Fault Detection of VVER-1000 NPP Using MLP Neural Network and Dynamic Sequential Windows

Hadad [1,2], K., Mortazavi[1] M., Nematollahi[1], M.R.
[1]Department of Nuclear Engineering, Shiraz University, Shiraz Iran
[2]Nuclear Safety Research Center, Shiraz University, Shiraz Iran

hadadk@shirazu.ac.ir, mortazavi.m.m@gmail.com, nema@shirazu.ac.ir

Abstract
This paper presents an approach for neural network based fault diagnosing system which allows either dynamic identification. The approach uses a multilayer perceptron neural network (MLP) trained with the resilient backpropagation algorithm. This neural network (NN) uses, as input, a short set (in a moving time window) of recent measurements of each variable avoiding the necessity of using starting events. The data used in this work contains 10 simulated transients by using VVER-1000 simulator data. For evaluation of the method robustness, white noise was added to data. This work has shown that the fault diagnostic advisor, developed by using ANNs, is effective in providing proper diagnoses for all 10 transients analyzed at various severity levels (even when the transients are degraded by noise).

Keywords: fault diagnosis, sequential windows, artificial neural network, multi-layer perceptron, resilient backpropagation, VVER-1000.

1. Introduction
Nuclear power plants (NPPs) are extremely complex systems that are operated and monitored by human operators. NPPs have special features, not only in the system's complexity, but also in the potential harmful results from control errors. When faced with transient, such as a plant accident scenario, equipment failure or an external disturbance to the system, the operator has to execute regular and suitable operations according to the operating procedures and corrective actions. The importance of human factors in NPPs considered since the 1980s. A total of 180 significant events in NPPs have been reported in the United States, it was found that 48% of the incidents were attributed to human factor failures (Hwang and Hwang, 2003)[1]. Therefore, failures detection, identification, and accommodation (FDIA) has always been an important aspect of fault tolerant control system design. In a nuclear power plant, for example, tens of alarms can occur in a few second after a fault. It necessitates developing a system that will assist the operator to identify such transients at the earliest stages of their developments. Early detection will help in

minimizing or even mitigating the negative consequences of such transients. It is equally important to identify the type of transient correctly.
The transient identification system for NPPs has been developed using techniques such as artificial neural networks (ANN). Initially, these works (Cheon and Chang, 1993[2]; Ohga and Seki, 1993[3]) explored multilayer Artificial Neural Networks (ANN) with backpropagation training, as this kind of ANNs has an excellent capacity of approximation and generalization.
The objective of the plant diagnostic system in any potentially unsafe scenario is to give the plant operators appropriate inputs to formulate, conform, initiate and perform the corrective actions. The event detection can be classified as a pattern recognition problem. When an event occurs starting from the steady-state operation, instruments' readings develop a time-dependent pattern and these patterns are unique with respect to the type of the event (Guo Z, Uhrig RE. 1992)[4]. Therefore, by properly selecting the plant parameters, the initial events (IEs) can be distinguished. Various neural network algorithms such as back propagation, resilient-back propagation, quick propagation, manhattan, etc. have been studied (Klerfors D., 1998) [5]. After carrying out an optimization study, it is observed that the resilient backpropagation algorithm converged to a solution with a minimum error compared to all other algorithms. Based on this algorithm, an event-identifying framework known as symptom based diagnostic system has been developed for diagnosing several events of a typical NPP (Santosh T.V. et al., 2006) [6]. This will help the operator in identifying the event in advance. In this study we have used backpropagation resilient algorithm for a MLP neural network to predict the transient event by looking into sequential windows of 12 main parameters of the system.

Table 1 (T.V.Santosh et. Al, 2003)
Result of optimization study

ANN algorithm	Number of epochs	Minimum error	Average error (%)	CPU time (s)
Resilient-back propagation(batch)	95463	3.82E-11	3.25	5472.204
Standard back propagation(pattern)	325611	9.95 E-11	8.98	21866.031
back propagation with momentum (pattern)	262335	9.96 E-11	7.59	12242.47
back propagation with momentum (batch)	550100	5.50 E-09	8.62	23561.66
Quick propagation (batch)	484793	3.60 E-10	10.92	25537.04
Manhattan (batch)	459126	4.90 E-09	11.74	23729.23
Delta bar delta (batch)	504532	8.20 E-10	9.67	27836.45

2. Artificial Neural Networks

2-1. Multilayer Feed Forward Neural Networks

ANNs consist of a great number of processing elements (neurons), which are connected with each other; the strengths of the connections are called weights. For the modeling of physical systems, a feed-forward multi-layered network is commonly used. It consists of a layer of input neurons, a layer of output neurons and one or more hidden layers. In a multi-layer perceptron (MLP) (Figure.1), there are no connections between the neurons in a given layer, so that the information is transferred from the (l–1)th layer to the lth one. External data enter the network through the input nodes and, after typically non-linear transformations, output data are generated by the output nodes.

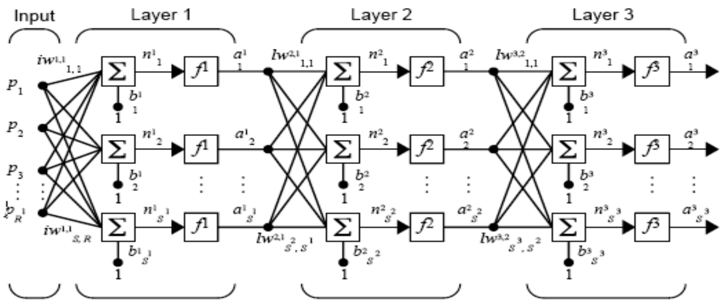

Fig.1. A multilayer perceptron (MLP) neural network.

In ANNs, the knowledge lies in the interconnection weights between neurons. Therefore, learning process is an important characteristic of the ANN methodology, whereby representative examples of the knowledge are iteratively presented to the network, so that it can integrate this knowledge within its structure (training phase). During the training phase, the weights are successively adjusted based on a set of inputs and the corresponding set of desired output targets. First, the inputs are presented to the network and propagated forward to determine the resulting signal at the output neurons; the difference between the computed output vectors and the desired output targets represent an error that is back-propagated through the network in order to adjust the weights. This process is repeated a great number of times and the learning continues until the desired degree of accuracy is achieved (Haykin, 1999) [7]. According to the back-propagation algorithm, when an input is presented to the network, the activation of each neuron is determined by:

$$o_i = y_i(\sum_j w_{ij} o_j + \theta_i) \tag{1}$$

where o_i is the activation of unit i, y_i is the activation function of unit i, w_{ij} is the weight from unit j to i, and θ_i is the bias for unit i. Back propagation is then invoked to update all the weights in the network according to the following rule:

$$\Delta w_{ij}(n+1) = \eta \cdot (\delta_i \cdot o_j) + m \cdot \Delta w_{ij}(n) \qquad (2)$$

where n is the iteration number, η is the learning rate, δ_i is the error signal for unit i, and m is the momentum factor. The error signal δ_k for an output unit k is calculated from the difference between the desired value d_k and actual value y_k for that unit, while the error signal δ_h for a hidden unit h is a function of the error signals of those units in the next higher layer connected to unit h and the weights of those connections. The two parameters η and m represent, respectively, an adjustment of step size and a weight on the 'memory' of previous steps. Assuming η and m are appropriately chosen, the back-propagation process will generally converge to a minimum that satisfies the criterion imposed by the user, usually the sum of the squares of the error of the output signals, $\sum_k (d_k - y_k)^2$, is less than a predetermined value.

2-2. The Resilient Backpropagation Learning Algorithm

Resilient-back propagation algorithm is a local adaptive learning scheme performing supervised batch learning in multi-layer perceptrons. The basic principle of this algorithm is to eliminate the harmful influence of the size of the partial derivative on the weight step. Consequently, only the derivative is considered to indicate the direction of the weight update.

There are two factors associated with the learning rate: the learning-rate increment factor η^+, and the learning-rate decrement factor η^-. The most suitable values for η^+ and η^-, found experimentally, are 1.2 and 0.5, respectively (Klerfors D., 1998) [5]. While learning starts, all the update values are set to an initial value Δ_0. In order to prevent the weights becoming too small, the lower bound is set to the minimum weight step by $\Delta_{\min}$, and to prevent the weights becoming too large, the upper bound is set to the maximum weight step by $\Delta_{\max}$. Here, the update value Δ_{ij}, for each weight, determines the size of the weight update. This adaptive update value evolves during the learning process based on its local sight of the error function E, according to the following learning rule:

$$\Delta_{ij}(t)=\begin{cases}\eta^{+}*\Delta_{ij}(t-1) & if\ \dfrac{\partial E}{\partial\omega}(t-1)>0,\\ \eta^{-}*\Delta_{ij}(t-1) & if\ \dfrac{\partial E}{\partial\omega}(t-1)<0,\\ \Delta_{ij}(t-1) & else,\end{cases} \tag{3}$$

$$0<\eta^{-}<1<\eta^{+}.$$

Once the update value for each weight is adapted, the weight update is done as under:

$$\Delta\omega_{ij}(t)=\begin{cases}-\Delta_{ij}(t) & if\ \dfrac{\partial E}{\partial\omega}(t)>0\\ +\Delta_{ij}(t) & if\ \dfrac{\partial E}{\partial\omega}(t)<0\\ 0 & else.\end{cases}$$

(4)

However, there is one exception: if the partial derivative changes its sign, it indicates that the previous weight step was too large and the minimum was missed. Hence, the previous weight update is reverted as

$$\Delta\omega_{ij}(t)=-\Delta\omega_{ij}(t),\quad if\ \frac{\partial E}{\partial\omega}(t-1)*\frac{\partial E}{\partial\omega}(t)<0.$$

(5)

In addition, $(\partial E/\partial\omega)(t)=0$ is set to avoid the update being done twice. The sigmoid activation and mean square error functions are adopted in this algorithm.[8]

3. Transient Identification Method

A study on various artificial neural network (ANN) algorithms for selecting a best suitable algorithm for diagnosing the transients of a typical nuclear power plant is done by T.V. Santosh et. al (2006) [6]. The objective of this study is to develop a neural network based framework that will assist the operator to identify such initiating events quickly and to take corrective actions. Optimization study on several neural network algorithms has been carried out. These algorithms have been trained and tested for several initiating events of a typical nuclear power plant. The study shows that the resilient-back propagation algorithm is best suitable for this application (Table 1) (T.V. Santosh et. al, 2006) [6]. The MLP has a good capacity of generalization. However, since this type of ANN does not have recurrence in its internal structure (where there is a direct dependence upon time), it is not able to directly work with dynamic systems. In this way, aiming at overcoming this limitation, it was adopted the system of ‘‘movable sequential window’’ (MSW), Table 2. In order to define the number of elements of this ‘‘window’’, it is necessary to consider that this window must be large enough to identify the dynamic behavior of the system. The method of fault detection is based on a type of ANN architecture represented in Figure.2. ,

Table 2
Movable sequential windows

Variable $x_1(t)$			…	Variable $x_k(t)$			Event
Input 1	Input 2 …	Input m		Input 1	Input 2 …	Input m	Output
$x_1(t)$	$x_1(t+1)$	$x_1(t+m)$		$x_k(t)$	$x_k(t+1)$	$x_k(t+m)$	1
$x_1(t)$	$x_1(t+2)$	$x_1(t+m+1)$		$x_k(t)$	$x_k(t+2)$	$x_k(t+m+1)$	1
:	:	:		:	:	:	:
:	:	:		:	:	:	:
$x_1(t+n-m)$	$x_1(t+n-m+1)$	$x_1(t+n)$		$x_k(t+n-m)$	$x_k(t+n-m+1)$	$x_k(t+n)$	1

k-number of input variables; m-number of elements of the ‘‘window’’; n-size of temporal series.

using as input the recent history of the process variables of the power plant. As this type of architecture allows that the presented value at its output can assume continuous values, and on the other hand, the transient classification system adopted represents every event by a discrete integer value (that is, event A is represented by the value 1, event B is represented by the value 2, and so on successively for all anticipated events), the values presented at the network output, for each event, oscillated around the anticipated discrete value.

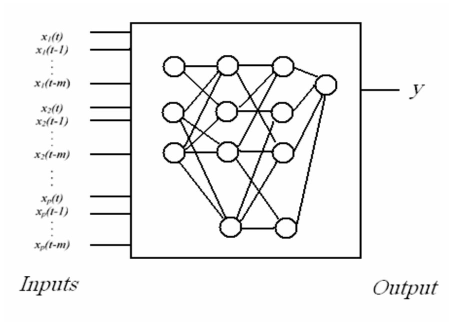

Fig. 2. ANN with movable sequential windows.

In this way, the acceptable tolerance (Eev) of each transient was determined as being the greatest deviation between the expected value and the value presented at the network output for the respective transient.

$$E_{ev} = Maximum \; |y_{ev} - \hat{y}_{ev}| \quad (6)$$

where: ev indicates the event, yev is the expected value, $\hat{y}_{ev}$ is the value obtained by the ANN. This method is applied for fault detection for a VVER-type NPP by data generated by VVER-1000 simulator, operating at 100% power level. This was achieved using a CEPAC simulator [9] of PWR and correcting its input database according to VVER-1000 system parameters. Besides the normal condition, the data of 9 transient events were generated including: (1) Hot leg break (HLB) ; (2) Cold leg break (CLB) ; (3) Upper head Control Element Assembly leak (UHCL); (4) Upper head seal leak (UHSL) ; (5) Stuck atmospheric dump valve (SADV); (6) MSSV leak (MSSVL) ; (7) Steam line break (SLB) ; (8) Steam generator tube rupture (SGTR) ; (9) Loss of main feed water on SG (LOMF). Each one of these transients was described by values of 12 state variables that are presented in Table 3. After defining the input variables and the group of accidents (output numbers) the training set for ANN was formed. The training set contained 110 patterns without noise and 440 patterns with noise. A white noise with SNR=70 to 90 were imposed on raw data. Based on the maximum and minimum possible values of the variables, raw data were normalized to a continuous range from 0 to 1 before training. The normalization made the ANN learning easier, because the original input data contained both small and large scale. In this system, MLP neural network with configuration of 150×200×100×50×25×5×1 has found to have the smallest error.

Table 3
State variables used in the test scenario

Variable number	State variable	Unit
1	Pressurizer pressure	MP
2	Pressurizer level	Percent (%)
3	Loop 1,2 Hot Leg temperature	^oC
4	Loop 1,2 Cold Leg temperature	^oC
5	Sub cooling	^oC
6	Steam generator (SG)1,2 pressure	MP
7	SG 1,2 Narrow range level	%
8	SG 1,2 Wide range level	%
9	Loop 1,2 Main feed water flows	Kg/sec
10	SG 1,2 Steam flow	Kg/sec
11	Core power	%
12	Core flow	%

In other words, the ANN model has 150 inputs consisting of 15 input variables of system, 5 hidden layers and 1 neuron in the output layer. Note that the activation functions used in the hidden layers and output layer are logarithmic and pure linear, respectively. The network is trained over 50000 epochs with error back propagation training. The network is evaluated by comparing the prediction to the true output, resulting in a prediction error. The training result is shown in Figure 3.

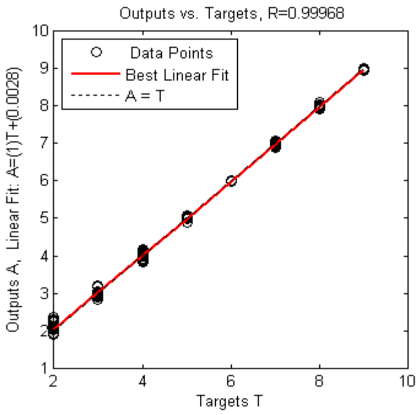

Fig.3 –The training results with the MLP network

The network was trained for all but HLB and LOMF accidents to evaluate its response for non-trained events. As shown in figure 4.a to 4.h, the program estimates the initial event correctly from its previous training. Although in HLB and LOMF events, the abnormality of system is detected, but the initial event remains unknown. This is an important outcome of the trained network, since a response similar to the previously trained events would mislead the operator in responding to the actual event.

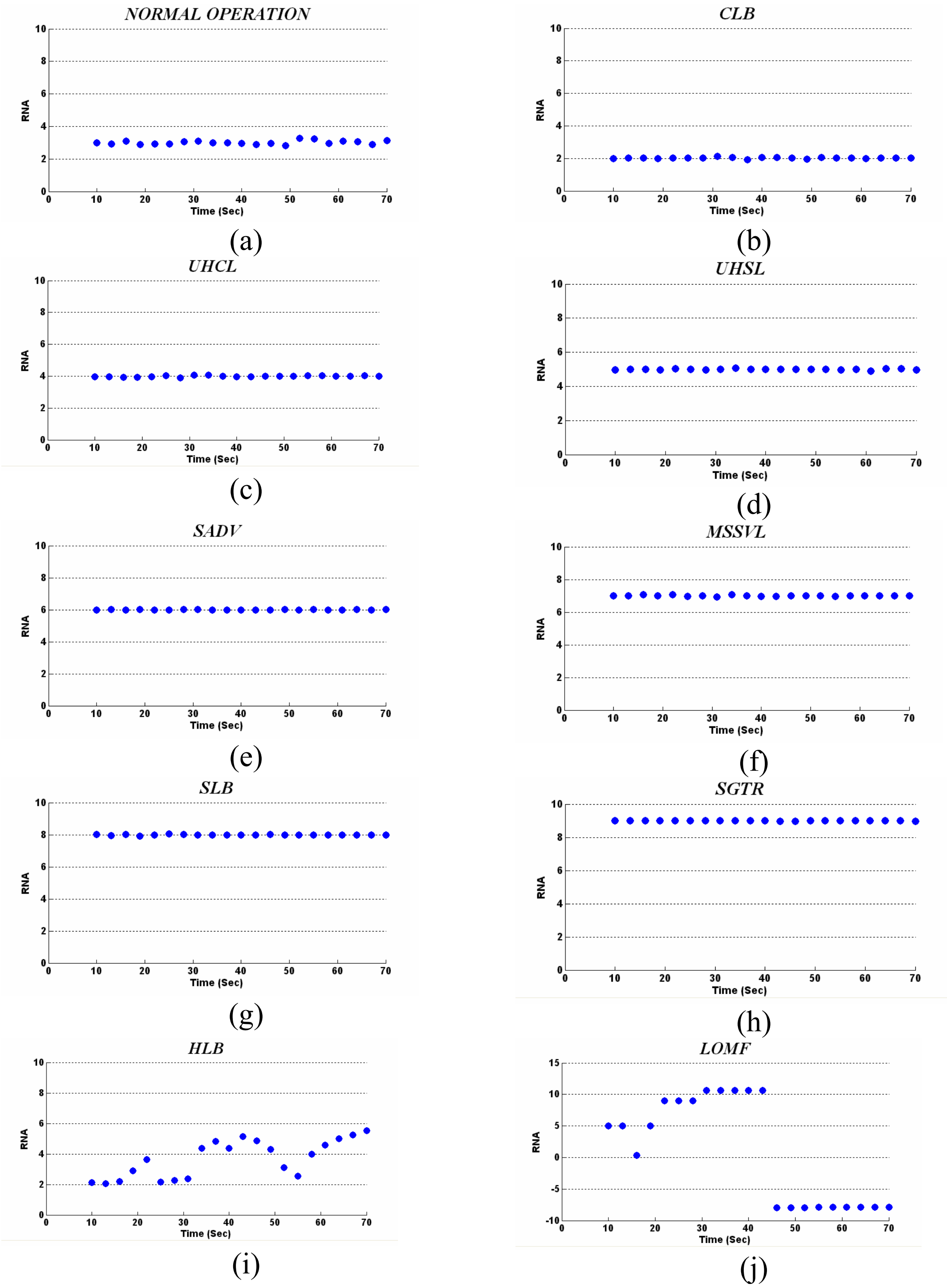

Fig.4. Response of ANN to fault detection

Once the ANNs trained, the acceptable tolerance (Eev) was obtained for the accidents and results shown in Table 4.

Table 4
acceptable tolerance of each accident

Event	ANN Code	Eev
CLB	2	0.1123
UHCL	4	0.1287
UHSL	5	0.1152
SADV	6	0.0099
MSSVL	7	0.0965
SLB	8	0.1112
SGTR	9	0.0305
Normal operation	3	0.2763

As a final test, the system response in 3 second intervals during the transient of different initial events is presented in table 5. As seen in this table, the system originally operates in normal condition. when the events start, the network response changes to UNKNOWN for few intervals and in the following intervals the initial event is detected correctly. However for non-trained events, the response continues to stay UNKNOWN.

Table 5
Response of the system to event

Time	CLB	SLB	SGTR	LOMF
1	_	_	_	_
3	_	_	_	_
6	_	_	_	_
9	NORMAL	NORMAL	NORMAL	NORMAL
12	NORMAL	NORMAL	NORMAL	NORMAL
15	NORMAL	NORMAL	NORMAL	NORMAL
18	NORMAL	NORMAL	NORMAL	NORMAL
21	UNKNOWN	UNKNOWN	UNKNOWN	UNKNOWN
24	UNKNOWN	UNKNOWN	UNKNOWN	UNKNOWN
27	UNKNOWN	UNKNOWN	UNKNOWN	UNKNOWN
30	UNKNOWN	UNKNOWN	UNKNOWN	UNKNOWN
33	UNKNOWN	UNKNOWN	UNKNOWN	UNKNOWN
36	CLB	SLB	SGTR	UNKNOWN
39	CLB	SLB	SGTR	UNKNOWN
42	CLB	SLB	SGTR	UNKNOWN
45	CLB	SLB	SGTR	UNKNOWN
.	.	.	.	.
.	.	.	.	.
.	.	.	.	.
66	CLB	SLB	SGTR	UNKNOWN
69	CLB	SLB	SGTR	UNKNOWN

4. Conclusion

In this work, based on ANNs, a fault detection system was presented. This system aims in helping the operator to make decisions relative to the procedures to be followed in accidents and transient conditions in a nuclear power plant. The results was verified that the MLP-type ANN, with ‘‘resilient backpropagation’’ training algorithm, was able to diagnose all the accidents postulated for a PWR nuclear reactor, in a reasonable time. Even with noise addition to the accident parameters data, identification of the accident in 15 seconds following initiating event was achievable. An important feature of this study is the use of sequential windows. This feature allows the system to became independent of a trigger signal (which indicate the beginning of the transient), thus making it robust in relation to this limitation.

The results are encouraging and indicate that the neural network techniques have the potential to enhance the performance and safety of complex systems in a cost effective way. It could be concluded that the proposed system could function as an operator support system and can assist operators during abnormal situations and minimize the operator errors.

5. Acknowledgement

The authors are grateful to the Shiraz University for supporting this research.

6. References

[1] Hwang F.H., Hwang S.L. Design and evaluation of computerized operating procedures in nuclear power plants. Ergonomics. 46, 271-284, 2003.

[2] Cheon S.W., Chang S.H. Application of neural networks to a connectionist expert system for transient identification in nuclear power plants. Nuclear Technology 102, 177, 1993.

[3] Ohga Y., Seki H. Abnormal transient identification in nuclear power plants using neural network and knowledge processing. Nucl. Technol. 101, 109, 1993.

[4] Guo Z, Uhrig R.E. Use of artificial neural networks to analyze nuclear power plant performance. Nucl. Technol. 99, 1992.

[5] Klerfors D. Artificial neural networks. What are they? How do they work? In what areas are they used?, An individual project within MISB-420-0, St.Louis, November 1998.

[6] Santosh T.V., Gopika Vinod, Saraf R.K., Ghosh A.K., Kushwaha H.S. Application of artificial neural networks to nuclear power plant transient diagnosis. Reliability Engineering and System Safety, 2006.

[7] Haykin, S., 1999. Neural Networks—A Comprehensive Foundation. Prentice-Hall Inc, New Jersey.
[8] Poularikas A. Handbook of neural network signal processing, CRC Press LLC, 2002.
Antonio C. de A. Mol, Aquilino S.M., Schirru R. A neural model for transient identification in dynamic processes with ‘‘don’t know’’ response. Annals of Nuclear Energy 30, 1365–1381, 2003.
[9] CEPAC 2.1, Combustion Engineering Plant Analysis Code, Combustion Engineering USA, 1995.