NEURAL NETWORKS WITH DECISION TREES FOR DIAGNOSIS ISSUES

Yahia Kourd[1], Dimitri Lefebvre[2] and Noureddine Guersi[3]

[1]Med Khider Biskra University, Department of Electrical Engineering, Algeria
ykourd@ yahoo.fr
[2]GREAH – University of Havre 25 rue Philippe Lebon – 76058 Le Havre – France dimitri.lefebvre@univ-lehavre.fr
[3] Badji Mokhtar University, Department of Electronic, Annaba, Algeria
guersi54@yahoo.fr

ABSTRACT

This paper presents a new idea for fault detection and isolation (FDI) technique which is applied to industrial system. This technique is based on Neural Networks fault-free and Faulty behaviours Models (NNFMs). NNFMs are used for residual generation, while decision tree architecture is used for residual evaluation. The decision tree is realized with data collected from the NNFM's outputs and is used to isolate detectable faults depending on computed threshold. Each part of the tree corresponds to specific residual. With the decision tree, it becomes possible to take the appropriate decision regarding the actual process behaviour by evaluating few numbers of residuals. In comparison to usual systematic evaluation of all residuals, the proposed technique requires less computational effort and can be used for on line diagnosis. An application example is presented to illustrate and confirm the effectiveness and the accuracy of the proposed approach.

KEYWORDS

Neural Network, Fault Detection and Isolation, Faulty Model, & Decision Tree

1. INTRODUCTION

In a process, early diagnosis of faults, that might occur, allows performing important prevention actions. Therefore, fault detection is a crucial task in the automatic control of large system as manufacturing systems (Diag, 2009). Moreover, it allows avoiding heavy economic losses due to production stop, replacement of spares parts, etc. The need of performing and reliable developed methods for the systems diagnosis becomes increasingly pressing. These methods should respects the following points: (1) Standards and quality improvement; (2) Diagnostic failure to improve the relationship and (3) The definition of new services as new technologies and economic interests. Most of the fault diagnosis methods found in the literature are based on linear methodology or exact models. Models of industrial processes are often very complex. It is difficult to accurately predict their behavior, especially with corrupted measures, and unreliable sensors. Therefore, a number of researchers have perceived artificial neural networks as an alternative way to represent knowledge about faults (Sorsa et al. 1992, Himmelblau 1992, Patton et al. 1994, Frank 1997, Patton et al. 1999, Calado et al. 2001, Korbicz et al. 2004).

This paper presents a FDI method that generates a large number of residuals depending on the set of candidate faults. The residuals are analyzed and evaluated according to their mean values. A decision tree is introduced to manage the residuals evaluation and to decide on line which residual

Sundarapandian et al. (Eds) : CSE, CICS, DBDM, AIFL, SCOM - 2013
pp. 29–39, 2013. © CS & IT-CSCP 2013 DOI : 10.5121/csit.2013.3304

must be evaluated or not. This technique is validated with the DAMADICS benchmark process, a European project under which several FDI methods have been developed and compared. This benchmark is based on industry requirements described in (Bartys et al. 2006). It is used with different approaches in the purpose of providing a training facility for both industry and academia. It gains a better understanding for the way in which the various FDI methods can perform in a realistic control engineering application setting. In the literature, different analytical FDI approaches have been developed. In (Puig et al. 2006), passive robustness fault detection method using intervals observers is presented. In (Previdi et al. 2006), authors introduce signal model based fault detection using squared coherency functions. An actuator fault distinguishability study is presented in (Koscielny et al. 2006). Several soft computing techniques used in FDI methods have been developed under the scope of DAMADICS. A data-driven method in FDI is presented in (Bocaniala et al. 2006), where a novel classifier based on particle swarm optimization was developed. Group method of data handling (GMDH) neural networks have been used in (Witczak et al. 2006) for robust fault detection. A computer-assisted FDI scheme based on a fuzzy qualitative simulation, where the fault isolation is performed by a hierarchical structure of the neuro-fuzzy networks is presented in (Calado et al. 2006). A neuro-fuzzy modeling for FDI, involving a hybrid combination of neuro-fuzzy identification and unknown input observers in the neuro-fuzzy and decoupling fault diagnosis scheme, has been proposed in (Uppal et al. 2006).

The paper is organized as follows: In section 2 the proposed technique for FDI issues is presented. This method uses models of faulty and fault-free behaviors with Neural Networks. The design of decision tree is proposed in section 3 to assist the diagnosis on line with early decision. Section 4 presents the DAMADICS actuator benchmark and application of our approach. Finally, in the last section a conclusion about the effectiveness of this approach and future research directions are presented.

2. NEURAL NETWORKS FAULTY AND FAULT-FREE MODELS

2.1. Fault-free Model

Physical processes are often very complex dynamic systems, having strong non linearities. As a consequence, knowledge based models are not easy to obtain. Simplifications are essential to formulate an exploitable model, but they may degrade the accuracy of the mathematical model. Other problems remain with some model parameters that are not easy to measure or estimate and that could be variable in time. Another problem lies in the systematic processing of data collected by sensors.

At this stage, unknown nonlinear systems are considered with input vector $U(t) = (u_i(t)),\ i = 1,...,q$ and output vector $Y(t) = (y_k(t)),\ k = 1,...,n$. The state variables are not measurable. Neural Networks (NN) are introduced to generate accurate models of the system in normal operating conditions (Kourd et al., 2008, 2010, 2011). The comparison between the output of the system and the output $Y_0'(t) = (y'_{k0}(t)),\ k = 1,...,n$, of the NN model gives the error vector $E(t) = (e_k(t)),\ k = 1,...,n$, with:

$$e_k(t) = y_k(t) - y'_{k0}(t) \qquad (1)$$

The Neural Network model is trained with data collected from the fault-free system utilizing Levenberg-Marquardt algorithm with early stopping that uses three data sets (training, testing and validation) to avoid overfitting. Moreover, this algorithm is known in its fast convergence. The obtained model is then tested and validated again with other sets of data. In order to get the best model, several configurations are tested according to a trial error processing that uses pruning methods to eliminate the useless nodes.

2.2. Faulty Models

When multiple faults are considered, the isolation of the detected faults is no longer trivial and early diagnosis becomes a difficult task. One can multiply the measurements and use some analysis tools (residuals analysis) in order to isolate them. In particular, a history of collected data can be used to improve the knowledge about the faulty behaviors. This knowledge is then used to design models of faulty behaviors and additional residuals. Such models will be used to provide estimations for each fault candidate. The decision results are then provided from the comparison between estimations with the measurements collected during system operations.

The design of faulty models is similar to the method described in previous section (2.A). The learning of faulty behaviors is obtained according to the Levenberg-Marquardt algorithm with early stopping. Each model is built for a specific fault candidate f_j that is considered as an additional input. The vectors $Y'_j(t) = (y'_{kj}(t))$, $k = 1,...,n$, $j = 1,\ldots p$ stand for the outputs of the Neural Networks models designed for the faults f_j, $j = 1,\ldots p$.

3. DECISION TREES FOR RESIDUAL EVALUATION

3.1. Fault detection technique

During monitoring, the direct comparison of the system's outputs $Y(t)$ and the outputs $Y'_0(t)$ of fault-free model leads to residuals $R_0(t) = (r_{k0}(t))$ and $k = 1,\ldots,n$ with:

$$r_{k0}(t)= y_k(t)-y'_k(t),\ k = 1,\ldots,n. \qquad (2)$$

The residual $R_0(t)$ provides information about faults for further processing. Fault detection is based on the evaluation of residuals magnitude. It is assumed that each residual $r_{k0}(t)$, $k = 1,\ldots,n$ should normally be close to zero in the fault-free case, and it should be far from zero in the case of a fault. Thus, faults are detected by setting threshold S_{k0} on the residual signals (Fig. 1 up, single residual and a single fault are considered for simplicity). The analysis of residuals $r_{k0}(t)$ also provides an estimate τ_k of the time of occurrence t_f used for diagnosis issue. When several residuals are used, the estimate τ of the time of occurrence of faults is given by:

$$\tau = \min\{\tau_k,\ k = 1,\ldots,n\} \qquad (3)$$

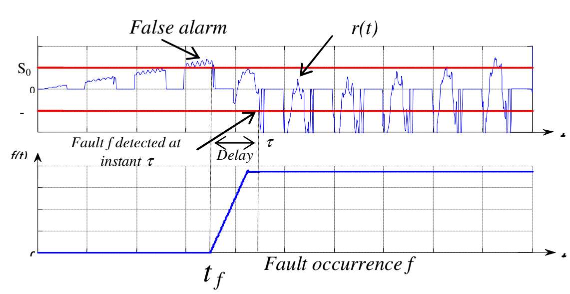

Figure 1. Fault detection by thresholding technique

The faults are detected when the magnitude of one residual $|r_{k0}(t)|$ augments the threshold S_{k0}:

$$\begin{cases} |r_{k0}(t)| \le S_{k0}: \text{No fault is detected at time } t \\ |r_{k0}(t)| > S_{k0}: \text{A fault is detected at time } t \end{cases} \qquad (4)$$

The main difficulty with this evaluation is that the measurement of the system outputs $y_k(t)$ is usually corrupted by disturbances (for example, measurement noise). In practice, due to the modeling uncertainties and disturbances, it is necessary to assign large thresholds S_{k0} in order to avoid false alarms. Such thresholds usually imply a reduction of the fault detection sensitivity and can lead to non detections. In order to avoid such problems, one can run also the models of faulty behaviors from t=0 and use the method described below. The idea is to evaluate the probability of the fault candidates at each instant. A fault is detected when the probability of one neural network faulty model *NNFM(j)*, $j = 1,..., p$ becomes larger than the probability of the fault-free model *NNFM(0)* (Kourd et al., 2010).

3.2. Fault diagnosis based on three valued residuals

The proposed approach is based on the analysis of the outputs obtained after applying the input $U(t)$ on the real system. Obtained outputs are also analyzed in parallel with fault-free and faulty behaviors models constructed by Neural Networks technique (Kourd et al., 2012). Detection and diagnosis are achieved from residuals generation $R_j(t)$, $j = 0,...,p$ according to a decision block.
The diagnosis results either from the usual thresholding technique or from the on-line determination of fault probabilities and confidence factors (Kourd et al., 2011). In the second method, the faulty models run simultaneously from time $t = \tau$ where τ is the time when the fault is detected. Each model will behave according to a single fault candidate and the resulting behaviors will be compared with the collected data to provide rapid diagnosis. In case of numerous fault candidates f_j, $j = 1,\ldots,p$, the output $Y'_j(t) = (y'_k(t, f_j, \tau))$ of the model *NNFM(j)* is compared with the measured vector $Y(t)$ to compute additive residual $R_j(t) = (r_{kj}(t, \tau))$, $k = 1,\ldots,n$. The most probable fault candidate is determined according to the comparison of all residuals $r_{kj}(t, \tau)$, $k = 1,\ldots,n$, $j = 1,\ldots,p$ resulting from the n outputs and p models of faults:

$$r_{kj}(t,\tau) = y_k(t) - y'_{kj}(t, \tau) \qquad (5)$$

According to the residual analysis of r_{kj} obtained by equation (5), and adopting positive and negative thresholds, threes values of these residuals are obtained (positive, negative or null). The comparison of the current residual with the signatures matrix leads to diagnosis (Chen et al. 1999).

3.3. Decision trees for residual evaluation

The aim of this section is to propose a hierarchical structure to simplify the diagnosis of automation systems when numerous residuals are computed. The idea is to organize the residuals in a decision tree. This tree is used to compute only the selection of residuals that are the most significant for the current signal. The tree starts with the evaluation of the residual that corresponds to the fault free model (step 1). The value of the residual is used to classify the fault candidates in several subgroups (figure 2, G_1 to G_m). Each subgroup limits the number of fault candidates. The algorithm continues by evaluating second residual that depends on the subgroup resulting from the first step (step 2). Another time the fault candidates are separated into subgroups (figure 2, G_{11} to G_{1t} for example).

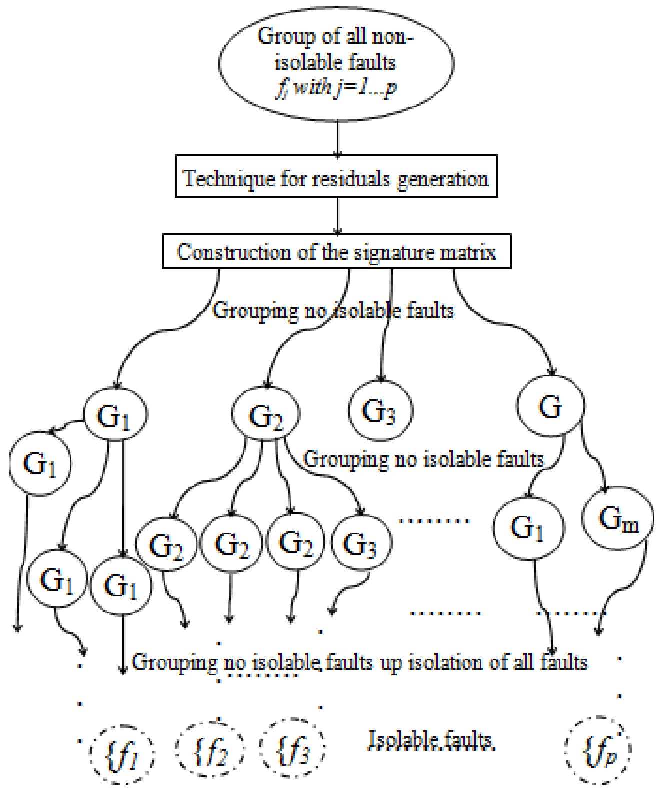

Figure 2. FDI decision tree proposed

According to that evaluation, the algorithm continues until a subset of faults with a single candidate is isolated (presented by the leaves of tree in figure 2). Finally the faults are isolated by the computation of selected residuals. Thus the computational effort is reduced in comparison with the systematic evaluation of all residuals. In practice, the use of hierarchical architecture is feasible online or offline depending on the complexity of the system.

4. APPLICATION IN INDUSTRIAL SYSTEM

To evaluate FDI methods proposed, The DAMADICS benchmark is an engineering research case-study that can be used. This is electro-pneumatic valve actuator in the Lublin sugar factory in Poland (Bartys et al., 2006). Its main characteristics are:
(a) The DAMADICS benchmark is based on the physical phenomena that give origin to faults in the system.
(b) The DAMADICS benchmark clearly defines the process and data sets; the fault scenarios are standardized. This is done in view of industrial applicability of the tested FDI solutions, to cut off methods that have no practical feasibility.

4.1. Actuator description

The actuator consists of a control valve, a pneumatic servomotor and a positioner (Figure 3). In the actuator, faults can appear in: control valve, servo-motor, electro-pneumatic transducer, piston rod travel transducer, pressure transmitter or microprocessor control unit. A total number of 19 different faults is considered (p = 19, Table 1). The faults are emulated under carefully monitored conditions, keeping the process operation within acceptable limits. Five available measurements and one control value signal have been considered for benchmarking purposes: process control external signal (*CV*), liquid pressures on the valve inlet (P_1) and outlet (P_2), liquid flow rate (*F*), liquid temperature (T_1) and servomotor rod displacement (*X*).

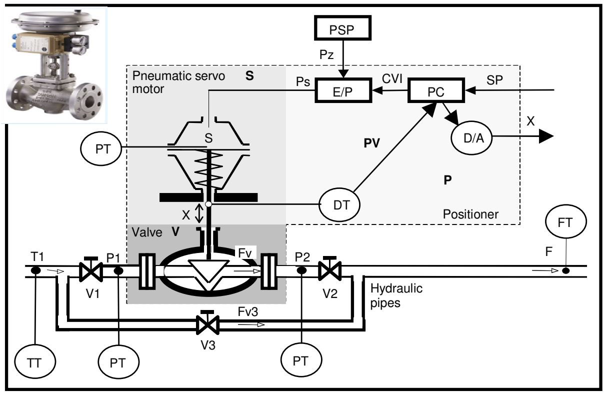

Figure 3. Electro-pneumatic valve Schema.

Within the DAMADICS project the actuator simulator was developed under MATLAB Simulink. This tool makes it possible to generate data for the normal and 19 faulty operating modes. The considered faults can be either of abrupt or incipient origins. Abrupt faults are of small (S), medium (M) or big (B) magnitude.

4.2. Neural Network Fault-free Model for Actuator

Two Multi Layer Perceptron (MLP) neural networks are designed to model the outputs $y_1(t) = X(t)$ and $y_2(t) = F(t)$ of the DAMADICS system in case of fault-free behaviors. $y'_{10}(t) = X'(t)$ and $y'_{20}(t) = F'(t)$ are the estimated values of $X(t)$ and $F(t)$ processed by *NNs*:

$$(X', F') = NNFM(0)\ \{CV, P1, P2, T1, X, F\} \qquad (6)$$

Where *NNFM(0)* stands for the double MLP structures. To select the structure of *NNFM(0)*, several tests are carried out to obtain the best architectures (with minimal number of hidden layers and number of neurons by layer) for modeling the operation of the actuator. The training, testing and validating data is simulated using Matlab Simulink actuator model. Validation is done by the measured data provided by 'Lublin Sugar Factory'.

4.3. Neural Network faulty Model for Actuator

The preceding method is applied to build NNs models corresponding to the 19 fault candidates that are considered in DAMADICS benchmark. For that purpose, it is necessary to create a data base that contains samples for all faults (Kourd et al., 2011) exposed to the DAMADICS system. The method is illustrated in Figure 4 for the fault f_j with $j=1$ *to 19 faults*. The network *NNFM(j)* learns the mapping from q=6 inputs to n=2 outputs when fault f_j is assumed to affect the system from time t = 0. Equation (7) holds:

$$(X'_j,\ F'_j) = NNFM(j)\ (CV,\ P_1,\ P_2,\ T_1,\ X_j,\ F_j) \qquad (7)$$

To select the structure of *NNFM(j)*, numerous tests are carried out to obtain the best architectures. The training and test data are generated using Matlab-Simulink DABLIB models (DAMADICS 2002). The best structure is a Neural Network with 6 nodes in the first hidden layer, 3 nodes in the second hidden layer and two output neurons. Validation is done with the measured data provided by the Lublin Sugar Factory in 2001 (DAMADICS 2002).

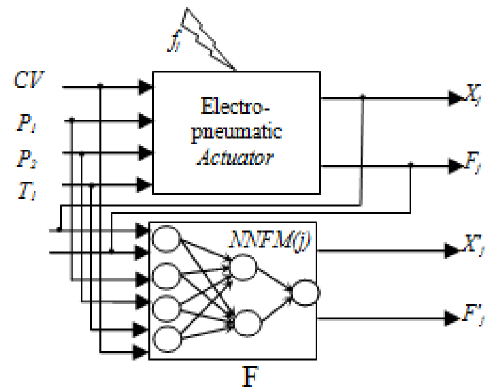

Figure 4. Schema of Neural Networks faulty model NNFM(j).

4.4. Residual by Three valued for Actuator system:

The residuals analysis is an essential step in FDI systems. The first parameter to be defined in the design of a fault detection system is the threshold value, which value allows the system to meet the required false alarm probability. The problem of the threshold selection is closely linked to the behavior of residuals and also to constraints that may be imposed such as security margins tolerance (Lefebvre et al., 2010). The residual vector $R_0(t) = (r_{k0}(t))$, k = 1, 2 for DAMADICS actuator is first considered for fault detection:

$$\begin{cases} r_{10}(t) = X(t) - X'(t) \\ r_{20}(t) = F(t) - F'(t) \end{cases} \quad (8)$$

Where X' and F' are the outputs of the NN model of fault-free behaviors. The detection is obtained by comparing residuals with appropriate thresholds. Three-valued signal are obtained (positive, negative and zero). The thresholds are figured out according to the standard deviation of the residual for fault-free case (Kourd et al., 2011). Let us notice that the choice of constant or adaptive thresholds strongly influences the performance of the FDI system. The thresholds must be thoroughly selected. For the continuation of our work, the thresholds $S_{10}=5*\sigma_1 = 0.0027$ and $S_{20} =5*\sigma_2 =0.004$ are selected where σ_1 and σ_2 are the standard deviations obtained from the learning process. Table 1 sums up the detection performances for the 19 types of faults according to the sign of the residual vector R_0.

Table 1 Signature matrix for DAMADICS Actuator with residual R0

G_0	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}	f_{17}	f_{18}	f_{19}
r_{10}	0	+1	-1	0	-1	0	0	+1	0	0	+1	-1	-1	-1	0	-1	+1	+1	0	0
r_{20}	0	-1	+1	-1	-1	+1	-1	-1	0	-1	-1	+1	+1	-1	0	+1	-1	-1	-1	+1

The signification of (+1) is the case where the residual is above the positive threshold, (-1) is the case where the residual is below the negative threshold and (0) when the residual is between both thresholds.

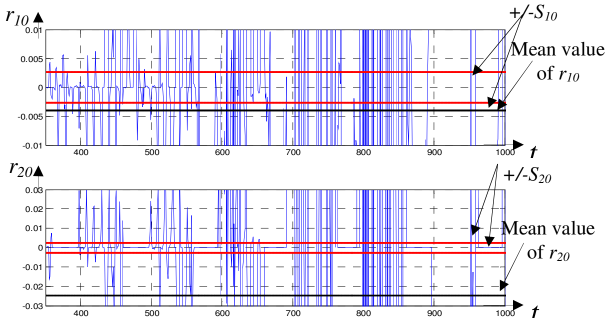

Figure 5. Residuals r10 and r20 in the case where f_1 was simulated during time interval [350s 1000s].

The evaluation of residual vector R_0 leads to a first stage in detection and isolation. From Table 1, six groups of faults with similar symptoms can be separated:

- Group 1 : $G_1 = \{f_3\ f_6\ f_9\ f_{18}\}$ with signature $\begin{pmatrix} 0 \\ -1 \end{pmatrix}$
- Group 2 : $G_2 = \{f_1\ f_7\ f_{10}\ f_{16}\ f_{17}\}$ with signature $\begin{pmatrix} +1 \\ -1 \end{pmatrix}$
- Group 3 : $G_3 = \{f_5\ f_{19}\}$ with signature $\begin{pmatrix} 0 \\ +1 \end{pmatrix}$
- Group 4 : $G_4 = \{f_2\ f_{11}\ f_{12}\ f_{15}\}$ with signature $\begin{pmatrix} -1 \\ +1 \end{pmatrix}$
- Group 5 : $G_5 = \{f_4\ f_{13}\}$ with signature $\begin{pmatrix} -1 \\ -1 \end{pmatrix}$
- Group 6 : $G_6 = \{f_0\ f_8\ f_{14}\}$ with signature $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$

The faults in groups G_1 to G_5 are detected but not isolated because the signatures over r_{10} and r_{20} are similar within the group. One can also notice that the faults in group G_6 have the same signature as the fault-free behaviors. Thus faults in group G_6 cannot be directly detected with residuals r_{10} and r_{20}.

For this reason other residuals generated by faulty behavior neural models *NNFM(j)* can be added for each group. This allows the building of signatures matrix. An evaluation step that uses same thresholding technique is followed up. Tables 2 to 7 sum up the detection performances for the 19 types of faults according to the sign of the residual vector $R_j(t)$ generated by *NNFM(j)* with $j= 1,...,19$.

Table 2

Signature matrix for group G_1

G_1	f_3	f_6	f_9	f_{18}
r_{13}	0	0	0	0
r_{23}	0	-1	-1	-1
r_{16}	0	0	0	0
r_{26}	-1	0	-1	-1
r_{19}	+1	+1	0	+1
r_{29}	-1	+1	0	+1
r_{118}	0	0	0	0
r_{218}	-1	-1	-1	0

Table 3

Signature matrix for group G_2

G_2	f_1	f_7	f_{10}	f_{16}	f_{17}
r_{11}	0	+1	+1	+1	+1
r_{21}	0	0	-1	-1	-1
r_{17}	-1	0	+1	+1	+1
r_{27}	0	0	-1	-1	-1
r_{110}	-1	-1	0	+1	+1
r_{210}	+1	+1	0	-1	-1
r_{116}	-1	-1	-1	0	+1
r_{116}	+1	+1	+1	0	-1
r_{117}	-1	-1	-1	-1	0
r_{117}	+1	+1	+1	+1	0

Table 4

Signature matrix for group G_3

G_3	f_5	f_{19}
r_{15}	0	0
r_{25}	0	-1
r_{119}	0	0
r_{219}	1	0

Table 5

Signature matrix for group G4

G_4	f_2	f_{11}	f_{12}	f_{15}
r_{12}	0	+1	+1	+1
r_{22}	0	+1	+1	-1
r_{111}	-1	0	+1	+1
r_{211}	-1	0	-1	-1
r_{112}	-1	-1	0	+1
r_{212}	-1	+1	0	-1
r_{1115}	-1	-1	-1	0
r_{215}	+1	+1	+1	0

Table 6

Signature matrix for group G_5

G_5	f_4	f_{13}
r_{14}	0	0
r_{24}	0	+1
r_{113}	-1	0
r_{213}	-1	0

Table 7

Signature matrix for group G_6

G_6	f_0	f_8	f_{14}
r_{10}	0	0	0
r_{20}	0	0	0
r_{18}	0	0	0
r_{28}	0	0	0
r_{114}	0	0	0
r_{214}	0	0	0

4.5. Decision tree for diagnosis DAMADICS Actuator:

The introduction of probabilities to evaluate the significance of each residual and the reliability of the decision is another component of our approach. The proposed method uses a time window that can be sized according to the time requirement. Diagnosis with a large time window includes a diagnosis delay but will lead to a decision with a high confidence index. On the contrary single diagnosis with a small time window leads to early diagnosis but with a lower confidence index. The diagnosis results either from the usual thresholding technique or from the on-line determination of fault probabilities and confidence factors (Kourd et al., 2011).

According to the values of residuals and selection of faulty model *NNFM (j)*, we introduce decision trees to set the path of the branch that must be followed for the isolation of all detected faults. From this standpoint, we can configure many decision trees that lead to various computational complexities.

5. CONCLUSION

The accurate and timely fault diagnosis of safety critical systems is important because it can decrease the probability of catastrophic failures, increase the life of the plant, and reduce maintenance costs. In this paper, a multiple-model FDI scheme is presented for a DAMADICS actuator. NNs technique is applied for residual generation. In the first step of FDI, the use of NNs method is considered as an alternative to the traditional model-based approach for residual generation. When quantitative models are not readily available, a correctly trained NNs model can be used as a non-linear dynamic model of the system. The proposed NNFMs presented in this paper can be used to diagnose the faults in the DANMADICS actuator. In the second step of the FDI task, a decision tree was introduced for residual evaluation. An evolutionary technique was used to isolate the faults step by step by separating the fault candidates into several groups with same signatures. This study shows that the developed approach can produce good diagnosis results for a complex system exposed to numerous faults. The use of decision trees divides the computational effort by 10 for the DAMADICS system and can be implemented on line.

Applying this scheme on other types of operating systems will be the key issue of our interests in future research, in which other labeled faults will be investigated. This work can be extended to diagnose new faults using integrated set of decision tree which looks to be a worthwhile direction for future case study. Implementing this decision tree for the online diagnosis issues as well as choosing its components is an open task for more optimization and quick isolation.

ACKNOWLEDGMENTS

The work was supported in part by Ministry of Higher Education and Scientific Research in Algeria.

REFERENCES

[1] Bartys M., Patton R., M. Syfert, Heras S. & Quevedo J., (2006) "Introduction to the DAMADICS FDI benchmark actuator study". Control engineering practice. 14. 577-596.

[2] Brown M. & Harris C. J., (1994) "Neuro-fuzzy adaptive modelling and control". Englewood Cliffs, NJ: Prentice-Hall.

[3] Browne A., Hudson B. D., Whitley D. C., Ford M.G. & Picton P., (2004) "Biological data mining with neural networks: implementation and application of a flexible decision tree extraction algorithm to genomic problem domains". Neurocomputing. 57. 275–293.

[4] Calado J., Korbicz J., Patan K., Patton R.J & Sa da Costa J., (2001) "Soft computing approaches to fault diagnosis for dynamic systems". European Journal of Control 7. 248–286.

[5] Calado J., & Sa da Costa J., (2006) "Fuzzy neural networks applied to fault diagnosis. In: Computational intelligence in fault diagnosis", Springer-Verlag, London.

[6] Chen J. & Patton R.J., (1999) "Robust Model-Based Fault Diagnosis for Dynamic Systems". Kluwer Academic Publishers, Berlin.

[7] Chen Y., Abrahama A. & Yanga B., (2006) "Feature selection and classification using flexible neural tree". Neurocomputing. 70. 305–313.

[8] DAMADICS, (2002) Website of the RTN DAMADICS. (http://diag.mchtr.pw.edu.pl/damadics).

[9] Frank P.M. & Koppen-Seliger B., (1997) "New developments using AI in fault diagnosis". Engineering Applications of Artificial Intelligence 10. 3-14.

[10] Gertler, J. (1998) "Fault detection and diagnosis in engineering systems", New York, Basel, Hong Kong: Marcel Dekker.

[11] Guglielmi G., Parisini T. & Rossi G., (1995) "Fault diagnosis and neural networks: A power plant application" (keynote paper). Control Engineering Practice. 3. 601–620.

[12] Himmelblau D. M., (1992) "Use of artificial neural networks in monitor faults and for troubleshooting in the process industries". In: Preprints IFAC Symp. On-line Fault Detection and Supervision in the Chemical Process Industries, Newark, Delaware. USA. 144-149.

[13] Isermann R., (1994) "Fault diagnosis of machines via parameter estimation and knowledge processing", A tutorial paper. Automatica. 29. 815–835.

[14] Isermann R., (1997) "Supervision, fault detection and diagnosis of technical systems". Special Section of Control Engineering Practice 5.

[15] Isermann R., (2006) "Fault Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance". Springer-Verlag, New York.

[16] Janczak A., (2005) "Identification of Nonlinear Systems Using Neural Networks and Polynomial Models". A Block-oriented Approach. Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin.

[17] Jang J. S. R., Sun C. T. & Mizutani E., (1997) "Neuro-Fuzzy and Soft Computing". Englewood Cliffs, NJ: Prentice-Hall.

[18] Korbicz J., Koscielny J., Kowalczuk Z. & Cholewa W., (2004) "Fault Diagnosis. Models, Artificial Intelligence, Applications". Springer-Verlag, Berlin Heidelberg.

[19] Korbicz J., Patan K. & Kowal M., eds., (2007) "Fault Diagnosis and Fault Tolerant Control. Challenging Problems of Science - Theory and Applications: Automatic Control and Robotics". Academic Publishing House EXIT, Warsaw.

[20] Koscielny J. M., Bartys M., Rzepiejewski P. & S. da Costa J., (2006) "Actuator fault distinguishability study for the DAMADICS benchmark problem". Control Engineering Practice, 14, 645-652.

[21] Kourd Y., Guersi N. & Lefebvre D., (2008) "A two stages diagnosis method with Neural networks". Proceeding ICEETD, Hammamet Tunisie.

[22] Kourd Y., Guersi N. & Lefebvre D. (2010), "Neuro-fuzzy approach for fault Diagnosis: application to the DAMADICS". 7th international Conference on Informatics in Control, Automation and Robotics. Proceeding ICINCO 2010. Funchal. Portugal.

[23] Kourd Y., Lefebvre D. & Guersi N. (2011) "Early FDI Based on Residuals Design According to the Analysis of Models of Faults: Application to DAMADICS". Advances in Artificial Neural Systems. doi:10, 1155/2011/453169.

[24] Lefebvre D., Chafouk H. & Lebbal M. (2010) "Modélisation et Diagnostic des Systèmes. Une Approche Hybride". Editions Universitaires Européennes.

[25] Maji P., (2008) "Efficient design of neural network tree using a new splitting criterion". Neurocomputing. 71. 787-800.

[26] Narendra K., Balakrishnan J. & Kermal M., (1995) "Adaptation and learning using multiple models, switching and tuning". IEEE Control Systems Magazine. 37–51.

[27] Osowski S. (1996) "Neural Networks in Algorithmitic Expression". WNT, Warsaw (in Polish).

[28] Patan K. & Parisini T., (2005) "Identification of neural dynamic models for fault detection and isolation: The case of a real sugar evaporation process". Journal of Process Control. 15.67-79

[29] Patton R.J., Chen J. & Siew T. (1994) "Fault diagnosis in nonlinear dynamic systems via neural networks". In: Proc. of CONTROL'94, Coventry, UK. 2. 1346–1351.

[30] Patton, R.J., Korbicz, J., (1999) "Advances in computational intelligence". Special Issue of International Journal of Applied Mathematics and Computer Science 9.

[31] Patton R.J., Frank P.M. & Clark R., (2000) "Issues of Fault Diagnosis for Dynamic Systems". Springer-Verlag, Berlin.

[32] Pomorski D., Perche P. B., (2001) "Inductive learning of decision trees: application to fault isolation of an induction motor". Engineering Applications of Artificial Intelligence. 14,155–166

[33] Previdi F. & Parisini T., (2006) "Model-free actuator fault detection using a spectral estimation approach: the case of the DAMADICS benchmark problem". Control Engineering Practice. 14, 635–644.

[34] Sorsa T. & Koivo H. N., (1992) "Application of neural networks in the detection of breaks in a paper machine". In: Preprints IFAC Symp. On-line Fault Detection and Supervision in the Chemical Process Industries, Newark, Delaware, USA. 162–167.

[35] Sugumaran V., Muralidharan V. & Ramachandran K. I., (2007) "Feature selection using decision tree and classification through proximal support vector machine for fault diagnosis of roller bearing". Mechanical Systems and Signal Processing. 21. 930-942.

[36] Uppala F. J., Patton R. J. & Witczak M., (2006) "A neuro-fuzzy multiple-model observer approach to robust fault diagnosis based on the DAMADICS benchmark problem". Control Engineering Practice. 14. 699-717.

[37] Witczak M. & Korbicz J., (2002) "Genetic programming in fault diagnosis and identification of non-linear dynamic systems". In: W. Cholewa, J. Korbicz, Koscielny & J.M. Kowalczuk, (Eds.), Diagnostics of processes. Models, Methods of Artifical Intelligence, Applications, Scientific Engineering Press, Warsaw.

[38] Witczak M., (2007) "Modelling and Estimation Strategies for Fault Diagnosis of Non-Linear Systems". From Analytical to Soft Computing Approaches. Lecture Notes in Control and Information Sciences. Springer–Verlag, Berlin.

[39] Witczak M., Korbicz J., Mrugalski M. & Patton R. J., (2006) "A GMDH neural network-based approach to robust fault diagnosis: Application to the DAMADICS benchmark problem". Control Engineering Practice. 14. 671-683.

[40] Zhang K., Li Y., Scarf P. & Ball A., (2011) "Feature selection for high-dimensional machinery fault diagnosis data using multiple models and Radial Basis Function networks". Neurocomputing. 74. 2941–2952.