NEURO-FUZZY TECHNIQUES IN FDI SYSTEM FOR SUGAR FACTORY ACTUATORS

Mário J. G. C. Mendes *, **Marek Kowal** **,
Józef Korbicz ** and **José M. G. Sá da Costa** ***

* *IDMEC/ISEL - Instituto Superior de Engenharia de Lisboa*
Polytechnic Institute of Lisbon, Dept. of Mechanical Engineering
Rua Conselheiro Emidio Navarro, 1949-014 Lisboa, Portugal
fax:+351 21 8317057, e-mail:mmendes@dem.isel.ipl.pt

** *Technical University of Zielona Góra, Institute of Control and Computation Engineering*
ul. Podgórna 50, 65-246 Zielona Góra, Poland
fax:+48 68 3254615,e-mail:{M.Kowal,J.Korbicz}@irio.pz.zgora.pl

*** *Technical University of Lisbon, Instituto Superior Técnico*
Dept. of Mechanical Engineering, GCAR/IDMEC
Av. Rovisco Pais,1049-001 Lisboa, Portugal
fax:+351 21 8498097, e-mail:sadacosta@dem.ist.utl.pt

Abstract: Fault diagnosis systems have an important role in industrial plants because the early fault detection and isolation (FDI) can minimize damages in the plants. The main aim of this work is to propose a two-stage neuro-fuzzy approach as a fault diagnosis system in dynamic processes. The first stage of the system is responsible for fault detection and is implemented using a neuro-fuzzy model. The second stage of the system is responsible for fault isolation and is built using an hierarchical structure of fuzzy neural networks. The FDI system is applied to fault diagnosis in the sugar factory actuators.

Keywords: Fault diagnosis, fuzzy neural networks, hierarchical structure, abrupt faults, incipient faults, actuators.

1. INTRODUCTION

Modern industrial systems are liable to numerous faults due to their complexity. In many applications, increased requirements on productivity and performance lead to plants operating near to the design limits and therefore, faults can occur in the components of the process or in the sensors and actuators. So, the early detection and isolation of faults can minimise damages in the plants and reduce the effects of the faults in the industrial environment. Systems that have the capacity of detect, isolate and identify faults are called fault diagnosis systems. These systems are very im-

[1] This research was supported by the EU FP V Research Training Network DAMADICS (Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems.

portant for the increase of safety, reliability and availability of processes.

In dynamical processes, faults may be divided into two main classes: abrupt faults and incipient faults. Abrupt faults give rise to jumps in the process parameters, resulting in an appreciable deviation from normal system behaviours. On the other hand, incipient faults affect the process behaviour slowly and may take a long time before being detected. The fault detection and isolation of these faults can be achieved with model-based processing of all measured variables, using either qualitative or quantitative modelling.

The most popular quantitative modelling approaches are based in the mathematical model of the process and use conventional design techniques supported by linear process models. However, requirements for accurate analytical model imply that any resulting modelling simplification or error will affect the performance of the resulting FDI system. This is more important for dynamically nonlinear and uncertain systems, which represent the majority of real processes (Patton *et al.*, 2000; Calado *et al.*, 2001). To overcome this problem the qualitative modelling methods became a good solution.

Qualitative modelling approaches make use of more abstract models based on Artificial Intelligence (AI) techniques like fuzzy logic or neuro-fuzzy networks. Fuzzy logic enables the system behaviour description with "*if-then*" rules and artificial neural networks can be trained to have the required mapping between inputs and outputs. They can be used to overcome the difficulties of conventional fault detection techniques to deal with nonlinear behaviours. In this way is constructed a robust FDI system that combine both quantitative and qualitative information (Patton *et al.*, 2000; Calado *et al.*, 2001).

In recent years the idea of using AI methods has been widely exploited in the fault diagnosis applications. The main part of research has been done for neural networks, expert systems, fuzzy and neuro-fuzzy systems (Patton *et al.*, 2000a; Calado *et al.*, 2001; Patton and Korbicz(Eds.), 1999). These methods seem to be a promising solution for high complex systems where other methods cannot be used. Especially, the neuro-fuzzy methods have received much attention because of their capability to use simultaneously qualitative and quantitative knowledge and ability to representation of some kinds of uncertainty present in real processes (Takagi, 2000; Rutkowska and Zadeh(Eds.), 2000; Nauck *et al.*, 1997; Yager and Filev, 1994). For these reasons this work is focused in qualitative model methods for FDI systems.

It has been constructed a fault diagnosis system, as described in the Fig. 1. This system have two stages, the first stage is responsible for fault detection and is based in neuro-fuzzy models for all subsystems considered for the fault diagnosis and in which the faults will occur. The fault detection is required for symptom generation and is the task where are calculated the residuals (fault indicator, based on deviation between measurements and model outputs). Neuro-fuzzy models have a multilayer structure like ordinary neural networks. It means that the learning algorithms for the neural networks can be applied to this system and after training, fuzzy rules can be extracted and analysed. However, the fault detection subsystem can determine only that a fault occurred in the process and because of that, it will be necessary a fault isolation subsystem to decide which kind and location of faults are present in the process.

The second stage is responsible for fault isolation, decision making or classification of residuals (with or without more measurement variables depending of the process) to determine the type, location and reasons of the faults. This decision is described by a discrete mapping from continuous symptom space to discrete fault space. Such task is carried out by a classifier, which determines what kind of faults are present in the process.

In this approach, the second stage has been done with an hierarchical structure of fuzzy neural networks (HSFNN) that combines the advantages of both fuzzy reasoning and neural networks. Fuzzy reasoning is capable of handling uncertainty and imprecise information, while the neural networks are capable of learning from examples. The adoption of an HSFNN for fault isolation aims at the development of an architecture that can localize abrupt and incipient single and multiple faults correctly, or at least with a minimum misclassification rate, from only single abrupt fault symptoms (residuals), and be easily trained (Mendes, 2001; Mendes *et al.*, n.d.; Calado and Sá da Costa, 1999; Watanabe *et al.*, 1994).

The proposed fault diagnosis system is applied to fault diagnosis of sugar factory actuators valves.

The paper is organised as follows. Section 2 provides a description of the fault detection system, where are described the neuro-fuzzy models used. Section 3 presents the description of the fault isolation system based in one HSFNN. Section 4 describes the case study, two valves of the evaporation station from Lublin sugar factory. Section 5 presents the simulation of faults and the training

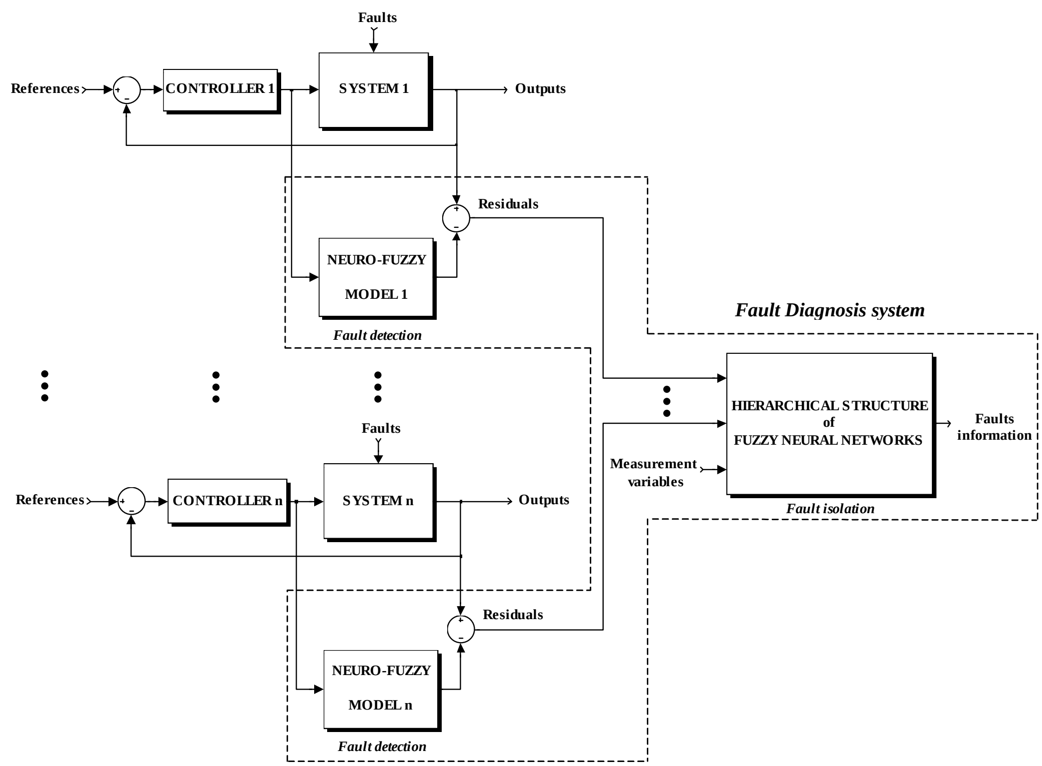

Fig. 1. Fault diagnosis system.

data. This section also presents the results of this fault diagnosis system applied to the sugar factory actuators. Finally, in Section 6 some concluding remarks are given.

2. FAULT DETECTION

The first stage of the fault diagnosis system is the neuro-fuzzy model for residuals generation.

2.1 *Neuro-fuzzy modelling*

Generally, the methods of fault detection can be divided into two groups: a simple process variable monitoring and a more complex model based methods. For a simple fault that can be detected by a single measurement, a conventional alarm circuit may be appropriated. However, since it is usually very difficult in complex industrial systems to directly measure the state of the process, more sophisticated solutions are needed. In this case a model based approach will be more suitable. This requires modelling of the process and may be called an analytical redundancy because the model and the process work in parallel. The idea of such fault detection is to compare output signals of the model and the process, thereby generating a residual or an output error, which is used to make the decision about the state of the process. This approach makes it possible, to detect small scale and incipient faults quickly and reliably. These are the reasons of using such systems in life-critical applications where even small defects can cause big damages.

Different methods of model design are available. The most popular are analytical and AI methods. Analytical methods (the Kalman filter, the Luenberger observer, etc.) can be applied to the process expressed by a mathematical model, usually of the linear type. On the other hand, artificial intelligence methods do not require the mathematical model of the process, using the process qualitative and quantitative knowledge to build the models. These methods have been investigated intensively in the recent years (Calado *et al.*, 2001; Mendes, 2001; Korbicz *et al.*, 2001; Kowal and Korbicz, 2000; Koscielny *et al.*, 2000; Calado and Sá da Costa, 1999).

Neural nets and neuro-fuzzy applications have received much attention due to their fast and robust implementation, their performance in learning arbitrary non-linear mappings, and their abilities of generalization. Especially, the neuro-fuzzy approach has been actively employed for modelling. It integrates fuzzy modelling techniques and neural nets learning abilities. The main feature of such approach is to join the strengths of both quantitative and qualitative modelling methodologies, in order to achieve a learning sys-

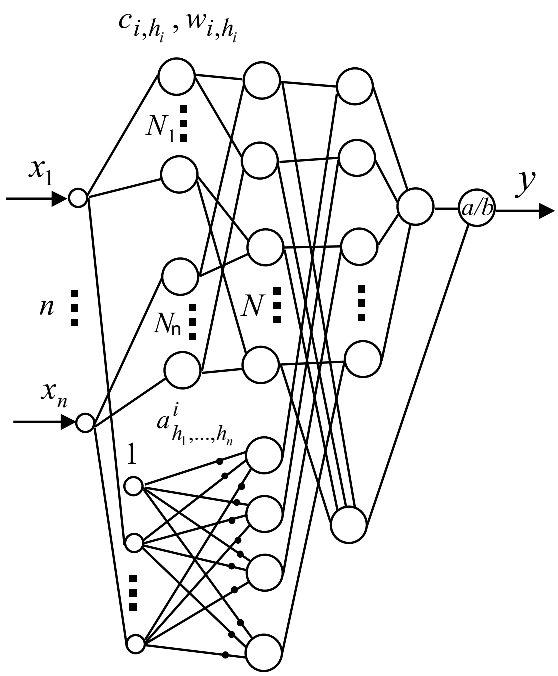

Fig. 2. The structure of TSK neuro-fuzzy network.

tem with transparent knowledge (Rutkowska and Zadeh(Eds.), 2000).

However, neuro-fuzzy techniques encounter the problem of exponential growth of the network structure when the dimension of input/output space increases or the number of fuzzy partition increases (Jin, 2000).

The complexity of the neuro-fuzzy network structure causes the drastic growth of the computational cost during the learning process. Moreover, knowledge coded as rule base is unclear due to a large number of rules. Such situation can reduce the usage of the neuro-fuzzy networks to simply industrial processes. Fortunately, there are several methods for reducing the neuro-fuzzy network structure without loss in their accuracy (Jin, 2000; Runkler and Bezdek, 1999). The most commonly used network structure for system modelling is Takagi- Sugeno-Kang (TSK) neuro-fuzzy network (Nelles *et al.*, 2000). The topology of such network is shown in Fig. 2, where the following notations are used: $x_1, ..., x_n$ are the inputs, y is the output, n is the number of inputs, N is the number of rules, N_i is the number of fuzzy partitions for ith input and w_{i,h_i} are the weights.

The linear conclusion, height method of defuzzification, Gaussian functions as membership functions of the fuzzy sets and aggregation operation defined as a product are used to build this network. Three types of weights are tuned during the learning process. The weights c_{i,h_i} and the weights w_{i,h_i} are the parameters of membership functions of the fuzzy sets. The parameters c_{i,h_i} are defined as centers of the Gaussian functions and the parameters w_{i,h_i} are defined as widths of the Gaussian functions. The weights $a^i_{h_1,...,h_n}$ are the parameters of the linear equations, which express conclusions.

Generally TSK neuro-fuzzy models have a better performance in modelling then other structures due to their possibility to representation of non-linear systems by several local linear models (Fig. 3).

2.2 *Structure simplification and learning of TSK neuro-fuzzy network*

The learning process of the neuro-fuzzy network has been divided into two phases. In the first step clustering methods have been used to optimise the network structure and prepare the initial values of the parameters. In the second step the gradient descent method has been used to tune all parameters. Two clustering algorithm have been used at the first step, the Mountain Method and the Fuzzy C-Mean (FCM) algorithm (Runkler and Bezdek, 1999; Yager and Filev, 1994). They allow the reducing the number of the rules and prepare better initial parameters than random. It is an important operation because the neuro-fuzzy networks encounter the problem of exponential growth of rules as the dimension of the input space increases.

Table 1. The problem of exponential growth of rules for the Neuro-Fuzzy networks.

Inputs	Partitions defined for each input	Rules
4	3	81
6	4	4096
10	4	1048576

The Table 1 describes some sample structures of the neuro-fuzzy networks. At the first the mountain method is used to determine an optimal number of the rules.

The basic idea of using the mountain method is to group the input data into clusters and use one rule for each cluster. The example in the Fig. 4 shows how the cluster type partitioning method reduce the number of the rules used.

The mountain method do not require predetermination of the number of the centers, but it needs the intersection points of the grid lines as candidates for cluster centers where the grid lines are drawn on the input space. The mountain method cannot determine the size or shape of the clusters. The FCM algorithm to determine initial values of

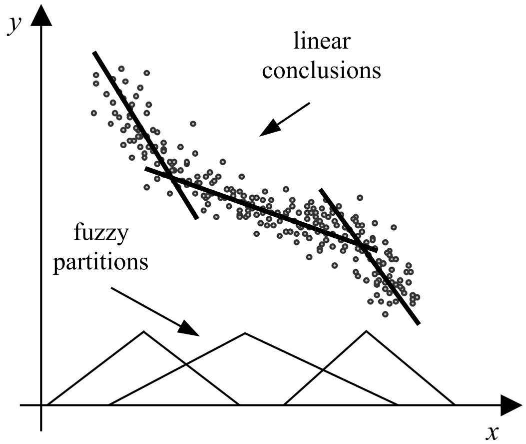

Fig. 3. The locally linear models to representation of non-linear system where x is the input and y is the output.

the parameters of the neuro-fuzzy network uses the results generated by the mountain method. The FCM algorithm obtains from this method the number of clusters and the initial cluster centers, as well determines the exact centers of the membership functions and their widths. At the second step the gradient descent method is used to tune all parameters.

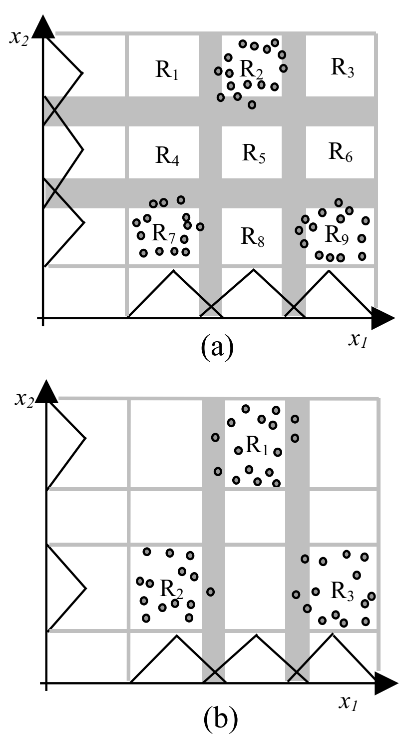

Fig. 4. Grid type partitioning method (a), cluster type partitioning method (b).

3. FAULT ISOLATION

The second stage of the fault diagnosis system of Fig. 1 is an HSFNN for fault isolation (classification of residuals).

3.1 *Hierarchical structure of fuzzy neural networks*

In the current approach an HSFNN has been developed for fault isolation purposes (Mendes, 2001; Calado and Sá da Costa, 1999; Watanabe *et al.*, 1994). It is aimed to isolate (classify) multiple simultaneous faults from only single abrupt fault symptoms. The hierarchical structure has three levels where several fuzzy neural networks (FNNs) are used, as shown in Fig. 5.

The lower level consists of one FNN where residuals (and also measurement variables, if necessary to diagnose some fault that aren't dependent from the neuro-fuzzy model) are used as inputs. At the medium level a number of FNNs (structurally identical) that is equal to the number of single fault scenarios considered, are used. Each FNN at the medium level is also fed with all the residuals and measurement variables, and each one is associated with an output of the FNN at the lower level, corresponding to a particular single fault. The upper level consists in one fuzzy OR operation between the outputs of the FNNs of the medium level. There are different fuzzy OR operation available (Klir and Folger, 1988), but in this approach is used the max-min operations to construct the fuzzy OR. So, the final fault vector is the result of this operation, the maximum values of each fault for all outputs from the medium level (1).

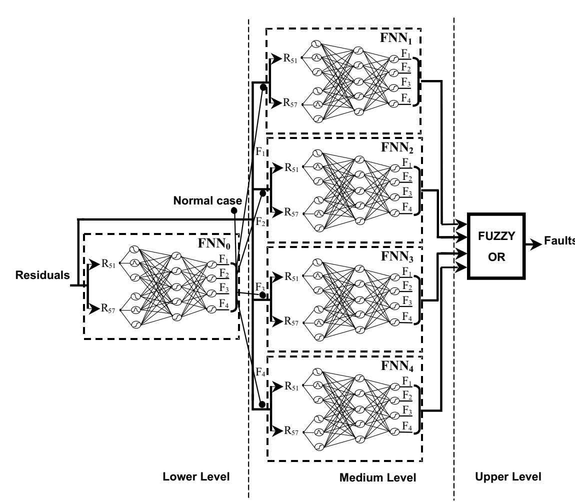

Fig. 5. Hierarchical structure of Fuzzy Neural Networks.

$$F_i = max[\mu(F_i^{FNN_1}), ..., \mu(F_i^{FNN_n})] \; , \; i \geq 1 \quad (1)$$

The elements of the set used in the fuzzy OR operation are determined by the outputs of the FNN at the lower level. Thus, if the ith and jth outputs of the FNN at the lower level is taking values greater than a specified threshold (e.g., 0.5), then the outputs of the ith and jth FNNs at the medium level form the elements used in the fuzzy OR operation. However, if only one output of the FNN at the lower level is taking a value greater than this specified threshold, then the corresponding FNN in the medium level is used to confirm that this fault is a single fault, or to diagnose multiple faults. Obviously, the multiple faults must include the one corresponding to the output of the FNN at the lower level.

In contrast to the conventional multi-layer feed-forward neural network, the adopted FNN has an additional fuzzy input layer that maps the increment of each residuum into fuzzy sets. Therefore, the fuzzification layer converts each input into the quantity space, q_f={decrease, steady, increase}, by association with three types of neurons. The processing elements of the fuzzification layer related to the fuzzy sets decrease and increase use the complement sigmoid function and the sigmoid function, respectively, as their activation functions. On the other hand, the other processing elements of the fuzzification layer related to the fuzzy set steady use the Gaussian function.

The membership functions used in the input fuzzy layer have been achieved by fuzzy clustering algorithm, the Gustafson-Kessel algorithm (Gustfason and Kessel, 1979), which is implemented in the "*Fuzzy Modeling and Identification Toolbox*" for MATLAB (Babuška, 1998) and have been adjusted with the parametric sigmoidal equations (2)-(4):

$$\mu = \begin{cases} \frac{1}{2}(\frac{x-a}{w_1})^2 & \text{if } a < x < a + w_1 \\ 1 - \frac{1}{2}(\frac{b-x}{w_1})^2 & \text{if } a + w_1 < x < b \\ 1 & \text{if } b < x < c \\ 1 - \frac{1}{2}(\frac{c-x}{w_2})^2 & \text{if } c < x < c + w_2 \\ \frac{1}{2}(\frac{d-x}{w_2})^2 & \text{if } c + w_2 < x < d \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$w_1 = \frac{b-a}{2} \quad (3)$$

$$w_2 = \frac{d-c}{2} \quad (4)$$

The parameters a, b, c and d are the parameters used in the fuzzy membership function adjustment, as is presented in the Fig. 6:

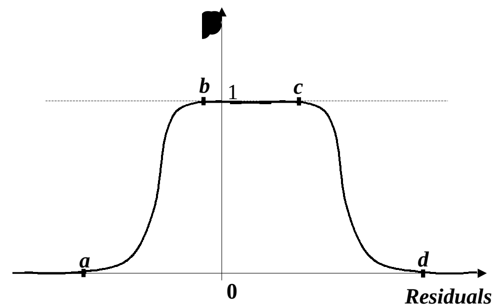

Fig. 6. Membership function parameters.

As can be seen in Fig. 7, the antecedent membership functions are generated by projecting the clusters onto the antecedent variables (residuals) and in this case, with the HSFNN, the consequents (faults) are generated by FNN learning.

The Gustafson-Kessel algorithm is a suitable method for identification because it is an algorithm based on an adaptive distance measure and because of this, it isn't so sensitive to scaling (normalization) of the data. It is also an algorithm that can detect clusters of different shapes and the fuzzy sets induced by the partition matrix are compact and easy to interpret (Babuška, 1998).

The hidden and output layers processing elements use the sigmoid function as their activation functions.

Both the lower level and the medium level networks are made up of three layers: a fuzzification

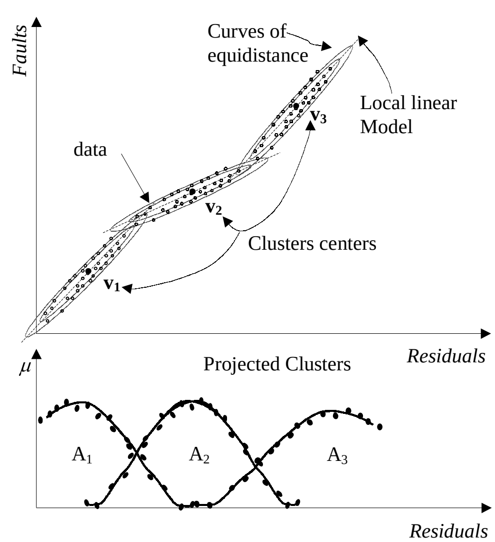

Fig. 7. Fuzzy clustering with hyperellipsoidal fuzzy clusters.

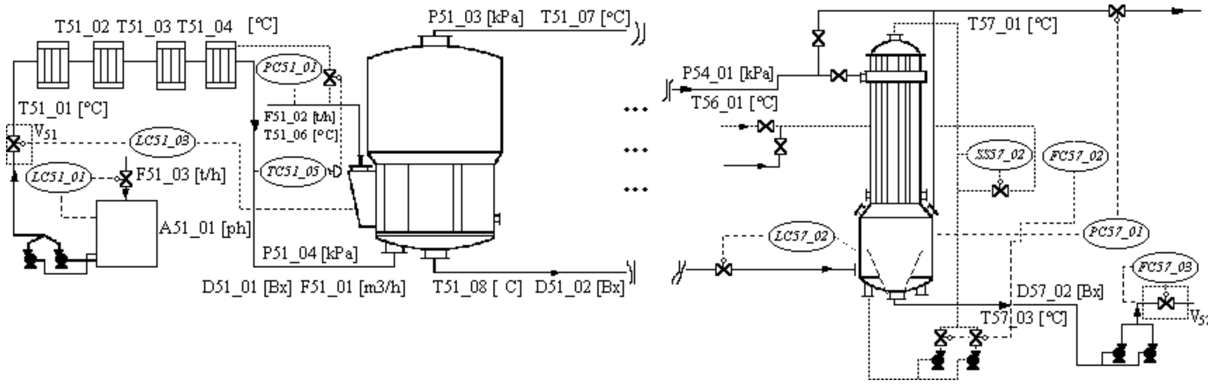

Fig. 8. Evaporation station from Lublin sugar factory.

layer, an hidden layer and an output layer. The FNNs have been trained using the resilient backpropagation learning algorithm (Riedmiller and Braun, 1993).

This algorithm is very fast in the training convergence by comparing with the standard backpropagation, and use the derivative sign and not the value, in the learning process. It is also an algorithm that require few resources of the computer, memory and storage of data.

The FNN_0 is trained with single abrupt fault symptoms and with stationary operational conditions symptoms. On the other hand, the FNN_i are trained using the data for one single abrupt fault (the fault associated with the corresponding FNN_i)) and for all possible double abrupt faults that the FNN_i net will be able to diagnose. This training data is obtained by adding the data for the corresponding single abrupt faults considered.

This structure will be used for fault isolation for all actuators in the industrial plant.

4. CASE STUDY

The evaporation station presented here is a part of Lublin Sugar Factory in Poland. In Fig. 8 the first and last sections of evaporation station are shown. The main technological task of an evaporation station is to thicken the beet juice being just after the filtering and cleaning processes. This station consists of seven evaporators grouped in five sections (sections I, IV and V consist of one evaporator each and sections II and III consists of two evaporators each). The first five evaporators work with natural juice circulation and the last two have another construction and work with juice circulation forced by pumps. The juice condensation process is performed using steam and vapour, which are the same quantities from physical point of view but they have different sources.

Table 2. Table of measurement tags.

Measurement tags	Description	Min-Max	Units
LC51_03.CV	control value	0-100	%
F51_01	juice flow	0-500	m^3/h
FC57_03.CV	control value	0-100	%
FC57_03.PV	juice flow	0-80	m^3/h

Steam is produced by water-steam-boiler and is mainly delivered to the I section of evaporation station. Whereas, vapour as recyclable medium, is produced in each evaporator and heat accumulator. The vapour is used as heating medium in many technological stations (e.g. steam heaters, consecutive evaporators, diffusers, syrup boilers etc.). Additional task of evaporation station is to produce the condensate, which is delivered next to the steam boiler.

Two valves connected with evaporation station have been chosen for research purposes. First one (valve V_{51}) situated on the inflow of thin juice into the evaporation station and the second one (valve V_{57}) situated on the outlet of thick juice from the V section of evaporation station (see Fig. 8). The Table 2 shows the variables that were used from the process.

5. FAULT DIAGNOSIS RESULTS

This section presents the results achieved with the FDI system proposed.

5.1 *Neuro-fuzzy models for sugar factory actuators*

Two valve models have been designed and implemented using described neuro-fuzzy techniques. The proposed models generate residuals which are used by the second stage of the FDI system (fault classification). The residuals are computed as a difference between system and model outputs Fig. 11.

The data sets from 14th and 23rd November 2000 have been used to design and train the neuro-fuzzy models. After this the models have been tested using data from 9th, 11th, 15th and 17th November 2000. All data, before use, have been filtered by a digital 2nd order Butterworth filter and so, cleaned data have been used to build models. The training set contains 10000 samples from 2 days and the performance index has been defined in the form of sum of the squared errors. The structures of designed models are presented in Table 3.

Table 3. The structures of the neuro-fuzzy models (C_V - control value, F - juice flow.)

Model	Inputs	Partitions	Output
valve V_{51}	$C_V(k)$	3	$F(k)$
valve V_{57}	$[C_V(k), C_V(k-15)]$	[3,3]	$F(k)$

After the learning process some simulations have been made using testing data. Some results are shown in the Fig. 9 and Fig. 10. The presented results contains 2000 samples from 11th November, thick lines represent the flow in the valves and the thin lines represents the results generated by the neuro-fuzzy models. The behaviour of the flow in the valve V_{57} is smoother than the behaviour of the flow in the valve V_{51} because it has been used different stop bands in the filtering. The error for the valve V_{51} is greater than for the valve V_{57} because the flow level for valve V_{51} is greater than the flow level for the valve V_{57}. The performance of the models can be improved if more measurements will be available and the data from the actuators for different working points will be collected. This will allow the construction of more accurate models for the actuators.

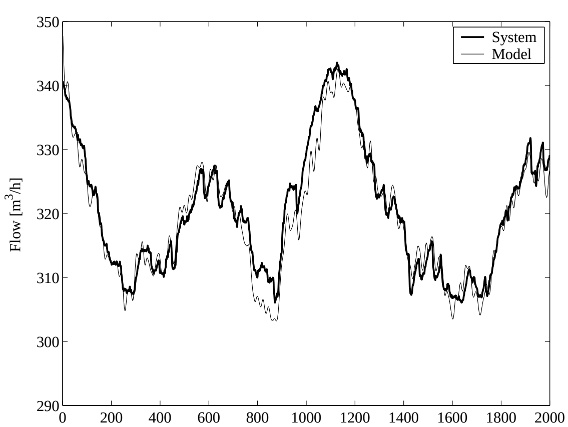

Fig. 9. Performance of TSK neuro-fuzzy model for valve V_{51} (SSE / $N.^o$ of samples = 6.774).

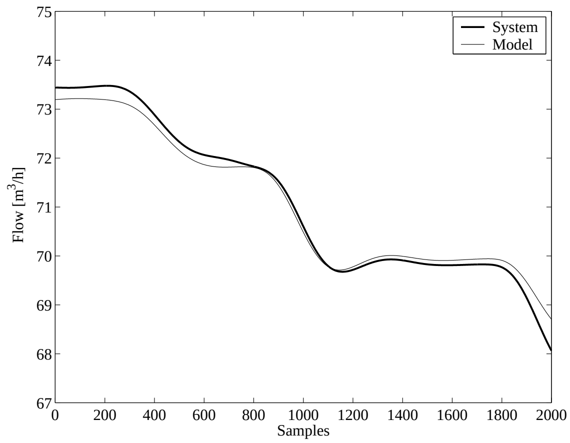

Fig. 10. Performance of TSK neuro-fuzzy model for valve V_{57} (SSE / $N.^o$ of samples = 0.0372).

5.2 *Simulation of faults and training data*

The fault isolation subsystem is based on an HSFNN with the characteristics previously presented (see Fig. 5). But, to construct this structure is necessary to define the fault set and the variables used in the input layer. Thus, two residuals have been used as input data to all the fuzzy neural networks. These residuals are the following: R_{51}, residuum from the valve V_{51}; R_{57}, residuum from the valve V_{57}.

The sugar factory data from the 2000 campaign have only control and flow values for both valves (see Table 2). Therefore, it isn't possible to use the set of faults defined for DAMADICS benchmark but only faults connected to the flow. For these reasons it has been considered 4 single abrupt faults:

- F_1, valve V_{51} blocked fully open
- F_2, valve V_{51} blocked fully closed
- F_3, valve V_{57} blocked fully open
- F_4, valve V_{57} blocked fully closed

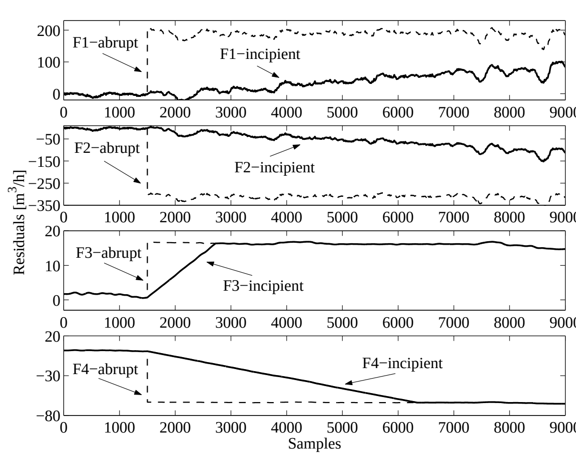

Fig. 11. Simulation of faults for both valves.

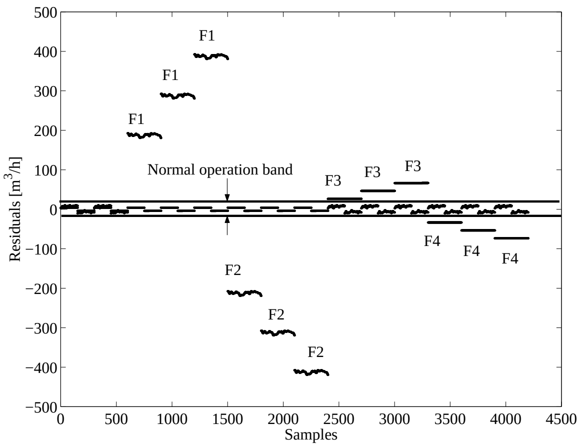

Fig. 12. Training data for the HSFNN.

The Fig. 11 show how this set of faults has been simulated for training and test data construction because the files from the sugar factory have only steady state data. Therefore, this set of faults (abrupt faults) and also the normal operation residuals have been used to construct the HSFNN training data, as can be seen in the Fig. 12. This figure show the training set for the lower level FNN_0 network. In this case, it has been also simulated data for 3 different installation set points, considering that the evaporation station do not work below approximately 100 m^3/h for valve V_{51} and below approximately 35 m^3/h for valve V_{57}. These values have been chosen after the inspection of the sugar factory data files and after the confirmation that, in all files, the working set points are around 300 m^3/h for valve V_{51} and around 65 m^3/h for valve V_{57}.

The Fig. 12 also show the normal operation band, which is defined from the normal operation residuals. This normal operation residuals band aim (jointly with the FNN_0 threshold) the elimination of false alarms and it is dependent from the model errors.

The training set for the FNN_0 network has 4200 vectors since it has been chosen, after some tests, that 300 vectors for each fault set point was sufficient to represent the residuum behaviour under faulty situation. The training set for the FNN_0 network include also the targets training set, which is represented by (5):

$$\mathbf{T_{FNN_0}} = \begin{pmatrix} 0\ldots\ 1\ldots\ 0\ldots\ 0\ldots\ 0\ldots \\ 0\ldots\ 0\ldots\ 1\ldots\ 0\ldots\ 0\ldots \\ 0\ldots\ 0\ldots\ 0\ldots\ 1\ldots\ 0\ldots \\ 0\ldots\ 0\ldots\ 0\ldots\ 0\ldots\ 1\ldots \end{pmatrix} \qquad (5)$$

and it has the same number of vectors (4200).

The training set for the FNN_i network has 2700 vectors, and has been achieved as explained before, considering that the fault symptoms are additive. For example, for the FNN_1 network, the targets training set is presented in (6):

$$\mathbf{T_{FNN_1}} = \begin{pmatrix} 1\ldots\ 1\ldots\ 1\ldots\ 0\ldots\ 0\ldots \\ 0\ldots\ 0\ldots\ 0\ldots\ 1\ldots\ 1\ldots \\ 0\ldots\ 1\ldots\ 0\ldots\ 1\ldots\ 0\ldots \\ 0\ldots\ 0\ldots\ 1\ldots\ 0\ldots\ 1\ldots \end{pmatrix} \qquad (6)$$

and it has the same number of vectors (2700).

5.3 *HSFNN for sugar factory actuators*

The HSFNN implemented for fault isolation on the sugar factory actuators is one structure where all the FNNs are equal, with a fuzzification layer consisting of 6 processing elements arranged in 2 groups, corresponding to the 2 residuals, with each group containing 3 neurons corresponding to the respective fuzzy sets, as can be seen in the Fig. 13. The number of neurons in the hidden layer is determined by the complexities of the relationships between the faults and the fault symptoms. During the current study, it was found that 5 hidden processing elements could give good performance. The output layer of each fuzzy neural network is up of 4 neurons, each one corresponding to a fault (see Fig. 5). In order to achieve a diagnosis of a fault or faults in the process, an analysis of the output values from the fuzzy neural networks FNN_0 is necessary. If the number of nonzero outputs (output ≥ 0.5) in FNN_0 is equal to 0, then it is assumed that no fault occurred in the process. Otherwise, the result of the HSFNN is considered to be the result of a fuzzy OR operation (upper level) on several FNN_i outputs in medium level, as previously described.

The membership function achieved and used in the fuzzification layer are:

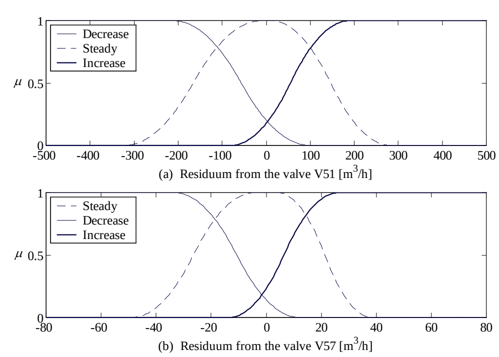

Fig. 13. Membership function used in the fuzzy layer.

This membership functions can be adjusted to improve the HSFNN performance under incipient faults scenarios (Mendes, 2001).

5.4 *Results of fault diagnosis*

The results achieved with the fault diagnosis system are presented in tables 4-9. These tables have a first column with the faults simulated. A second column, where the classification values (F_3=0.63, for example) are presented, corresponding to the results achieved with the lower level (FNN_0 network). Furthermore, they also have four columns associated with the four FNNs at the medium level and finally, a column with the final diagnosis results of the HSFNN (achieved after the fuzzy OR operation). The results achieved so far have shown that the system proposed in this paper is a potential tool for fault diagnosis system of single/multiple abrupt (Tables 4, 5) and incipient faults (Table 6, 8).

Tables 4 and 5 show the diagnosis results for single and double abrupt faults when the process is under the fault scenarios shown in the Fig. 11.

Table 4. Results with single abrupt faults.

Faults	Lower level	Medium level				Upper level
	FNN_0	FNN_1	FNN_2	FNN_3	FNN_4	OR
F_1	F_1=1	F_1	-	-	-	F_1=1
F_2	F_2=1	-	F_2	-	-	F_2=1
F_3	F_3=0.63	-	-	F_3	-	F_3=1
F_4	F_4=0.99	-	-	-	F_4	F_4=1

In the case of single abrupt faults (Table 4), it has been observed 100 % of correct diagnosis.

Table 5. Results with double abrupt faults.

Faults	Lower level	Medium level				Upper level
	FNN_0	FNN_1	FNN_2	FNN_3	FNN_4	OR
F_1F_3	F_1=0.98 F_3=0	F_1	-	-	-	F_1=1 F_3=0.43
F_1F_4	F_1=0.82 F_4=0	F_1F_4	-	-	-	F_1=1 F_4=0.98
F_2F_3	F_2=0.99 F_3=0	-	F_2	-	-	F_2=1 F_3=0.40
F_2F_4	F_2=0.99 F_4=0	-	F_2F_4	-	-	F_2=1 F_4=0.99

In the case of double abrupt faults (Table 5), it has been observed 50 % of correct diagnosis. But, it can be seen that the responsible for these results is the isolation of fault F_3 (with values near to good isolation F_3=0.43 and F_3=0.40). This happen due to the fact of this fault, when are used the real data from the sugar factory for FDI tests, generate a very small residuum (see Fig. 11). This residuum is near to the normal operation band and because of that the network learning, for this fault, is insufficient. This fault symptom is also hidden by the other fault symptoms (F_1 and F_2), see Fig. 11.

To improve the fault isolation subsystem, under double abrupt faults, the FNN_0 threshold can be change to 0.4 or the models in the fault detection subsystem must be improved, to reduce the model errors and with this improvement the normal operation band will be smaller. The network learning can be also improved but is possible that overlearning occurs.

Tables 6 and 8 show the diagnosis results for single and double incipient faults when the process is under the incipient fault scenarios shown in the Fig. 11. The slope used in the incipient fault simulation was $slope = \frac{Residuum}{Time} = 0.013\ [\frac{m^3}{h \cdot samples}]$.

Table 6. Results with single incipient faults.

Faults	Lower level	Medium level				Upper level
	FNN_0	FNN_1	FNN_2	FNN_3	FNN_4	OR
F_1	F_1=0.99	F_1	-	-	-	F_1=1
F_2	F_2=0.99	-	F_2	-	-	F_2=1
F_3	F_3=0.99	-	-	F_3	-	F_3=1
F_4	F_4=0.99	-	-	-	F_4	F_4=1

In the case of single incipient faults (Table 6), it has been observed 100 % of correct diagnosis, with this slope and other slopes tested.

Table 7. Slopes and diagnosis time with single incipient faults.

Incipient fault	Slope $[\frac{m^3}{h \cdot samples}]$	Fault start [samples]	Diagnosis time [samples]
F_1	0.013	1500	5620
F_2	0.013	1500	8490
F_3	0.013	1500	2597
F_4	0.013	1500	2480

The Table 7 show the slopes, fault start and diagnosis time for all single incipient faults tested.

Table 8. Results with double incipient faults.

Faults	Lower level	Medium level				Upper level
	FNN_0	FNN_1	FNN_2	FNN_3	FNN_4	OR
F_1F_3	F_1=0.98 F_3=1	F_1F_3	-	F_1F_3	-	F_1=1 F_3=1
F_1F_4	F_1=0.85 F_4=0.98	F_1F_4	-	-	F_1F_4	F_1=1 F_4=1
F_2F_3	F_2=1 F_3=0.99	-	F_2	F_2F_3	-	F_2=1 F_3=1
F_2F_4	F_2=0.99 F_4=0.98	-	F_2F_4	-	F_4	F_2=1 F_4=1

Table 9. Slopes and diagnosis time with double incipient faults.

Incipient fault	Slope $[\frac{m^3}{h \cdot samples}]$	Fault start [samples]	Diagnosis time [samples]
F_1F_3	F_1=0.013	F_1=1500	F_1=7048
	F_3=0.013	F_3=1500	F_3=2564
F_1F_4	F_1=0.013	F_1=1500	F_1=7055
	F_4=0.013	F_4=1500	F_4=7440
F_2F_3	F_2=0.013	F_2=1500	F_2=8560
	F_3=0.013	F_3=1500	F_3=2575
F_2F_4	F_2=0.013	F_2=1500	F_2=8560
	F_4=0.013	F_4=1500	F_4=2535

In the case of double incipient faults (Table 8), it has been observed 100 % of correct diagnosis, with this slope. As can be seen from the Tables 6 and 8, the fault diagnosis system can diagnose incipient faults with a smaller slope because the diagnosis values are around 1 (F_2=0.99 for example) and the threshold value in the FNN_0 network is 0.5. With different slopes for the two valves it is possible to see some false alarms.

6. CONCLUSIONS

This paper deals with the soft computing methods in fault diagnosis. The proposed approach integrates fuzzy system and neural network techniques in order to combine the advantages of both methods. Neuro-fuzzy networks and an HSFNN have been used to design and implement the two-stage FDI system. The neuro-fuzzy models of actuators carry out the detection task and the HSFNN isolate the faults. The main advantage of the proposed approach is that no mathematical model of the process is required and the construction of the FDI system can be realized using quantitative and qualitative knowledge. The neuro-fuzzy networks appear as an effective tool for fault detection. This approach allows achieving a transparent structure of the models, which can be easily tuned using gradient descent algorithm. Another advantage of the model based fault diagnosis approach is that faulty data are not needed during the learning and the model is able to detect all faults in the process.

The fault isolation subsystem combines the advantages of both, fuzzy reasoning and neural networks. The developed system was successfully applied to fault diagnosis in sugar factory actuators. It has been demonstrate that the current fault isolation approach is able to diagnose multiple simultaneous abrupt and incipient faults from only single fault symptoms. During the current study, it has been observed that the neural network's generalization ability has a great importance in the diagnosis of incipient faults since the training patterns only include single abrupt fault symptoms.

As it has been demonstrated this two-stage FDI system is a suitable approach for this kind of fault diagnosis systems.

7. REFERENCES

Babuška, R. (1998). *Fuzzy Modeling for Control.* Kluwer Academic Publishers. London.

Calado, J.M.F. and J.M.G. Sá da Costa (1999). An expert system coupled with a hierarchical structure of fuzzy neural networks for fault diagnosis. *Int. J. Appl. Math. and Comp. Sci.* **9**(3), 667–687.

Calado, J.M.F., J. Korbicz, K. Patan, R.J. Patton and J.M.G. Sá da Costa (2001). Soft computing approaches to fault diagnosis for dynamic systems. *European Journal of Control* **7**(2-3), 169–208.

Gustfason, D.E. and W.C. Kessel (1979). Fuzzy clustering with a fuzzy covariance matrix. *In Proceeding IEEE CDC* pp. 761–766.

Jin, Y. (2000). Fuzzy modeling of high-dimensional systems: complexity reduction and interpretability improvement. *IEEE Trans. on Fuzzy Systems* **8**(2), 212–221.

Klir, J.G. and A.T. Folger (1988). *Fuzzy Sets, Uncertainty and Information.* Prentice-Hall. New York.

Korbicz, J.K., K. Patan and A. Obuchowicz (2001). Neural network fault detection system for dynamic processes. *Biulletin of the Polish Academy of Sciences. Technical Sciences* **49**(2), 301–321.

Koscielny, M.J., A. Ostasz and P. Wasiewicz (2000). Fault detection based on fuzzy neural networks - application to sugar factory evaporator. *Proc. 4th IFAC Symp. Fault Detection, Supervision and Safety for Technical Processes, SAFEPROCESS'2000* **1**, 337–342.

Kowal, M. and J. Korbicz (2000). Neuro-fuzzy detector for industrial process. In: *Advances in Soft Computing. Fuzzy Control* (R. Hampel, M. Wagenknecht and N. Chaker, Eds.). pp. 294–301. Physica-Verlag. New York.

Mendes, M.J.G.C. (2001). Fault diagnosis based on hierarhical fuzzy techniques. Master's thesis. Instituto superior tecnico, Technical University of Lisbon.

Mendes, M.J.G.C., J.M.F.Calado, J.M.C.Sousa and J.M.G.Sá da Costa (n.d.). Industrial actuator diagnosis using hierarhical fuzzy neural networks. European Control Conference ECC 2001 (accepted).

Nauck, D., F. Klawonn and R. Kruse (1997). *Foundations of Neuro-Fuzzy Systems.* John Wiley & Sons, Inc.. New York.

Nelles, O., A. Fink, R. Babuška and M. Setnes (2000). Comparison of two construction algorithms for Takagi-Sugeno fuzzy models. *Int. J. Appl. Math. and Comp. Sci.* **10**(4), 835–855.

Patton, R.J. and J. Korbicz(Eds.) (1999). Advances in computational intelligence for fault diagnosis systems. *Special issue of Int. J. Appl. Math. and Comp. Sci.* **9**(3), 471–711.

Patton, R.J., F.J. Uppal and C.J. Lopez-Toribio (2000a). Soft computing approaches to fault diagnosis for dynamic systems: A survey. *Proc. 4th IFAC Symp. Fault Detection, Supervision and Safety for Technical Processes SAFEPROCESS'2000* **1**, 298–311.

Patton, R.J., P.M. Frank and R.N.Clark (2000). *Issues of Fault Diagnosis for Dynamic Systems.* Springer. London.

Riedmiller, M. and H. Braun (1993). A direct adaptive method for faster backpropagation learning: The RPROP algorithm. *Proc. of the IEEE Intl. Conf. on Neural Networks* pp. 586–591.

Runkler, T.A. and J.C. Bezdek (1999). Alternating cluster estimation: A new tool for clustering and function approximation. *IEEE Trans. on Fuzzy Systems* **7**(4), 377–393.

Rutkowska, D. and L.A. Zadeh(Eds.) (2000). Neuro-fuzzy and soft computing. *Special issue of Int. J. Appl. Math. and Comp. Sci.* **10**(4), 647–877.

Takagi, H. (2000). Fusion technology of neural networks and fuzzy systems: a chronicled progression from the laboratory to our daily lives. *Int. J. Appl. Math. and Comp. Sci.* **10**(4), 647–673.

Watanabe, K., S. Hirota, L. Hou and D.M. Himmelblau (1994). Diagnosis of multiple simultaneous fault via hierarchical artificial neural networks. *AIChE Journal* **40**(5), 839–848.

Yager, R.R. and D.P. Filev (1994). *Essentials of Fuzzy Modelling and Control.* John Wiley & Sons, Inc.. New York.