5 th International Conference on Control & Signal Processing (CSP-2017)
Proceeding of Engineering and Technology –PET
Vol.26 pp.139-144

Diagnosis of hybrid systems through Neural Networks and Timed Automata

Lotfi MHAMDI[#1], Ayachi ERRACHDI[*2], Hedi DHOUIBI[#3]

[#]*LARATSI, Engineering School of Monastir, Tunisia*
[1]lotfienim@yahoo.fr
[3]hedi.dhouibi@laposte.net

[*]*Laboratoire de Recherche Automatique, Université Tunis El Manar*
[2]errachdi_ayachi@yahoo.fr

***Abstract*— Despite technological advances and progress in industrial systems, the fault diagnosis of a system remains a very important task. In fact, an effective diagnosis contributes not only to improve reliability but also to decrease in maintenance costs. This paper, presents a diagnosis approach of hybrid systems thanks to the use of Timed Automata and Neural Networks. Dynamic models (in normal and failing mode) are generated by a Timed automata based methods as well as through state equations generated by Neural Networks (NN) model. The procedure of fault localization through a method based on the Neural Networks does not allow locating faults with the same signature of failure. Thus the diagnosis technique for the localization of these defects will be based on the time analysis using Timed Automata. The proposed approach is then validated by simulation tests in a water tank.**

***Keywords*— Fault Diagnosis, Timed Automata, Fault Localization, Neural Networks.**

I. INTRODUCTION

Improvement in the dependability of systems rests essentially on algorithms of detection and isolation of defects. These algorithms mainly consist in comparing the actual behavior of a system with a behavior of reference systems describing normal functioning (in order to detect defects) or describing different kinds of defects (In order to analyze and isolate faults), while reducing false alarms, non-detections as well as delays in detection of defects. In surveillance approaches based on quantitative models performances of detection procedures and localization of failures strongly depend on the used model. Once the last is generated, failure indicators can be deducted. Obtaining such model is a complex and difficult task more particularly for the process engineering systems because of their diversity and coupling energies which characterize them.

Classical techniques for detection, localization and diagnosis show their limits, especially for systems which become increasingly complex, which Hybrid Dynamic Systems (HDS). HDSs [1] are systems composed of dynamics of a continuous and discrete nature interacting between them; continuous dynamics is represented by differential equations and discrete dynamics by state transitions.

To obtain good performance in terms of coverage and high quality of isolation, research is directed towards coupling approaches and using their complementarily. The coupling of continuous / discrete approaches must achieve good performance.

In this paper, we propose an approach based on the use of Neural networks well known in continuous field, attached to timed automata used in the field of discrete events systems [2]. For a complex system, neural networks can solve non-linear and multi-variable problems, then store knowledge compactly and finally learn online and in real-time [3]. Timed automata [4, 5] allow to take into account the dynamic evolution of the system and their failures propagation. The diagnoser built by automaton can refine the location of the fault [6]. As shown in the flowchart of Figure 1.

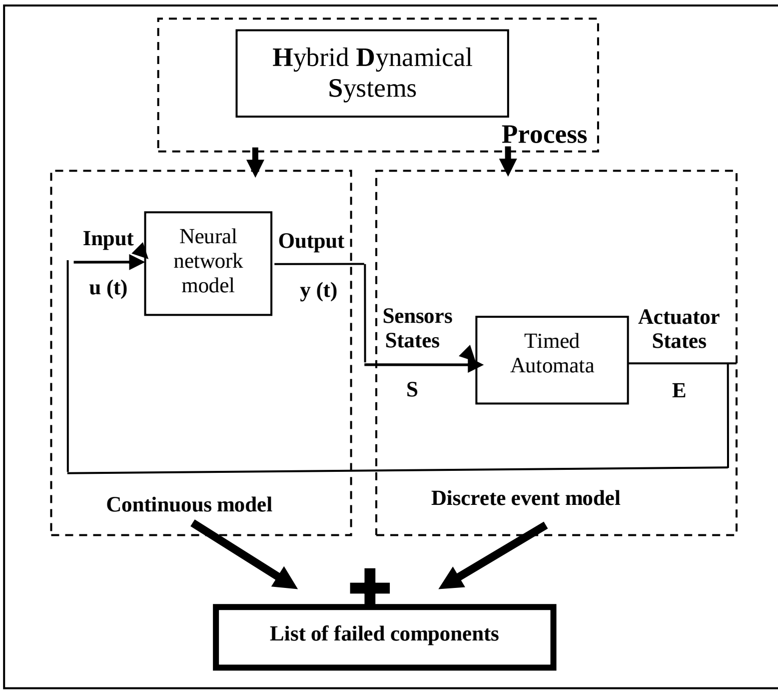

Fig.1. Flowchart of the proposed approach

ISSN 2356-5608

II. DIAGNOSTIC APPROACH BASED ON A HYBRID MODEL

Many industrial processes are hybrid in nature, which means that their behavior results from the evolution and interaction of continuous variables and discrete variables. For this type of system, little work has been devoted to detecting, locating or diagnosing failures [7]. The literature in this field is abundant and numerous solutions have been proposed for continuous and discrete systems, linear and nonlinear.

The mixed approach, proposed in this paper, is based on a combination of two models (continuous and discrete). The continuous component is described by a set of differential equations obtained by neural networks and the discrete component by a finite state automaton. This approach evolves through an alternation of continuous steps, where state variables and time evolve continuously, and discrete steps where several discrete and instantaneous transitions can be crossed. The diagnostic method combines the advantages of the both approaches (Neural Networks and Timed Automata) for best performance, particularly in the fault locating phase. Each step is described in a conventional form.

A. Fault Diagnosis Approach

The neural networks model under the normal condition is established in the equation (2) and (3). Therefore, a fault can be detected by observing residual values, which are defined as the difference between the actual measured values under a fault condition and the expected values under the normal condition.

The output residual of a dynamic model can be calculated by:

$$e(k) = y(k) - yr(k) \tag{1}$$

The sum of square residual can be calculated as:

$$J = \frac{1}{N}\sum_{k=1}^{N}(e(k))^2 \tag{2}$$

Where N, is the length of the observation.

Assume that the threshold value ε, then the fault can be determined by:

$$\begin{cases} J \le \varepsilon, & \text{Normal} \\ J > \varepsilon, & \text{Fault} \end{cases} \tag{3}$$

and $\varepsilon = 2J_0$, where J_0 is the sum of square residual under normal condition.

In this paper, the learning rate is $\eta = 0.42$, the regularization coefficient is $\lambda = 1$, the hidden neurons are 25, and the output neuron is equal to 1.

We notice that the two real and estimated curves are combined, which shows the effectiveness of the neuronal estimator to give a value close to the reality, and this gives us the confidence to use this type of estimator in this applications.

B. Principle of neural modeling

In this part, the principle of the dynamic neural modelling of the nonlinear multivariable systems is proposed. This principle is given in Figure 2.

A nonlinear multivariable system given by the following form:

$$y(k+1) = f\left[y(k),...,y(k-ns+1),...,u(k),...,u(k-nu+1)\right]$$

with:

f: unknown function of model process

$U = [u_1(k) \quad u_2(k) \quad ... \quad u_{nu}(k)], (1\times nu)$ being the input of the process,

$Y = [y_1(k) \quad y_2(k) \quad ... \quad y_{ns}(k)]^T, (ns\times 1)$ being the output of the process,

$Yr = [yr_1(k) \quad yr_2(k) \quad ... \quad yr_{ns}(k)]^T, (ns\times 1)$ being the output of the NN,

$E = [e_1(k) \quad e_2(k) \quad ... \quad e_{ns}(k)]^T, (ns\times 1)$ being the error vector,

$e_i(k) = y_i(k) - yr_i(k)$: error between the $i-th$ measured output and the $i-th$ NN output,

x: The NN input vector, $(t\times 1), (t = nu + ns)$,

ncc: The number of nodes of the hidden layer,

W: The synaptic weights of the layer towards the hidden layer, $(ncc\times t)$,

Z: The synaptic weights of the hidden layer towards the output layer, $(ns\times ncc)$,

s: The activation function,

η: The learning rate,

λ: The scaling coefficient used to expand the range of NN output,

TDL: The Tapped Delay Line block.

The output of the $l-th$ hidden node $(l = 1,...,ncc)$:

$$net_l = \sum_{j=1}^{l} w_{lj}x_j = w_l x \tag{4}$$

$s(net_l)$: output of the $l-th$ node of hidden layer.

The $i-th$ NN output $(i = 1,...,ns)$ is given by the following equation:

$$\begin{aligned} yr_i(k+1) &= \lambda s\Big(\sum_{l=1}^{ncc} s\big(\sum_{j=1}^{t} w_{lj}x_j\big)z_{il}\Big) \\ &= \lambda s\Big(\sum_{l=1}^{ncc} s(net_l)z_{il}\Big) \end{aligned} \tag{5}$$

Finally, the compact form is defined as:

$$Yr(k+1) = \lambda s\left[Z^T S(Wx)\right] \tag{6}$$

with

$$x = [x_j]^T \in R^{t\times 1}; j = 1,...,t$$

$$Z = [z_{il}]^T \in R^{ns\times ncc}; i = 1,...,ns \text{ and } l = 1,...,ncc$$

$W = \left[w_{lj} \right] \in R^{ncc \times t}; l = 1,...,ncc \; and \; j = 1,...,t$

$S(Wx) = \left[s(net_l) \right]^T \in R^{ncc \times 1}; l = 1,...,ncc$

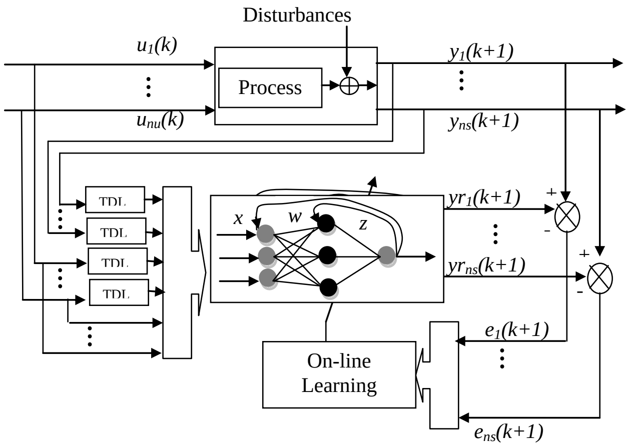

Fig.2. Principle of the neural modelling of the multivariable system

C. Neural networks estimator

On the basis of the input and output relation of a system, the nonlinear system can be expressed by a NARMA (Nonlinear Auto-Regressive Moving Average) model [8, 9], that is given by the following from:

$$y(k+1) = f\left[y(k),\ldots,y(k-n_y+1),u(k),\ldots,u(k-n_u+1) \right] \quad (7)$$

Where $f(.)$ represents the non-linear function mapping specified by the model, $y(k)$ and $u(k)$ are real, ($y(k) \in \mathbb{R}$ and $u(k) \in \mathbb{R}$) are the outputs and the inputs of the system respectively. k is the discrete time index. n_y and n_u are the number of the past output and input samples required for the prediction.

The used neural networks, in this paper, is a multilayer perceptron with tapped delay lines (TDL) of both the input and the output. The delay elements are used to introduce delayed inputs and outputs that are then fed to a static network as the repressor vector, so that the predicted neural network output will follow the target output.

The estimated ratio between the open circuit voltage and the standard open circuit voltage using neural networks $yr(k)$ is given by the following equation:

$$yr(k+1) = \lambda s(\sum_{l=1}^{N_1} s(\sum_{j=1}^{N_0} w_{lj} x_j) z_l) = \lambda s(\sum_{l=1}^{N_1} s(net_l) z_l) \quad (8)$$

with $s(net_l) = s(\sum_{j=1}^{N_0} w_{lj} x_j)$, $l = 1,\ldots,N_1$, s is a sigmoid function, w_{lj} and z_l are respectively the hidden synaptic weights and the output synaptic weights, x_j is the input vector of neural network, λ is a regularization coefficient, N_0 is the number of the input neurons and N_1 is the number of the hidden neurons. In the compact form:

$$yr(k+1) = \lambda s\left[z^T S(Wx) \right] \quad (9)$$

With $S(Wx) = \left[s(net_1) \;\; \ldots \;\; s(net_l) \right]$

and $S'(Wx) = diag\left[s'(net_1) \;\; \ldots \;\; s'(net_l) \right]$.

The update of the hidden synaptic weights and the output synaptic weights are given as follows:

$$w_{lj}(k+1) = w_{lj}(k) + \eta \Delta w_{lj}(k) \quad (10)$$

$$z_l(k+1) = z_l(k) + \eta \Delta z w_l(k) \quad (11)$$

With:

$$\Delta w_{lj} = \eta \lambda s'(net_l) S'(Wx) z_l x^T e(k) \quad (12)$$

$$\Delta z_l = \eta \lambda s'(net_l) S(Wx) e(k) \quad (13)$$

D. Discrete event model: Fault diagnosis through the Timed Automata

The timed automata tool [10, 11] is defined as a finite state machine with a set of continuous variables named clock. These variables evolve continuously in each location of the automata, according to an associated evolution function. As long as the system is in one state Si, the clock xi is continuously incremented. Its evolution is described by $\dot{X} = 1$. The clocks are synchronized and changed with the same step.

An invariant is associated to each state. It corresponds to conditions needed to remain in the state. The number of clocks depends on the parallelism in the system. The automata can stay in one state as long as the invariant condition is checked. Each transition of an automata is conditioned by an event or temporization called “guard” and its execution determines the discrete evolution of the variables according to its associated assignment.

Let us consider the timed automata given in Figure 3. This automata has two clocks x and y. The continuous evolution of time in this model is represented by $\dot{x} = 1$ and the labeled arcs in the graph represent the model of discrete evolution. The guard in each arc is a transition labeling function that assigns firing conditions with the transitions of the automata. The affectation is a function that associates with each transition of the automata with relation that allows actualizing the value of continuous state space variables after the firing of a transition. The invariant in the state S0 and S1 are respectively y ≤ 5 and x ≤ 8. The initial state of this system is represented by an input arc in the origin state (S0). In the dynamic model active clocks are found in each state. A graphical interpretation of the timed automata is the automata graph (Figure 3).

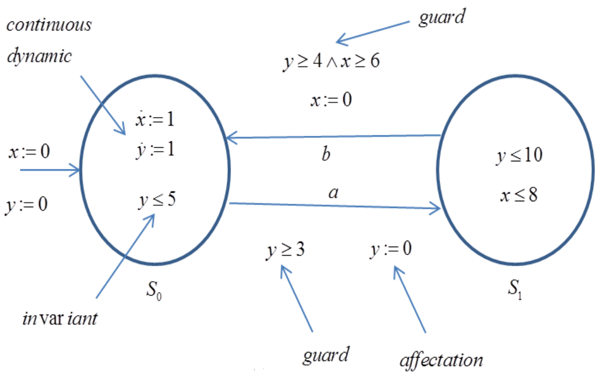

Fig. 3. Example of Timed Automata.

Our objective, thanks to the use of timed automata, is to build a diagnosis system called diagnostician which allows to analyze, detect and locate a fault in a system. The construction of the diagnostician is based on a dynamic model representing different functioning modes of the monitored system (normal and failing). The dynamic model is neither more nor less a copy of a control-command program of the system to diagnose with added time information such as the duration of different steps of functioning, the execution order of tasks and the date of event appearance.

III. Application Example

A. Description of the system

We consider as example a one tank hydraulic system, Figure 4. The tank filling system allows the mixing of ingredient A through the valve VA with ingredient B through the valve VB. The valve VS makes it possible to drain the tank. The sensors L2 and L3 give information on the tank level. Finally, the L1 level sensor detects the moment when the tank is empty or not, while the L4 level sensor activates an alarm in case of an overflow.

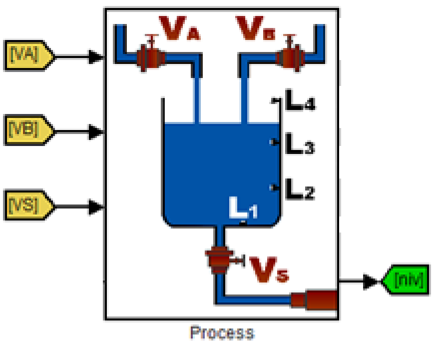

Fig. 4. One tank hydraulic system

S0: The process is initialized, consequently the tank should be empty.

S1: Firstly, the valve VA is opened, the ingredient A flows into the tank.

S2: If the level L2 is reached, then the valve VA is closed and the valve VB is opened.

S3: If the level L3 is reached, then the valve VB is closed and the valve VS is opened in order to drain the tank.

There are two potential defects of the system that concern the valves, and two potential defects of the system that concern the sensors.

- The two defects that concern the valves are: The valve remains opened in spite of the closing request. The valve remains closed in spite of the opening request.
- The two defects that concerns the sensors are: The sensor failed to detect higher level. The sensor does not detect the lower level.

B. Simulation results

Figure 5 shows the Diagnostic the valve VA stucks closed

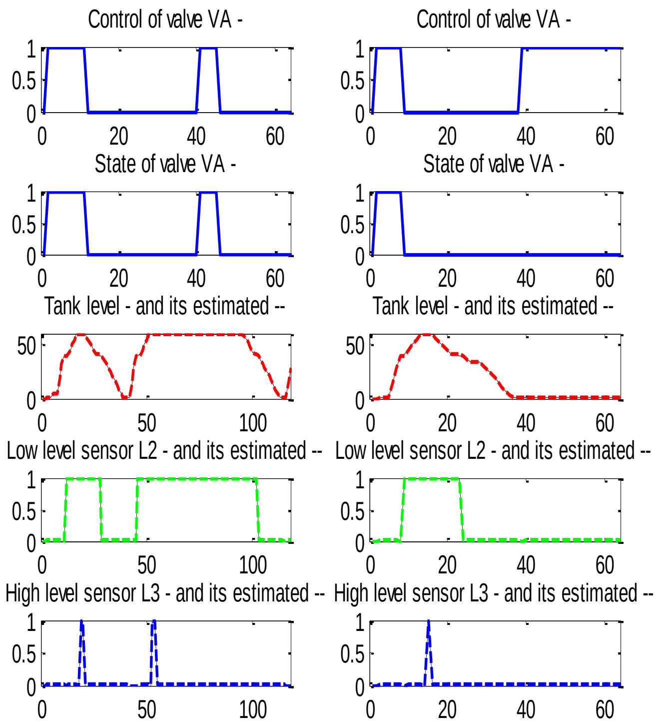

Fig. 5. -a. Functioning mode results, -b. Failure mode results

In figure 5, we present the neural networks model of the system with and without fault. It is noticed a good corresponding between the system and the neural model.

For a better understanding of the principle of the diagnostic, the figure 5 makes it possible to compare a normal process with a faulty operating condition. On the figure 5 a, we have the standard functioning of the process and on the figure 5 b, an operation failed.

In figure 6, we present the estimate error in the two cases with and without fault.

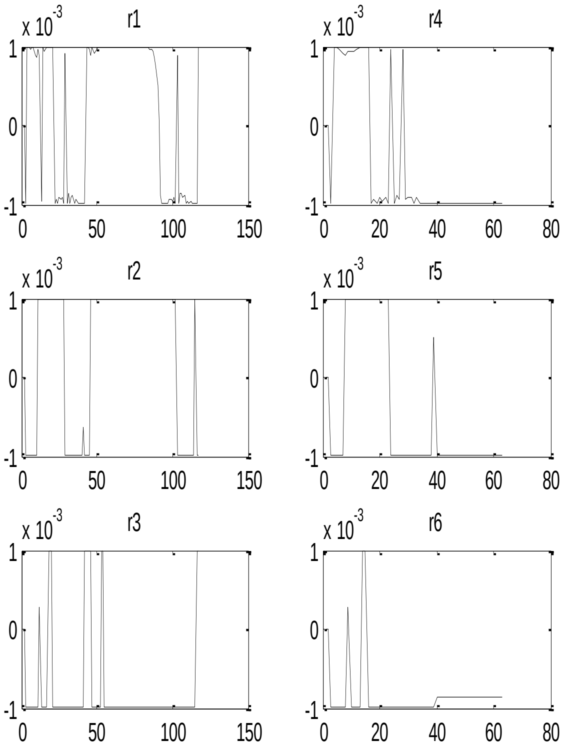

Fig. 6. Residual results

- Functioning mode results

r1 = tank level- estimated tank level
r2 = Low level sensor L2 - estimated Low level sensor L2
r3 = High level sensor L3 - estimated High level sensor L3

In the table 1, we present the different characteristics of the estimate error. i=1, 2, 3.

TABLE I
FUNCTIONING MODE RESULTS

	min(ri)	**max(ri)**	**MES(ri)**
r1	-9.9998e-04	9.9896e-04	8.9506e-07
r2	-9.9902e-04	9.9449e-04	9.5301e-07
r3	-9.9738e-04	9.9954e-04	9.5337e-07

- Failure mode results

r4 = tank level- estimated tank level
r5 = Low level sensor L2 - estimated Low level sensor L2
r6 = High level sensor L3 - estimated High level sensor L3
i=4, 5, 6.

TABLE III
FAILURE MODE RESULTS

	min(ri)	**max(ri)**	**MES(ri)**
r4	-9.9983e-04	9.9338e-04	9.0834e-07
r5	-9.9392e-04	9.9449e-04	9.3885e-07
r6	-9.9292e-04	9.9954e-04	8.4556e-07

IV. CONCLUSION

In this paper we tackled the problem of fault diagnosis in a hybrid system using Neural Networks and Timed Automata. The dynamic models (in normal and failing mode) are generated by the Neural Networks.

The Neural Networks based method consists of generate residuals which are calculated by making the difference, eventually filtered between actual output and those estimated. Today this very powerful method allows accomplishing objectives for detection and localization of faults in an effective and fast way. Nevertheless, the fault location procedure by the Neural Networks based method does not allow locating faults having the same signature failure. To fix this problem, diagnosis technique for location of these defects is based on analysis time using timed automata.

A perspective of this work is to extend our approach to take into account the diagnosis problem when the system is affected simultaneously by actuators and sensors faults. Another problem not addressed in this paper would be study and the mastery of propagation of defects in a hybrid system.

REFERENCES

[1] Michael Stephen Branicky, “*Studies in hybrid systems: Modeling, Analysis and Control*,” Ph.D thesis, Department of Electrical Engineering and Computer Science – MIT Cambridge, Juin1995.

[2] Mhamdi L., Dhouibi H., Simeu-Abazi Z., Liouane N., "Using constrained Petri nets and timed automata for modelling of discrete event systems," *IJE*, vol. 27, pp. 113-122, January 2014.

[3] P.L. MAISONNEUVE, Utilisation des réseaux de neurones pour la modélisation fiabiliste des pièces mécaniques, mémoire de Master de l’école nationale d’ingénieurs de Saint-Etienne, sept 2006.

[4] Bengtsson J. and Wang Yi, "Timed automata: semantics, algorithms and tools," In lecture Notes on Concurrency and Petri Nets, W. Reisig and G. Rozenberg (eds), LNCS 3098, *Springer-Verlag,* 2004.

[5] Saeed Ghasemi, Leili Mohammad Khanli, Mina Zolfi, GhaderTahmasebpour, "Modeling and Simulation of NFC Logical Layer Peer-to-Peer Mode using CPN and TA," *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 4, No. 2, pp. 162~168,April 2014.

[6] Zineb Simeu-Abazi et al., "Fault diagnosis for discrete event systems: modelling and verification," *Reliability Engineering et systems Safety*, Volume 95, Issue 4, PP. 369-378, Avril 2010.

[7] Panos Antsaklis, Xenofon Koutsoukos, Janan Zaytoon, “*On hybrid control of complex systems: a survey*,” ADPM’98, Reims, France, March 19-20 1998.

[8] Errachdi, Ayachi., and Benrejeb, Mohamed., 2016. Online identification using radial basis function neural network coupled with KPCA, International Journal of General Systems, DOI: 10.1080/03081079.2016.1151421.

[9] Mahmoud, Ines., Errachdi, Ayachi., and Benrejeb, Mohamed., 2017. Disturbance influence on neuronal system modelling, International Journal of Computer Applications in Technology, 55(1).

[10] Sava. A.T. "Sur la synthese de la commande des systemes a evenements discrets temporises." PhD thesis, Laboratoired’Automatique de Grenoble, November 2001.

[11] Simeu-Abazi Z., Rayhane H., Bennani T., Bouredji. Z., "Optimisation des temps de detection dans la surveillance des systemes," *5eme Congres International de Genie Industriel*, pp. 25-29, Octobre 2003.