

NEURAL NETWORKS USED TO MONITOR AN EXPERIMENTAL TEST WORKBENCH

Davi Almeida Moraes and Iraci Martinez Pereira

Instituto de Pesquisas Energéticas e Nucleares (IPEN / CNEN - SP)
Av. Professor Lineu Prestes 2242
05508-000 São Paulo, SP
dmoraes@dk8.com.br
martinez@ipen.br

ABSTRACT

This work presents the application of neural networks in an experimental workbench. This bench was developed with the purpose of conducting real time tests and data acquisition. The method applied for this work allowed to generate faulty data in a gradual and controlled way through the binary combination of double action valves. Using the SCADA application (Supervisory Control and Data Acquisition), it became possible to acquire data for analysis in Matlab / Simulink software. This bench has two reservoirs: a reservoir that has sensors for recording pressure and temperature variables for later analysis, and another reservoir that has level sensors. Four models were used to develop the respective practical experiments. In the first model, it was possible to perform all practical tests of the plant, as well as mechanical changes like repositioning of some mechanical components, piping, sensors and electrovalves. In the second model, it was noticed that the positioning of the flow meter, located after the pump output, prevented a good measurement of the flow variable. In the third model, it was perceived that the number of failures initially adopted, made the data too confusing for the neural network analysis. In the last model, it was possible to obtain a performance of 96.6% of hits after the reconfiguration for 4 controlled faults.

1. INTRODUCTION

A nuclear power plant needs a control room where all the information necessary for its monitoring and control. However, the increase in size and complexity of nuclear power plants has greatly influenced the operational requirements of control rooms, because of the large number of information available, the number of indicators, alarms, etc., has increased substantially. On the other hand, the current trend of the control rooms is to reduce the complexity of the panels to provide information in a more convenient way so that it is adequate for each particular situation. In this way several types of dedicated monitoring systems for specific functions have emerged, such as monitoring and vibration analysis systems, monitoring of loose parts and monitoring of material fatigue. These systems are called Operator Support Systems (OSS) and are intended to facilitate the actions of the operator, whose main function has been to supervise and anticipate undesirable situations [1].

Operators use support systems that provide more processed and integrated information than those available in conventional instrumentation media. Although they are not required for operation or safety, support systems provide high-level information. They assist and advise operators and allow them to make better strategic decisions during both normal and abnormal operating conditions. They also provide a supervisory management capability during emergency conditions [2].

The operator also needs another important support system which is the fault detection and diagnostic system. This system alerts operators about sensors and actuators problems and assists in diagnostics before normal alarm limits are reached. Failure and diagnostic system tasks can range from simple monitoring of the variables, checking if their values are within a pre-established threshold, until a process anomaly is detected. More sophisticated monitoring and diagnostics systems can also perform the identification of the component responsible for the irregularity observed in the system. In the latter case, and as long as this does not compromise process performance, the system may still generate a signal that will replace the failed sensor or actuator signal, so that the process can continue in progress until further servicing.

In traditional monitoring and diagnostic systems the so-called physical redundancy (or hardware redundancy) is used, where two or more similar sensors perform the same measurement. A logic voting scheme can then be used to identify a faulty instrument. While this is a reliable method, there are a number of disadvantages, such as the high cost of installing and maintaining a greater number of sensors. In addition, there is not always room for the installation of redundant sensors.

Faults can occur in processes, sensors, actuators and process instruments independently or simultaneously. When a simple fault occurs, a conventional alarm circuit is sufficient to make the diagnosis, however in the more complex industrial processes it is necessary to use automatic systems to assist the operator in making decisions [3].

2. OBJECTIVE

The purpose of this work is to study the sensitivity and robustness of Fault Detection methods using Neural Networks methodology. This study is unprecedented because the System of Monitoring and Detection of Failures developed will have the possibility to provide a fast and reliable detection of the failures when the control plant is in conditions of disturbance, even when the mathematical model on which the diagnosis is made cannot reproduce the plant's dynamic operation.

In order to do so, Fault Monitoring and Detection systems will be developed and applied to an experimental bench specifically developed to generate faulty data.

As a result, it will be possible to analyze the sensitivity and robustness of the fault detection system that was developed using different computational intelligence techniques.

3. METHODOLOGY

The method used in the development of the monitoring and fault detection system in this paper is Neural Network.

The Advancements in neurophysiology have uncovered several mechanisms on the flow and processing of information that occur in the human brain. Some of these mechanisms were modeled mathematically allowing the elaboration of computational algorithms that simulate, albeit in a simplified way, the most basic of the brain structures: the neuron.

The chosen algorithm for the training of the Neural Networks in this work is the backpropagation. This algorithm was chosen because it presents satisfactory results for the proposed application.

The computational implementation of the backpropagation algorithm presents the following steps:

Step 1: Initialize all parameters weights.

Step 2: Forward phase:

- Calculate all inputs and outputs for all layers of the network;
- Calculate the output error of the network.

Step 3: Backward phase:

- Carry out the calculation of the weight updates between the layers of the network, starting from the last layer, until reaching the input layer.

The backpropagation algorithm consists in the use of the weights, considering the propagation of the error of the output of the network for its input. This algorithm is also known as a generalized delta rule.

Although the total error E is defined by the sum of the errors of the output neurons for all the standards, it will be considered no loss, that the minimization of the error for each standard individually will lead to the minimization of the total error for all the standards. Therefore, the error is defined according to equation (11).

$$E(w) = \frac{1}{2} \times \sum_{k=1}^k (Yd(k) - Y(k))^2 \quad (11)$$

Where E is the measure of the error committed by the network for each standard individually, the outputs generated by the network $Y(k)$ differ from the desired outputs $Yd(k)$, where k is the number of output units of the network.

As in the delta rule the weight variation is defined according to the descending gradient, starting from equation (12), the equation (13) for the generalized delta rule is calculated mathematically:

$$W(t+1) = W(t) - \eta \times \frac{\partial E}{\partial W} \quad (12)$$

$$W(t+1) = W(t) - \eta \times \delta_j \times X_i \quad (13)$$

For an output neuron, the variable δ_j (error) is calculated according to equation (14):

$$\delta_j = (Yd_j - Y_j) \times f'(u_j) \quad (14)$$

$$u_j = \sum_{i=1}^m x_i \cdot w_{ij}$$

For the other neurons, other than the output neurons, the variable δ_j (error) is calculated according to equation (15):

$$\delta_j = f'(u_j) \cdot \sum_l \delta_l \cdot W_{jl} \quad (15)$$

At where:

W (t + 1): updated weight

W (t): old weight

$f'(\)$: The derivative of the neuron activation function

Ydj: desired output over the network

Yj: output from the network

$\frac{\partial E}{\partial W}$

: Gradient

η : Rate of learning

x_i : Neuron input

The term δ_j corresponds to the derivative of the error function with the weights, being responsible for the propagation of the error of the output of the network to its intermediate layers, through a recursive calculation.

4. DATA GENERATION AND ANALYSIS PLAN

Using a process plant to simulate a nuclear power reactor, it's possible to carry out the tests of control systems in different situations of normal and faulty operation.

Figure 1 shows the Engineering Flowchart of the PWR Power Reactor Plant. The plant will be composed of elements that will simulate the control behavior of the pressure vessel, steam generator, primary and secondary circuit pumps, an additional tank for heat exchange, sensor and actuator technologies, digital networks or connectivity using a Fieldbus system [4], signal processing devices (Programmable Logic Controllers), electro-electronic assembly of all system, as well as the mechanical structure in carbon steel, steel and aluminum.

In a second level of communication, the communication system in a reactor, allows to make effective the safety in the status of the sensors and actuators, as well as makes effective the maintenance of the system.

The database will interface with the supervisory system of the plant, which will be a SCADA (Supervisory Control and Data Acquisition) and it will make the data available through a standard interface. SCADA system is a software that interfaces with industrial processes and / or machines of the most varied sectors. [5]

The system developed for plotting the graphics will bring the following benefits to the system:

- Recording and temporal analysis of the process variables of pressure, level, temperature and flow in the pressure vessel;
- Recording and temporal analysis of the variables of the process of pressure, level, temperature and flow in the steam generator;
- Analysis and storage of more than 40 electrical variables considering the pressure vessel and steam generator;
- Analysis of useful life and predictive maintenance of sensors and actuators;
- Application of SCADA supervision system for remote monitoring, via web of the control systems, by I / O variables.

As for the monitoring and interaction to the temperature variable, a system will be implemented that allows proportional operation of the signal through a PWM (Pulse Width Modulation) system.

Engineering Fluxogram - PWR Power Reactor Simulation Plant

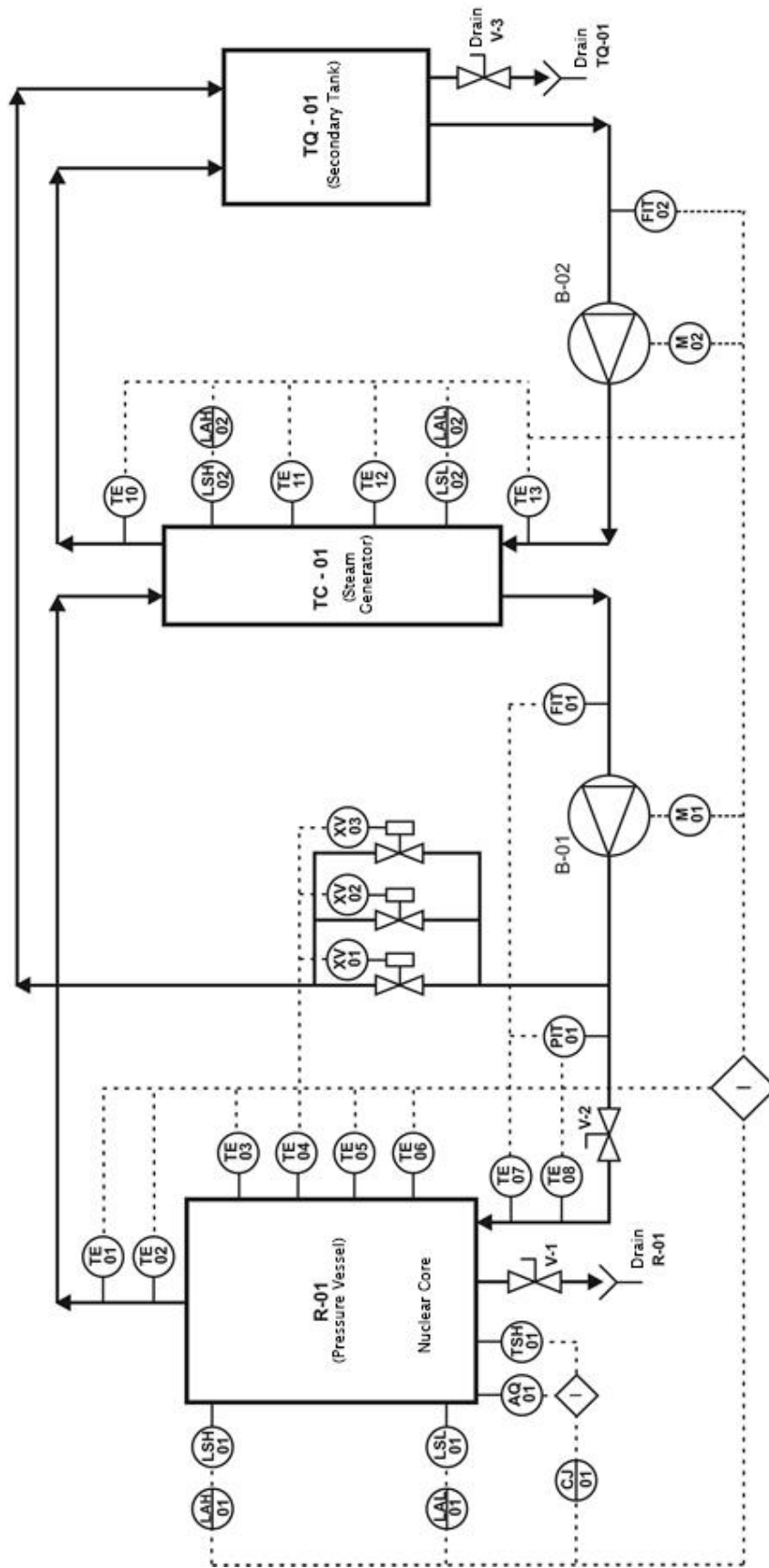


Figure 1. Engineering flowchart – PWR power reactor simulation plant

Control systems can occur through any of the variables, including pressure, level, temperature and flow (Figure 1).

- Temperature: it's possible to control the temperature through all the measuring elements installed in the Pressure Vessel and the Steam Generator. In the case of the Pressure Vessel, the performance for the development of the control will be through PWM power controller.
- Pressure: The measurement will occur through PIT-01 located between the pump and the pressure vessel inlet. The pressure in the system will act by varying the speed of the pump B-01.
- Level: Both LSH-01, LSL-01, LSH-02 and LSL-02 devices will be installed in both the Pressure Vessel and the Steam Generator. The actuation will be possible by pumping the pumps B-01 and B-02.
- Flow: The flow shall be measured at two points in the primary circuit and the secondary circuit. In the primary circuit, the measurement will occur through the FIT-01 meter and the secondary circuit, through the FIT-02 meter. The actuation will be possible by controlling the motor speed of the B-01 and B-02 pumps.

With the insertion of one or more faults, heating problems may occur in the Pressure Vessel. The flexibility of varying the cooling water flow of the primary circuit will act as a simulation of refrigerant fluid loss, known as LOCA (Loss of Coolant Accident), which is an important consideration in design criteria for nuclear power reactors. [6]

5. WORK DEVELOPMENT

The first stage of the work consisted in assembling the first prototype of the plant, thus allowing the acquisition of data with real characteristics for later processing of information using neural networks.

When creating the physical model, data acquisitions were made, but at first there was no information on what data would be needed for use in the control. Through training and testing of the acquired data, it was possible to make physical modifications to the system in a way that allowed the recognition of all information relevant to the analysis of errors.

After the assembly of the plant, it was used to train the Multiple Layer Networks with the retro propagation algorithm. All the tests performed in this work were done using the Matalab R2016a Neural Networks toolbox.

5.1. Architecture of MLP Networks Developed

The MLP network developed in this work consists of: an input layer, a hidden layer and an output layer.

- Input layer: In monitoring the number of neurons in the input layer depends on the inputs used in each model.
- Hidden layer: For the hidden layer of the developed networks, we studied and tested cases with different numbers of neurons. It was decided to use ten neurons since a larger number did not influence on the final results.
- Output layer: The output layers developed in this work present the response regarding the monitored variable.

6. OBTAINED RESULTS

The data acquisition system consists of the input variables of level, flow, temperature, time and output variable pump setpoint. The data were generated for different operating conditions of the system.

In the first physical model built, the variables of level, flow, time, temperature and setpoint of the pump were used. Later through tests performed, conditions were found that the temperature and time variables did not contribute to the correct use of the neural network in the system (temperature and time), so during the tests they were excluded in the data analysis. For example, the fluid temperature did not significantly influence the system, so it would only make the neural network more confuse when analyzing the data.

After removing the temperature variable, in the second model it was noticed that the system flow meter was positioned before the faults, so that the measurements made by the system did not have a significant difference to the neural network, thus causing a high degree of confusion in the network.

In the third model, the location of the flowmeter was installed in a position where there would be failures before and after it. Even so, the percentage of correctness in the system using the neural network was very low. After plotting the measurements graphs it was noticed at several moments that it was not possible to identify if there were any errors after the meter, since the measured values were very similar to the situation of non-existence of faults with a slight variation in the flow. Another point to note is that the time variable used in the system could not be considered, since the actual system would not know how long the physical fault started.

The fourth model was made with the installation of the meter as close as possible to the fluid reservoir, so that all faults inserted in the system were located before the measurement, thus having the maximum influence on the flow of the same. Thus, after measurements were made and the neural network was used to compare results as a physical model, it was possible to achieve a considerable percentage of accuracy in the tests performed

Table 1: Model variables

Model	Variable
1	Level, time, temperature, flow, pump setpoint.
2	Level, time, flow, pump setpoint.
3	Level, time, flow, pump setpoint.
4	Level, flow, pump setpoint.

From the second model onwards, the data read by the system with the sampling divided by 60% of training, 20% of validation and 20% of the test were analyzed in all the networks plotting the confusion matrix to make sure the answers of the network neural.

6.1. Model 1

Figure 2 represents the system of the first physical model used to read the variables in the system. When analyzing the data during the tests for this model, it was noticed that the temperature variable did not influence in any of the other variables in a significant way so that it was calculated together in the analysis of errors.

Thus, model 1 was important for the test of the physical parts where it was verified the equipment and the method of reading the plant variables.

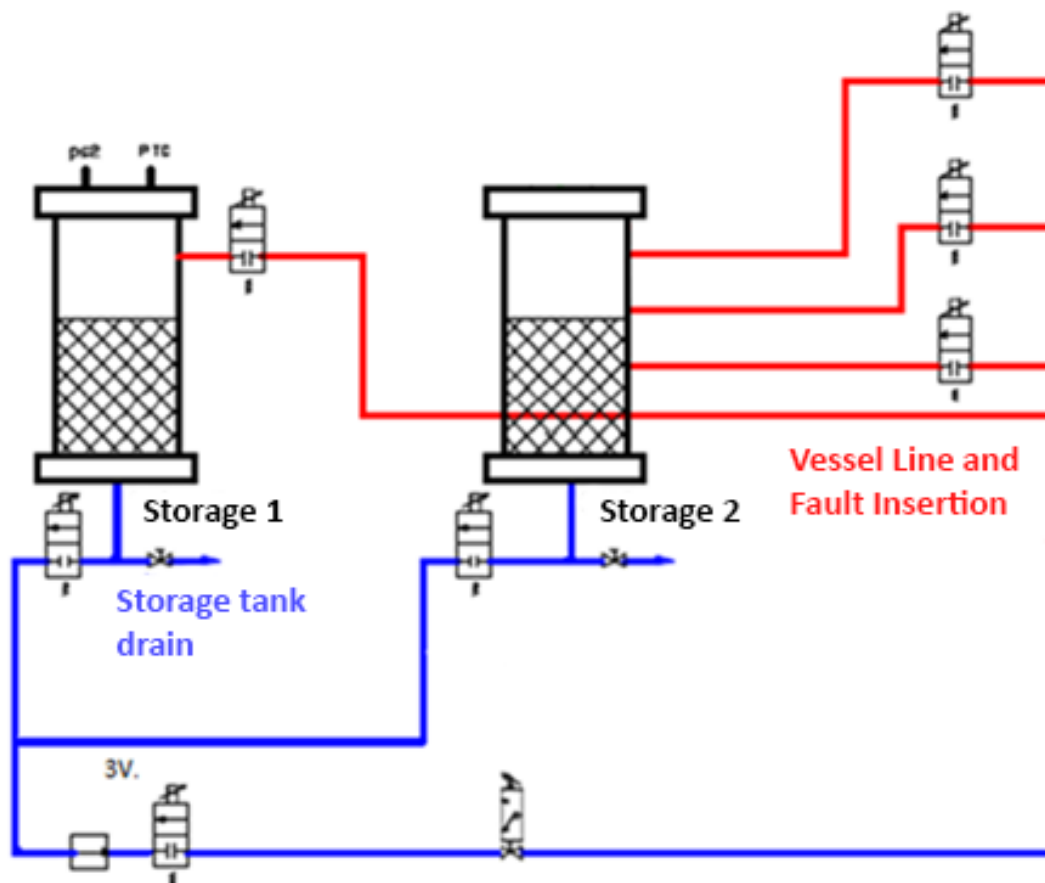


Figure 2. Pneumatic and hydraulic scheme used for model 1

6.2. Model 2

Model 2 follows the same design as model 1, however, without using the temperature variable. During the data analysis of the second test, a low performance training was verified. Due the fact that the meter was installed before the faults, all variations of input values were very small, thus making it only 23.3% certain during the tests.

6.3. Model 3

Because of the high degree of confusion in which the previous model was trained, it became necessary to change the position of the flow meter as well as the position of one of the valves (which generate one of the faults), according to figure 12.

In this way, the system was left with 2 valves before the meter and an another one after the meter.

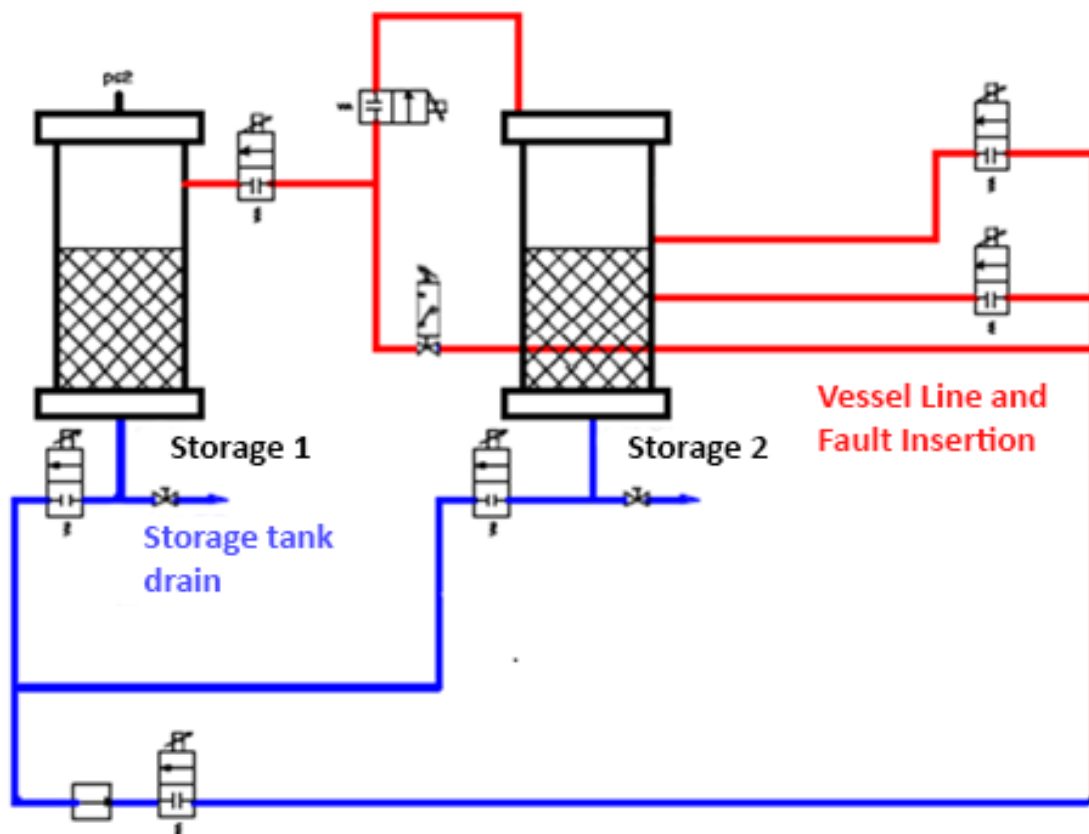


Figure 3. Pneumatic and hydraulic scheme used for model 3

The purpose of Model 3 was to verify the possibility of performing the detection of some failure after the meter since the valves located before the meter would generate a considerable difference in the flow according to the setpoint of the pump. For data acquisition, all possible combinations of failures (8 in total) were used. This became possible through the activation of the three valves.

In the training, good results were obtained according to the neural network, however, regardless of the confounding rate being in a low value, when comparing the physical test with the response given by the neural network, it was noticed that the new values did not correspond with the tested ones Using simulink.

6.4. Model 4

Once the tests of the previous model were finished, it was confirmed that the failure analysis could not be performed if there was any valve after the flow sensor. In this way the project was changed again so that the flowmeter was installed as close as possible to the entrance of Reservoir 1.

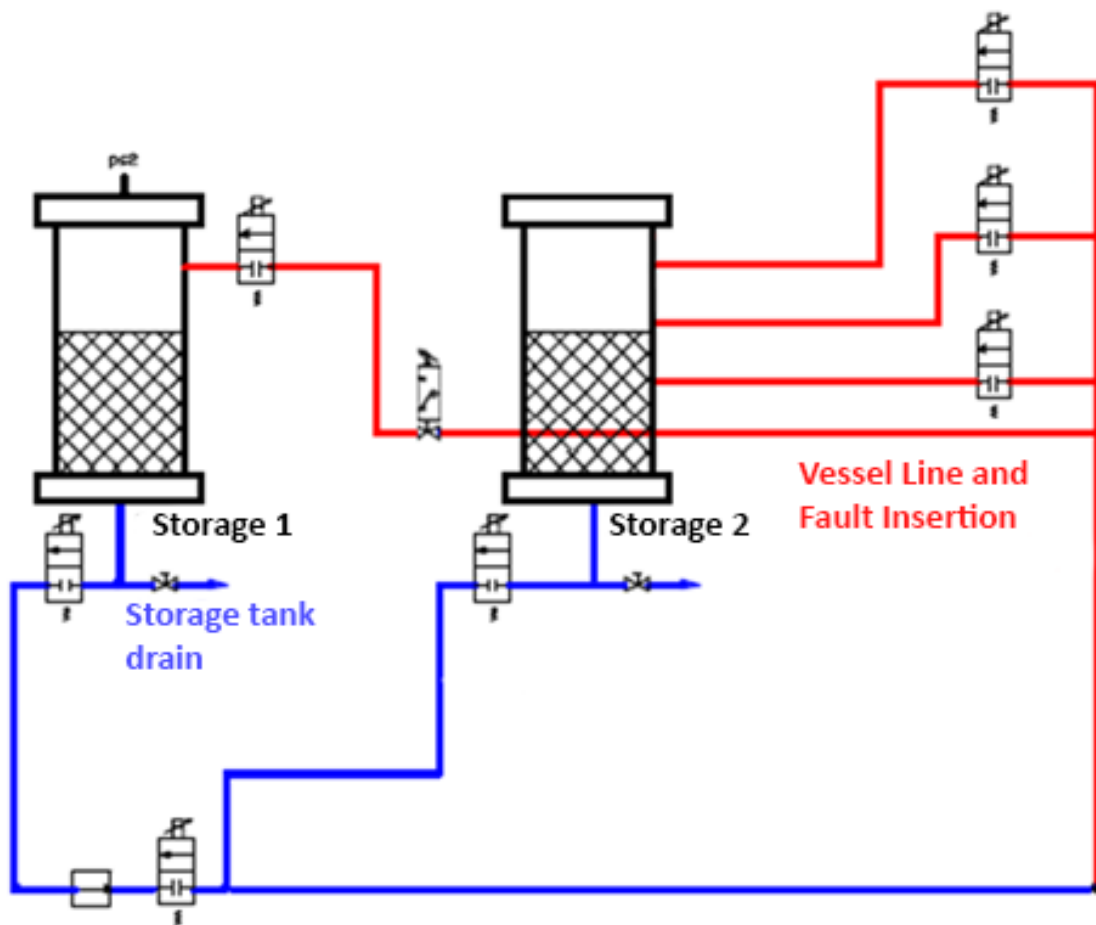


Figure 4. Pneumatic and hydraulic scheme used for model 4

The measurements were made so that there were 4 main cases in which there would be normal operation (without errors), one open valve (low leakage), two open valves (medium leakage) and three open valves (high leakage).

Thus, regardless of which valve is opened, the system flow has changed perceptibly to the flow sensor. The measurements showed an improvement in the performance of the neural network.

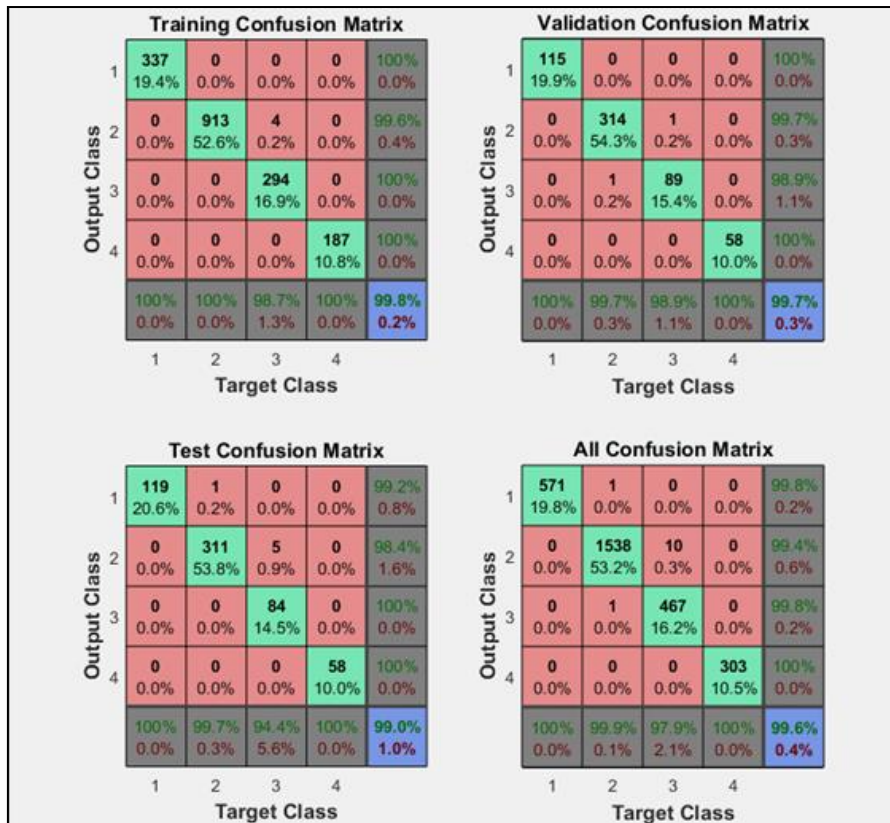


Figure 5. Validation of training performed in model 4

6.5. Analysis of Results

Figure 5 shows the Confusion Matrix for the results of model 4. In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm, typically a supervised learning one (in unsupervised learning it is usually called a matching matrix). Each column of the matrix represents the instances in a predicted class while each row represents the instances in an actual class (or vice versa). The name stems from the fact that it makes it easy to see if the system is confusing two classes (i.e. commonly mislabelling one as another).

It is a special kind of contingency table, with two dimensions (“actual” and “predicted”), and identical sets of "classes" in both dimensions (each combination of dimension and class is a variable in the contingency table). [7]

Training using fourth model allowed getting a good precision with the results obtained as shown on Figure 5, regardless of the time variable, which was being used to indicate when the system error was initiated. It was observed that when doing tests with different values in simulink, the result was always the same.

For the best possible performance, only the values of Setpoint for the fluid pump, level of the reservoir 1 and the flow measured by the sensor located at the fluid inlet of the reservoir 1 were used.

The network was also successful in tests performed with values outside the pre-defined setpoints 50%, 75% and 100%, thus demonstrating the network's ability to deduce the degree of leakage through values not found in the database. Neural network (measurements taken during all tests).

7. CONCLUSIONS

For these first experiments, the data acquisition system consists of 4 types of variables: reservoir level, system flow, fluid temperature and pump setpoint, which directly influence the fluid cycling in the system.

Through the analysis performed on the results obtained in each of the models, it is possible to observe the importance of choosing the appropriate input variables for ANNs. The data were generated for different operating conditions of the system.

In order to continue the work, the following studies will be carried out:

- Required setpoint values to stabilize the leakage independent system;
- Calculation of the amount of fluid that will be restored after leakage losses;
- Insert new variables into the system so that more types of leak failures can be identified.

REFERENCES

1. On-line Monitoring for Improving Performance of Nuclear Power Plants, Part 1: Instrument Channel Monitoring, IEA Nuclear Energy Series No. NP-T-1.1 , Vienna, 2008.
2. International Atomic Energy Agency (IAEA). Modern Instrumentation and Control for Nuclear Power Plants: a Guidebook. Technical Reports Series No. 387, Vienna, 1999.
3. SIMANI, S.; FANTUZZI, C. Fault diagnosis in power plant using neural networks. Information Sciences, v. 117, p. 125-136, 2000.
4. FRENCH ASSOCIATION FOR STANDARDIZATION. FIP. Bus for Exchange of Information Between Transmitter, Actuators and Programmable Controllers, NF C46 601-607, Mar. 1990.
5. <http://www.scadabr.org.br/?q=node/1> O que é um sistema SCADA. access on March, 2014.
6. NA, M. G., SHIN, S. H., JUNG, D. W. H., KIM, S. P., JEONG, J. H., LEE, B. C., Estimation of break location and size for loss of coolant accidents using neural networks. Nuclear Engineering and design 232, p295, Coreia do Sul, 2004
7. https://en.wikipedia.org/wiki/Confusion_matrix access on July 28, 2017.