse Switch status, and Trunk Lid status.

The values of the monitored variables are allowed to vary within two intervals (warning and alarm) centered at nominal values. If a value is outside these predefined intervals then a warning or an alarm is issued. Three consecutive alarms constitute a failure. A typical section of the main trunk is depicted in Figure 1. Each amplifier in the network has a name, as well as a location, connectivity and functionality attributes

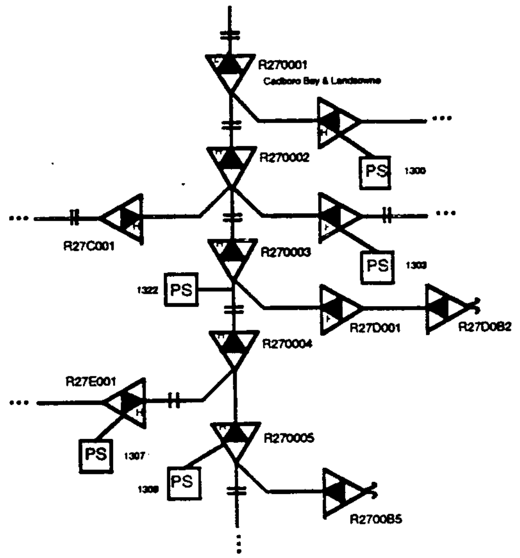

Fig. 1. A typical Section of a Main Trunk (curtesy Roger's Victoria)

B. *Diagnosis Problem*

There are several modalities of failure. Some are discussed below.

A single amplifier may fail, whereupon the signal is fed through unamplified to the subsequent stage of the network. The subsequent amplifiers, equipped with automatic gain control, will boost the signal back to its normal level after two to three stages. Because of the failure, all amplifiers located between the failing amplifier and the stage at which the signal was boosted to its correct level, report alarms or failures. Because of the tree structure of the network, the reporting amplifiers are not typically polled sequentially and the reports appear at seemingly random locations in the report.

A power grid failure, will affect a number of amplifiers. The affected amplifiers fail to communicate, while the deteriorated signal, cause downstream nodes to report failure until the signal is boosted again to its typical values.

Age may cause the amplifier to drift away from its optimal operating point. This in turn results in some or all of the measured parameters to have values which cause alarms or warnings. Such a situation may exist for a large number of amplifiers for long periods of time. This in turn result in a multitude of warnings/alarms which make the interpretation of the status of the network extremely difficult.

We have developed an expert diagnosis environment [2] which monitors the amplifier network, analyzes the failure modalities and reports the location and cause of faults in real time. This diagnosis environment has been successful in locating and diagnosing fault modalities which involve significant changes in the measurements. Examples are loss of power, damage in the network.

There are several other modalities which manifest themselves as changes of behavior rather than significant changes in the measurements.

The focus of this work is in detecting the onset of such behavior changes and providing a diagnosis based on the properties of the ensuing behavior.

We use neural network techniques in our effort to classify and identify the behavior of the network of amplifiers.

This work is divided into the following sections.

Section II introduces a class of recurrent stable neural networks which can be used to identify the behavior of dynamical systems in general and amplifiers used in transmitting cable TV signals in particular. Section III presents the results of the application of the neural networks discussed in section II to identify the dynamical behavior of monitored parameters in main trunk amplifiers, detect the onset of failures and provide a diagnosis Finally section V concludes the work.

II. STABLE RECURRENT NEURAL NETWORKS

It has been shown [5] that asymptotic stability is ensured for neural networks which are described by the differential equation

$$\dot{O} = -TO + Wf(O) + b \tag{1}$$

In (1), there are N neurons divided into k classes, and

$$O = \begin{bmatrix} O_1 & O_2 & \dots & O_k \end{bmatrix}$$
$$= \begin{bmatrix} o_1 & o_2 & \dots & o_N \end{bmatrix}$$ is the state of the neural network,

$$W = \begin{bmatrix} W_{11} & W_{12} & \dots & W_{1k} \\ \vdots & & & \\ W_{k1} & W_{k2} & \dots & W_{kk} \end{bmatrix} \tag{2}$$

is the network connectivity matrix, $T = \mathrm{diag}(\tau_i)$ is the diagonal matrix of neural relaxation constants, b is the input to the neural network, and $f(O)$ belongs to the class of so-called neuromime functions, which are essentially positive and monotonically non-decreasing. The condition on W that guarantees asymptotic behavior is that it must contain all of its positive entries on one side of the main diagonal [1]. This gives an easy way to check whether a neural network is stable. For instance, the neural network shown in Figure 1 is stable provided that the connection weights in submatrices W_{23} and W_{34} are non-positive (i.e., inhibitory). This result is extremely useful in the area of identification and control. The most important feature of a controller or model is that it must be stable. By starting with a model as defined by (1), stability is ensured.

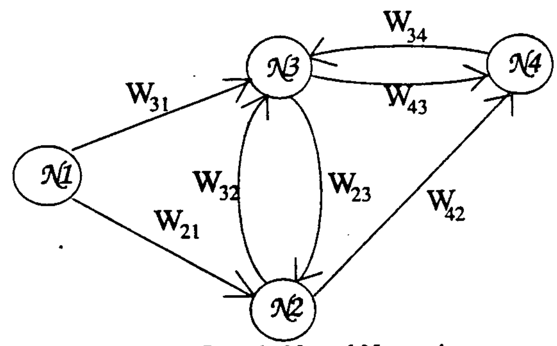

Fig. 2. Sample Neural Network

A. Parameter Adjustment In Stable Neural Networks

This section discusses a method for adjusting the weights and other parameters of neural networks that are stable in the sense described in Section 2. The general approach that is used here is to define some a criterion and then adjust the parameters in a direction that will decrease this cost. In this sense the technique is similar to linear recursive adaptive methods [4] and to classical back propagation [5]. However, since the stable neural networks described in section 2. have certain restrictions on the polarity of the connection of classes, a straightforward gradient adjustment is not possible. A solution for this is also presented here.

B. Gradient of Cost Function

The general equation for calculating the behavior of the class of neural networks of interest here is

$$\dot{O} = -TO + Wf(O) + b \tag{3}$$

using the same notation introduced in section 2. One possible criterion for measuring the performance is the quadratic cost function

$$J(e) = 1/2(O-O_d)^T A(O-O_d) = 1/2e^T Ae \qquad (4)$$

where O_d is the desired state of the neural network. Matrix A is used to eliminate from the cost any neurons whose state is not crucial. A is a diagonal matrix with ones corresponding to output neurons and zero's elsewhere. As in other recursive adaptive methods [1],[6], parameters θ in the neural network are adjusted along the negative gradient of this cost, i.e., $\frac{d\theta}{dt} = -\eta\frac{\partial J}{\partial\theta}$. The chain rule for differentiation is used to allow for the calculation of this gradient for parameters associated with neuron j:

$$\frac{\partial J}{\partial\theta} = \frac{\partial J}{\partial o_j}\frac{\partial o_j}{\partial\theta} = \gamma_j\frac{\partial o_j}{\partial\theta} \qquad (5)$$

The notation γ_j is used to denote the derivative of the cost with respect to the activation of neuron j. If neuron j is an output neuron, this derivative is simply

$$\gamma_j = o_j - o_{d_j}. \qquad (6)$$

In a manner analogous to traditional back propagation of the error [5], this gradient may be calculated for units that are not output neurons by using the values of the gradient in all the neurons k that have neuron j as inputs:

$$\gamma_j = \sum_k \gamma_k\frac{\partial o_k}{\partial o_j} = \sum_k \gamma_k\Delta_{kj} \qquad (7)$$

Here, the notation Δ_{kj} has been introduced to represent the partial derivative $\partial o_k/\partial o_j$. To calculate Δ_{kj}, it is necessary to use the differential equation which defines the behavior of the neural network. Rewriting (4) specifically for neuron k, and using the operator D to represent differentiation results in

$$(\tau_k + D)\,o_k = \sum_j w_{kj}f(o_j) + b_k \qquad (8)$$

Differentiating (8) with respect to o_j results in

$$\dot{\Delta}_{kj} = (-\tau_k)\,\Delta_{kj} + w_{kj}f'(o_j)\,. \qquad (9)$$

All the derivatives required in (5) to adjust a parameter θ have now been obtained, except for the derivative $\partial o_j/\partial\theta$. The next section discusses the case when θ is a connecting weight. A similar technique can be used to obtain a formulae for adjusting any of the other variables that parameterize the neural network such as the relaxation constant τ or parameters of the activation function $f(\;)$ [3],[4].

C. *Weight Adjustment*

Let θ represent a connecting weight w_{ji} which connects neuron i (input) to neuron j. Use the notation $\xi_{ji} = \partial o_j/\partial w_{ji}$. Differentiating (10) with respect to w_{ji}, the differential equation for ξ_{ji} is obtained:

$$\dot{\xi}_{ji} = -\tau_j\xi_{ji} + f(o_i) \qquad (10)$$

Using this equation and the results of the previous section, equation (7) may now be written as

$$\frac{dw_{ji}}{dt} = -\eta\gamma_j\xi_{ji} \qquad (11)$$

with γ_j calculated using (6) or (7) as appropriate.

D. *Weight Clamping*

Section II. describes a class of neural networks that are asymptotically stable. This condition is guaranteed provided that the connectivity matrix W has all of its positive entries on one side of the diagonal [5]. However, (11) gives a formula for adjusting the connection weights that may violate this condition. To combat this, it is necessary to check the polarity of certain crucial weights after each weight adjustment. For instance, as discussed in section 2, if the weights labeled W_{23} in Figure 1 are guaranteed to be non-positive, then the neural network will be stable. Thus after any weight in W_{23} is adjusted using (13), the weight should be checked to ensure that it is not positive. If it is, then it should be clamped at 0. This ensures that inhibitory weights stay inhibitory throughout the training procedure.

III. IDENTIFICATION

The term identification is used in this section to refer to the process of developing a model of an unknown system by observing its input/output behavior [1],[6].

This section uses the results of the previous section to identify some unknown systems. A suitable neural network architecture is proposed and some motivation for this configuration is given.

A. *Identification Architecture*

Consider the simple nonlinear system described by the relation $\dot{y} = u - \frac{y}{1+4y^2}$

If y remains relatively constant near some value y_{ss}, then this system can be approximated by a first order linear system that has a pole at $(1+4y_{ss}^2)^{-1}$. If y varies from this value significantly, then the 'pole' can be thought of as roving in some sense. Although this is not an exact description of the behavior of the system, it does illustrate one of the more common types of nonlinearity which is encountered in real systems.

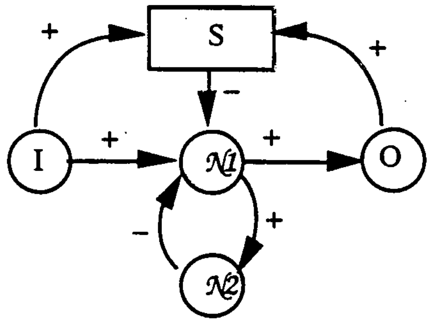

Fig. 3. An Architecture for System Identification

To take advantage of this type of nonlinearity, the architecture of Figure 3. is proposed for general system identification. Labels I and O refer to the input and output of the system, and $\mathcal{N}1$ and $\mathcal{N}2$ refer to two classes of neural networks. The block marked S is a special connection of classes called the 'scheduler class'. The idea of this class is that it controls or schedules which neurons will be active and when, thereby emulating the movement of the 'pole' for large variations of the state variable or input. Neurons in the scheduler class have a peaked response as shown in Figure 4. Each neuron in the class has a peak ρ that occurs at a different value. Figure 3. shows that the scheduler class receives input from I and O. Thus, depending on the value of the input and output, different neurons in $\mathcal{N}1$ and $\mathcal{N}2$ will be active.

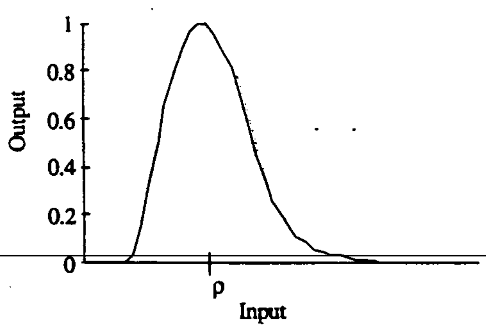

Fig. 4. Response of Scheduler Neurons

This allows the neural network to take advantage of the type of nonlinearity discussed above. The example described by (19) is well suited to this kind of architecture since neurons with different relaxation constants may be activated depending on the level of the output.

B. Identification and Diagnosis of the Pilot Measurements

In this section, we are describing our attempts to accurately determine the onset of a "fault' and to diagnose it. Our diagnosis domain is a large cable television network comprising several hundred main trunk high frequency amplifiers and described in section I.

Eventhough, the amplifiers are temperature compensated, it has been observed that all the measurements vary with the temperature.

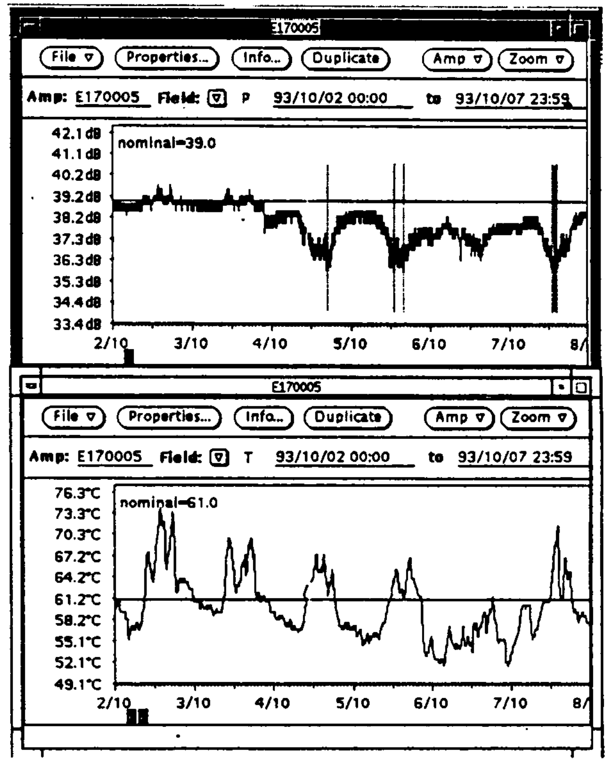

Fig. 5. Pilot and Temperature data over several days in October 1993. The shaded regions in the pilot data indicate alarms (i.e. measurements outside preset limits). Observe the fault initiation on October 3^{d}.

Of course the typical variation is small when the amplifier is properly adjusted. Because of drifts or malfunctions, the amplifiers evolve to a "faulty" state where they exhibit an altered pattern of behavior. A typical example is presented in Figure 5. The behavior pattern depicted during the first two days (until October 3^{d} at 21:40) is representative of a well tuned high-pilot amplifier. Observe the small, temperature correlated variations of the pilot level around the nominal value of 39 db. Suddenly, on October 3^{d} at 21:40, a significant change in the pattern of behavior occurs. The standard fault detection techniques which are normally in use, alert the user only if the values of a measured parameter exceed some preset limits. In this case the pilot level did not exceed its threshold until the following day, some eighteen hours after the start of the new behavior in the evening of October 3^{d}, and then it stayed outside its nominal range for only a limited duration.

Such a belated reporting in conjunction with the short duration of the time during which a measurement stays outside its nominal range, makes an accurate diagnosis very difficult.

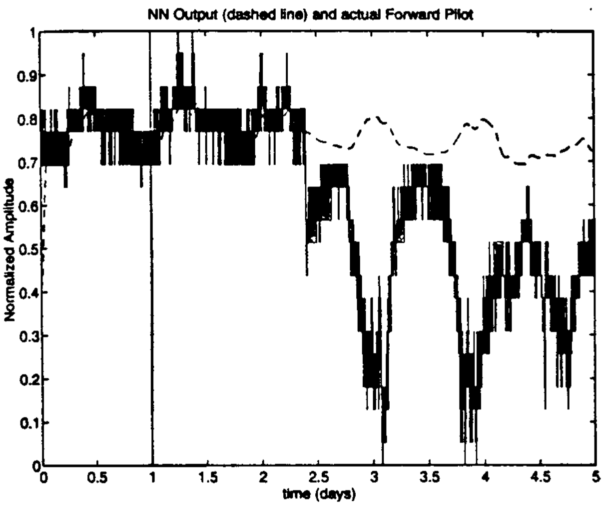

Fig. 6. Actual and Neural Network (dashed line) response for the pilot of E170005. The training set is delineated by the vertical line at day 1.

One may observe therefore that establishing a nominal range of values, is not the best way of detecting the onset of a "faulty" behavior pattern. In addition it does not accurately identify periods during which the behavior is "faulty", and thus it makes an accurate diagnosis of the fault problematic.

An accurate detection of a "fault" initialization and detection requires a model of the behavior of the measurement in time and its dependencies. Any deviation from the model, would denote the onset of a "fault", while the model of the "faulty" behavior pattern, if it could be established, would contribute to the diagnosis.

We have used recurrent neural networks, as presented in sections II and III above, to model behavior of the pilot measurement and its dependence on the temperature of the enclosure. Figure 6. presents the response of the trained neural network (incorporating consisting of two classes each comprising 10 neurons, one being a scheduler class) together with the actual readings for the pilot.

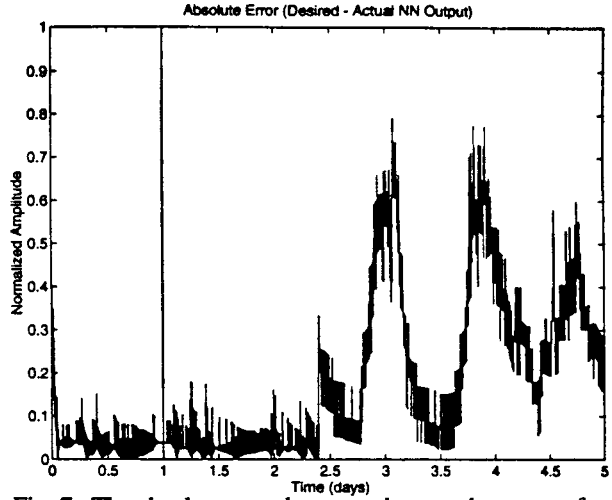

Fig. 7. The absolute error between the actual response for the pilot of E170005 and that of the trained Neural Network. Observe the sizeable increase after the fault

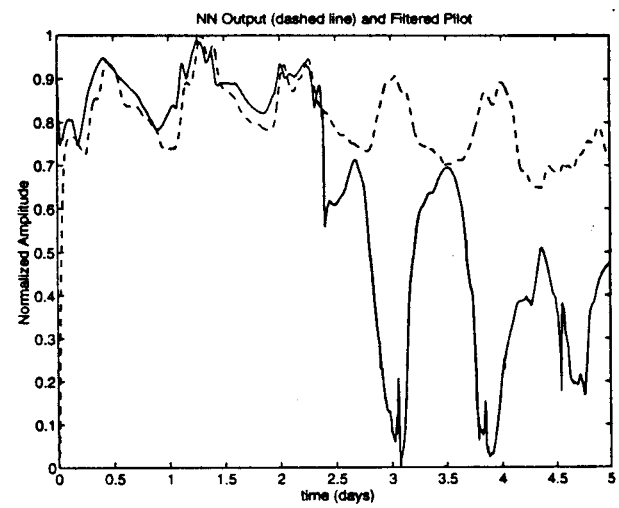

Fig. 8. Filtered and Neural Network (dashed line) response for the pilot of E170005

The neural network was trained with data from October 1, and this period is delineated by the vertical line on day 1 in the plot. Figure 7. presents the difference between the response of the trained neural network and the actual measurements. Observe the abrupt increase of the error at the fault.

Our assumption that the pilot level is affected by the temperature of the enclosure, has been verified through by the measurements taken from a model amplifier in a temperature chamber. Figure 9. presents the pilot measurements of the model amplifier together with the response of the trained neural network, as discussed above, on the same temperature data. The differences in behavior are attributed to the fact that the model amplifier was not driving any load.

As it can be seen in Figure 5. the pilot measurements contain a sizeable noise component, comprising thermal and roundoff. We have used wavelet filters [7] to remove this noise component. Wavelet filters are ideal for this purpose since they easily preserve the discontinuities which are indicative of the onset of the "fault". Figure 8. presents the thus filtered pilot measurement.

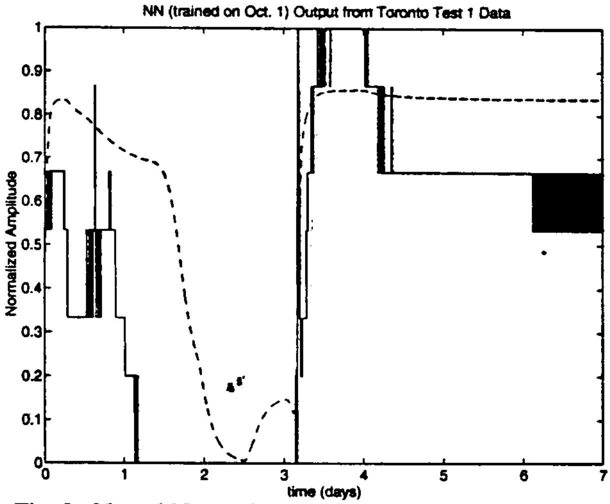

Fig. 9. Neural Network trained with data from E170005 compared with measurements from a model amplifier taken in a temperature chamber.

C. *Fault Diagnosis*

After detecting the onset of the "fault", we attempted to characterize the new behavior pattern. The same neural network was retrained with data from the fault behavior pattern. Figure 10. and Figure 11. show the result of the training. t is obvious that the neural network learned the behavior of the training set, but then as time progressed, the behavior changed and the Neural Network was not able to accurately track the measurements. This behavior is consistent with that of a nonstationary process, and it is in turn consistent with behavior of exhibited by an amplifier which has lost its agc (automatic gain control) section.

We were able to confirm that this diagnosis is the most likely explanation of the behavior of the pilot measurements with the manufacturer of the said amplifier. We are planning to perform a "postmortem" analysis of the affected amplifier to fully quantify the diagnosis.

IV. Conclusions And Discussion

In this work we have presented our efforts in modelling the behavior of the constituent components of a large Cable Television network. Our aim is to devise a methodology through which we shall be able to accurately detect and diagnosed of "faults".

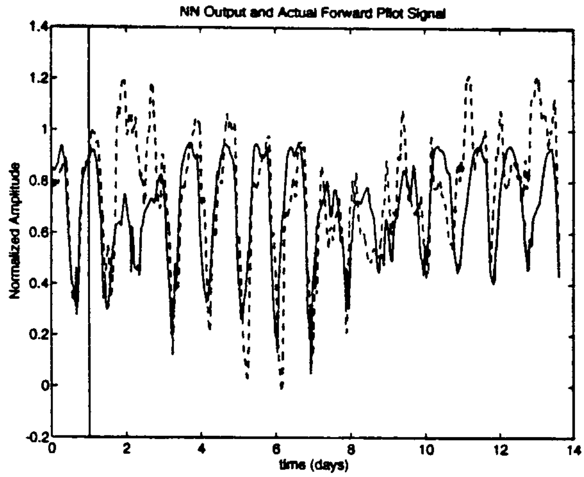

Fig. 10. Neural Network trained after the onset of the fault.

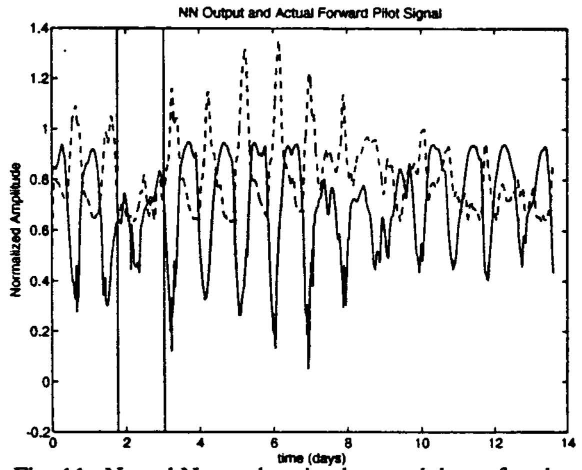

Fig. 11. Neural Network trained several days after the onset of the fault.

We have used stable recurrent neural networks which were trained to model the behavior of certain measured parameters. Deviations from the modelled behavior indicate the onset of a "fault". The characteristics of the behavior pattern after the fault are indicative of the "fault" modality.

We are currently proceeding with the verification of our diagnosis procedures. To this end, we have enlisted the help of domain experts from both the equipment manufacturer and the operator.

Acknowledgement

This work was supported through a grant by the Canadian Cable Labs Fund.

REFERENCES

[1] K.J. Astrom and B. Wittenmark, *Adaptive Control*, Addison-Wesley Publishing Company, 1989.

[2] N. J. Dimopoulos, A. Watkins, S. Neville, K. F. Li "An Expert Network Analyzer: Knowledge Acquisition, Fault Diagnosis and Prediction" *Proceedings 1993 DND Workshop on Advanced Technologies in Knowledge Based Systems and Robotics* Nov. 14-17 1993, Ottawa, Canada.

[3] C. M. Jubien, N. J. Dimopoulos "Recurrent Neural Networks in Systems Identification" *Proceedings 1993 IEEE International Symposium on Circuits and Systems* pp. 2458-2461 (May 1993)

[4] C. M. Jubien, N. J. Dimopoulos "Identification of a PUMA-560 Two Link Robot Using a Stable Neural Network" *Proceedings 1993 International Conference on Neural Networks* (Apr. 1993).

[5] N. Dimopoulos, "A Study of the Asymptotic Behavior of Neural Networks," *IEEE Transactions on Circuits and Systems*, Vol. 36, No.5, pp. 687-694, May 1989.

[6] K.S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Transactions on Neural Networks*, Vol. 1, No. 1, pp.4-27, March 1990.

[7] P. Steffen, P. N. Heller, R. A. Gopinath, C. S. Burrus "Theory of Regular M-Band Wavelet Bases" *IEEE Trans. on Signal Processing* Vol. 41, No. 12, pp. 3497-3511, Dec. 1993.