Chapter Number

Fault Localization Upon Non-Supervised Neural Networks and Unknown Input Observers for Bounded Faults

Benítez-Pérez H.[1] and Ortega-Arjona J. L.[2]

[1]Departamento de Ingeniería de Sistemas Computacionales y Automatización, Instituto de Matemáticas Aplicadas y en Sistemas, Universidad Nacional Autónoma de México, Apdo. Postal 20-726., Admón. No. 20, Del. A. Obregón, México D. F., CP. 01000
[2]Departamento de Matemáticas, Facultad de Ciencias, UNAM, Ciudad Universitaria, CP. 04510, México City,
México

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. 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. (Venkatasubramanian V., et al., 2003a) (Venkatasubramanian V., et al., 2003b) (Venkatasubramanian V., et al., 2003c) 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).

Several approximations have reviewed this constraint like (Su T., et al., 2008) where function approximation is pursued using a hybrid artificial neural network where data analysis becomes crutial for this purpose. In a similar manner (Zhong & Wang, 2008) presents a support vector regression where data uncertainty is studied, giving a good idea about the inherent characteristics of the data in order to by analysed. Several issues need to be addressed in order to study data analysis for system identification and representation, most of there are out of the scope of this paper. The goal of the approach followed here is to

enhance fault localization by local model identification (Billings & Wei, 2005) (Wang X., et al., 1999) (Li D., et al., 2005) and classification to unknown scenarios through a self organizing map as shown in Fig. 1. Similar approaches haven been studied by Benitez-Perez (Benítez & García, 2005) (Benítez P., et al., 2007).
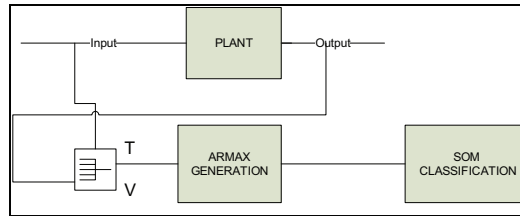

Fig. 1. Schematic approximation.

Furthermore (Benítez P. H. & Benítez P. A., 2009) present a novel strategy based upon the combination of parameter modeling (to extract several characteristics of the observed system) and the self organizing map (to classify the type of parameters extracted from the parametrization stage) in order to detect and classify online unknown behaviour.

For the purpose of this work it is necessary to define that 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. 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 D., et al., 2000) proposes 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 W., 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.

On the other hand, the use of feature extraction by neural networks only presents the disadvantage of robustness in terms of inherent data uncertainties and large quantity of necessary data. Different authors have explored similar strategies previously, like (Abe S., 2001) where feature extraction using local parametric models are proposed giving valuable results, however, with the drawback of bounded system response. A strategy for fault diagnosis that integrates an ART2A network and a Kohonen neural network can be feasible since the objective is to combine both strategies in order to generate two subsystems capable to overcome glitches and redundant data representations (Abe S., 2001). The attention and oriented subsystems, following the ART2A topology and the Kohonen Neural Network, are used to perform a learning strategy from the evaluation result of the Kohonen Network. This strategy allows diagnosing faults on-line with the inherent uncertainty to SOM variation due to the plasticity-stability dilemma. A fundamental work has been introduced by (Kiviluoto, 1995) where an extended review is given about topics related to sensors patterns and stability-plasticity trade-off inherent to the ART2A network. Interesting comments are included in here on how time window data can be monitored in order to identify abnormal situations, as well as, how data should be treated in terms of normalization, time scaling and filtering and their comparison prior to declare a winner selection. Further developments are addressed (García H. & Machón G. I. 2004) focusing on

the use of a parallel ART2A network approach based on wavelet decomposition where clustering is defined on wavelet domain although it is not proposed for a dynamical system. A complementary strategy for fault diagnosis is based on feature extraction following wavelets decomposition. Feature extraction based upon wavelets for dynamical systems presents the advantage of scales decomposition allowing several possibilities of fault detection depending on the scale of the fault. Similarly, fault detection can be easily engage if a source of information is decomposed in several fruitful components. These components can be taken as parameter vectors where several signal conditions are highlighted depending on the resolution. Moreover, these need to be combined in a fair strategy in order to classify similar behaviors. To do so, these are classified using a Self Organizing Map (SOM) where each vector is processed as consecutive input. The result of this classification would give a number of selected patterns depending on the learning rate with respect to a time window. Nevertheless, the plasticity-stability is not overcome by this technique (Benítez P. H. & Benítez P. A. 2010).

Following this review this work is focused on load systems modeling and nonlinear classifiers. 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.

This work 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. 2.1. 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.

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.
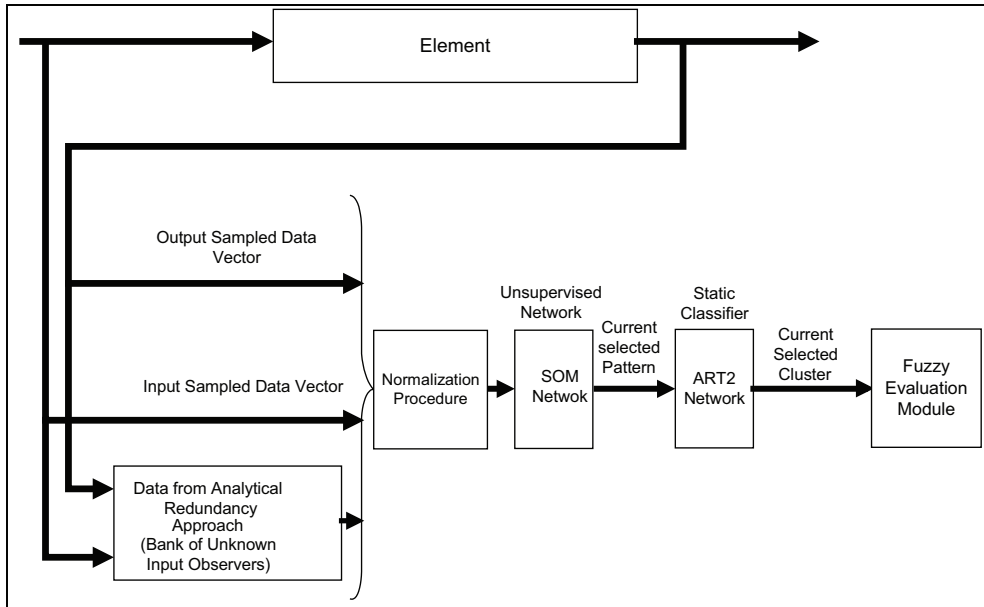
Fig. 2.1. Process diagram for Fault Localization

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. Each type of variable has M samples, organized as rows. The whole bunch of variables are integrated by three scenarios, organized as columns.

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.

$$\text{frq}_{\text{fault}} \geq \frac{4}{\Delta t} \tag{2.1}$$

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.

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 relays 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 (Billings & Wei, 2005). Proposals such as an entropy-based fuzzy clustering method defines cluster based on the entropy of each point with respect to a center. In the case of SOM and ART2A, these have the peculiarity to classify unknown scenarios in a predictable behavior. 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.

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.

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. 2.2).
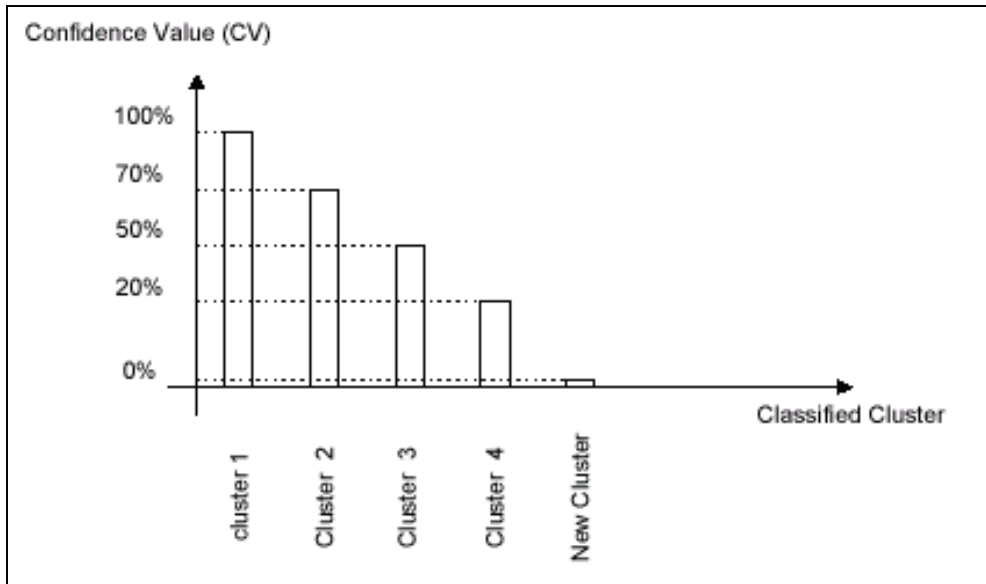

Fig. 2.2. Confidence Value Representation

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. 2.3).

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. 2.3). 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.
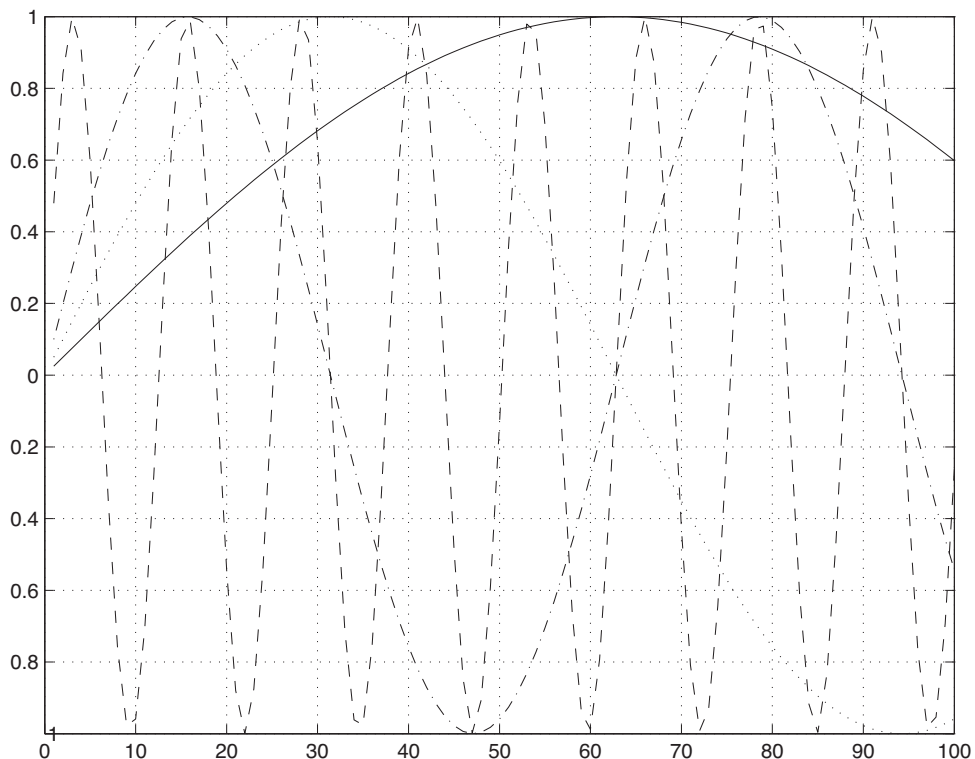
Fig. 2.3. First scenario used to evaluate the approach (a section of first 100 seconds)

		SOM		ART2A	
η (Parameter related to SOM)	ρ (Parameter related to 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. 2.4 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.
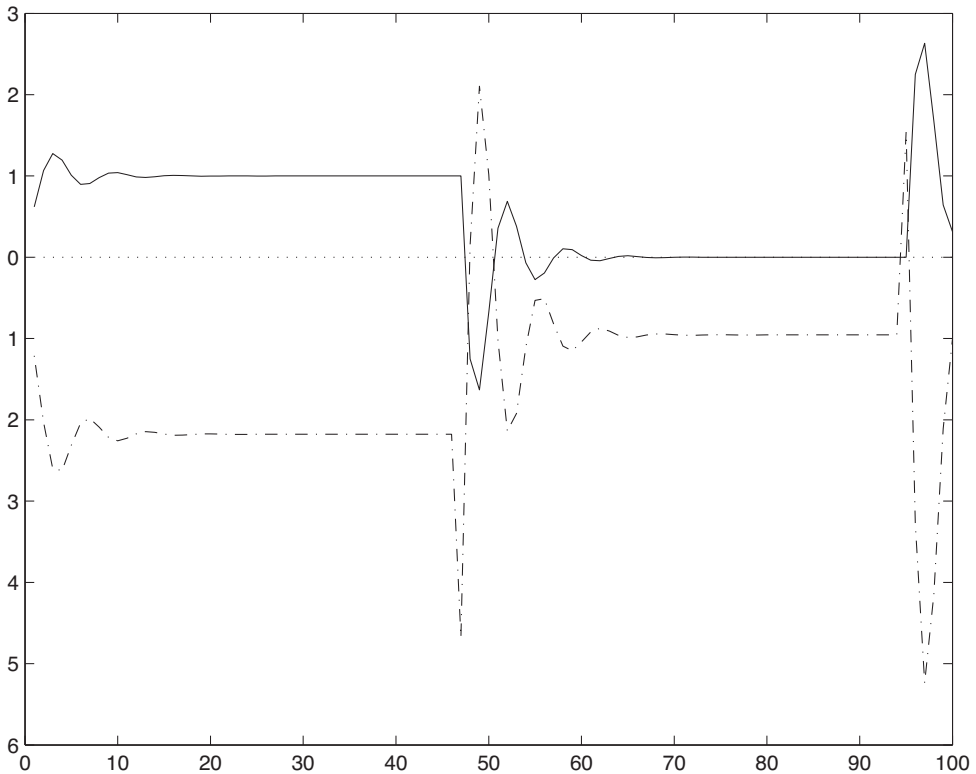

Fig. 2.4. Second scenario used to evaluate the approach

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.

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 (Wang X., et al., 1999) (Li D., et al., 2005) has been performed, obtaining a topographic error calculated as follows (eqn. 2.2).

| 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
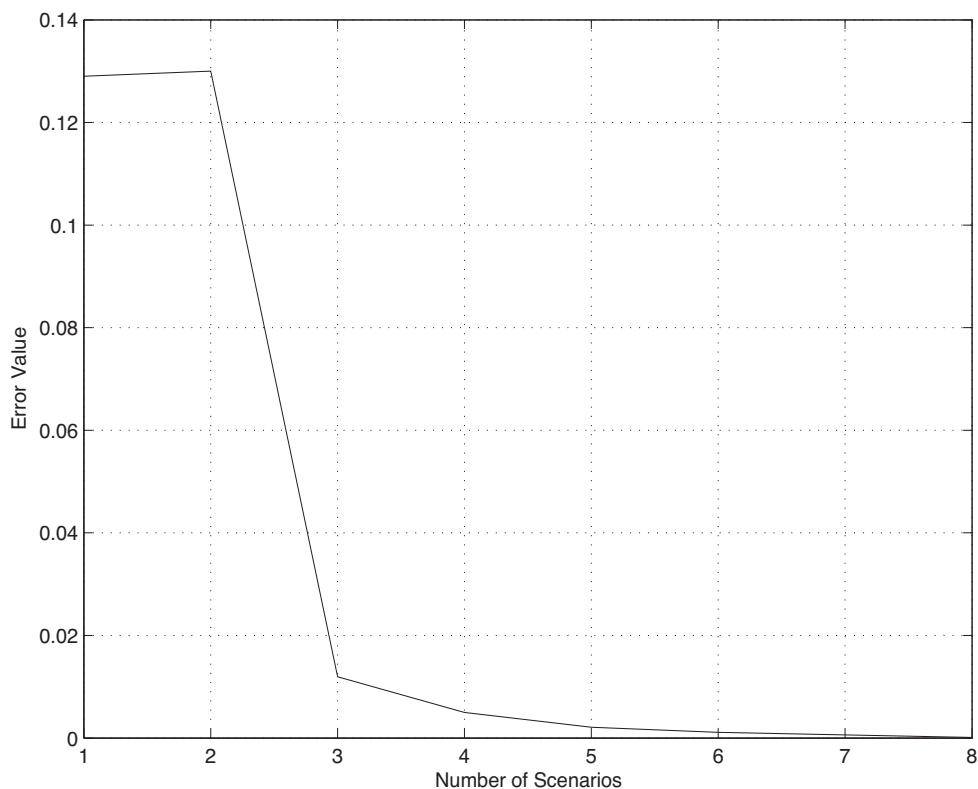
Fig. 2.5. Error Measure Performance with respect to η and ρ numbers

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

where N is the number of samples, x_k is the kth 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. 2.5). 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.

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.

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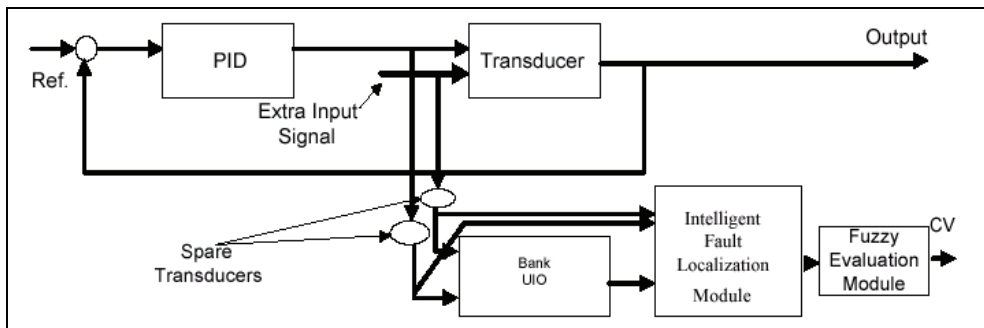C &= \begin{bmatrix} 0 & 1 \end{bmatrix}
\end{aligned}
$$ (3.1)

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 (Su T., et al., 2008). 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).

$$
\begin{array}{cc}
\text{First Observer} & \text{Second Observer} \\
\end{array}
$$

$$
H = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \qquad H = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
$$

$$
F = \begin{bmatrix} 1.1 & -0.1 \\ 0.0 & 20.0 \end{bmatrix} \qquad F = \begin{bmatrix} 1.21 & -0.1 \\ 0.0 & 20.0 \end{bmatrix}
$$

$$
K = \begin{bmatrix} 0.0001 \\ 0.0001 \end{bmatrix} \qquad K = \begin{bmatrix} 0.0001 \\ 0.0001 \end{bmatrix}
$$

$$
G = \begin{bmatrix} 1.8 & -2.1 \\ 0.0 & 0.0 \end{bmatrix} \qquad G = \begin{bmatrix} 1.91 & -2.1 \\ 0.0 & 0.0 \end{bmatrix}
$$

(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).

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

(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.
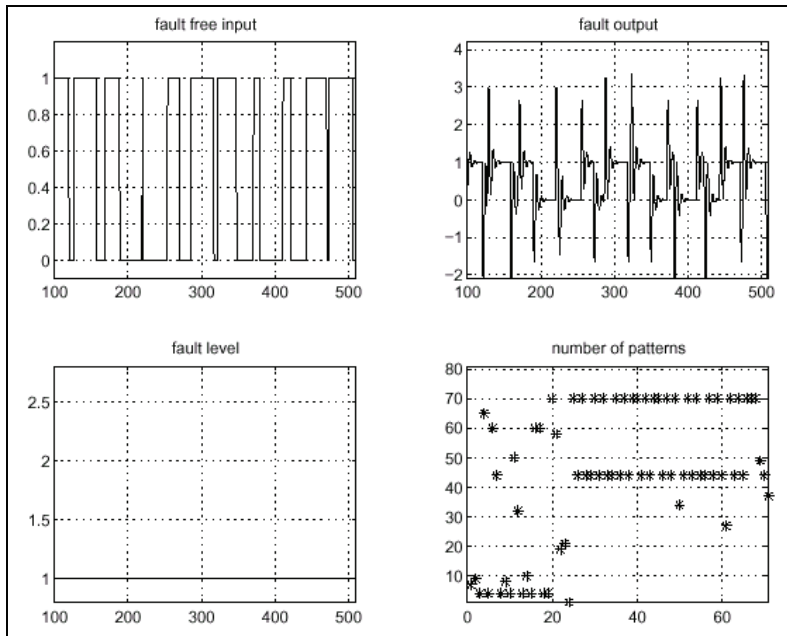

Fig. 4.1 Fault-Free Scenario with Time Variance Behaviour
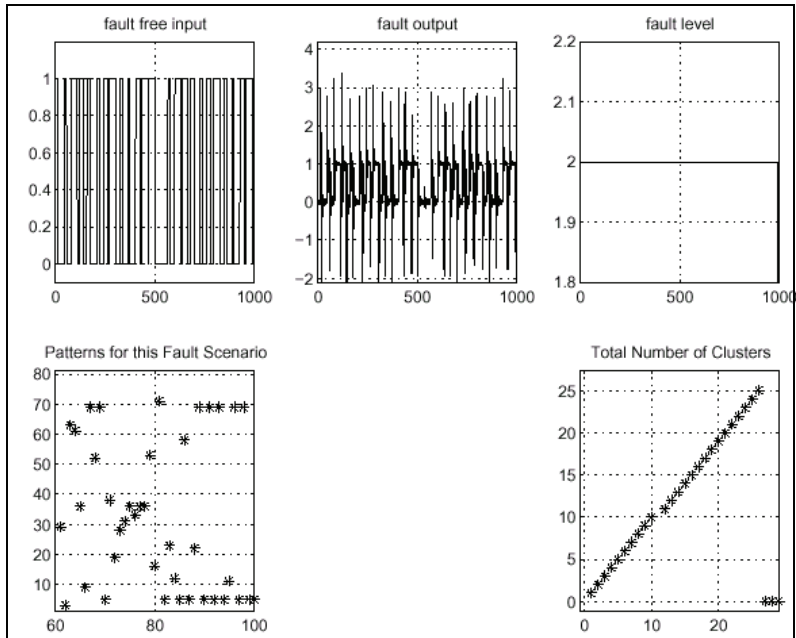
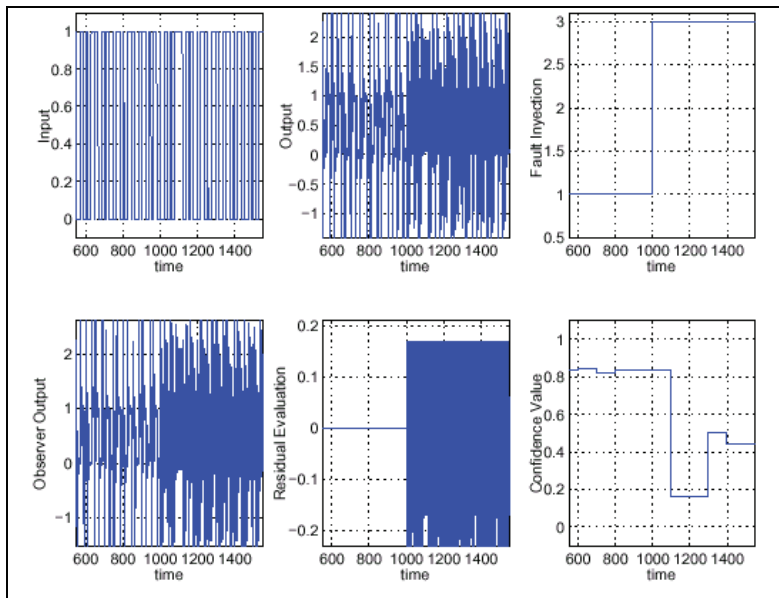Fig. 4.2. Fault Scenario without Residual Evaluation

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

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.

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.
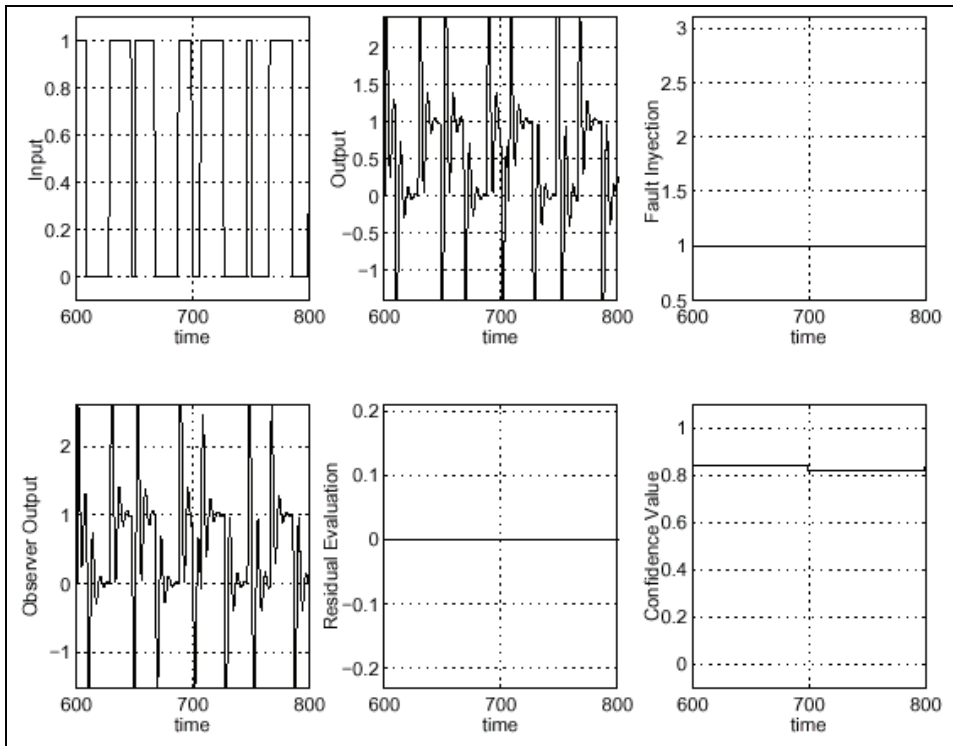

Fig. 4.4. Fault-Free Scenario with Time Variance Input Behaviour with Residual Evaluation

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 frq_{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.

6. Acknowledgments

7. References

Nelles, 0. (2001). *Non-Linear Systems Identification*, Springer-Verlag, Berlin, Germany.

Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K.. (2003a). Computers and Chemical Engineering. *A review of process fault detection and diagnosis Part I: Quantitative model-based methods*, vol. 27, pp. 293-311.

Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K.. (2003b)., Computers and Chemical Engineering. *A review of process fault detection and diagnosis Part II: Qualitative models and search strategies*, vol. 27, pp. 313-326.

Venkatasubramanian V., Rengaswamy R., Kavuri, S., and Yin K. (2003c). Computers and Chemical Engineering. *A review of process fault detection and diagnosis Part III: Process history based methods*, vol. 27, pp. 327-346.

Liling Ma Yinghua Y., Fuli W. and Ningyun Lu. (2002). 15th Triennial World Congress, IFAC, Session slot T-We-M10. *A Neural Network Observer Approach for Actuator Fault Detection and Diagnosis in Non-linear Systems*, Barcelona, Spain.

Su T, Jhang J., and Hou C. (2008). International Journal of Innovative Computing Information and Control. *A Hybrid Artificial Neural Networks and Particle Swarm Optimization for Function Approximation*, Vol. 4, No. 9, pp. 2363 -2374.

Zhong P. and Wang L. (2008). International Journal of Innovative Computing Information and Control. *Support Vector Regression with Input Data Uncertainty*, Vol. 4, No. 9, pp- 2325-2332.

Billings S., and Wei H. (2005). IEEE Transaction on Neural Networks. *A New Class of Wavelet Networks for Nonlinear System Identification*, Vol. 14, No. 4, pp. 862-874.

Wang X., Chen B., Yang S. and McGreavy C. (1999). Computers and Chemical Engineering. *Application of Wavelets and Neural Networks to Diagnostic Systems Development, 2, an Integrated Framework and its Application*, Vol. 23, pp. 945-954.

Li D., Pedrycz W. and Pizzu N. (2005). IEEE Transaction on Biomedical Engineering. *Fuzzy Wavelet Packet based Feature Extraction Method and Its Application to Biomedical Signal Classification*, Vol. 52, No. 6, pp. 1132 -1139.

Benítez-Pérez, H. and García-Nocetti, F. (2005). *Reconfigurable Distributed Control*, Springer Verlag.

Benítez-Pérez H., García-Nocetti F. and Thompson H., International Journal of Innovative Computing, Information and Control. *Fault Classification Based upon Self Organizing Feature Maps and Principal Component Analysis for Inertial Sensor Drift*, Vol. 3, April 2007, Issue 2.

Benítez-Pérez H. and Alma Benítez-Pérez. (2009). International Journal of Innovative Computing, Information and Control, IJICIC. *The use of ARMAX strategy and Self Organizing Maps for Feature Extraction and Classification for Fault Diagnosis*, Vol. 5, No. 12, Diciembre, pp. 4787-4796.

Lee D., Thompson H. A., and Bennett S. (2000). *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.

Wang W. H., Liu K. D., Zhou D. H. and Wang C. H. (2002) 15th Triennial World Congress, IFAC. *A Fuzzy and Rough sets Integrated Approach to Fault Diagnosis*, pp. 1031-1037, Barcelona, Spain.

Abe S. (2001). *Pattern Classification: Neuro- Fuzzy Methods and their Comparison*, Springer-Verlag, UK.

Kiviluoto K.. (1995) *Topology Preservation in Self-organizing Maps*, Technical Report A29, Helsinki University of Technology, Laboratory of Computer and Information Science.

Lopez-García, H., and Machón González I. (2004). Engineering Applications of Artificial Intelligence. *Self-Organizing Map and Clustering for Wastewater treatment monitoring*, Vol. 17, pp. 215-225.

Benítez-Pérez H. and Alma Benitez-Pérez. (2010). Accepted en International Journal of Innovative Computing, Information and Control, IJICIC. *The use of WAVELET strategy and Self Organizing Maps for Feature Extraction and Classification for Fault Diagnosis.*