2011 International Nuclear Atlantic Conference - INAC 2011
Belo Horizonte,MG, Brazil, October 24-28, 2011
ASSOCIAÇÃO BRASILEIRA DE ENERGIA NUCLEAR - ABEN
***ISBN:* 978-85-99141-04-5**

FAULT DETECTION OF SENSORS IN NUCLEAR REACTORS USING SELF-ORGANIZING MAPS

Paulo Roberto Barbosa[1], Graziela Marchi Tiago[1], Elaine Inacio Bueno[2], Iraci Martinez Pereira[3]

[1] Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP - *Campus* São Paulo
Rua Pedro Vicente, 625
01109-010 São Paulo, SP
barbosapr@gmail.com, grazielamarchi@gmail.com

[2] Instituto Federal de Educação, Ciência e Tecnologia de São Paulo – IFSP - *Campus* Guarulhos
Av. Salgado Filho, 3501
07115-000 Guarulhos, SP
elainebueno@gmail.com

[3] Instituto de Pesquisas Energéticas e Nucleares, IPEN - CNEN/SP
Av. Professor Lineu Prestes 2242
05508-000 São Paulo, SP
martinez@ipen.br

ABSTRACT

In this work a Fault Detection System was developed based on the self-organizing maps methodology. This method was applied to the IEA-R1 research reactor at IPEN using a database generated by a theoretical model of the reactor. The IEA-R1 research reactor is a pool type reactor of 5 MW, cooled and moderated by light water, and uses graphite and beryllium as reflector. The theoretical model was developed using the Matlab GUIDE toolbox. The equations are based in the IEA-R1 mass and energy inventory balance and physical as well as operational aspects are taken into consideration. In order to test the model ability for fault detection, faults were artificially produced. As the value of the maximum calibration error for special thermocouples is ± 0.5 º C, it had been inserted faults in the sensor signals with the purpose to produce the database considered in this work. The results show a high percentage of correct classification, encouraging the use of the technique for this type of industrial application.

1. INTRODUCTION

The studies in Fault Detection have been encouraged because of the increasing demand on quality, reliability and safety in industrial processes. This interest is justified due to complexity of some industrial processes, as chemical, power plants, and so on. In these processes, the interruption of the production due to some unexpected change can bring risk to the operator's security besides provoking economic losses, increasing the costs to repair some damaged equipment. Because of these two points, the economic losses and the operator's security, it becomes necessary to implement Fault Detection Systems [4] [5] [12] [16]. The Artificial Intelligence (AI) techniques have been successfully applied in the development of this kind of system because fault detection is a complex reasoning activity.

There are different procedures in Fault Detection Systems, including heuristic knowledge and mathematical models to the AI methods. The detection can be performed using different elements based on analytical methods, expert systems, artificial neural networks (ANN) and fuzzy logic [1] [14] [19].

The use of analytical methods applied in fault detection is not always possible because it requires depth knowledge of a process model. False alarms can occur due to the estimation errors of the process parameter because of the imprecise system model [11] [17] [18].

When heuristic expert system is applied, it is necessary to use human knowledge and experience. This method is much easier and more useful in comparison with analytical method, but it is difficult for automatic realization.

On the other hand, the use of ANN is rather easy to develop and to perform [6]. ANN can be applied when there is a database containing the process measurements, which later can be used in the training of ANN. The advantage of this method is the possibility to obtain on-line information about the kind and the size of a fault without developing very complicated mathematical models [6] [16] [18].

The purpose of this work is to develop a Fault Detection System based on the self-organizing maps methodology which was applied to thermocouples fault detection. The fault detection model was implemented through many computational simulations in offline form using a database generated by a theoretical reactor model [7] where faults were artificially inserted in the sensor signals database.

2. IPEN RESEARCH REACTOR IEA-R1 THEORETICAL MODEL

The Ipen nuclear research reactor IEA-R1 is a pool type reactor using water for the cooling and moderation functions and graphite and beryllium as reflector. Its first criticality was in September 16th, 1957. Since then, its nominal operation power is 2 MW. In 1997 a modernization process was performed to increase the power to 5 MW, in a full cycle operation time of 120 hours, in order to improve its radioisotope production capacity. Figure 1 shows a flowchart diagram of the Ipen nuclear research reactor IEA-R1.

A Ipen research reactor theoretical model was built in order to generate data in different reactor operation conditions, allowing flexibility in situations where it is not possible to obtain data experimentally because of restrictions due to the nature of a nuclear reactor operation. Using the model, data was generated both under normal and faulty conditions. The IEA-R1 theoretical model performs the following tasks:

- Generation of data in different reactor operation conditions
- Setting the input variable values in an easy and fast way using a graphic interface
- Setting the noise level for the input variables
- Selecting a faulty variable from a list
- Visualization of the results in a dynamical way

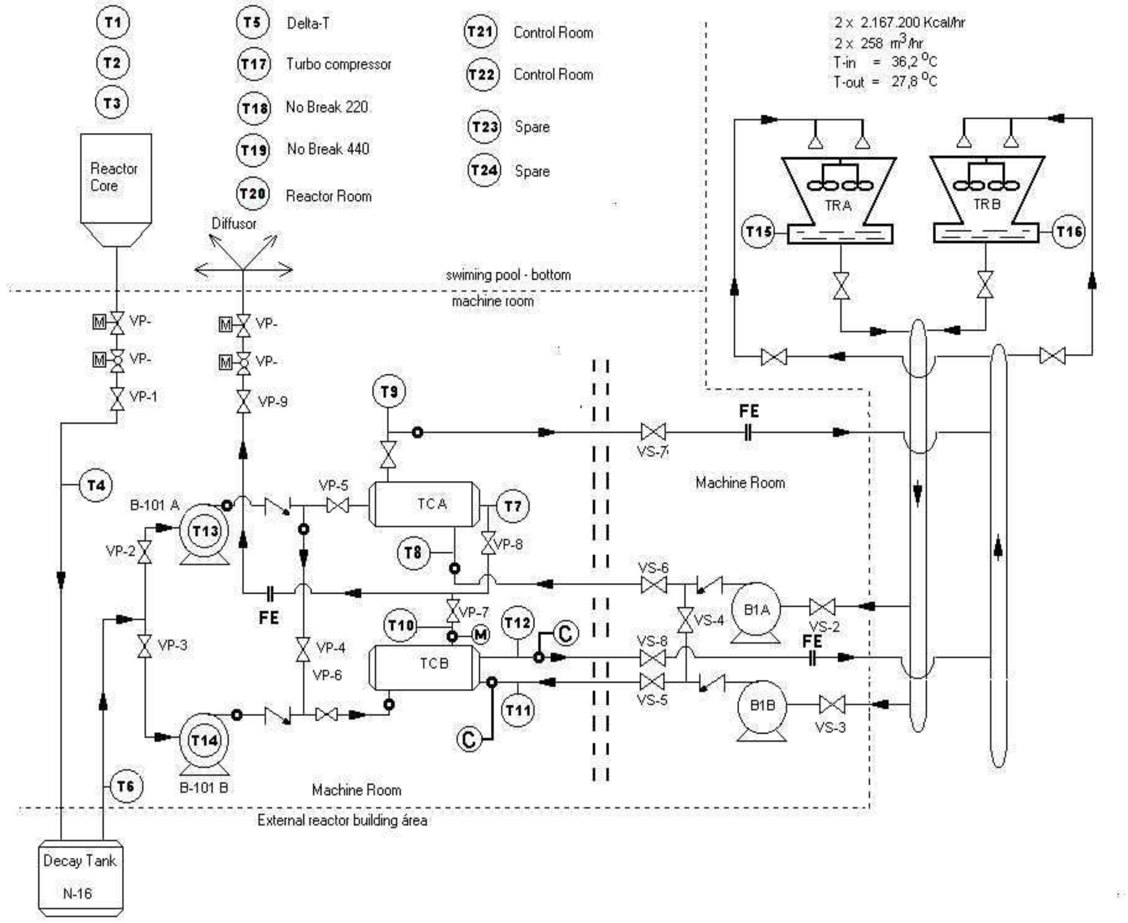

Figure 1. Flowchart diagram of the Ipen nuclear research reactor IEA-R1

The model represents the basic relationships among the different process variables. The system process equations are based on the IEA-R1 mass and energy inventory balance [2] and [9], and the physical and operational parameters, such as pipe length and diameter, relationships among the flow rate, temperatures and pressure drop are taken into consideration.

The Ipen research reactor model was built using the Matlab GUIDE toolbox [3]. The GUIDE (Graphical User Interface Development Environment) toolbox is a set of functions designed to develop interfaces in an easy and fast way. One can add plots, sliders, frames, editable texts and push buttons that are related to other Matlab functions.

The interface layout was built to look like the reactor process flowchart. Figure 2 shows the program interface. The reactor core is represented immersed in the water pool. The temperatures T1, T2 and T3 are the temperatures above the core near the pool surface, at mid high and close to the core, respectively. The nuclear power is an input data and a nuclear power of 100% corresponds to the maximum operation power of 5 MW.

The reactor coolant system is represented in the interface. The primary loop water flows through the reactor fuel elements and leaves the pool through a nozzle under the core. Then, the water passes through the decay tank: T4 which is the reactor core outlet temperature and T6 is the outlet temperature. B101-A is the primary loop pump. The heat exchanger is also represented. T7 is the heat exchanger outlet temperature (primary loop side). FE 01 is the primary loop flowmeter. The primary water loop flows out of the heat exchanger and then returns to the pool. The secondary loop is partially represented by the secondary side of the heat exchanger.

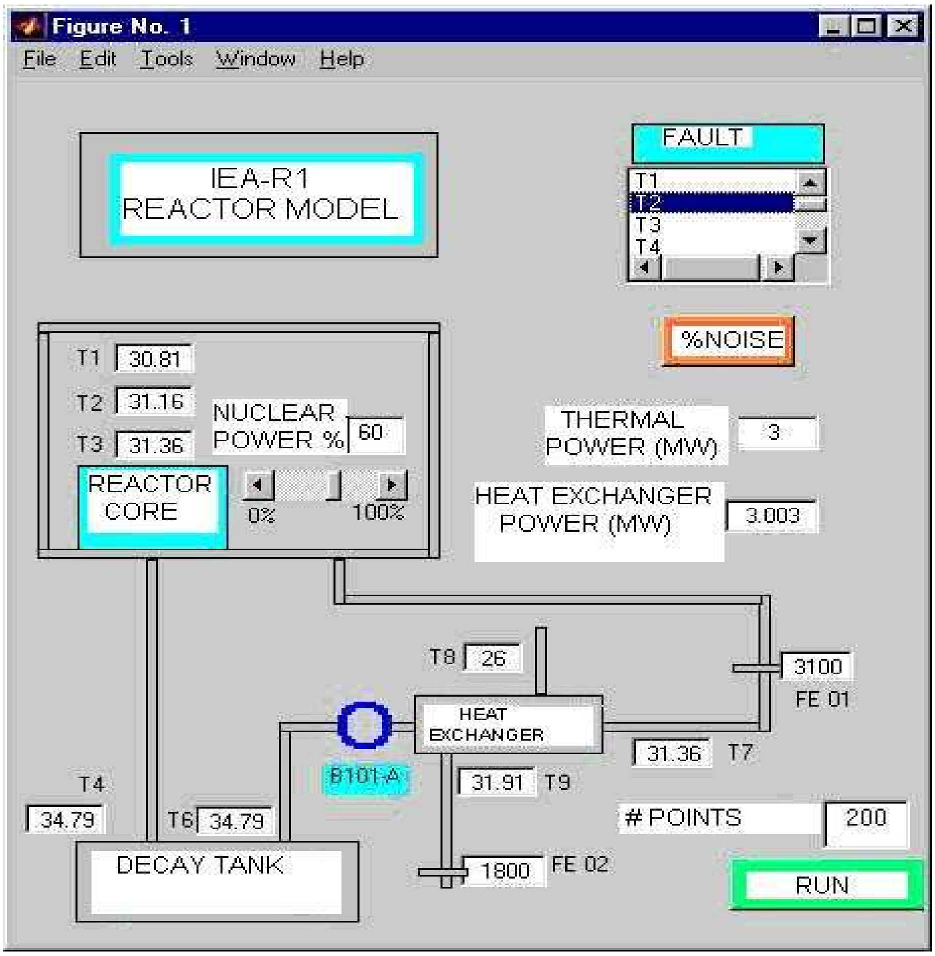

Figure 2. Program interface developed to compute the IEA-R1 nuclear reactor model variables.

The pump in the secondary side and the cooling towers are not represented. T8 is the inlet temperature of the heat exchanger secondary side, and T9 is its outlet temperature. The secondary loop flow is measured by the FE 02 flow meter. The units of temperature and flow are the same used in the reactor data acquisition system that is Celsius degrees and gallons per minute.

The user can define the time interval by defining the total number of points and the time step where the variables are to be calculated by the model for a given operational condition. In this case the program calculates for one point, refreshes the values and restarts the computation for the next point.

The user defines the desired variable values for the temperatures, flow rate or nuclear power directly in the interface editable dialog box. After entering the variable values, the noise level, the fault condition and the number of data points, pressing the button *calculate* initiates the program, which calculates the thermal power according to the mass and energy inventory balance equations.

The fault data was generated by adding random values to the signal in normal operation. The amplitude of these values was varied to allow a study of the sensitivity of the neural model proposed. As the value of the maximum calibration error for special thermocouples is ± 0.5°C [13], it had been inserted values ranging from 0 to 0.5, 1.0, 1.5, 2.0, 3.0, 4.0 and 5.0, respectively, above to the original signal.

3. SELF-ORGANIZING NEURAL NETWORKS

An artificial neural network model can be defined as a large number of simple interconnected processing units used to establish an input/output relationship. The self-organizing system considered here belongs to a special class of artificial neural networks (ANN) known as feature maps.

A Self Organizing Map (SOM) consists of neurons organized on a regular low-dimensional grid (usually one or bi-dimensional). Figure 3 shows a schematic diagram of a bi-dimensional grid, frequently used as a discrete map. Each neuron in the grid is fully connected to the neurons at the input layer.

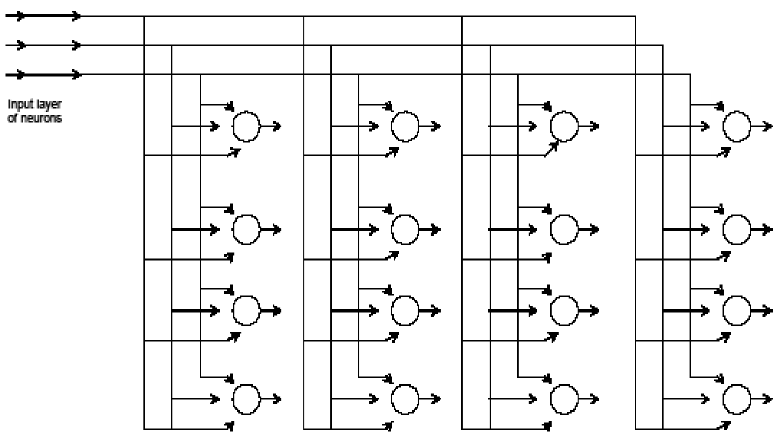

Figure 3. Bi-dimensional grid of neurons [8]

From the point of view of the information and how it is visualized, the self-organizing nature of the mapping implies that the statistical and nonlinear metric relations among the n-dimensional input data are converted into simple geometric relations between variables located at the nodes of a bi-dimensional net [10]. In other words, a self-organizing map projects the information contained in the primary data space on to a bi-dimensional network, without altering significantly the topological relations. It may be regarded as a tool capable of creating abstractions. These two attributes - visualization and abstraction of data - are of great importance in complex information-analysis applications, such as the problem of fault detection of sensors in nuclear reactors. These networks are characterized by competitive learning, a process in which the output "neurons", or nodes of the map, compete among themselves to become activated while a data-pattern is presented to the inputs. Eventually, just one output neuron, or one in each local group, becomes the "winner" of the competition and remains active.

The neurons are selectively composed according to the many input patterns or input pattern classes in the context of a competitive learning process. The winner neuron location is arranged according to the other neurons in a significant way inside the coordinate system. It creates a grid for different characteristics of the input patterns.

Let m be the input vector dimension. Let p be any input vector selected from the input space, represented as:

$$p = [\, p_1, p_2, ..., p_m \,]^{\mathrm{T}} \tag{1}$$

The weight vector of which neuron has the same dimension as the input vector. Let the weight vector of the neuron j denoted by:

$$\mathbf{W_j} = [w_{j1}, w_{j2}, ..., w_{jm}]^{\mathrm{T}}, \;\; j = 1, 2, ..., l \tag{2}$$

where l is the total number of neurons. In order to find the best competition of input and weight vectors, we compare the internal product $\mathbf{w}^{\mathrm{T}}{}_{\mathrm{j}}\mathbf{p}$ for j = 1,2,...,l . Then, the one with the best result is selected. Furthermore, this neuron fixes the location where the topologic neighborhood of excited neurons is centered.

The best competition criterion, based on the internal product maximization is mathematically equivalent of Euclidean distance (between **p** and $\mathbf{w_j}$) minimization.

The neuron *i(p)* identifies the closest neuron from input vector **p**, and *i(p)* can be determined applying the condition:

$$i(p) = \mathrm{argmin}_{\mathrm{j}} \left\| \mathbf{p} - \mathbf{w}_{\mathrm{j}} \right\|, \;\; j = 1,2,...,l \tag{3}$$

This procedure is the essence of the competition process of neurons. The specific neuron i which satisfy this condition is the so called winner neuron for the input vector **p.** We can verify this in Eq. (3).

A continuous input space of activation patterns is mapped on an output discrete space by a competition process of neurons. Depending on the application, the neural network output can be the winner neuron index (i.e., its grid position) or the weight vector near the input vector or both [8]

A topographical map is formed in the SOM from the input patterns, in which the spatial locations (coordinates) of the neurons in the grid reflect intrinsic statistical features within the input patters - hence the name self-organizing maps [8]. Each input pattern presented to the network is equivalent to a certain region of input space. The position and nature of that region usually vary from one input pattern to the next. All neurons in the SOM should be exposed to a sufficiently large number of different patterns to ensure that the self-organizing process has the chance of evolve correctly and develop a complete feature map. The layer of nodes in a SOM is arranged initially in physical positions, in conformity with the topology adopted for the map: a hexagonal bi-dimensional grid was used in this work.

One of the main neural network features is the generalization ability, i.e, successfully classify the patterns not presented before. Self-organizing maps generalize, placing in the same class, similar patterns to the ones previously classified at training procedure. It means that, signal identification by a representative data set is a feasible proposal, since the data set has been correctly classified by the net. This work focuses on analyzing how this feature can be used in the task of detecting faults in sensors.

4. RESULTS

In this work tests were performed with the signals from the temperature sensor T4 which is the reactor core outlet temperature (Fig. 4). In each test 10 examples were presented to the neural model, 5 being the original signal and 5 the fault ones. Each sample consisted of a portion of 100 points of the temperature signal. So, in a same test, the neural network received one portion of the original signal and the same portion of the signal with faults. It was performed by adding random values to the original signal whose amplitude was varied in order to produce a study on the sensitivity of the model. The range was varied from 0 to 5. Let the constant (C) represents the range of the random values that in this study was considered as 0.5, 1.0, 1.5, 2.0, 3.0, 4.0 and 5.0, respectively. Thus, for example, considering one sample, which had a minimum value equal to 39.2 ° C and a maximum equal to 40.4 ° C, began to show a value of minimum and maximum temperature of 39.2 ° C and 40.88 ° C, respectively, using the constant 0.5. In the case of the constant 5.0, the same portion has now a range of 39.2 ° C to 45.32 ° C. The Fig. 4 shows the original and the fault sample cited above.

The signals produced with constant 0.5 correspond to the values of fluctuation inherent in the process of thermo-hydraulic data found in the reactor operation. The other cases, i.e., constants 1.0, 1.5, 2.0, 3.0, 4.0 and 5.0, are considered faults.

Considering the problem of classification into two classes, ie, normal operation and the presence of sensor failure, the network architecture was designed with two neurons. To develop the computer program was used the Matlab toolbox with the parameters already described in the previous section. The other parameters were not changed. Several tests of neural network training have been performed and all showed similar results to table 1. This table shows the percentages of accuracy of the neural model including the test of generalization, where different signals from ones used in training procedure were presented to the model.

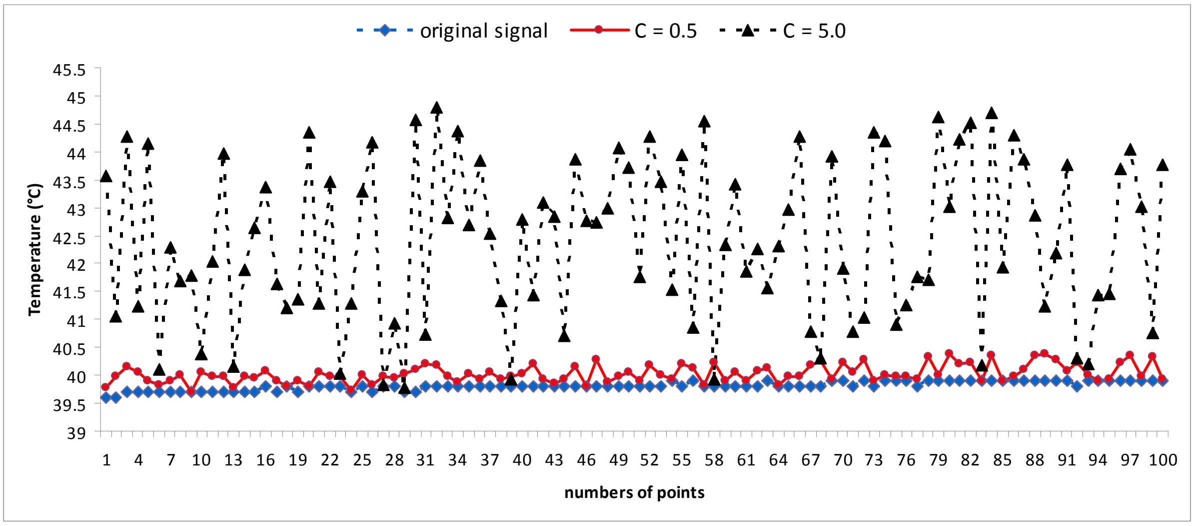

Figure 4: Examples of original and fault signals.

Through analysis of Table 1, it is possible to see the great sensitivity of the neural model to detect faults with amplitudes above 1 ° C. For example, results obtained with constants 0.5 and 1.0 show a small accuracy percentage during training and generalization processes. This fact suggests that for the neural model, very probably, these signals are related with fluctuations. For the others cases (1.5 to 5), the neural models were able to represent the different simulated faults with a very significant percentage of correct classification, both in training and generalizations processes.

Table 1: Neural model accuracy percentage

	0.5	1.0	1.5	2.0	3.0	4.0	5.0
training	60%	60%	80%	90%	100%	100%	100%
generalization	50%	80%	90%	90%	90%	100%	100%

5. CONCLUSION AND FUTURE WORK

A neural model based on self-organizing map to detect faults in temperature sensors found in nuclear reactors has been proposed in this paper. The main idea was to investigate the ability of the model to detect faults that may be considered, for example, a not calibrated sensor. Data from a theoretical model of the IEA-R1 research reactor of IPEN were used for the tests.

The computational tests performed with various faults levels showed a great percentage of model accuracy with respect to the classification of data with and without faults. The results therefore demonstrate the feasibility of using this model in nuclear industrial application. Future work should include extensive testing with theoretical data and experimental data, including data from other sensors in the reactor.

REFERENCES

1. Angeli, C., Chatzinikolaou, A., “On-Line Fault Detection Techniques for Technical Systems: A Survey”, *International Journal of Computer Science & Applications,* v. 1, n. 1, pp. 12 – 30, (2004).
2. Bassel, W. S., “*Balanço de Massa e Energia do Circuito Secundário do Reator IEA-R1”. Technical Report*, IPEN, (1996).
3. Building GUIs with MATLAB Version 5, The MATH WORKS Inc, MA, USA, 1997.
4. Clark, R. N., “Instrument fault detection”, *IEEE Transactions on Aerospace Electronic Systems,* v. 14, pp. 456-465, 1978.
5. Echendu, J. E. A., Zhu, H., “Detecting changes in the condition of process instruments”, *IEEE Transactions on Instrumentation and Measurement,* v. 43 (2), pp. 355-358, 1994.
6. Filipetti, G., et. al., “Recent developments of induction motor drives fault diagnosis using AI techniques”. *IEEE Transactions on Industrial Electronics,* v. 47, n. 5, pp. 994 – 1004, 2000.
7. Gonçalves, I. M. P., Ting, D. K. S., “A theoretical model for the IPEN research reactor IEA-R1”. *INAC 2005 – International Nuclear Atlantic Conference Proceedings* (Cdroom), 2005.
8. Haykin, S., “Neural Networks - A comprehensive foundation”. USA: Prentice Hall, 2001.
9. Kern, D. Q., “Process Heat Transfer*”*, Mc Graw-Hill Book Company, Inc, New York, USA, 1950.
10. Kohonen, T., "Self-Organizing Maps", 3rd Ed., Springer-Verlag, 2000.
11. Kowalski, C. T., Kowalska, T. O., “Neural networks application for induction motor fault diagnosis”. *Mathematics and Computers in Simulation,* v. 63, pp. 435-448, 2003.
12. Maki, Y., Loparo, K. A., “A Neural Network approach to fault detection and diagnosis in industrial processes”. *IEEE Transactions on Control System Technology,* v. 5 (6), pp. 529-541, 1997.
13. Manual on the use of thermocouples in temperature measurement. ASTM STP 470B. American Society for Testing and Materials, Fevereiro 1983.
14. Puig, V. et. al., “A GMDH neural network-based approach to passive robust fault detection using a constraint satisfaction backward test”**.** *Engineering Applications of Artificial Intelligence*, v. 20, pp. 886-897, 2007.
15. Sydenham, P. H., Thorn, R., “Strategies for sensor performance assessment”**.** *IEEE Instrumentation and Measurement Technology Conference Record*, p. 353 – 358, 1993.
16. Shoen, R. R., “An unsupervised, on-line system for induction motor fault detection using stator current monitoring”. *IEEE Transactions on Industrial Applications*, v. 31, pp. 1280 – 1286, 1995.
17. Vas, P., “Parameter Estimation, Condition Monitoring, and Diagnosis of Electrical Machines”. Oxford University Press, 1993.
18. Vas, P., “Artificial Intelligence based electrical machines and drives”. Oxford University Press, 1999.
19. Zupan, J., Novic, M., Ruisanchez, I., “Kohonen and counterpropagation artificial neural networks in analytical chemistry”. *Chemometrics and Intelligent Laboratory Systems*, v. 38 (1), pp. 1-23, 1997.