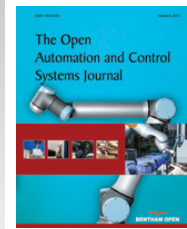




The Open Automation and Control Systems Journal

Content list available at: www.benthamopen.com/TOAUTOCJ/

DOI: 10.2174/1874444301709010015



RESEARCH ARTICLE

Monitoring of Dynamical Systems Using Hybrid Automata with Stopwatch

B. Tolbi* and H. Tebbikh

LAIG, University of Guelma, BP.401, 24000 Guelma, Algeria

Received: October 30, 2016

Revised: February 15, 2017

Accepted: March 30, 2017

Abstract:

Objective:

The complexity and size are increasingly important in industrial processes; they lead to the supervision of a growing number of variables whose knowledge is generally based on measurements and/or on processes models.

Introduction:

Different monitoring methods are generally based on output measurements and/or process modeling.

Materials and Methods:

With the lack of precision and the monitoring speed in most work due to the model or measurement noise, for this we aim to develop a new and more powerful detection model based on clocks, to ensure a rapid detection and determine fault duration to facilitate the fault identification.

Result:

This paper proposes a monitoring approach for interruption, permanent and/or intermittent faults in hybrid dynamical systems, whose principle is based on system modeling using hybrid automaton with stopwatch. The obtained results show that the monitoring system is able to detect rapidly the considered types of faults.

Conclusion:

A classic example is dedicated to illustrate our approach; the obtained results confirm the effectiveness of the proposed work.

Keywords: Hybrid dynamical systems, Modeling, Intermittent fault, Permanent fault, Monitoring, Hybrid automaton with stopwatch.

1. INTRODUCTION

Hybrid dynamic systems (HDSs) are currently attracting a lot of attention. The behavior of interest of these systems is determined by the interaction of a continuous and a discrete event dynamics. The hybrid character of a system can owe either to the system itself or to a discrete controller applied to a continuous system. Several works have been devoted to the modeling of HDSs. In this work we are interested in hybrid models formalisms which combine explicitly a discrete event model and a continuous model. The most known model of this category is hybrid automata (HA). HA constitute a powerful analysing tool. They were introduced by Alur *et al.* [1], this tool can model the widest variety of continuous dynamics; it also allows a formal analysis of HDSs. HA are finite state automata, extended with real value

* Address correspondence to this author at the LAIG, University of Guelma, BP.401, 24000 Guelma, Algeria; Tel: +213775.45.59.13; E-mails: bilalcdsi@gmail.com, tolbi.bilal@univ-guelma.dz

variables evolving according to differential equations; they are the most general HDSs formalism, since they can model the largest variety of HDSs.

The current economic context has made the requirements of performance, reliability, and security of HDSs. Considering the growing complexity of these systems, it is necessary to establish a monitoring mechanism that can detect the dysfunction of the system. There are two monitoring methods: the method without model and the method based on a model. The first one uses only heuristic knowledge on the system; we shall mention the methods based on expert systems [2], or neurons' networks [3]. Based on qualitative models, the second one makes it possible to represent the behavior of the process with a certain degree of abstraction, through models that are no longer mathematic but of a symbolic order, this method relies on the comparison of the expected behavior of the system (described by model) and the actual observations. So, a fault occurrence is declared when some discrepancies occur between the two mentioned outputs. In order to have the monitoring system able to detect the deviations of normal functioning, the qualitative models have to represent systems that are qualitatively continuous [4], discrete and hybrid [5, 6]. We consider a normal functioning of a system, any functioning of the latter that does not lead to a faulty state.

The purpose of a monitoring system used on an industrial process, is to emit an alarm by analyzing the information sent out by the captors or signals coming from the command process. The issue of the monitoring of industrial systems has been dealt with in several works; both on the continuous systems [7] and on the systems with discrete event and hybrid systems [4]. Very few works, however, are to be found on HDSs. These monitoring methods are based on models of normal function and/or the malfunction of system which are found in the literatures [3, 8 - 13]. The major issue we are concerned with here consists in studying a system in real time that is able to develop in several different functioning processes. Each of these processes having a distinct evolution dynamics. In this paper we focus on the second category in which the normal behavior of the system is modeled.

Most of the works are not based on time aspect of faults and operating system regardless of the normal functioning or faulty functioning. For this we aim to develop a new and more powerful detection model based on time aspect. This paper proposes a monitoring approach for permanent and/or intermittent faults in hybrid dynamical systems, whose principle is based on system modeling using hybrid automaton with stopwatch this model is proposed with two aspects: data acquisition and fault detection. The data collected during the actual functioning of the system (in presence of faults) and they will be compared with normal functioning.

We will build a monitoring system based on HA with stopwatch to trigger an alarm in the case where an intermittent or permanent fault occurs. Firstly we have modeled the system in normal functioning with HA, and basing on transitions time, we have determined a new transition which leads us to the alarm state and thereafter we have built the monitoring system able to detect the two types of faults previously mentioned. Finally a case study will be used subsequently to illustrate the idea.

Outline of the paper: Section 2 introduces fault modelling techniques; in this section the two types of faults will also be presented. Section 3 explains our monitoring approach. In Section 4 our approach with a case study namely hydraulic system will be illustrated. Finally, we conclude with our on-going work and perspectives in Section 5.

2. FAULT MODELING

Fault modeling involves the acquisition for prior knowledge of faults to be detected. We note that a system fault is a faulty state, while a failure or a fault source can lead to a faulty state. In the context of HDSs, the occurrence of a fault is, also, the passage towards a faulty state. This passage can be modeled by a transition on a fault, if we consider models-based events [14, 15]. If we consider a state-based model [16], a partition of system states in nominal conditions and faulty states are first established. In this last case, a system is declared faulty if it reaches a malfunction state.

In permanent faults, the passage towards a state of incipient fault is due to the occurrence of a fault event. Then the system will move to other states of dysfunction. In case of intermittent faults, the system can return to a normal operating state after occurrence of an event back in normal operation. In this work, we focus on two types of faults which are defined as follows.

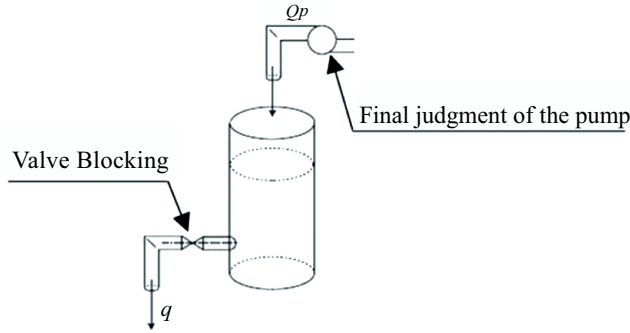


Fig. (1). Filling system.

2.1. Permanent Faults

Permanent faults are defined as malfunctions of components that need to be replaced or repaired. Therefore, a fault is permanent if recovery occurs after the repair or replacement of the faulty component. For example: The final judgment of the pump is shown in Fig. (1).

2.2. Intermittent Faults

An intermittent fault is usually the result of a partial and gradual degradation of a system component, can lead to permanent fault. In this case, the system can regain its nominal operation after occurrence of the event. Therefore, fault is intermittent if the recovery occurs automatically (without resorting to replace the components). For example, valve blocking in opening or closing mode (Fig. 1).

The difference between the above fault types is that the intermittent fault leads the system to switch between the offending state and the normal state (no faults), however, the permanent faults are always associated with events of overlap and the system cannot automatically switch from fault state to the normal state.

3. MONITORING APPROACH

The monitoring approaches in which the abstraction of continuous dynamics of HDS has been utilized (through the use of DES models) leading to a considerable loss of information and so are not reliable for the fault detection. In other words, in some cases, failed behavior manifested by a trajectories deviation of continuous dynamics of system. Therefore, using the diagnostic approach based on a purely discrete abstraction of the system evolution is inadequate for the fault detection purposes.

To overcome this problem, a monitoring approach is proposed in this article based on the work presented by Karoui [17] and Derbel [18] illustrated in Fig. (2-4). Our goal is to develop a new approach for monitoring purposes utilizing the stopwatch hybrid automaton model (SHA). Fig. (5) illustrates the overall scheme of our proposed monitoring approach. This approach takes into consideration the temporal time aspect in the evolution of HDS to be monitored. It relies on the use of a complete model of the system in the normal operation in the form of a hybrid automaton (HA). HA allow both modeling of discrete and continuous parameters that make up the HDSs. The continuous component is described by a set of ordinary differential equations, and discrete component is described by a finite automaton. This model describes the time evolution of system events through the use of a set of timers. These fictitious variables are used to abstract the interactions between the continuous dynamics (variables) and discrete dynamics (events) of a system, using a set of temporal constraints (guards) conditioning the continuous evolutions and discrete evolutions.

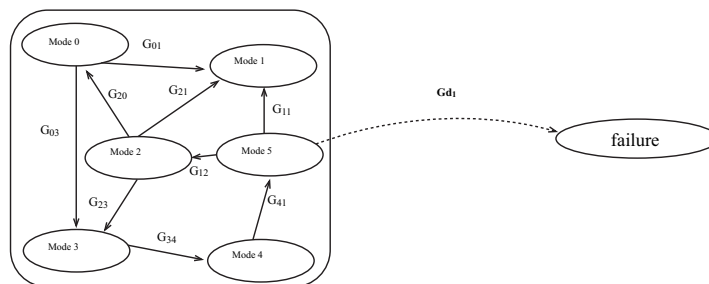


Fig. (2). General structure of a monetoring model.

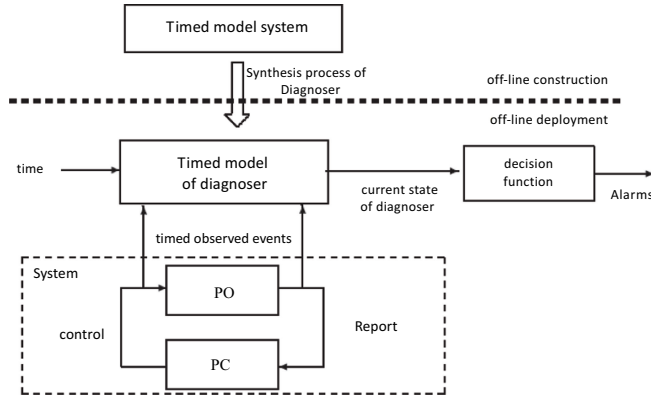


Fig. (3). Overall approach of diagnostic based on timed automata.

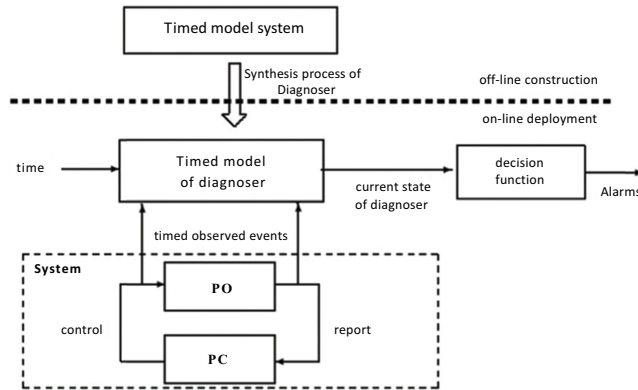


Fig. (4). Diagram of on-line diagnostic based on rectangular hybrid automata.

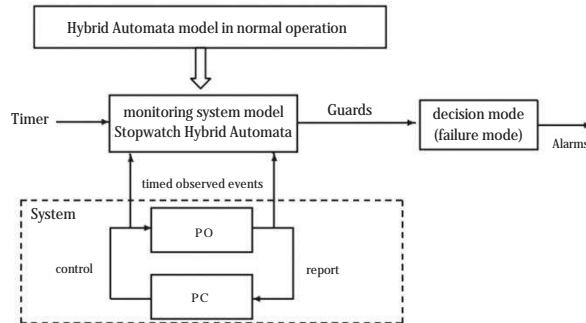


Fig. (5). Overall scheme of our proposed monitoring approach.

In the sequel, first, some required definitions are presented; then, the developed monitoring algorithm is detailed.

Definition 1. A linear hybrid automaton [19] with stopwatch is 8-uplet $H = (S, X, T, \Sigma, t, dif, Inv, S_o)$ in which the following variables are incorporated:

- S : the finite set of locations (also known as localities, situations);
- X : the finite set of real variables (continuous state vector components);
- T : the finite set of transitions. We note that $a = (s, g, \sigma, R, s') \in T$ where s is a location source, g is the guard, σ is the event associated, R is the assignment and s' is the destination location;
- t : the stopwatch timer which calculates the time at each operating cycle;
- Σ : finite set of labels (i.e. set of event-actions related to the transitions crossing);
- Dif : the function associating with every location $s \in S$ the set of continuous behavior (also called as activities);
- Inv : the function associating with every location $s \in S$ an invariant $inv(s)$;
- $S_o \in S$: the initial location.

Theorem 1: Assume that $ev_j, j \in \{1, \dots, N\}$ is the set of all related events, where N represents the number of events in the system. We consider t as a timer that calculates the elapsed time from the beginning of the operating cycle of the system. This timer reset to zero after each cycle execution. Also assume that $T_j, j \in \{1, \dots, N\}$ is a set of all occurrence times associated with ev_j each, According to the mentioned assumptions, we can derive the following condition:

If the event ev_j appears when $t = T_j$, then the system is in normal operation,

Else, the timer indicates a violation of specific time associated with each ev_j .

Example 1. Let $j = 2$, therefore they are: Two events ev_1 , and ev_2 ; Two moments of occurrence, T_1 and T_2 .

If ev_1 appeared when $t = T_1$ and ev_2 appeared when $t = T_2$, the system works normally.

Else, the timer indicates a violation of the specific time associated with ev_1 and/or ev_2 .

Theorem 2: Assume that $X_i, i \in \{1, \dots, n\}$ is the set of all state variables of continuous dynamics of HDS, where n is the variables number, $M_i, i \in \{1, \dots, m\}$ is the finite set of modes (also called locations) and m is the number of modes in normal operation. Crossing conditions fixed in the HA are calculated using $x_i \in X_i$, which is an external variable of HA. We consider $T_s, s \in \{1, \dots, S\}$ as the set of intervals in which S is the number of occurrence intervals of modes $M_i, i \in \{1, \dots, m\}$ and assume that T_d is the occurrence interval of the fault d_i , According to the mentioned assumptions, we can derive the following condition:

If the condition C_i on x_i that keeps the input transition associated with each mode M_i is true

($t \in T_s$), then the system normally switches from the previous mode to the current mode,

Else, the timer indicates a violation of the specific time for each C_i that control the input transition of the current mode.

Example 2. Let $m = 2$ and $i = 2$, $C(x_1 \geq 0.5)$ and the occurrence interval of the mode M_2 is $[50 \ 60]$ therefore:

If $(x_1 \geq 0.5)$ and $t \in [50 \ 60]$, the system switches from the mode M_1 to the current mode M_2 , else, the timer indicates a violation of the specific time $[50 \ 60]$ on the condition $(x_1 \geq 0.5)$ which keeps the input of the current transition mode M_2 .

From the last two theorems, we distinguished input transitions guards associated with each mode M_i . Accordingly, the monitoring algorithm can be developed as follows.

3.1. The Monitoring Algorithm

The monitoring process is formally described in Algorithm 1. We begin by initializing the timer t then run the SHA. The variable t is used to measure the time elapsed between the occurrence time and disappearance time of the observable event. It can also measure the length of stay in each mode. This variable is reset after each execution cycle. This algorithm could be simplified as the architecture presented in Fig. (6). Our algorithm is only valid for this type of systems, and then they cannot identify the two faults that occur at the same time, it remains as a perspective of our research.

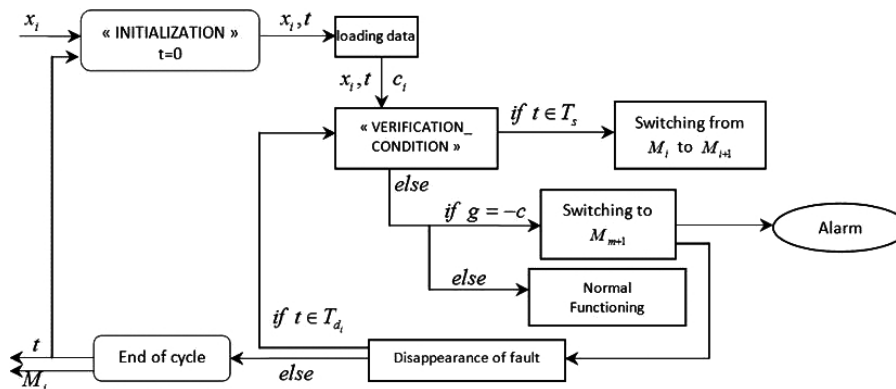


Fig. (6). Algorithm architecture.

Algorithm 1 proposed monitoring Algorithm

```

Inputs:  $x_i$ 
Outputs:  $M_i$ ,  $t$  of timer
1: «INITIALIZATION», time initialization,  $t=0$ ;
2: Loading data in SHA
3: If «VERIFICATION_CONDITIONS», the condition  $C_i$  do
4:   verification of timer  $t$ ;
5:   If the timer  $t \in T_s$  do
6:     system normally switches from  $M_i$  to  $M_{i+1}$ ;
7:   Else if the guard  $g = -C_i$  is true when  $t \in T_s$  do
8:     the system switches to the malfunctioning mode  $M_{m+1}$ ;
9:     announce the occurrence of a fault  $d_i$ ;
10:    until the fault  $d_i$  disappeared do
11:      If  $t \in T_{d_i}$ 
12:        return «VERIFICATION_CONDITIONS»;
13:      else
14:        End of the operating cycle;
15:      End
16:    until end
17:  else
18:    announce « normal », the system operates in the normal mode;
19:  end
20: end
21: else
22:   the operating system staying in the normal mode  $M_i$ ;
23: End.
    
```

In the following, a hydraulic system has been used as dedicated example to illustrate the process implementation.

4. CASE STUDY

We consider the hydraulic system [7] depicted in Fig. (7). It consists of [12]:

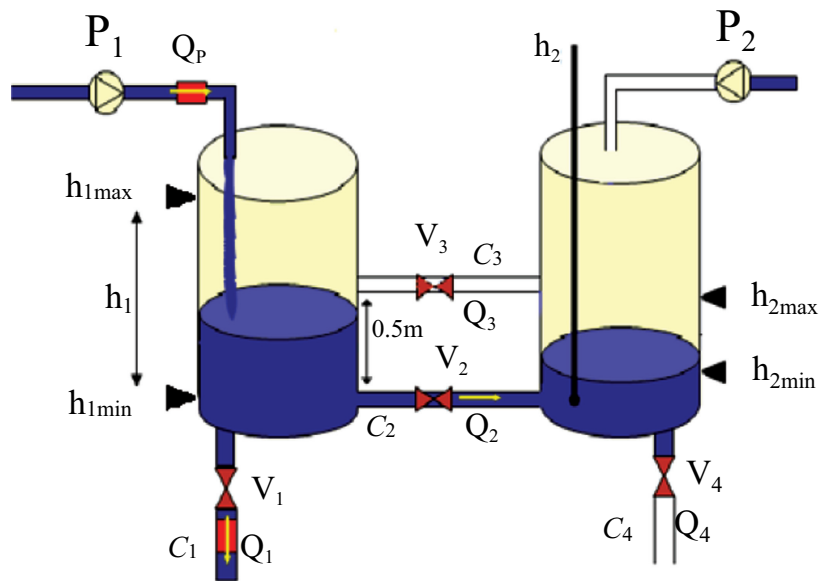


Fig. (7). Hydraulic system.

- Two tanks R_1 and R_2 , with cross sections $S_1 = S_2 = S$;

- Four cylindrical pipes C_1, C_2, C_3 and C_4 have the same cross section A . the two tanks are connected by pipes C_2 and C_3 placed respectively at the levels $h_{12} = 0m$ and $h_{12} = 0.5m$. The pipes C_2 and C_3 are equipped with V_2 and V_3 valves.
- Two valves V_1 and V_4 allowing liquid evacuation for the use;
- Two pumps P_1 and P_2 have the same flow rate Q_p ;
- Four sensors: two sensors measure the flow rates Q_p and Q_{V_1} (flow rate through the valve V_1) and the other ones measure h_1 and h_2 that are the heights of liquid in tanks R_1 and R_2 respectively.

Remark 1. To simplify the study, we consider that the valves V_1, V_2 and V_3 are kept constantly open.

4.1. System Modeling During Normal Operation

The modeling is an important step to understand the system behaviors, we use hybrid automaton which represents an extended version of finite state automaton associated with differential equations. Thus, the overall state of a hybrid automaton, at a given time is defined by a pair (q, x) in which q represents the situation (discrete state) and x the state vector (in the sense of continuous). This global state changes for two reasons:

- The crossing of discrete transition is changed abruptly or directly by the evolution of continuous state. This crossing happens when an appropriate event occurs and/or if a condition becomes true;
- The temporal evolution that affects x following the differential equation associated to the current situation. This situation remains unchanged.

Flow rates expressions given in [7, 12] by *Torricelli law* are:

$$\begin{cases} Q_1(t) = S_1 \sqrt{2g \cdot h_1(t)} ; \\ Q_2(t) = A_2 \text{sign}(h_1(t) - h_2(t)) \cdot \sqrt{2g \cdot |h_1(t) - h_2(t)|} ; \\ Q_4(t) = S_4 \sqrt{2g \cdot h_2(t)} . \end{cases} \quad (1)$$

Q_3 could be calculated using three expressions depending on liquid level in the tanks R_1 and R_2

$$Q_3 = \begin{cases} A_3 \sqrt{2g \cdot (h_1(t) - h_2(t))} , & \text{if } h_1 > h_2 \text{ and } h_2 < h_{12} \\ -A_3 \sqrt{2g \cdot (h_2(t) - h_1(t))} , & \text{if } h_1 < h_2 \text{ and } h_2 > h_{12} \\ A_3 \text{sign}(h_1(t) - h_2(t)) \sqrt{2g \cdot |h_1(t) - h_2(t)|} , & \text{if } h_1 > h_2 \text{ and } h_2 > h_{12} \end{cases}$$

To simplify writing, we rewrite Q_3 by the following expression:

$$Q_3 = A_3 \cdot \text{sign}(H_1(h_1) - H_2(h_2)) \sqrt{2g \cdot |H_1(h_1) - H_2(h_2)|} \quad (2)$$

Where H_1 and H_2 are respectively the non-linear functions of h_1 and h_2

$$H_1(h_1) = \begin{cases} 0 & \text{if } h_1 < h_{12} \\ h_1 - h_{12} & \text{if } h_1 \geq h_{12} \end{cases} \quad (3)$$

$$H_2(h_2) = \begin{cases} 0 & \text{if } h_2 < h_{12} \\ h_2 - h_{12} & \text{if } h_2 \geq h_{12} \end{cases} \quad (4)$$

After developing and linearizing, the following equations can be derived:

$$\begin{cases} Q_1 = A \cdot \sqrt{2g} \cdot \sqrt{h_1} \\ Q_2 = A \cdot \sqrt{2g} \cdot \sqrt{|h_1 - h_2|} \\ Q_3 = A \cdot \sqrt{2g} \cdot \sqrt{|h_1 - h_2|} \\ Q_4 = A \cdot \sqrt{2g} \cdot \sqrt{h_2} \end{cases} \quad (5)$$

Table 1 shows the probable events (spontaneous or controlled) that may be happened [20].

Table 1. Controls and spontaneous events generated by the system.

Notation	Description
ev_1	Event according to opening of valve V_4
ev_2	Event according to closing of valve V_4
$h_1 \geq h_{12}$	Level threshold h_{12} crossed by h_1 in ascending order
$h_1 \leq h_{12}$	Level threshold h_{12} crossed by h_1 in descending order

We consider the following specifications:

- the valves V_1 , V_2 and V_3 are always open,
- the pump is controlled to maintain h_2 in a fixed interval, the pump is switched on when $h_2 = h_{2max}$ ($h_{2max} = 0.1m$). The pump is stopped when $h_2 = h_{2min}$ ($h_{2min} = 0.2m$).
- the valve V_4 can be opened by the operator, but the action is only performed if the liquid level in the tank R_1 is greater than h_{12} . The valve V_4 is held closed when the liquid level in the tank R_1 is less than h_{12} .

In normal operation, only two discrete states are considered: the state of the C3 pipe that can take the terms V (The two levels h_1 and h_2 are lower than h_{12}) or P (at least one level h_1 or h_2 is greater than h_{12}); and the state of the valve V_4 which can take the modality O (opened) or F (closed). The initial state corresponds to discrete state $q_0 = (V; F)$ and the continuous state $x = (0.4 \ 0)^T$. Thus, the system initially starts with the levels h_1 and h_2 greater than h_{12} .

Remark 2. We consider a safety threshold: $h = h_{1min}$ ($h_{1min} = 0.001m$) at which the pump should be started and superior threshold $h_1 = h_{1max}$ ($h_{1max} = 2cm$) above, the pump should be stopped.

Using these behaviors that are developed based on considered specifications, we can represent each behavior of this system by mode i , knowledge that switching from one mode to another is controlled by inequality constraints that depend on h_1 , ev_1 or ev_2 .

Fig. (8) illustrates the hybrid automaton in normal operation.

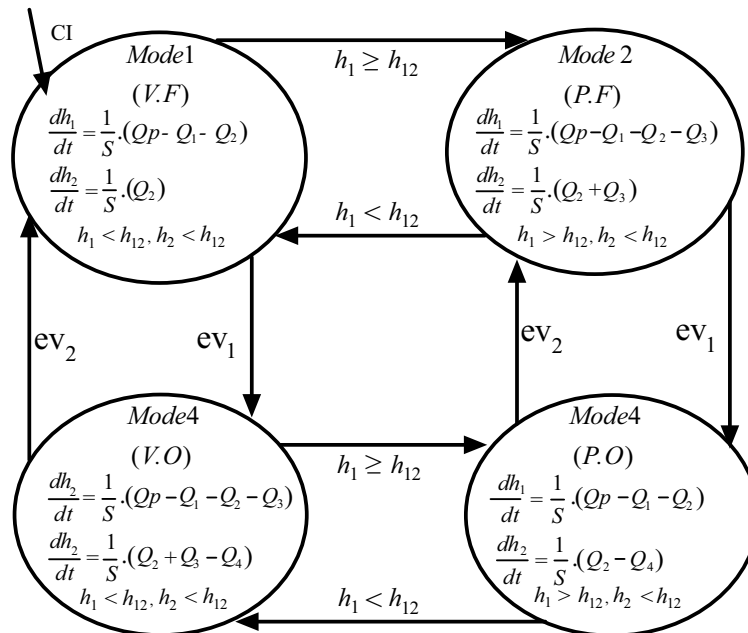


Fig. (8). Hybrid automaton in normal operation.

Fig. (9a, b) illustrates the levels h_1 and h_2 and the evolution of modes respectively.

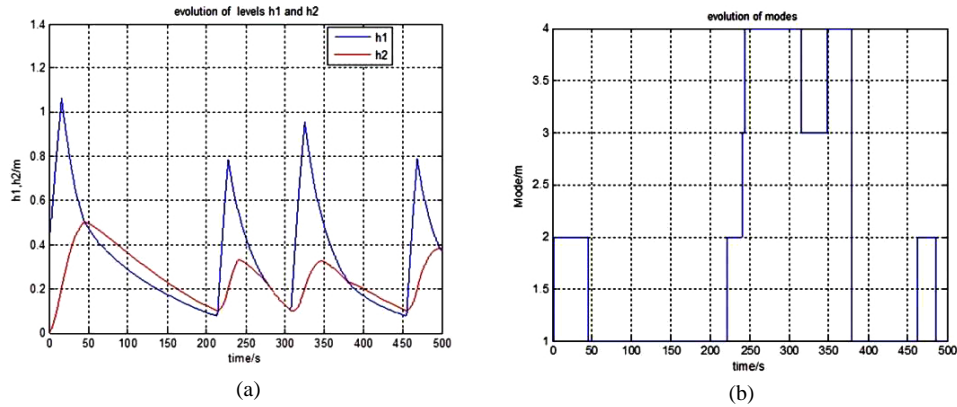


Fig. (9). (a) level of h_1 and h_2 . (b) evolution of modes.

4.2. Modeling of Monitoring System

Based on this part of the algorithm that we proposed previously, in which we have modeled the system behavior during normal and faulty functioning by SHA. This algorithm allows us to extract the guards of input and output transitions of faulty operation. Therefore, we have been able to achieve monitoring system (Fig. 10).

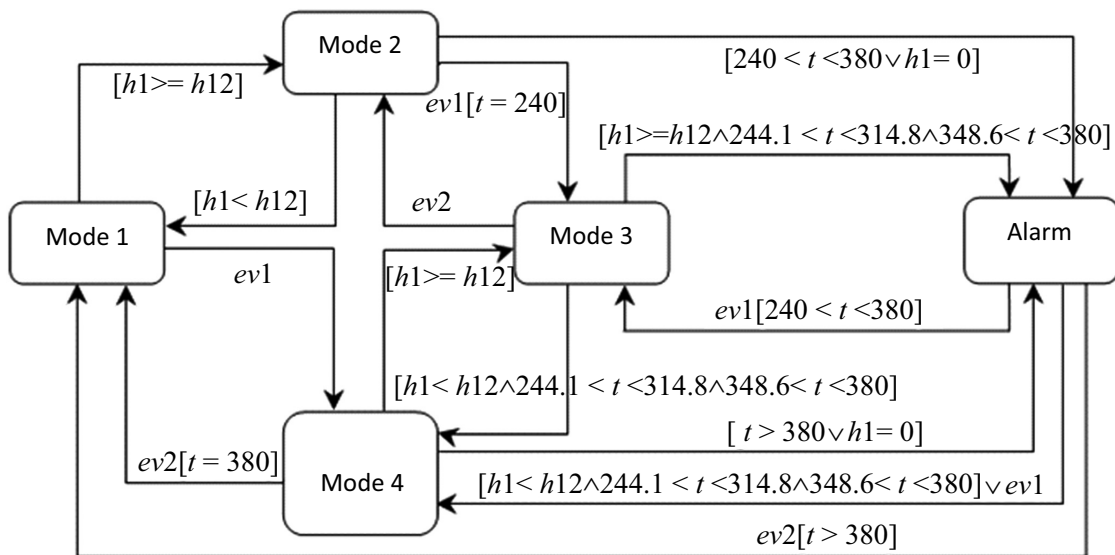


Fig. (10). System monitoring model.

As part of the formalization of our monitoring approach, we consider intermittent and permanent faults, that are presented on Table 2, *vis-à-vis* which we analyzed behavior of our monitoring system.

Table 2. Considered faults in the system.

Faults	Type	Description
d_1	Intermittent	Blocking in opening of valve V_i .
d_2	Intermittent	Blocking in closing of valve V_i .
d_3	Permanent	Final judgment of pump P_i .

4.3. Results

Fault 1: The fault d_1 is characterized by blocking of the valve V_i in opening mode until time $t = 385$ s, opening time exceeds the specified time $t \in [240, 380]$. Fig. (11) shows that an intermittent fault has been announced by the monitoring system (illustrated by the red circle in this figure). The alarm turns on quickly when $t = 380$ s for a few moments to tell us that the system has violated specific time (Fig. 11b).

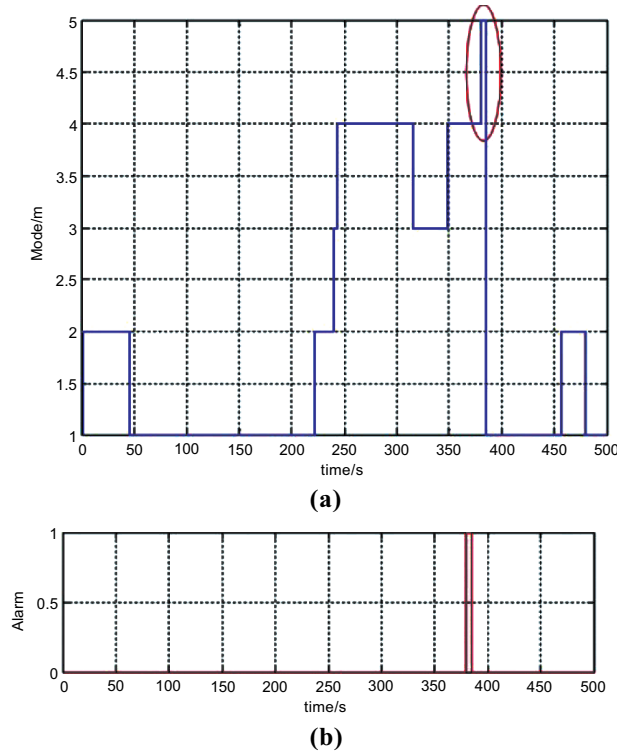


Fig. (11). (a) SHA behavior in the presence of d_3 . (b) alarm behavior.

Fault 2: The fault d_2 is characterized by blocking of the valve V_4 in closing mode until time $t = 243$ s. In this case, an intermittent fault has been again declared by the developed monitoring system which it can be observed in Fig. (12). The alarm turns on quickly when $t = 240$ s for a few moments (Fig. 12b) to tell us that the system has violated specific time.

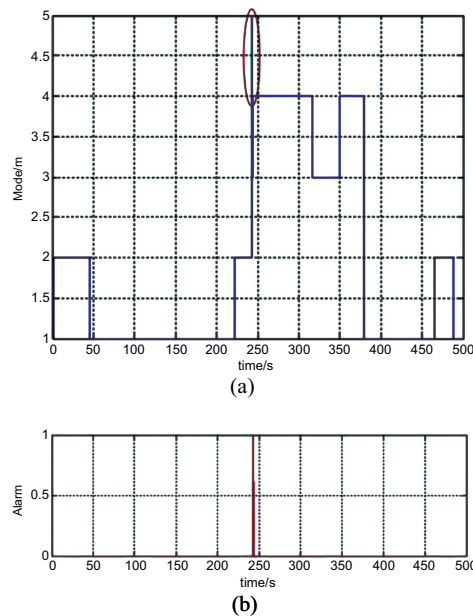


Fig. (12). (a) SHA behavior in the presence of d_2 . (b) alarm behavior.

Remark 3. The two faults d_1 and d_2 do not affect the continuous part of our system, because the system is provided with a control of pump to keep level h_2 between h_{2min} and h_{2max} .

Fault 3: The fault d_3 occurs at Final judgment of pump at time $t = 243$ s. This type of fault affecting the continuous part of the system is shown in Fig. (13a). As it can be observed from Fig. (13), a permanent fault has been correctly detected by the monitoring system (shown by the red circle in this figure). the alarm turns on quickly when $t=243$ s for

an unlimited delay (Fig. 13c) to tell us that system is broken down, Which requires us to repair or to replace the pump.

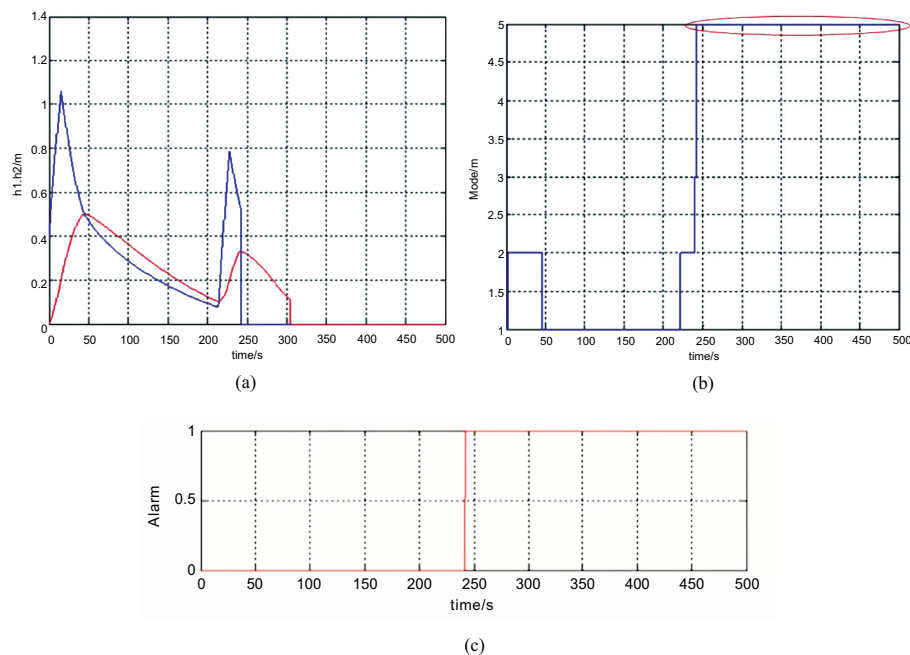


Fig. (13). (a) Continuous evolution in the presence of d_3 . (b) SHA behavior. (c) alarm behavior.

5. CONCLUSION

In this article we proposed a monitoring system for HDS whose model is a stopwatch hybrid automaton. It takes into account the dynamic changes that may appear during the process execution; this change may be definitive or spontaneous. The guards of automaton represent the crossing conditions, these conditions are timed with a timer that calculates elapsed time of each execution cycle. Authorized system behavior (normal) is controlled by variables whose constraints are expressed by inequalities defining the acceptable space of evolution system.

The current work is devoted to develop a stopwatch hybrid automaton structure for a monitoring system able to detect, identify and locate intermittent and/or permanent faults quickly, by combining monitoring approach based on the model presented in this paper with the methods which are cited in section III.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

- [1] R. Alur, C. Courcoubetis, T.A. Henzinger, and P-H. Ho, "Hybrid Automata: An Algorithmic Approach to the Specification and Verification of Hybrid Systems", In: *Hybrid Systems I, Lecture Notes in Science 736*, Springer-Verlag, 1993, pp. 209-229.
- [2] M. Combacau, "Discrete approach of the monitoring of the systems of production", In: *Control of the Risks and Reliability of the Systems of Production.*, Hermes, 2001.
- [3] G. Zwingelstein, "Diagnostic des défaillances. Traité des nouvelles technologies", série Diagnostic et Maintenance, *Hermes (Wiesb.)*, 1995.
- [4] J. Lunze, "Diagnosis of Quantised Systems by Means of Timed Discrete-Event Representations", In: *Lecture Notes in Computer Science*, BerlinSpringer, 2000.
[http://dx.doi.org/10.1007/3-540-46430-1_23]
- [5] X. Koutsoukos, K.X. He, M. Lemmon, and P. Antsaklis, "Timed Petri nets in hybrid systems; stability and supervisory control", *Discrete*

- Event Dyn. Syst.*, pp. 137-173, 1998.
[<http://dx.doi.org/10.1023/A:1008293802713>]
- [6] R. David, and H. Alla, "Discrete, Continuous and Hybrid Petri Nets", *Springer*, 2005.
- [7] V. Cocquempot, T. El Mezyani, and M. Staroswiecki, "Fault detection and isolation for hybrid systems using structured parity residuals", In: *Asian Control Conference*, 2004.
- [8] E.Y. Chow, and A.S. Willsky, "Analytical redundancy and the design of robust failure detection systems", *IEEE Trans. Automat. Contr.*, vol. 29, no. 7, pp. 602-615, 1984.
[<http://dx.doi.org/10.1109/TAC.1984.1103593>]
- [9] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Failure diagnosis using discrete event models", *IEEE Trans. Automat. Contr.*, vol. 4, no. 2, pp. 105-124, 1996.
[<http://dx.doi.org/10.1109/87.486338>]
- [10] S. Tripakis, "Fault diagnosis for timed automata", *Proceeding 7th International Symposium on Formal Techniques in Real-Time and Fault Tolerant Systems* (FTRTFT'02), 2791 of Lecture Notes in Computer Science, pp. 205-224, 2002.
[http://dx.doi.org/10.1007/3-540-45739-9_14]
- [11] S.H. Zad, R.H. Kwong, and W.M. Wonham, "Fault Diagnosis in Discrete-Events Systems: Incorporating Timing Information", *IEEE Trans Automat Contr*, vol. 50, no. 7, 2005.
- [12] T. El Mezyani, "*Méthodologie de Surveillance des Systèmes Dynamiques Hybrides*", M.S. thesis, Université des Sciences et Technologies de Lille, Lille, France, 2005.
- [13] A. Allaham, "*Surveillance des systèmes à événements discrets commandés: Conception et implémentation en utilisant l'automate programmable industriel*", université Joseph Fourier, Grenoble, France, 2008. M.S. thesis, université Joseph Fourier, Grenoble, France, 2008.
- [14] F. Lin, and W.M. Wonham, "Diagnosability of discrete event systems and its applications", *Discrete Event Dyn. Syst.*, vol. 4, no. 2, pp. 197-212, 1994.
[<http://dx.doi.org/10.1007/BF01441211>]
- [15] M. Sampath, R. Sengupta, S. Lafortune, K. Sinnamohideen, and D. Teneketzis, "Diagnosability of discrete event systems", *IEEE Trans. Automat. Contr.*, vol. 40, no. 9, pp. 1555-1575, 1995.
[<http://dx.doi.org/10.1109/9.412626>]
- [16] S.H. Zad, R.H. Kwong, and W.M. Wonham, "Fault diagnosis in finite-state automata and timed discrete-event systems", In: *Proceedings of the 38th IEEE Conference on Decision and Control*, 1999.
[http://dx.doi.org/10.1007/978-1-4471-0543-5_5]
- [17] M.F. Karoui, "*Surveillance des processus dynamiques événementiels*", M.S. thesis, université Joseph Fourier, Grenoble, France, 2011.
- [18] H. Derbel, "*Diagnostic à base de modèles des systèmes temporisés et d'une sous-classe de systèmes dynamiques hybrides*", M.S. thesis, université Joseph Fourier, Grenoble, France, 2009.
- [19] R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.h. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine, "The algorithmic analysis of hybrid systems", *Theor. Comput. Sci.*, vol. 138, pp. 3-34, 1995.
[[http://dx.doi.org/10.1016/0304-3975\(94\)00202-T](http://dx.doi.org/10.1016/0304-3975(94)00202-T)]
- [20] L. Ghomri, "*Synthèse de contrôleur de systèmes hybrides à flux continu par réseaux de Petri hybrides*", M.S. thesis, université Abou-bekr Belkaïd, Tlemcen, Algeria, 2012.