

Artificial Neural Network Model for Intermediate Heat Exchanger of Nuclear Reactor

Subhra Rani Patra
Junior Research
Fellow
Computer division
E & I Group
IGCAR (DAE)
Kalpakkam-603102

R.Jehadeesan
SO/F, CSS Computer
division
E & I Group
IGCAR (DAE)
Kalpakkam-603102

S.Rajeswari
Head, CSS
Computer division
E & I Group
IGCAR (DAE)
Kalpakkam-603102

S.A.V
Satyamurthy
Head, Computer
division
E & I Group
IGCAR (DAE)
Kalpakkam-603102

ABSTRACT

In this article we illustrated the applications of artificial neural network in a nuclear reactor. We have designed and developed an Artificial Neural Network (ANN) model for process modeling of Intermediate Heat Exchanger (IHX) subsystem in nuclear reactor. Based on the theoretical calculations carried out for primary and secondary outlet temperatures of IHX the multi layer neural network is implemented with back propagation algorithm for modeling of the IHX. It has several advantages as compared to the conventionally calculated results. The results show that neural network model gives faster and more accurate prediction of output for a given data.

Keywords

Neural network, intermediate heat exchanger, multi layer perceptron, back propagation algorithm

1. INTRODUCTION

Nonlinear empirical systems can be modeled using powerful algorithms of neural network. Neural network is a powerful tool for identification of relevant physical parameters which are difficult to measure conventionally for a complex system like nuclear reactor [1]. Artificial neural network need to know little about the process itself, therefore a network can be constructed and trained with a set of representative data from the process. Hence it is also referred as Black Box modeling [7]. The feed forward neural network model learns about the reactor in steady state conditions and performs nonlinear regression analysis. The main objective of regression analysis is to predict the value of continuous variables even if the variable interactions are not completely understood [7, 9].

A neural network has great potential for parallelism, since the computations of the components are largely independent of each other. ANN offers many benefits in processing information such as

- Parallelism (many neurons compute at once).
- Ability to learn how to solve a task from examples.
- Ability to adapt and generalize over time.
- Fault tolerance, resistance to noisy data

2. NEURAL NETWORK BASICS

A neural network (NN) model is composed of many simple processing elements, or units called neurons, whose functionality is compared with human brain. The NN is a parallel, distributive structure whose main objective is to transform the inputs into meaningful outputs. In NN the information or knowledge is encoded in the interconnection weights between the processing elements. The multilayer perceptron (MLP) is most commonly used NN model consisting of an input layer, a hidden layer and an output layer. A NN model having one hidden layer is most frequently used though by having two hidden layers a model can solve any type of complex problems. Back propagation algorithm (recursive) used is the simplest among all in MLP model. At first, the inputs are presented to the input nodes and propagated forward through the hidden layer to compute the corresponding outputs. Then the difference between the measured and predicted output is back propagated so as to adjust the weights after each iteration. This process is repeated so that finally the difference between the measured and predicted outputs, represented as mean square error (MSE), would approach to zero.

The learning process is very important in neural network. It is represented as “search process” in a multidimensional weight space for a solution, which governs how exactly the weight change occurs to give an optimized solution. The NN becomes more knowledgeable about its environment as the iteration process continues [8]. The different types of learning process are

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

Supervised learning consists of a series of inputs with known expected output and the network attempts to minimize error. Unsupervised learning only supplies inputs. The neural network adjusts its own weights so that similar inputs cause similar outputs. There are no desired outputs in this case. In case of reinforcement learning instead of waiting until the output is evaluated, the neural network is given feedback on its progress throughout its evaluation of the data. This feedback allows the network to strengthen or weaken links between nodes. The back propagation algorithm calculates the direction of steepest

descent on the error surface and jumps down the surface a distance proportional to learning rate. It also considers a momentum parameter to maintain a consistent direction in the error surface.

3. INTERMEDIATE HEAT EXCHANGER (IHX)

The IHX transfers the heat from primary sodium which is radioactive to non radioactive secondary sodium in PFBR. The IHX is a vertical shell and tube counter flow unit featuring single pass on both shell and tube with primary sodium on shell side and secondary sodium on tube side. The tube bundle is placed in a fixed shell immersed in sodium pool. The IHX is supported on the roof slab and is removable from the top.

The IHX passes from hot pool to cold pool through a stand pipe of the inner vessel and the sealing between the hot pool and cold pool is by mechanical seal. A manually operated sleeve type

valve is provided on each IHX for primary sodium side isolation in the event of reactor operation with one secondary sodium circuit.

The main characteristics of IHX are:

Nominal power	: 314.7 MWt.
Tube size	: 19mm OD* 0.8 mm WT
No. of tubes	: 3600
Heat transfer area	: 1612 m ²
Primary sodium inlet temperature	: 827k
Primary sodium outlet temperature	: 670k
Primary sodium flow	: 1649 kg/s
Secondary sodium inlet temperature	: 628k
Secondary sodium outlet temperature	: 798k
Secondary sodium flow	: 1450 kg/s

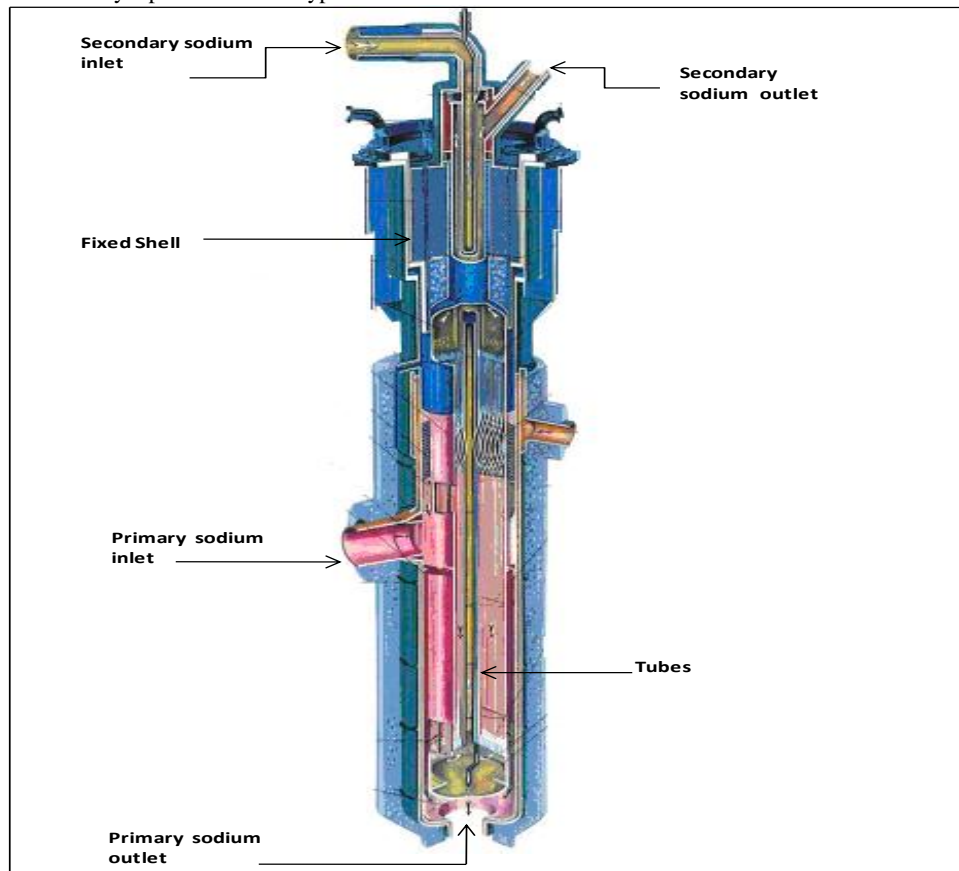


Fig 1: Architecture of Intermediate Heat Exchanger

4. ARCHITECTURE OF NEURAL NETWORK MODEL FOR IHX

For process modeling of IHX, multilayer perceptron (MLP) with feed forward neural network and back propagation algorithm is used. The MLP is having three layers: one input layer, one hidden layer and output layer. Variable selection for

process modeling of IHX includes identification of the parameters having the most relevant information about the desired output parameters [5]. Here the input layer has 5 input nodes (including bias) and they are primary inlet temperature, primary flow, secondary inlet temperature and secondary flow. The output layer has 2 output nodes and they are primary outlet temperature and secondary outlet temperature. Back propagation algorithm is a supervised learning method where both inputs and

desired outputs are known. The multilayer perceptron must have a nonlinear activation function such as sigmoid function. The conventional data available has a wide range. So the data are normalized between 0 to 1. We have modified the transfer function to

$$[2 / (1 + \exp(-x))] - 1$$

In order to make convergence faster.

The main advantage of neural network is that any unknown plant conditions, which the input sample set never learnt before can easily be calculated. The NN with adaptive learning finds the weights and consequently the model is changed by updating the weights. The iterative process is continued until the convergence is found. So when any unseen data is given to the

NN it generalizes the characteristics and produces an output corresponding to the input. The back propagation algorithm tries to minimize the mean square error value between the desired output and actual output. To make the algorithm converge faster

we need to use a learning rate coefficient (lr). We have to carefully select the learning rate because if the learning rate is too small, the convergence is extremely slow and if learning rate is too large, the NN may not converge at all. The learning rate and the number of hidden nodes varied one by one such that finally we get a result having minimum mean square error. We have taken 150 sample sets of data. The coding has been done in C language. Figure 3 represent the neural network architecture for the Intermediate Heat Exchanger.

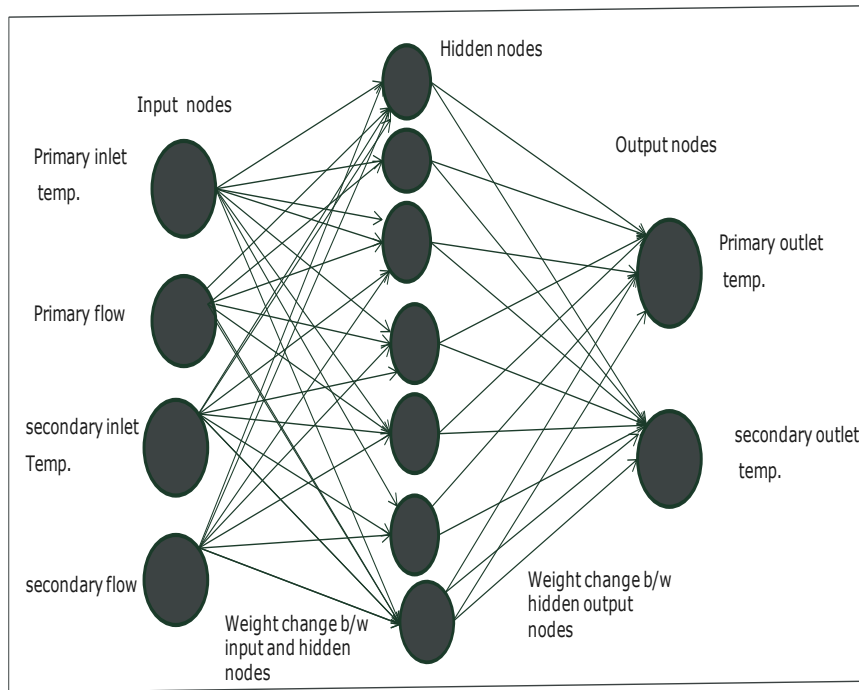


Fig 3: ANN architecture for Intermediate Heat Exchanger

5. TRAINING AND FINE TUNING

1. Present training samples (x, t) to the neural network and initialize random weights for input to hidden nodes and for hidden to output nodes.
2. For each hidden unit calculate

$$net_j = \sum_{i=1}^{NIN} W_{ji} * X_i \quad ;$$

$$Out_j = \frac{1}{[1 + \exp(-net_j)]} \quad \text{for } j= 1, 2, \dots, NHN$$

3. For each output unit calculate

$$net_{ok} = \sum_{j=1}^{NHN} W_{kj} * out_j ;$$

$$Ok = \frac{1}{[1 + \exp(-net_{ok})]}$$

4. For each neuron, calculate what the output should have been, and a *scaling factor*, how much lower or higher the output must be adjusted to match the desired output.

5. Adjust the weights of each neuron to lower the local error by

$$W_{kj} = W_{kj} + \delta W_1$$

$$W_{ji} = W_{ji} + \delta W_2$$

from that sample. Calculate the error in each output neuron.

$$MSE = \frac{1}{tsn} \sum_{t=1}^{TSN} \sum_{k=1}^{NON} (dout_{kt} - O_{kt})^2$$

6. Go to the next training sample of step 1.
7. After completing the total set of training samples compare the network's output to the desired output

8. If the MSE approaches to zero save the weights for testing purpose else repeat from step 1 till the error reduces to zero.

Table: Overview of the trial results

Serial number	No. of hidden nodes	Learning rate	No. of epochs	Mean square error (MSE)
1	5	0.01	100000	3.16E-04
2	6	0.01	100000	2.60E-04
3	7	0.01	100000	2.16E-04
4	8	0.01	100000	2.75E-04
5	9	0.01	100000	2.30E-04
6	10	0.01	100000	3.00E-04
7	11	0.01	100000	2.74E-04
8	5	0.01	200000	2.82E-04
9	6	0.01	200000	1.92E-04
10	7	0.01	200000	1.36E-04
11	8	0.01	200000	1.64E-04
12	7	0.02	100000	1.68E-04
13	7	0.02	200000	1.64E-04
14	7	0.01	500000	8.22E-05
15	9	0.01	500000	8.53E-05

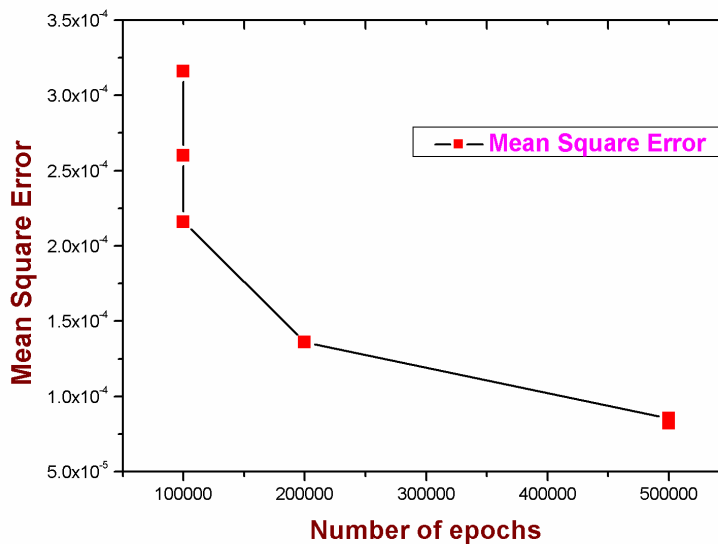


Fig 4. Graph plotted between mean square errors with respect to no. of epochs

The training process is summarized in the table. It shows that by varying the number of hidden layers, learning rate and the number of iterations we are able to reduce the mean square error (difference between desired and actual output). Figure 4 shows the number epochs versus the mean square error. After training the ANN with learning rate=0.01, no. of hidden nodes=7 and 5 lakh iterations we have arrived with MSE value=8.217e-05 and hence the case 14 is considered as optimal. The above graph represents the fine tuning of the neural network.

6. NEURAL NETWORK MODEL VALIDATION

The graph is plotted between the total number of training sample set and the corresponding outputs. It is seen from the graph of figure 4 and figure 5 that the actual output approaches to the desired output.

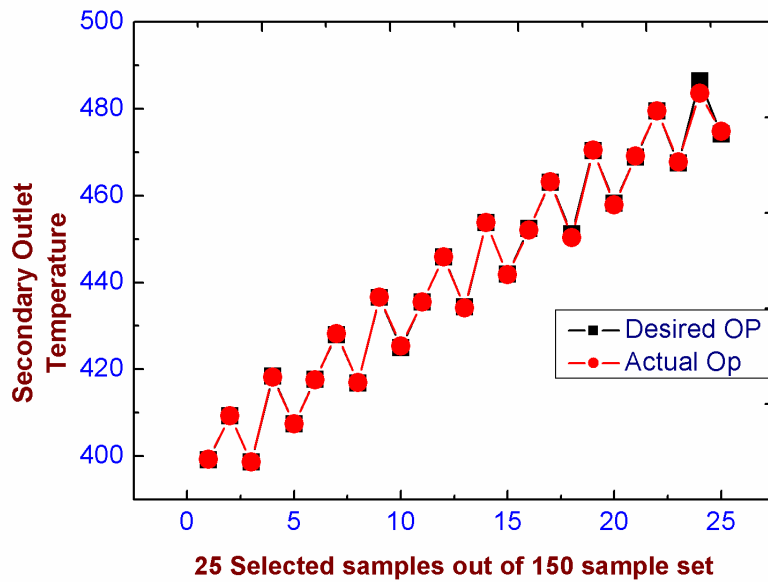
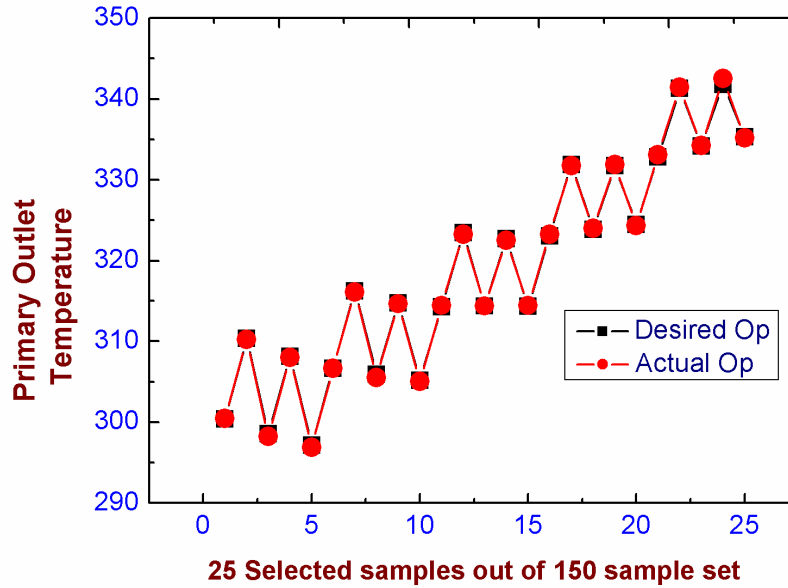


Fig5: Regression analysis for Training sample set

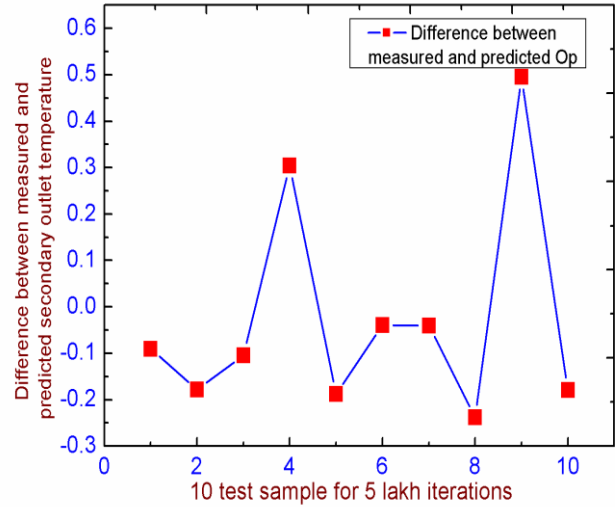
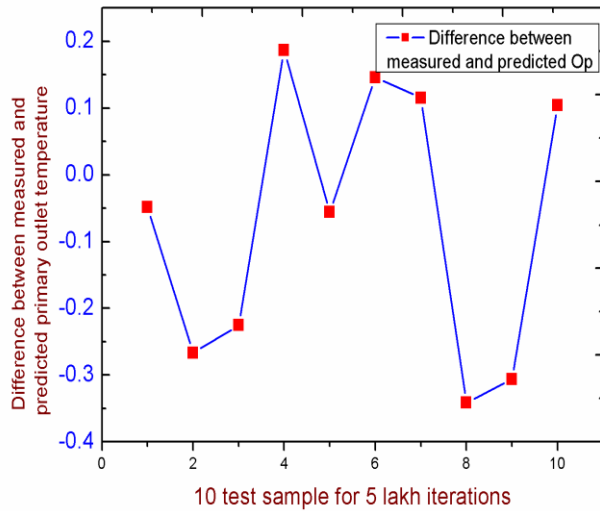
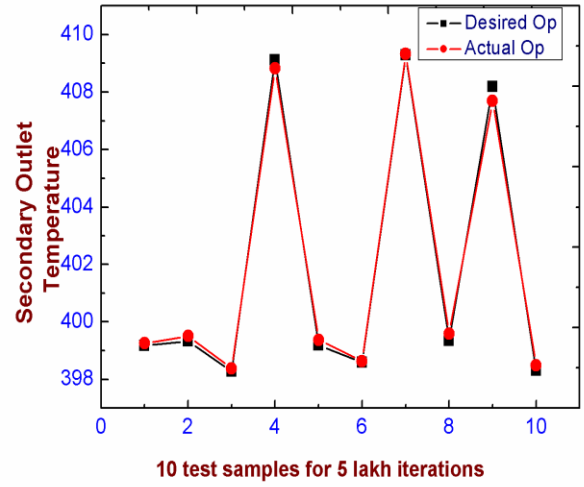
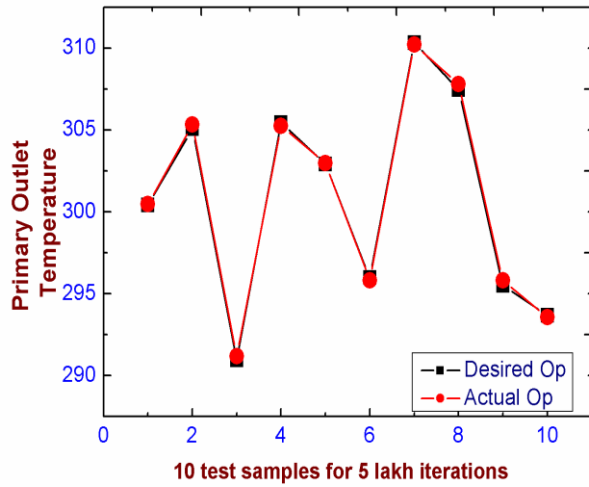


Fig 6: Regression analysis for testing sample set

Validation of the back propagation algorithm can be proved by testing a set of data which is within the range of training sample data. Here we have taken 10 set of testing data. From the figure 6 it has been found that the trained neural network model is able to generalize and give a meaningful output for the unseen data. The below graph shows the testing results.

7. RESULTS AND DISCUSSION

As compared to the traditional modeling techniques NN model has advantages including faster and more accurate prediction capability. We have implemented NN model for IHX in a nuclear reactor. We have trained 150 sample sets of data with an error margin of $8.21e-05$. Neural network model for IHX has

been trained and tested. The test result shows that this model can easily predict the expected output parameters.

8. ACKNOWLEDGMENTS

We are immensely thankful to Sri S.C Chetal, Director, IGCAR, Kalpakkam for his constant support and guidance for this project.

9. REFERENCES

- [1] Vasudevan M.*, Rao B.P.C, Venkatraman B., Jayakumar T., Baldev Raj, 2005, Artificial neural network modeling for evaluating austenitic stainless steel and Zircaloy-2 welds, Journals of Materials Processing Technology 169 (2005) 396-400

- [2] Mehrdad B.*, Mohammad Ghofrani B., Lucas C., 2002, Identification of a nuclear reactor core (VVER) using recurrent neural networks, *Annals of Nuclear Energy* 29 (2002) 1225-1240
- [3] Nabeshima K.*, Kurniant K, Surbakti T., Pinem S., Subekti M., Minakuchi Y., Kudo K., Online Reactor Monitoring with Neural Network for RSG-GAS, *Computational Intelligence Methods and Applications*, 2005, Digital Object Identifier: 10.1109/CIMA.2005.1662354
- [4] Malik N., 2005, Artificial neural network and their applications, National conference on 'Unearthing technological developments and their transfer for serving masses' GLA ITM, Mathura, India.
- [5] James Stich T., Sporre Julie K., Velasco T., The applications of Artificial neural networks to monitoring and control of an induction hardening process, *Journals of Industrial Technology*, 16, 1999-2000.
- [6] Choi Y J., Kim H K., Back W P., Chang S H., 2004, Hybrid accident simulation methodology using artificial neural networks for nuclear power plants, *Information Sciences—Informatics and Computer Science: An International Journal*, 60, 207-224 (2004).
- [7] Attieh I.K., Gribok A.V., Hines J.W., Uhrig R.E., Pattern recognition techniques for transient detection to enhance nuclear reactor's operational safety, *Nuclear engineering and design*, 207, 207-221, 2001.
- [8] Vallejo I., Barrio F., Neural networks applied to pressure sensor monitoring, *Progress in nuclear energy*, 29, 371-373, 1995.
- [9] Neural networks, *Electronics Textbook Statsoft*.
- [10] "Multilayer Feed Forward Neural Network and its Applications", Athinarayanan S., Jehadeesan R., Seminar on Automation in Process Engineering and Manufacture (SA PREM 97), 1997, IE (I), IGCAR, kalpakkam