

Fault Localization using Neural Networks and Observers for Autonomous Elements

Benítez-Pérez H.*+, Cárdenas-Flores F.+, Ortega-Arjona J. L.** and García-Nocetti, F.+

**Departamento de Matemáticas, Facultad de Ciencias, UNAM, Ciudad Universitaria, CP. 04510, México City, México.

*+Departamento de Ingeniería de Sistemas Computacionales y Automatización, IIMAS, UNAM, Apdo. Postal 20-726., Admón. No. 20, Del. A. Obregón, México D. F., CP. 01000, México.

Fax: ++52 55 5616 01 76, Tel: (*) ++52 55 5622 36 39

Email: (*) hector@uxdea4.iimas.unam.mx

Abstract: Fault detection and isolation (FDI) has become a useful strategy for determining fault appearance and on-line reconfiguration. However, unknown scenarios during on-line performance are still an open field for research. Different methods, such as knowledge-based techniques or analytical redundancy, have been followed. Nevertheless, both methods present inherent drawbacks for isolation. The present paper introduces a combined approach of model- and knowledge-based methods, using an autonomous element for isolation of unknown scenarios during on-line stage. The contribution is to integrate both methods in order to accomplish fault localization for unknown scenarios, based on previous information. Faults are constrained to certain bounded frequency response.

Keywords: Fault Diagnosis, Autonomous Elements, Self-Organizing Maps, Unknown Input Observers

1. Introduction

The task of fault diagnosis consists of determining the type, size and location of the fault as well as its time of detection. The use of knowledge-based techniques for fault localization and diagnosis allows on-line recognition of abnormal scenarios. These are based upon data treatment (Nelles, 2001), albeit these techniques require large amounts of data in order to obtain a valid representation of different scenarios. Alternatively, analytical redundancy allows a highly accurate detection of faults, based on a model of the observed system. It presents very well established methodologies, such as unknown input observers (Chen et al., 1999). Nevertheless, analytical redundancy requires a very accurate model of the system in order to locate a fault. Both, knowledge-based techniques and analytical redundancy, allow localization and classification of unknown scenarios as abnormal situations. The advantages of both methods depend on the type of information obtained, such as heuristic knowledge or model-based implementation. However, for abnormal situations, they have the disadvantage of not providing accurate results. In general, both methods require two important features: (a) the capability to determine faults, and (b) its sources of information.

Several different approaches attempt to combine knowledge-based techniques and analytical redundancy. Polycarpou et al., (1995) propose the use of Radial Based Functions, which converge into certain kind of dynamical system, based on a non-linear estimation from a case study. Chiang et al., (2001) present a complete review of fault diagnosis, focusing on the use of Principal Component Analysis (PCA) as a dimensionality reduction technique. Venkatasubramanian et al., (2003a,b,c) present an extended overview of fault localization and diagnosis based on model- and knowledge-based techniques. In general, the combination of both methods is feasible, although presenting undesirable glitches when used simultaneously, as discussed by Liling et al., (2002).

An autonomous element is a device that is able to communicate, self-diagnose, and make decisions. The main goal of this device is to obtain as much information as possible in order to produce self-calibration and compensation. By monitoring autonomous elements of a system, several approaches can be followed for fault detection, identification and localization. Several authors, like Isermann (1994), Masten (1997) and Henry et al., (1993), define typical autonomous elements. Moreover, the use of local control within autonomous elements is expected to attenuate the effects of disturbances and non-linearities inherent to local model. Lee et al., (2000)

propose the use of parameter estimation in order to self-tune a PID control where the response of the self-tuning procedure is restricted to a fairly linear model in order to response on time. Other approaches, like Wang et al., (2002) present a strategy based on robust control, which is feasible for highly non-linear models, although having a drawback of computational cost.

The objective of this work is to define a different approach to combine knowledge-based methods and analytical redundancy for on-line classification, using non-supervised neural networks and a bank of unknown input observers (UIO's) for self-diagnosis of autonomous elements. The novelty of this approach is the classification of non-well defined fault scenarios during on-line performance of an autonomous element. In order to locate unknown scenarios, input and output data is periodically sampled from the autonomous element, using a bank of unknown input observers, which generate residual data. Two neural networks are used to process data, input output and residual, in order to determine the autonomous element's behaviour. As this is inherently time varying, the signature of its faults may also change over time. Hence, one neural network is proposed to cope with changes in the signature of autonomous element's faults, within certain boundary restrictions. The other neural network is used to classify autonomous element's behaviour, according with a number of defined scenarios. Nevertheless, for detecting time varying faults, sampling time plays a key issue.

The paper is organized as follows: Section 2 describes the actual proposed approach for fault localization for an autonomous element, based on the integration of neural networks and unknown input observers. Section 3 presents a case study for testing the approach. Section 4 presents some of the most valuable results, as well as the correspondent analysis. Finally, Section 5 presents the concluding remarks.

2. Fault Localization for an Autonomous Element

2.1 General Description of the Approach

The actual approach proposes an integration of two neural networks and a bank of unknown input observers for fault localization, as presented in Fig. 3.1. A non-supervised neural network samples the data from an element, processing it in order to obtain a pattern. Then, a second non-supervised neural network, using the winning weight vector (related to the winning pattern) classifies any abnormal situation.

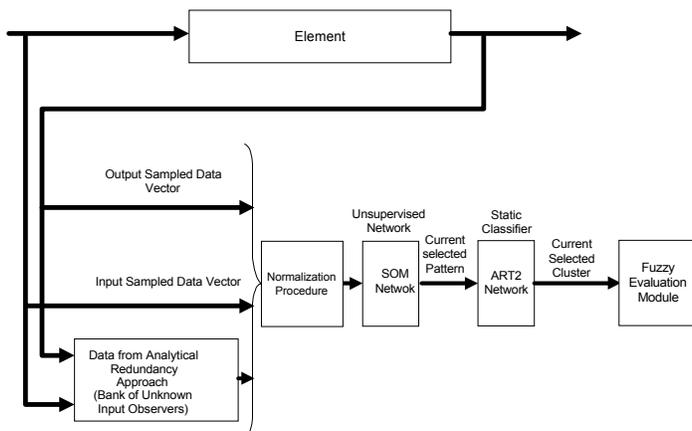


Fig. 2.1 Process diagram for Fault Localization

The idea of using two consecutive neural networks is to avoid miss-classification during the presence of unknown scenarios, using a self-organizing map (SOM) and adaptive resonance theory algorithms. SOM categorizes the behaviour of the monitored element. Then, the results are evaluated by a second neural network (an ART2A) in order to avoid glitches between similar categories.

The data is divided into three types: input, output and residual data, this last one obtained from analytical redundancy. Data is used in two stages: an off-line stage in order to train both neural networks, and an on-line stage for testing this approach. During the first stage, a training matrix is build. Such a training matrix consists of

three types of variables input, output and residual data, normalized between 0 and 1. In terms of scenarios, this matrix is divided into three areas as presented in Fig. 2.2. Each type of variable has M samples, organized as rows. The whole bunch of variables are integrated by three scenarios, organized as columns.

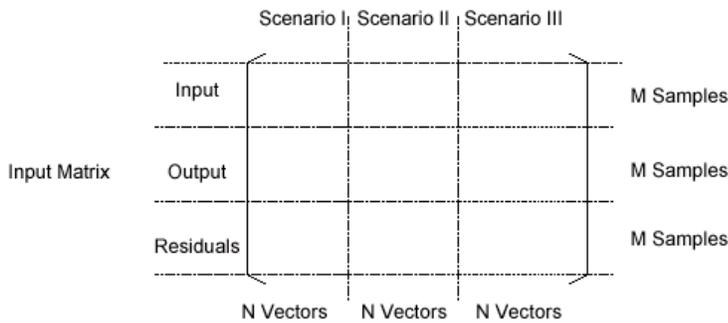


Fig. 2.2 Input Training Matrix

During training stage, each sample time window is composed of M samples directly related to a time window (Δt). The frequency of the fault has a bottom boundary, shown in eqn. 2.1.

$$frq_{fault} \geq \frac{4}{\Delta t} \quad (2.1)$$

where frq_{fault} is the frequency of the monitored fault, and Δt is the sampled time window.

Experimentally, a quarter of Δt has been chosen as the bottom boundary since this frequency is fast enough to distinguish sampled fault information between patterns. Therefore the frequency of the fault can be larger than this quarter of Δt . Alternatively, the top limit in terms of fault sampling is unlimited, although, the approach proposed here would be useless to classify a fault much faster than a Δt sampling window. At the time that this fault localization approach produces a result, it is highly possible that the current fault can be in another stage. This top bound is still open for further research and, in principle, is based on the relation between the frequency of case study and the Δt time window. Thus during on-line stage, sampling time is reduced to one sample evaluated every time as depicted in Fig. 2.3.

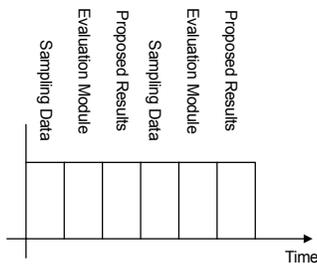


Fig. 2.3 Sampling time during on-line stage

Both neural networks are trained in cascade as shown in Fig. 2.1. Each has its own weight matrix, which are initialized randomly.

For UIO design, formal knowledge of the element behaviour during fault scenarios is crucial, since these scenarios are defined in terms of element response during the presence of certain unknown input. Hence, it is necessary to have access to several sources of information from the monitored element. Notice that any fault localization approach relies on the dynamic characteristics of the monitored element.

2.2 Integration of Non-supervised Neural Networks

Non-supervised Neural Networks are able to implement cluster algorithms. The main idea behind any cluster algorithm is to define centers as points within a data space. Centers serve as focal points for initial data representation. They are used for classifying non-linear behavior within non-supervised neural networks, such as SOM and ART2A. These networks present a fast response for non-linear and abnormal scenarios, although there is no guarantee for glitch presence in case of transitions. Therefore, the integration following a sequential mode allows the elimination of non-desirable transitions between scenarios due to “cluster” classification performed by SOM and pattern integration performed by ART2A.

There are various methodologies to build clusters (Abe, 2001). Proposals such as an entropy-based fuzzy clustering method defines cluster based on the entropy of each point with respect to a center (Höpnner et al., 1999). In the case of SOM and ART2A, these have the peculiarity to classify unknown scenarios in a predictable behavior (Fig. 2.4). In fact, the defined clusters are the representation of several scenarios (ART2-A results) whereas those classified patterns (SOM classified patterns) are the representation of the local behaviour of the element. The integration of UIO, SOM and ART2-A allows several advantages, such as availability of measured states and the capability to classify abnormal situation, avoiding undesired glitches during on-line performance.

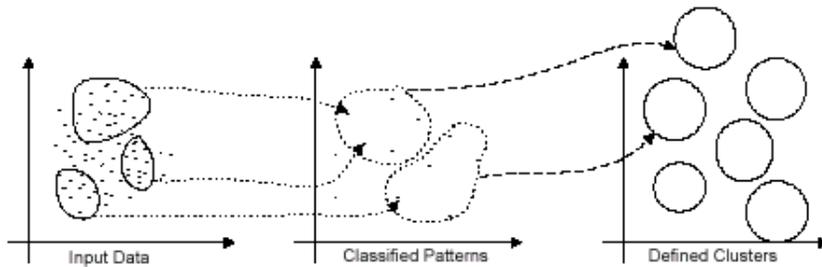


Fig. 2.4 The Method in terms of Patterns

During the off-line stage, SOM is trained using fault and fault-free scenarios with certain frequency, using different parameters, which are tuned in order to produce a valid and unique response. An important assumption, which impacts on the structure of SOM, is the use of a rectangular grid for data classification. This has been chosen due to its regularity when comparing between patterns, even in case of unknown faults. The regular grid allows a distribution of winner patterns. However, when a scenario is classified between the winner pattern and other devious patterns, miss-classification is present. This sort of case is defined as “glitch”, and it is related as a transition from one scenario to another. There are various ways to avoid this behaviour like a better training procedure, or defining winner patterns during transitions. However, glitches are not completely and certainly classified by SOM. Hence, glitches are classified using an ART2A. This network is trained to identify the response of SOM during the evaluation of one scenario with one particular pattern. This means that one particular pattern (from ART2A network) represents those patterns from SOM related to the same scenario. Fig. 2.5 gives a graphical representation of the procedure.

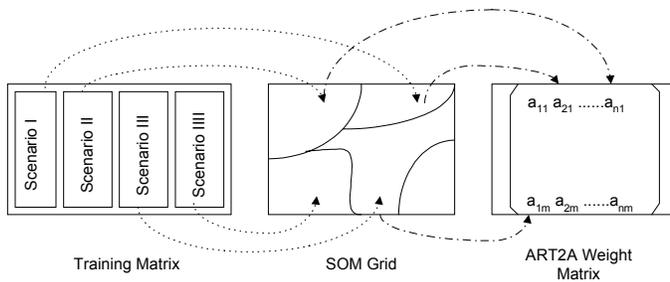


Fig 2.5 Pattern classification using SOM and ART2A

The objective of this cascade is to eliminate miss-classification of time variant faults and transitions between fault-free scenarios. This approach relays on certain boundary with respect to the similitude between patterns from SOM and those from ART2A.

2.3 The Fuzzy Evaluation Module

After defining the use of two neural networks as an approach to classify unknown scenarios, a heuristic measure is required as a final step to determine how a particular scenario has been degraded. This measure, known as confidence value, is generated by a fuzzy logic module. This module evaluates the winning weight vector related to the classified pattern from ART2A, in order to produce a percentage representation of current behaviour. The confidence value classifies the behaviour of peripheral element under the presence of a fault. It shows the degradation of the element with respect to the output, input and residuals. The procedure by which the fuzzy logic acquires knowledge is a key issue. Different methodologies can be followed. The confidence value has a continuous range from zero (catastrophic situation) to one (fault-free scenario) (Fig. 2.6).

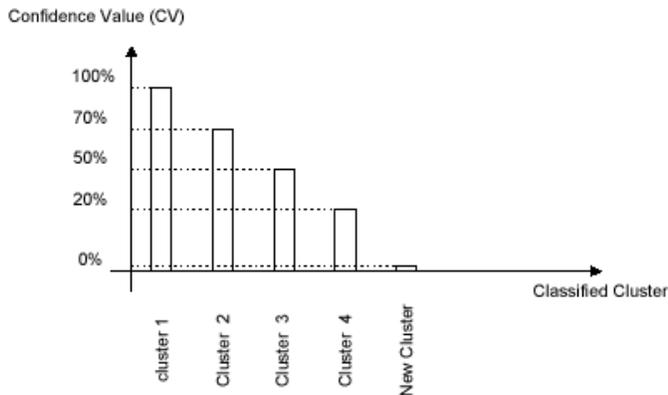


Fig. 2.6 Confidence Value Representation

2.4 Evaluation of the Approach

The evaluation of this approach is carried out using two scenarios: the first scenario is composed of four similar signals with different frequencies (Fig. 2.7).

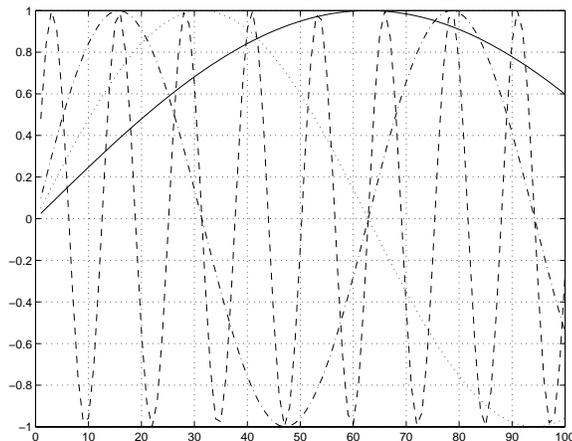


Fig. 2.7 First scenario used to evaluate the approach (a section of first 100 seconds)

These four signals have different frequencies: 0.005 Hz (continuous line), 0.01 Hz (dotted line), 0.06 Hz (dash-dotted line), and 0.1 Hz (dashed line). This scenario has a time window of 1000 seconds. The evaluation of SOM+ART2A approach is performed every sample during this time window. First, a learning stage is accomplished by training both neural networks, using this scenario during 100 seconds (Fig. 2.7). During this learning stage, the parameters η and ρ are 0.02 and 0.021, respectively. These two parameters, η and ρ , correspond to SOM and ART2A networks respectively. During the next stage (classification stage) the

SOM+ART2A approach is tested using the rest of the time window. In this case, the parameters η and ρ are changed, resulting in different numbers of patterns for the same evaluated scenario.

In the classification stage, these patterns are considered as *fail* patterns (as extra patterns) because SOM and ART2A have failed to classify them as similar to the originally recognized patterns, as shown in Table 2.1.

η (Parameter related to SOM)	ρ (Parameter related to ART2A)	SOM		ART2A	
		New Patterns	Number of Fail Patterns	New Patterns	Number of Fail Patterns
0.011	0.011	424	-	135	-
0.015	0.015	424	-	135	-
0.02	0.02	432	-	135	-
0.07	0.07	450	378	135	369
0.1	0.1	480	477	140	463
0.12	0.12	463	480	142	463
0.15	0.15	450	378	140	380
0.18	0.18	455	380	141	385

Table 2.1 Evaluation using the First Scenario

These results suggest that the increment on both parameters permit the increment of *fail* patterns. However, the number of patterns from both neural networks does not suffer a substantial increase. The meaning of this *failure* is that some patterns are miss-classified within different scenarios.

In the second scenario, the approach is evaluated keeping both weight matrices. In this case, signals are conformed by the element's response during different situations, such as transitions from different operating points. The case study, presented in Section 4, is used to generate these signals. Fig. 2.8 shows the initial 1000 seconds of the second scenario. The continuous line is referred to the output temperature, the dashed-dotted line is the response of pressure, and the dotted line is the residuals.

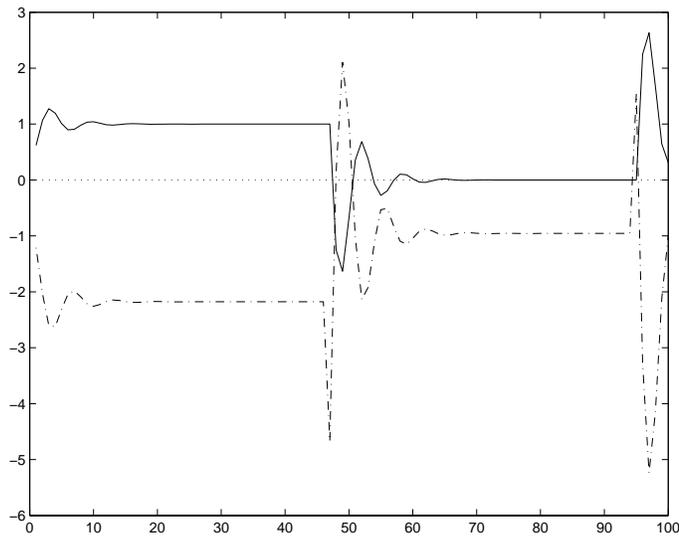


Fig. 2.8 Second scenario used to evaluate the approach

The response of this evaluation is shown in Table 2.2. During this scenario the number of new patterns is increased by SOM. Nevertheless, this behaviour is not presented in ART2A. Thus, this result confirms one of the goals of this paper, which is defining a strategy capable to cope with unknown scenarios without further appearance of new patterns. However, the number of *fails* patterns considerably increases.

Number of Scenarios	η (Parameter related to SOM)	ρ (Parameter related to ART2A)	SOM		ART2A	
			New Patterns	Number of Fail Patterns	New Patterns	Number of Fail Patterns
1	0.011	0.011	457	-	135	-
2	0.015	0.014	465	451	135	462
3	0.02	0.02	465	451	135	462
4	0.07	0.07	617	472	135	479
5	0.1	0.1	658	470	139	483
6	0.12	0.12	658	470	139	483
7	0.15	0.15	618	472	138	479

Table 2.2 Evaluation using the Second Scenario

For this scenario, the best η and ρ for unknown scenarios are 0.015 and 0.014 respectively. In order to confirm this result, a validation measure (Kiviluoto, 1995 & Lopez-García, et al., 2004) has been performed, obtaining a topographic error calculated as follows (eqn. 2.2).

$$e_t = \frac{1}{N} \sum_{k=1}^N u(x_k) \quad (2.2)$$

where N is the number of samples, x_k is the k th sample of the data set, and $u(x_k)$ is 1 if the first and second best matching patterns are not adjacent units, otherwise zero. The error is evaluated with respect to classified patterns from SOM. This error shows how separate are classified patterns between each other (Fig. 2.9). Every scenario presented has a very low error performance, such as scenarios 6 and 7, where error is neglected. However, the number of patterns during these scenarios considerably increases. This is an undesirable response for on-line performance, due to the increase of time consumption during the classification stage. Alternatively, second scenario has the largest error, but the number of patterns has not presented the previously referred increase. The conditions presented in second scenario for η and ρ are preferable for classification stage, rather than any other respective value.

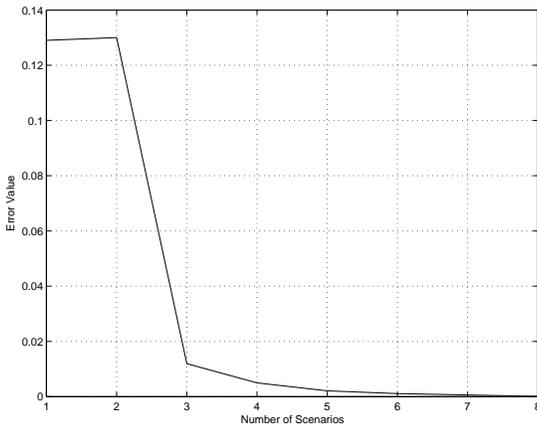


Fig. 2.9 Error Measure Performance with respect to η and ρ numbers

3. Case-Study

In order to validate the present fault localization approach, this section introduces an example related to a basic implementation of the autonomous element (Fig. 3.1). This case study is based on a pressure sensor composed of three similar transducers, which have been linearised to a nominal value. The dynamic model is presented in eqn. 3.1. It consists of a bank of UIO's, an Intelligent Fault Localization Module, a local control law, and a Fuzzy Evaluation Module.

$$\begin{aligned}
 A &= \begin{bmatrix} 1.1 & 0.0 \\ 0.01 & 2.1 \end{bmatrix} \\
 B &= \begin{bmatrix} 1.8 & -2.1 \\ 0.9 & 0.86 \end{bmatrix} \\
 C &= [0 \quad 1]
 \end{aligned} \tag{3.1}$$

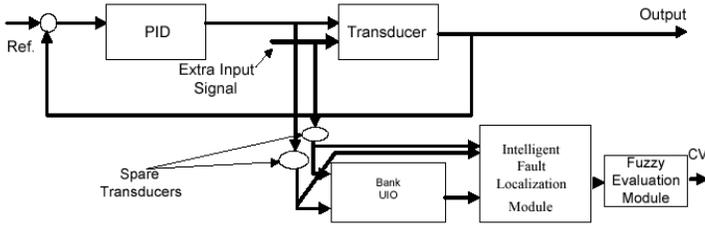


Fig. 3.1 Pressure Sensor divided in Three Modules

The input vector is composed of data from pressure demand and temperature. The output signal presents the delivery pressure. Based upon these available measures, the feedback relation is proposed in order to attenuate some disturbances. The proposed control is based upon the PI architecture (Lee et al., 2000). The dynamics of spare transducers are not modeled. Two additive faults are considered. Both injected faults are related to a backlash, variable time delays, and a dead zone (Table 3.1).

Fault I	Backlash=0.01, Dead Zone= (-0.01, 0.01), Time Delay = 0.001
Fault II	Backlash=0.09, Dead zone= (-0.051, 0.032), Time Delay = 0.012

Table 3.1 Fault Scenarios

The presence of the faults is established in two injection points at the output of case study. These are present during specific times in order to demonstrate the proposed approach. Hence, it is necessary to implement two UIO's, sensible to each fault. Both observers conform their matrices as follows (Eqn 3.2).

$$\begin{aligned}
 \text{First Observer} \quad H &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} & \text{Second Observer} \quad H &= \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\
 F &= \begin{bmatrix} 1.1 & -0.1 \\ 0.0 & 20.0 \end{bmatrix} & F &= \begin{bmatrix} 1.21 & -0.1 \\ 0.0 & 20.0 \end{bmatrix} \\
 K &= \begin{bmatrix} 0.0001 \\ 0.0001 \end{bmatrix} & K &= \begin{bmatrix} 0.0001 \\ 0.0001 \end{bmatrix} \\
 G &= \begin{bmatrix} 1.8 & -2.1 \\ 0.0 & 0.0 \end{bmatrix} & G &= \begin{bmatrix} 1.91 & -2.1 \\ 0.0 & 0.0 \end{bmatrix}
 \end{aligned} \tag{3.2}$$

Each observer responds to a particular fault. Furthermore, fault scenarios not considered for both observers are classified as different patterns by the fault localization module. In the case of glitches and transitions, SOM classifies this behaviour as “weak” patterns, meaning patterns that belong to a certain cluster in a distance manner. If this behaviour keeps its presence, a new cluster is declared. Current values of local PID control law are $k_1=0.91$, $k_2=0.05$ with respect to following equation (3.3).

$$pid = k_1 * e + k_2 * \int_{t=0}^{t=0.1} edt \quad (3.3)$$

where e corresponds to current error, and pid current control output.

The characteristics for both neural networks are selected as shown in Table 3.2. Specifically, the sampling window is equal to 100 samples. Therefore detectable fault have a minimum frequency equal to 100 Hz.

SOM	Size of Sampling data (M samples)	100 samples
	Size of Initial Output vector	4 data
	Input Vector	4 data
	Initial Population of Neurons	76 neurons
	Learning Value	0.015
ART2A	Input Vector	4 data
	Size of Initial Output vector	4 data
	Initial Number of Neurons	100
	Vigilance Parameters	0.014
	Learning value	0.02

Table 3.2 Technical Characteristics of Neural Networks

4. Results and Analysis

This section presents the results related to fault and fault-free scenarios. Three different scenarios are considered, two known scenarios (Fault and Fault-Free) and one unknown scenario (Unknown Fault). For the fault-free scenario both neural networks and UIO have been already trained and designed. The element response is presented in Table 4.1, where time delay gives an approximation of how long it takes to obtain a trustable response.

Name of Scenario	Number of Selected Patterns from ART2A (New Patterns)	Response Time Delay
Known Fault-Free Scenario	5	Immediate Response

Table 4.1 Fault Free Scenario

For the case of second known scenario, where a fault is present, the selected patterns and time delay response are shown in Table 4.2

Name of Scenario	Number of Selected Patterns from ART2A (New Patterns)	Response Time Delay
Known Fault Scenario	7	100 seconds

Table 4.2 Known Fault Scenario

Alternatively, an unknown scenario is used for fault localization procedure. This scenario consists of saturation at the output of case study. Therefore it is expected an increment in the number of patterns and time delay. Table 4.3 shows this behaviour, taking into account starting time and detection time.

Name of Scenario	Number of Selected Patterns from ART2A (New Patterns)	Response Time Delay
Unknown Fault Scenario	60	200 seconds

Table 4.3 Unknown Fault Scenario

The graphical representations of these results are presented in Fig. 4.1 for fault-free scenario. Four different graphs are shown: current input, its respective output, the injected fault according to the decision making module, and the number of patterns selected as result of this evaluation. This fault-free scenario has a time variance of $\sin(0.5*t)$. This time variance behaviour is depicted as current output of case study. The selected

patterns are presented in a consecutive manner with respect to the horizontal axis. Although, the number of patterns increased to 70 (vertical axis), those selected are no more than 10. The first 20 patterns have been selected as part of the setting of both neural networks. From the final 50 patterns, two are predominant. These are pattern number 70 and pattern number 45. Both patterns are the representation of this fault-free scenario with a limited time variance. The number of patterns is related with the final position within the weight matrix from ART2A. An important issue with respect to the number of patterns is the very low number of *fail* patterns.

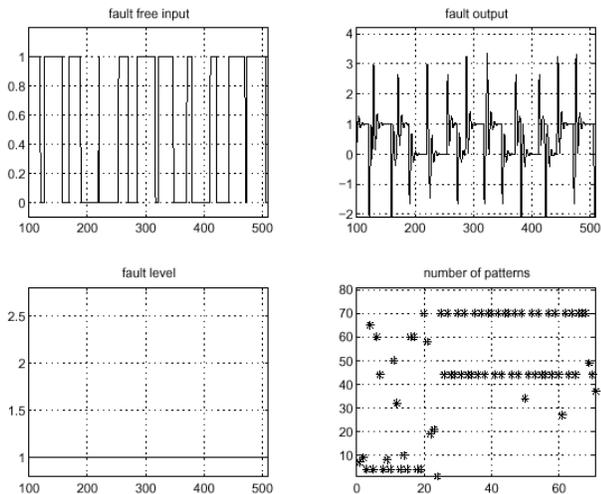


Fig. 4.1 Fault-Free Scenario with Time Variance Behaviour

In the case of a fault scenario (Fig. 4.2), the response of the element presents a small perturbation due to an increment of time delay (Table 3.1, Fault II). The number of patterns increases to 25. There is no predominant pattern during this test. However, some patterns have been already selected during fault-free scenario.

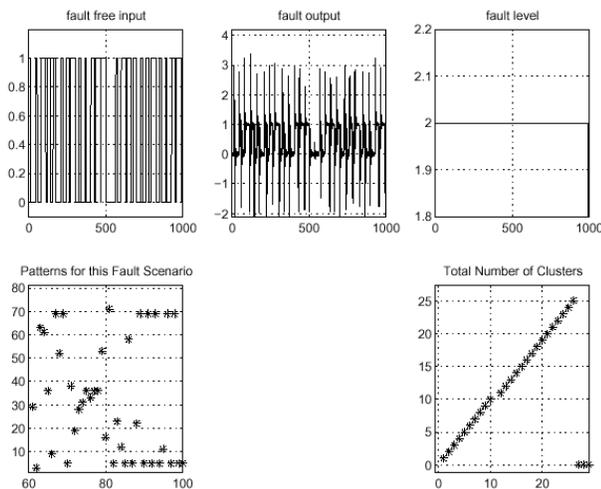


Fig. 4.2 Fault Scenario without Residual Evaluation

The fault scenario takes into account a time variance of 0.12s as well as confidence value responses (Fig. 4.3). In this case, the fault is a time delay at the output of case study. This fault modifies the residual value at the output of UIO, therefore the behaviour of selected patterns is modified. This results in a decrement of the confidence value, keeping a response of 82% during fault free scenarios, and a response of 19% and 41% during fault scenario.

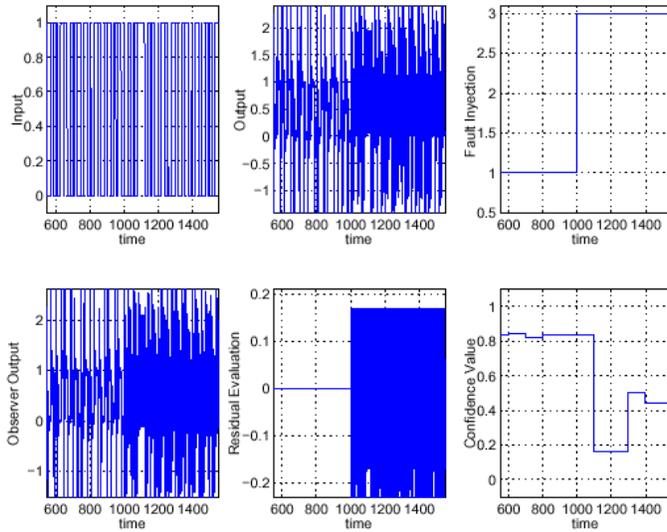


Fig. 4.3 Fault and Fault-Free Scenarios with Residual Evaluation and Time Invariant Behaviour

In the fault-free scenario (Fig. 4.4), several types of patterns are classified. However, confidence value keeps a regular result, around 80%. In this case, 100% trust has not been achieved due to inherent time variant. As expected, residual value remains null during this scenario. Although, element response has not been accurately controlled by the local control law.

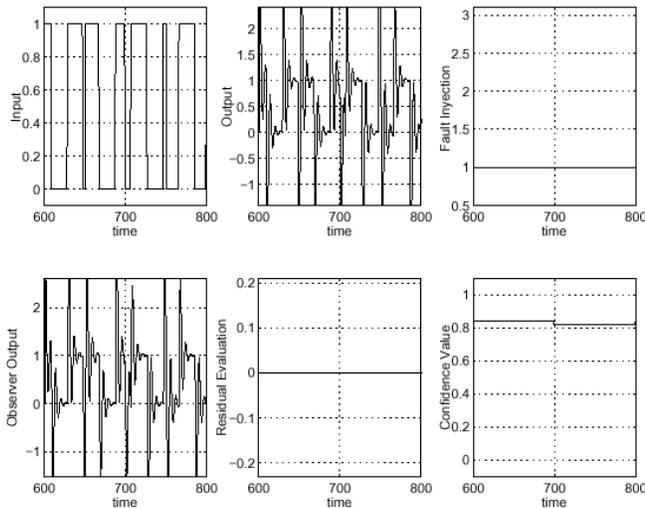


Fig. 4.4 Fault-Free Scenario with Time Variance Input Behaviour with Residual Evaluation

5. Conclusions

The combination of neural networks and analytical redundancy enhances the capabilities for fault localization. The key issue here is how data tends to be processed by the neural networks, in order to classify patterns. The integration of two neural networks in cascade allows the classification of time variant behaviour even during fault presence. This is possible due to an ART2A is used to determine the boundary between clusters from SOM output.

An important restriction of this approach is the sampling time window. This is inherent to the sampling technique. It is necessary to define a lower bound in terms of the number of samples, which is stated here as M . This value has a direct effect over Δt and $freq_{fault}$. However, in terms of sampling, there is no restriction regarding an upper bound. Its only practical restriction has to do with the response time. In such a case, faults that occur

faster than this fault localization approach give a useless classification. Moreover, there is another clear restriction regarding to the possible *explosion* of the number of patterns.

The use of a bank of observers presents a formal approach in order to determine an isolated fault. This allows the isolation of fault-free and fault-specific scenarios, with very low time consumption for on-line performance. However, when an unknown scenario appears, it cannot declare a specific performance. Thus, neural networks represent an advantage as geometric classifiers. The integration of both, bank of observers and non-supervised neural networks, enhances the classification of abnormal scenarios such as unknown faults even with time variation.

Acknowledgments

The authors wish to thank projects PAPIIT-UNAM Num. 106100 and Num. 105303 for their financial support.

References

- Abe, S.; "Pattern Classification: Neuro- Fuzzy Methods and their Comparison"; Springer-Verlag, UK, 2001.
- Chen J., and Patton, R.; "Robust Model-Based Fault Diagnosis for Dynamic Systems"; Edit. Kluwer Academic Press, USA, 1999.
- Chiang, L. H., Russell, E. L. and Braatz R. D.; "Fault Detection and Diagnosis in Industrial Systems"; Springer-Verlag, UK, 2001.
- Isermann, R.; "Integration of Fault Detection and Diagnosis Methods", IFAC SAFEPROCESS'94, Espoo, Finland, Vol. 2, pp. 597-612, 1994.
- Kiviluoto K.; "Topology Preservation in Self-organizing Maps"; Technical Report A29, Helsinki University of Technology, Laboratory of Computer and Information Science, 1995.
- Lee, D., Thompson, H. A., and Bennett, S.; "PID control for a distributed system with a smart actuator"; Digital Control: Past, Present and Future of PID Control (PID'00). Proceedings IFAC Workshop, pp.499-504, 2000.
- Liling, Ma, Yinghua, Y., Fuli, W. and Ningyun Lu; "A Neural Network Observer Approach for Actuator Fault Detection and Diagnosis in Non-linear Systems"; 15th Triennial World Congress, IFAC, Session slot T-We-M10, Barcelona, Spain, 2002.
- Lopez-García, H., and Machón González I.; "Self-Organizing Map and Clustering for Wastewater treatment monitoring"; Engineering Applications of Artificial Intelligence, Vol. 17, pp. 215-225, 2004.
- Masten, M. K.; "Electronics: The intelligence in Intelligent Control"; IFAC Symposium on Intelligent Components and Instrument for Control Applications, Annecy, France, pp. 1-11, 1997.
- Nelles, O.; "Non-Linear Systems Identification"; Springer-Verlag, Berlin, Germany, 2001.
- Polycarpou M., and Vemuri A.; "Learning Methodology for Failure Detection and Accommodation"; IEEE Control Systems, June, pp. 16-24, 1995.
- Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K.; "A review of process fault detection and diagnosis Part I: Quantitative model-based methods"; Computers and Chemical Engineering, vol. 27, pp. 293-311, 2003a.
- Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K.; "A review of process fault detection and diagnosis Part II: Qualitative models and search strategies"; Computers and Chemical Engineering, vol. 27, pp. 313-326 2003b.
- Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K.; "A review of process fault detection and diagnosis Part III: Process history based methods"; Computers and Chemical Engineering, vol. 27, pp. 327-346, 2003c.
- Wang, W. H., Liu, K. D., Zhou, D. H., and Wang C. H.; "A Fuzzy and Rough sets Integrated Approach to Fault Diagnosis"; 15th Triennial World Congress, IFAC, pp. 1031-1037, Barcelona, Spain, 2002.