Neural Network for Fault Detection and Isolation of the Three-Tank System

Panagiotis Tzionas
Technological Educational Institute of Thessaloniki, Greece
Thessaloniki 57400, Greece
ptzionas@teithe.gr

Dimitrios Krontsos
Technological Educational Institute of Thessaloniki, Greece
Thessaloniki 57400, Greece
krontsosd@yahoo.gr

Simira Papadopoulou
Technological Educational Institute of Thessaloniki, Greece
Thessaloniki 57400, Greece
shmira@teithe.gr

***Abstract*-This paper presents the design, training, verification and validation of a neural network architecture capable of early fault detection and fault isolation in a typical three – tank system. Certain fault types are induced to the system and its behavior is monitored. Parameters such as water-level and temperature in the tanks, together with delayed samples are used to design, train and validate the neural network architecture. The neural network is further tested on a set of signal values derived from subsequent operation of the system, with considerable success.**

***Index Terms* -Neural networks, fault detection, fault interpretation.**

I. INTRODUCTION

Faults can be defined as non-permitted deviations of some characteristic properties of a process that will cause a certain level of deterioration in the performance of the process [1]. Faults are generated since the mechanical parts and materials used in devices and processes undergo aging, wear, etc., and, briefly, their properties are time dependent and tend mostly in the direction of lowering the operational capabilities, safety and reliability [2].

Fault detection procedures are called to decide whether a system is in normal operating conditions or in faulty ones, on the basis of real-time observations. On-line (real time) procedures are necessary for fault tolerant control, while off-line procedures can be used for maintenance purposes [3]. A variety of approaches have been proposed in recent years, for the design of efficient real-time fault detection and isolation procedures by both the control and artificial intelligence communities [4-8].

Real-time fault detection and isolation procedures can only make use of the system observables, i.e. the system inputs and outputs, along with their derivatives for a continuous-time model or along with their delayed (memorized) values on a given time horizon, for discrete-time models.

Using these observables, fault detection and isolation is essentially a two-level procedure:

i) the first level is that of *detection and alarm generation* (decision whether the system is in normal conditions or not) and
ii) alarm interpretation, i.e., deciding which faults are present among a pre-defined fault set and which are their characteristics (occurrence time, fault size, class, conesqu-ences etc.).

In this paper we restrict our attention to the class of physical processes, making use of the following two hypotheses:

a)the system may be described by a set of state variables: let $x \in R^n$ be the state vector at time t,
b)then, the true behavior obeys a set of differential equations:

$$\frac{dx}{dt} = \phi\left(x,u,v,\theta^*\right) \tag{1}$$

where $u \in R^m$ and $v \in R^l$ are, respectively, the control and the perturbation inputs, and $\theta^* \in R^q$ is a vector of the 'true' parameters.

Three basic models have been proposed for the description of the normal operation of the system:

1) The *behavioral model* describes the way the system state evolves in time, as a consequence of the system inputs (controls and perturbations). The model closest to equation (1) would be a set of differential equations of the form:

$$\frac{dx}{dt} = f\left(x,u,v,\theta\right) \tag{2}$$

where f is an approximation of φ and θ is the vector of the previously defined model parameters.

2) The *measurements model* describes the measurements that are available, in the form :

$$y=g(x,u,v,\theta,\varepsilon) \tag{3}$$

where $y \in R^p$ is the output vector and $\varepsilon \in R^p$ is the measurements noise. This model expresses the way under which the sensors transform some states of the process into output signals that can be used for fault detection and isolation.

3) The *operating range model* defines the values the system variables are allowed to take under normal conditions. A direct representation is given by:

$$h(x,u) \leq \eta \tag{4}$$

where $\eta \in R^k$, and equation (4) defines a domain in the state and control space in which the system operates safely.

According to which form of a model is used, system theory, signal processing and artificial intelligence approaches were used extensively in the literature [1-7].

The approach proposed in this paper uses, essentially, the measurements model applied to a three-tank system. A multi-layered artificial neural network is employed for the detection and isolation of faults that are induced to the system. Measurable quantities that are sensitive and informative about the system operation are used as symptoms that discriminate particular faults.

II. THE THREE-TANK SYSTEM

II.A. System Description

The three-tank system is a commonly used process [9], consisting of three cylindrical water tanks connected by pipes of circular cross-section. Usually, the first tank has an incoming flow that can be controlled by means of a pump and the outflow is located in the last tank. The relative positions of the tanks and the existence of additional features such as water heating or cooling results in a variety of configurations for three-tank processes. The three-tank process used in this paper is shown in Figure 1.

Two electronic valves control the water flow between the tanks, whereas, the pump recirculates water from the bottom tank to the top tank. A heating element (resistor) located at the bottom tank heats up the water, whereas a cooling fan placed in a perpendicular direction to the water flow cools down the water. A PID controller controlling the resistor and cooling fan is responsible for attaining an almost steady water temperature. Two water level sensors are placed in the two top tanks measuring water level and three thermocouples monitor the water temperature of each tank. The overall process is monitored and controlled through a Supervisory Control and Data Acquisition interface implemented using LabView [10].

Let q_u be the incoming water volumetric flow to the top tank, s_i the cross-section of tank i, h_i the water level of tank i, q_{ii} a volumetric flow due to possible leakage in the i-tank and q_l is the water volumetric flow after the pump. Let, also, T_i be the temperature in tank i, Tu the temperature after the fan, Q_{ii} the heat losses in tank i, Q_F the heat removal by the fan and Q the heat supply provided by the resistor. Assuming that the density ρ *and the specific heat* c_p are constants and the incoming and outgoing water volumetric flows are independent since they are controlled by the valves and not by hydrostatic pressure the mathematical model of the system is:

$$
\begin{aligned}
& s_1 \cdot \frac{dh_1}{dt} = q_u - q_{12} - q_{11} \qquad s_2 \cdot \frac{dh_2}{dt} = q_{12} - q_{23} - q_{22} \\
& s_3 \cdot \frac{dh_3}{dt} = q_{23} - q_l - q_{33} \qquad q_u = q_l \\
& s_1 \cdot \frac{dT_1}{dt} = \frac{q_u}{h_1}\left[T_u - T_1\right] - \frac{1}{\rho c_p}\frac{Q_{11}}{h_1} \\
& s_2 \cdot \frac{dT_2}{dt} = \frac{q_{12}}{h_2}\left[T_1 - T_2\right] - \frac{1}{\rho c_p}\frac{Q_{22}}{h_2} \\
& s_3 \cdot \frac{dT_3}{dt} = \frac{q_{23}}{h_3}\left[T_2 - T_3\right] + \frac{1}{\rho c_p}\frac{Q}{h_3} - \frac{1}{\rho c_p}\frac{Q_{33}}{h_3} \\
& T_u = T_3 - \frac{1}{\rho c_p}\frac{Q_F}{q_l}
\end{aligned} \tag{5}
$$

Together with possible leakages, a number of additional uncertainty factors influence the operation of the system. The incompleteness of the knowledge about a component's behavior and ageing process needs to be examined. The pipes, in the proposed system are only partially known, for example it is not exactly clear what chemical reactions are actually happening between the pipe walls and the inside flow. On the other hand, the conditions of use of a pipe can have large variations in the temperature of the flow [5]. Moreover, in a hot-water vessel, the temperature of the main water volume increases rapidly, and the temperature of the water in front of the cooling fan decreases rapidly (uneven temperatures of the flows). The heated water results in the dissolved mineral particles solidifying into a scale deposit in the heated tank.

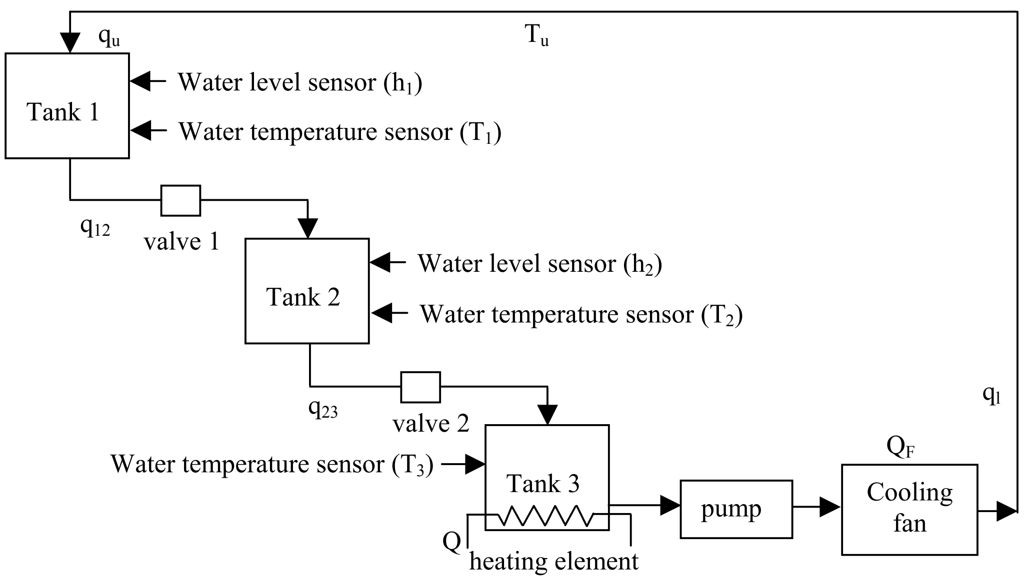


Figure 1: The three-tank system with sensors

Finally, erosions at the valve plugs, variations in water hardness, pipe and valve clogging, measuring sensor inaccuracies and system non-linearities when combined with the above mentioned inaccuracy factors make the system very difficult, or even impossible, to be expressed analytically in the form of equation (5) [1,9,11].

II.B. System operation and fault types

The normal operation of the proposed system is depicted in Figure 2(a). A simple water-level control algorithm uses the level sensor inputs to control the operation of the valves. If the water level in tank 1 exceeds $l1_{high}$ cm then valve 1 is opened (kept closed up to this level) and water flows to tank 2. Valve 1 is closed when the water level falls below $l1_{low}$ cm. Similarly, if the water level in tank 2 exceeds $l2_{high}$ then valve 2 is opened and water flows to tank 3. Valve 2 is closed when the water level falls below $l2_{low}$. A PID controller is employed to set water temperature at 35^{o}C, with the aid of the heating resistance and cooling fan.

In order to investigate the behavior of the proposed system under faulty conditions, a series of deliberate faults were induced. The abnormal behavior of the system under faulty conditions was monitored through the level and temperature sensors and it is shown on Figures 2(b-h), for the different fault types. After fault manifestation, the system is restored to normal operating conditions by removing the corresponding fault cause. This is in accordance to the most common faults scenarios used for such systems [7,5,11] and these scenarios are briefly described as follows:

i) *Fault type 1:* Valve 1 stuck closed. This fault is shown in Figure 2(b) and it is clear the water level in the first tank exceeds by far the normal value.
ii) *Fault type 2:* Valve 1 stuck open. This fault is displayed in Figure 2(c) and the water level in tank 1 reaches its minimum value.
iii) *Fault type 3:* Valve 2 stuck open: Figure 2(d) displays the minimum water level reached for tank2 and the temperature drop (due to lack of water).
iv) *Fault type 4:* Valve 2 stuck closed. Obviously, in this case the water level in tank 2 is raised far above the maximum value, as shown in Figure 2(e).
v) *Fault type 5:* Valves 1 & 2 both stuck open. In this case the water levels reach their minimum values in both tanks (first in tank 1 and then in tank 2, Figure 2(f)).
vi) *Fault type 6:* Pump switched off. In this case, water levels will also reach their minimum values, as long as the valves are operating (Figure 2(g)).
vii) *Fault type 7:* While the pump is switched off, valve 1 is closed. This results to a minimum level attained in tank 1 and an intermediate level attained in tank 2 (Figure 2(h)).

A fault detection and isolation procedure is required in order to detect faults as early as possible and to proceed to the necessary correcting procedures that will restore the process to its normal operation.

Due to the inherent uncertainties and measuring errors described in the previous section, the *measurements model* approach is adopted in this paper. Input /output measurements of the system under normal and faulty operating conditions are used to train a multi-layered feed-forward neural network that is capable to generalize and discriminate among normal and faulty system behavior. This neural network can then be used for fault prediction and identification.

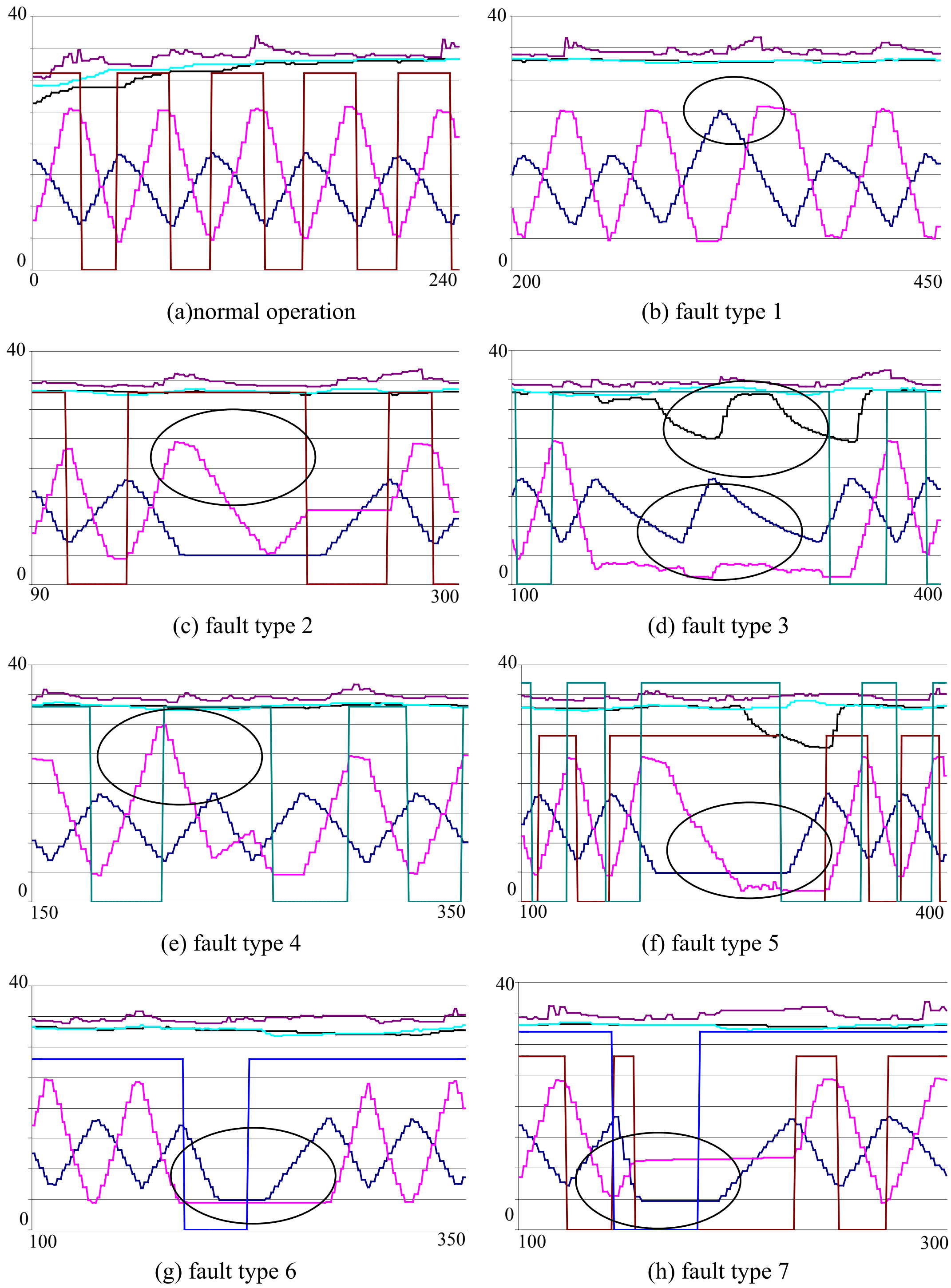
III. NEURAL NETWORK ARCHITECTURE

III.A. System Identification

The determination of the input signals that influence the system output in such a way that different output behaviors can be differentiated (i.e. differentiate among the different behaviors depicted in Figure 2) is, essentially, a parameter identification problem [11,12]. The choice of input signals influencing the output can be made by observing the output behavior. Additionally, input delays can be determined by observation of the time delay occurring between a change in the input signal and the related reaction of the output. This leads to the system model shown in Figure 3. However, the exact measurement of the delay time may be difficult, especially in the presence of noise [12]. Moreover, since in our case the system structure is assumed unknown, the choice of the model's structure is not evident.

In this paper, the determination of the delayed sample points that are needed for fault type discrimination was achieved by comparing the estimation error achieved by a large number of possible neural network architectures. The maximum number of samples per variable k was increased, starting from k=1. For each k, a large number of possible combinations were evaluated by the respective neural network architectures and k was increased until the minimal estimation error $\varepsilon_{min}(k)$ is achieved.

It has been shown [12] that these methods may lead to local minima, which means, that the obtained model structure is satisfactory for the identification data, but does not describe the system's behavior correctly in other nominal operational modes. In the neural network architecture proposed in this paper, the risk of choosing such a local model was minimized by model validation [13] for sets of measurements not used in the training set. Moreover, model checking for new data sets, after inducing the same faults, served as measure of the neural network performance [13,14].



Water level Tank 1 *(h_1)*:
Water level Tank 2 *(h_2)*:
Water temperature Tank 1 *(T_1)*:
Water temperature Tank 2 *(T_2)*:
Water temperature Tank 3 *(T_3)*:
(diagrams not to scale)

Horizontal axes: sampling time (s)
Vertical axes: levels (cm)
Temperatures (^{o}C)

Figure 2: a) normal operation
b-h) fault types

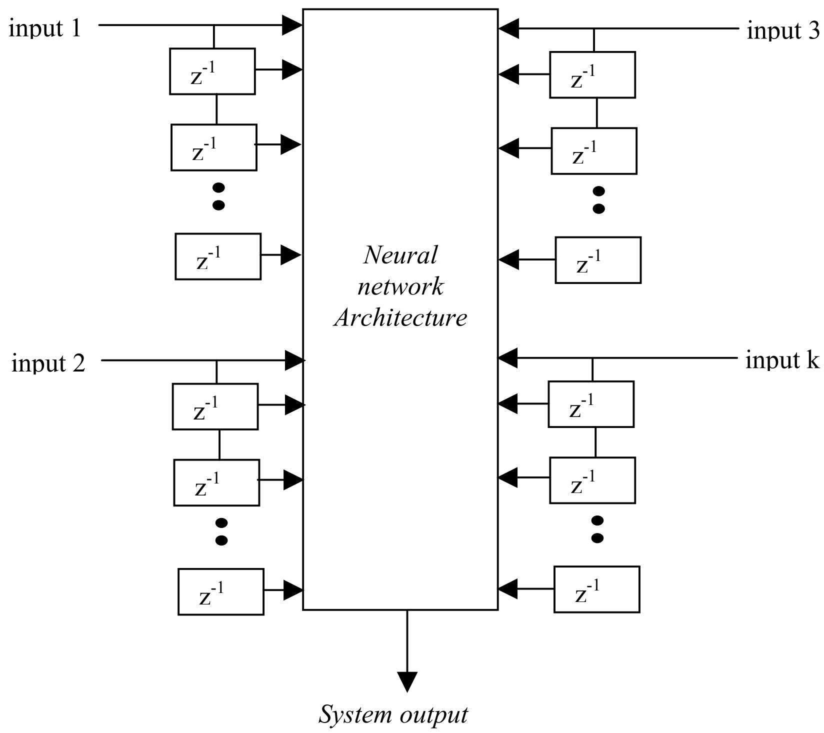


Figure 3: Identifying the network structure

III.B. Neural network design

In real-time systems, one cannot wait for a large number of delayed samples in order to provide an accurate fault prediction, since fault detection must be achieved as soon as possible (and trigger the proper alarms). In this paper, the number of previous samples for each of the 5 input parameters (water level in tank1, tank2, and water temperatures in tanks 1,2,3, respectively) was kept within the range *n-1* to *n-10* (moreover, this limits somehow the enormous number of possible input combinations, which is still, very high). By repetitive design, training, verification and validation of a large number of neural network architectures it was revealed that the delays in the temperature signals did not play significant role in fault discrimination, whereas, the temperature signals themselves, are quite important. This was as expected, since the temperature signals do not exhibit significant gradient changes in normal operation, except for the initial 'warming up' phase, as it is shown in Figure 2(a). In faulty conditions, their behavior is also distinctive. For the water level signals, it was expected that similar samples should be equally important for both signals, owing to their qualitatively symmetrical (out of phase) behavior depicted in Figure 2. It was found that although the combinations of 2 delayed samples for each water level signal (*n-2, n-5*, for both signals) resulted in small classification errors in the validation set, they were not very successful in the processing of the checking set (a data set derived from subsequent operation of the system). Further design and training of neural networks revealed the fact that by using the *n-2, n-5* and *n-8* samples for both *l1* and *l2*, the training set, validation set and checking set fault classification errors can be minimized.

A standard accelerated backpropagation training algorithm with momentum was used. Thus, the minimization of the sum squared error [13,14] for the elementary multi-layered architecture shown in Figure 4 is achieved according to:

e=d-y where e is the error and d the desired output.
f,g are the neuron activation function and its derivative.
$\sigma_1,\sigma_2,\sigma_3$ are the weighted sums of the inputs for each neuron.

$$\frac{\partial e^2}{\partial(w_{ij})} = 2e\frac{\partial e}{\partial(w_{ij})}$$

$$\frac{\partial e}{\partial w_{11}} = -\frac{\partial y}{\partial w_{11}} = -\frac{\partial f(\sigma_3)}{\partial \sigma_3}\frac{\partial \sigma_3}{\partial w_{11}} = \tag{6}$$

$$= -\frac{\partial f(\sigma_3)}{\partial \sigma_3}\frac{\partial \sigma_3}{\partial f(\sigma_1)}\frac{\partial f(\sigma_1)}{\partial \sigma_1}\frac{\partial \sigma_1}{\partial w_{11}} = -g(\sigma_3)g(\sigma_1)v_1 x_1$$

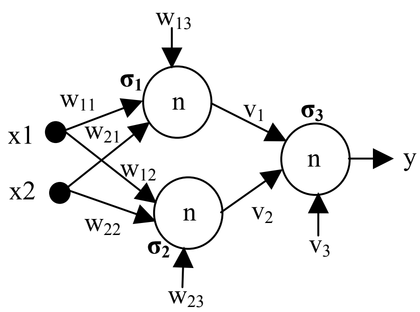


Figure 4: Elementary two-layered feed forward neural network

The characteristics of the best two-layered, feed-forward neural network architecture obtained after the large number of input combinations are shown on Table 1. A pre-processing step of input /output normalization preceded the training phase, normalizing input/output vectors for zero mean and standard deviation of one [14]. The samples were almost evenly distributed among the different fault types (including the normal operation) and 3 output neurons were used as a binary classification label for each of the fault types and normal operation. The number of neurons in the hidden layer was chosen by a repetitive - direct search algorithm implemented to choose the network structure with the best performance. The bipolar sigmoid activation function (denoted as *tansig* in MatLab) was used both for the hidden and output layers. It can be expressed as:

$$f(\sigma) = \frac{2}{(1+\exp(-\sigma))} - 1 \qquad (7)$$

Table 1: Neural Network characteristics

Inputs: h_1, h_2, T_1, T_2, T_3, h_1 *delayed samples :n-2, n-5, n-8* h_2 *delayed samples :n-2, n-5, n-8*
Number of hidden layer neurons: *50*
Output neurons: *3*
Number of Training set samples: *1044*
Number of Validation set samples: *1044*
Number of Checking set samples: *500*
Validation set success rate: *99%*
Checking set success rate: *99%*

The validation vectors were used to stop training early since further training on the primary vectors will hurt generalization to the validation vectors.

Thus, the proposed neural network architecture is capable of discriminating between the different types of faults shown in Figure 2. After completing the training and validation phases, the computational speed of the proposed neural network is quite fast since only feed forward calculations are employed. Therefore, faults can be detected in real-time and this is a significant advantage since the fault restoration procedures can begin immediately. The computational cost related to fault detection by using high order correlation functions, multi-spectra density functions and the Fourier Transform [2,15], is much higher.

The proposed method avoids complex matrix inversion problems met in the design of fuzzy relational models for fault detection [12]. Moreover, although there exists some overlap of fault features (e.g. fault type 5 and fault type 6, for the interval during which both water levels are almost at minimum values, as shown in Figure 2), the use of delayed samples manages to provide enough information for fault discrimination, in contrast to [7] where feature overlap is not completely resolved.

Thus, the proposed neural network can be used for early detection and alarm generation in case the system deviates from normal operation. Furthermore, it is capable of alarm interpretation i.e., deciding which fault is present among a pre-defined fault set.

IV. CONCLUSIONS

A physical system is specifically at risk if it is not monitored, if some of its components need regular maintenance, if some of its components are insufficiently known, regarding their dynamical behavior and ageing process, or its conditions of use are not controlled and can widely fluctuate. This paper presented the design, training, verification and validation of a neural network architecture capable of early fault detection and fault isolation in a typical three–tank system i.e., deciding which fault is present among a pre-defined fault set. Faulty conditions were deliberately induced to the system and its' behavior was monitored by appropriate sensors.

In terms of system parameter identification, a number of delayed samples were required in order to built a neural network model that minimizes both training and validation errors. The proposed architecture compares favorably to other methods in terms of complexity and speed. It was also further tested on a set of 'checking' signals, derived from subsequent operation of the system, with remarkable success.

REFERENCES

[1] R.P. Leger, Wm. J. Garland and W.F.S. Poehlman, Fault detection and diagnosis using statistical control charts and artificial neural networks, Artificial Intelligence in Engineering, 12, 1998, 35-47.

[2] H.G. Natke and C. Cempel, The symptom observation matrix for monitoring and diagnosis, Journal of Sound and Vibration, 248 (4), 2001, 597-620.

[3] M. Staroswiecki, Quantitative and Qualitative models for fault detection and isolation, Mechanical Systems and Signal Processing, 14(3), 2000, 301-325.

[4] R. Iserman, Process fault detection based on modeling and estimation methods – a survey, Automatica, 20, 1984, 387-404.

[5] D. Rozier, A strategy for diagnosing complex multiple-fault situations with a higher accuracy/cost ratio, Engineering Applications of Artificial Intelligence, 14, 2001, 217-227.

[6] I.Y. Tumer, K.L. Wood and I.J. Busch-Vishniac, Monitoring of signals from manufacturing processes using the Karhunen-Loeve Transform, Mechanical Systems and Signal Processing, 14(6), 2000, 1011-1026.

[7] P. Balle and D. Fuessel, Closed-loop fault diagnosis based on a nonlinear process model and automatic fuzzy rule generation, Engineering Applications of Artificial Intelligence, 13, 2000, 695-704.

[8] S. Wang, Zh. Jiang, Valve fault detection and diagnosis based on CMAC neural networks, Energy and Buildings 36, 2004, 599–610.

[9] M.A. Sainz, J. Armengol, J. Vehi, Fault detection and isolation of the three-tank system using the modal interval analysis, Journal of Process Control, 12, 2002, 325-338.

[10] National Instruments Corporation, Labview: Graphical Programming for Instrumentation, 2000.

[11] F. J. Garcia, V. Izquierdo, L.J. de Miguel, J.R. Peran, Fault-diagnostic system using analytical fuzzy redundancy, Engineering Applications of Artificial Intelligence, 13, 2000, 441-450.

[12] P. Amann, J.M. Perronne, G.L. Gissinger and P.M. Frank, Identification of fuzzy relational models for fault detection, Control Engineering Practice, 9, 2001, 555-562.

[13] S. Haykin, Neural networks, IEEE Press, 1994.

[14] Mathworks Inc., The Neural Network Toolbox, MatLab ver. R14, 2004.

[15] C.K. Mechefske, Objective Machinery Fault diagnosis using Fuzzy Logic, Mechanical Systems and Signal Processing, 12, 1998, 855-862.