5 th International Conference on Control & Signal Processing (CSP-2017)
Proceeding of Engineering and Technology –PET
Vol.25 pp.93-96

Multiple fault diagnosis method for discrete event systems

Bochra Maaref [#1], Hedi Dhouibi [#2], Zineb Simeu Abazi [*3], , Hassani Messaoud[#4]

[#]*LARATSI, National Engineering School of Monastir, Tunisia*
[1]bochra.maaref.87@gmail.com
[2]hedi.dhouibi@laposte.net
[4]Hassani.Messaoud@enim.rnu.tn
[*]*Grenoble-INP / UJF-Grenoble 1 / CNRS, G-SCOP UMR5272 Grenoble, F-38031, France*
[3]zineb.simeu-abazi@ grenoble-inp.fr

***Abstract*— This paper proposes a diagnostic method based on a timed automata model for discrete event systems. Indeed, the default detection is based on the temporal knowledge of each step of the process. Next, the simple or multiple fault isolation procedure consists of analyzing the fault signatures. A fault (simple or multiple) can be distinguished from each other by a set of variables, called diagnostic variables. The choice of these particular variables is essential to effectively diagnose a system. The results obtained were applied to a Robucar vehicle system.**

***Keywords*— Fault detection, fault isolation, characteristic times, discrete event system, timed automata.**

I. INTRODUCTION

[1] Fault diagnosis (Fault Detection and Isolation) of industrial systems is defined as the operation to detect a fault, locate its origin and determine its causes. Interest in fault diagnosis can be explained by the increasing complexity of industrial systems, which are increasingly demanding in terms of safety constraints, reliability, availability and performance. In fact, the possibility that a system will fail will increase in spite of handling precautions. Consequently, a diagnostic module is necessary to prevent fault propagation and to limit their consequences which can be catastrophic not only economically but also environmental. Several diagnostic approaches have been developed in recent decades and can be categorized into two main families: model-based approaches based on the existence of a model of the system to be monitored and non-model approaches based on analysis of monitoring variables and human expertise for identifying the exact cause of a failure. Discrete event systems (DES) represent the class of dynamic systems whose states and transitions are modeled discretely, for example by a finite state automaton. Any behavior of the system is then represented by a path on this automaton. The notion of diagnosis of DES has been introduced over the years in several studies [2]-[5]. In this context, numerous researches have been carried out in the field of the diagnosis of DES. In some applications, time information is essential and must be explicitly taken into account by the model. Models that have this characteristic are called timed. In the description of DES, these are models where time is deterministic (temporal Petri nets, timed automata) and models where time is random (Markov chain). We present a single tool among the modeling possibilities it is the timed automata [6], [7].

In this paper, a model-based method for discrete event systems will be presented. This approach is based on the use of characteristic system times. Our goal is to design a diagnostic system, called diagnoser, which allows analyzing, detecting and locating a fault in a system. And more precisely we have focused our research on the defects coming simultaneously from the sensors and actuators of the system. The method used is based on the use of timed automata (TA).

In this first part, the subject and context of the paper have been introduced. The second part deals with the tools used to analyze and model a system. In a third part, the measurements necessary for the construction of the diagnoser are detailed. The results and the validation of our method are presented in the fourth part. A conclusion is given at the end of the paper.

II. MODELING BY TIMED AUTOMATA

The modeling approach to which we have been interested in our work is a model-based method. The principle of model-based methods is based on a comparison of the behavior predicted by the model with the actual observed behavior of the system. Any deviation between these two behaviors will be synonymous with failure. The various steps of modeling a system are represented by Fig. 1.

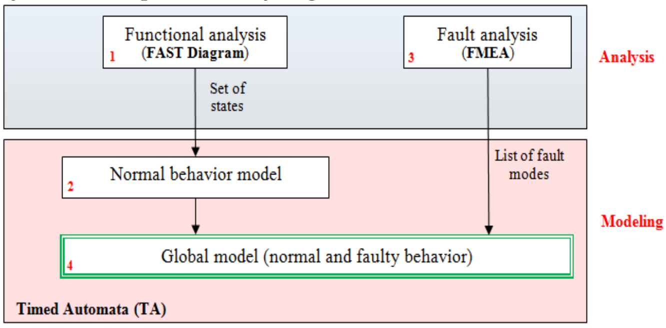

Fig.1. Different steps of modeling by Timed Automata (TA)

The first part of this work consists in analyzing the system by a functional analysis (1) using FAST diagram (**F**unction **A**nalysis **S**ystem **T**echnique). This analysis makes it possible to construct a model of normal system behavior (2) based on the use of timed automata. Then, in order to model the

ISSN 2356-5608

consequences of the failures on the system and the analysis of the defects (3), the list of failure modes is identified by the use of the FMEA (**F**ailure **M**odes and **E**ffects **A**nalysis). Finally, the global model (4) of the system is constructed.

A. Example: A Robucar vehicle

A "Robucar" electric vehicle (Fig.2) is considered for use in different types of environments to perform specific missions such as demining or guided tours.

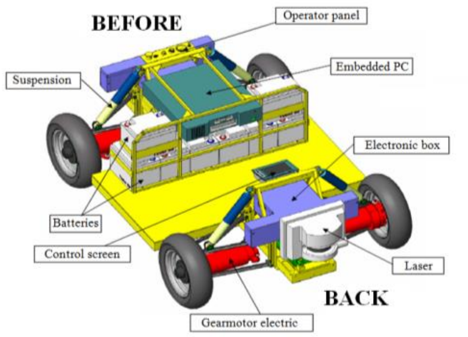

Fig.2. Scheme of ROBUCAR

This vehicle moves without operator, but guiding by variables (white lines on the ground, internal programming of a road ...). An on-board computer gives instructions to the actuators (engines) according to the values of the sensor (accelerometer, gyroscope, laser ...) and the program it contains. In this example, the route of the system (Fig.3) consists of four phases where the Robucar rolls and two breaks (50 s).

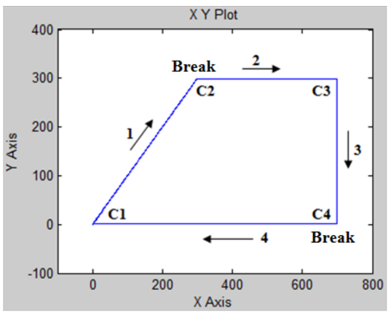

Fig.3. Circuit diagram that the Robucar must follow

Four sensors (C1 to C4), placed on the road, make it possible to know if the vehicle is present at the end of a stage (1 if present, 0 other). The vehicle moves at a constant speed (5 m / s). Interesting variables are the speed and steering angle of the vehicle. In the Cartesian coordinates (Fig.4), the dynamic equations are the following:

$$\begin{cases} x(t) = \int_0^t v\cos(\alpha)d\tau \\ y(t) = \int_0^t v\sin(\alpha)d\tau \end{cases} \quad (1)$$

Where v: the speed of the vehicle; α: the steering angle of the vehicle.

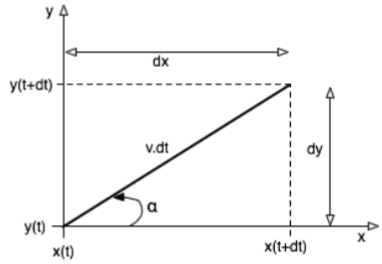

Fig.4. Angle and speed in Cartesian coordinates

The functional analysis for the electric vehicle "Robucar" will be carried out for a quarter of the system. The FAST diagram of the subsystem is shown in Fig.5. For each component, it is now easy to know its role and associated function.

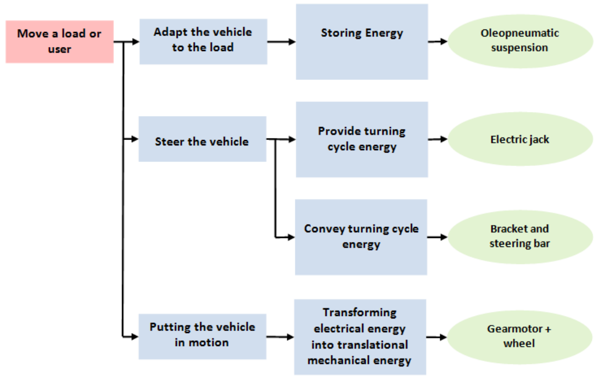

Fig.5. FAST diagram of the Robucar subsystem

The control variables for the electric vehicle "Robucar" are speed and angle. From these values, the dynamics calculates the coordinates (x, y) of the vehicle (system variables). In this way the movement of the Robucar on the ground was modeled.

In the case of this system, there are 18 failures related to the components of the subsystem and 8 related to the addition of the five sensors. Table 1 shows some of the lines of the FMEA.

TABLE I
SOME LINES OF THE FMEA VEHICLE ROBUCAR

n° FM	*Item*	*Function*	*Failure Mode*	*Causes*	*Effects*
1	Axle-Wheel block	Transform electric energy in mechanical energy	Damaged wheel	nails, pebble, tire wear, ...	-Transmitted mechanical energy does not correspond to request -Altered wheel motion -Altered vehicle direction -Reduced speed -Important energetic consumption
11	Steering block	Convey turning cycle energy	Crooked steering bar	Too high pressure on the bar	Insufficient turning cycle energy
18	Power	System power	No power	Short circuit	No power in the system, no motion

Regarding the injection of faults that the system is modelling correctly in its normal operation, it will be necessary to model the consequences of the failures on the speed and the direction of the vehicle.

With regard to the effect of the injection of simple defects and multiple defects in the model, the problem is divided into

ISSN 2356-5608

5 th International Conference on Control & Signal Processing (CSP-2017)
Proceeding of Engineering and Technology –PET
Vol.25 pp.93-96

two parts: the effects on the sensor values and on the Speed and Angle control variables.

For faults affecting control variables (speed and angle). We consider that the speed has several degrees of alteration: low increase, no change, low decrease, high decrease and no speed. In the same way, the angle can be unchanged (from the normal route), low deflected, high deflected and no change of direction.

As previously stated, there are 26 defects related to the components of the subsystem and sensors and 325 multiple defects ($C_{26}^{2} = 325$). Table 2 and Table 3 show some lines of simple and multiple faults.

TABLE III
SOME LINES OF SIMPLE FAULT

n° SFM	*Item*	*Failure Mode*	*Speed*	*Angle*
1	**Axle-Wheel block**	**Damaged wheel**	**High (-)**	**High**
4	Axle-Wheel block	Engine blocked	Low (-)	High
6	Axle-Wheel block	Out-of-order suspension	High (-)	Low
17	Power	Empty battery	Nothing	No change of direction
19	Sensors	C1 remains off	Unchanged	No change of direction

TABLE IIII
SOME LINES OF MULTIPLE FAULT

n° MFM	*Failure Mode*	*Speed*	*Angle*
1 &9	**Damaged wheel & Twisted bracket**	**High (-)**	**High**
12&5	Broken steering bar & Suspension seized	Low (-)	Low
1&10	**Damaged wheel & Broken bracket**	**High (-)**	**High**
3&13	Engine that runs continuously at maximum & Steering cylinder seized	Low (+)	High
1&19	**Damaged wheel & C1 remains off**	**High (-)**	**High**

Finally, the global model of the system has been successfully built: the system is modeled in its normal behavior but also when a failure occurs. Now we will use this model to diagnose failures in the system. With only the behavior of the system and its characteristic times, the objective is to detect and isolate the defect.

III. CONSTRUCTION OF DIAGNOSER

In this section, detection and isolation are explained. Then, the concept of fault signature is introduced, which is essential in the method of diagnosis in this paper. Finally, the construction of diagnostic is explained for the example of the Robucar vehicle.

A. Characteristics time of the system

The diagnostic method proposed in this paper is based on the characteristic times of the system. If these times are not respected, a fault is detected. Fig. 6 shows the behavior of the system over an operating cycle.

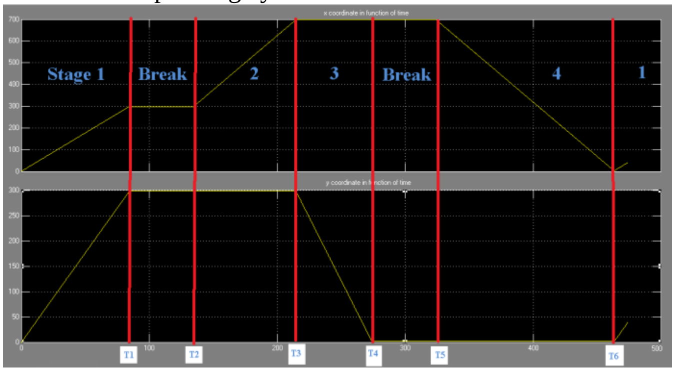

Fig. 6. Characteristic times of the vehicle on the road

B. Detection principle

When a failure occurs in the system, the first step is to detect it. It is possible by comparing the system with its model in normal behavior for the same command. If a difference between the two behaviors is found, a fault is detected.

C. Isolation principle

Once the fault has been detected, it must be isolated. First, the defective component must be found: it is the location. Then, a component can be associated with several failures. So all these defects are studied to determine which is occurring in the system: it is the identification.

D. Fault Signature Analysis

In order to isolate from a fault, the method described in this paper uses the concept of fault signature. It considers that a fault can be distinguished from each other by the values of certain system variables. A fault can be represented as a diagnostic matrix.

In Fig.7, you can see an example of a diagnostic matrix. S1 is the diagnostic matrix for a variable v1 (for example the speed) which can take three values X1 to X3 (for example a low increase, no change ...), in a system with four failures f1 to f4. If v1 can take the value Xj in the presence of the defect fi, the matrix element sij = 1 (i^{th} line, j^{th} column), else sij = 0. Then the fault signature fi corresponds to the i^{th} row of the matrix.

In addition, it is interesting to use several system variables in order to rethink the diagnosis. Indeed, in the example of Fig. 7, if v1 = X3, the possible faults are f1 and f3. But to distinguish the two faults, we must add a new variable that does not take the same value for the two defects. By adding

ISSN 2356-5608

5 th International Conference on Control & Signal Processing (CSP-2017)
Proceeding of Engineering and Technology –PET
Vol.25 pp.93-96

the variable v2 (for example the angle) and its diagnostic matrix, it becomes possible to differentiate the two failures.

Thus, the principle remains the same for multiple defects.

$$S_1 = \overbrace{\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}}^{X_1\ X_2\ X_3} \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{matrix} \qquad S_2 = \overbrace{\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}}^{Y_1\ Y_2\ Y_3} \begin{matrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{matrix}$$

Fig.7. Example of diagnostic matrices for variables v1 and v2

But in the case where the simple defect having the same signature as the multiple defect (example, Table 2 and Table 3, the default signature n°1 having the same signature as the multiple defect n° 1&9), it is necessary first of all to isolate the simple fault (fault n° 1). As soon as this fault is isolated and repaired, if another fault does not detect, then the fault which has been isolated is a simple fault, but in the case of multiple faults, the second fault (fault n°9) is detected and isolated automatically.

And in the case where the simple defect having the same signature as the multiple defects that have a multiple signature (example, multiple defects, in Table 3, which have in red, n°MFM: **1**&9; **1**&10; **1**&19), it is first necessary to isolate the defect which is common in a class where all the defects have the same signature (fault n°**1**). As soon as this fault is located and repaired, the second fault (fault n°9 or fault n°10 or fault n°19) is automatically detected and isolated (in case of multiple faults).

In conclusion, the system variables used for the diagnosis must be chosen in order to correctly isolate all the failures, Fig.8.

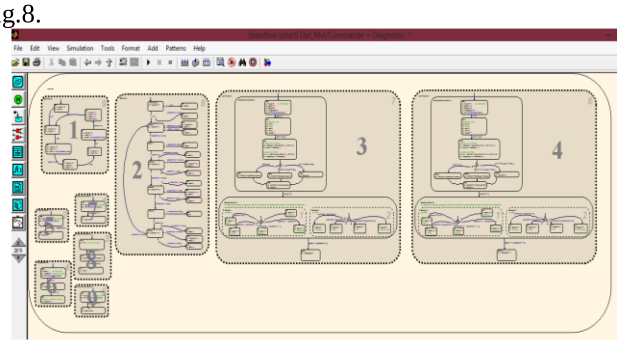

Fig.8. Diagnostic construction by TA with Matlab / Stateflow

1: control model - 2: detection - 3, 4: variable calculation - 5, 6: injection, detection times – 7, 8, 9: isolation time

IV. Simulation Results

In order to test the effectiveness of the proposed diagnostic approach, we injected faults randomly. Table 4 shows the simulation results obtained.

For single and multiple faults, the diagnosis is able to distinguish all faults and returns a single fault (single or multiple).

This means that the variables used for diagnosis are fairly accurate. About the speed of diagnosis, this method depends on the injection time. If the injection occurs at the beginning of a new state, detection may take a long time: the failure is detected when the state is changed in normal behavior or in the worst case. However, the isolation is quite fast. This essentially depends on the calculation of the variable values.

TABLE IVV

Simulation results of the diagnostic approach proposed for several faults injections

	Fault injection number	Injection time	Detection time	Isolation time	Diagnostic time	Results (failure number)
Simple Fault	(4)	200	215.2	217.2	17.2	(4)
	(1)	175	215.4	217.4	42.4	(1)
Multiple Fault	(3&13)	250	274.8	276.8	26.8	(3&13)
	(1&9)	197	215.4	**Isolation time 1** 217.4	20.4	(1)
				Isolation time 2 369.8 (with 132.6s is the repair time of the first fault)	172.8	(9)

V. Conclusion

In this paper, we discussed the problem of fault diagnosis on a discrete event system using the timed automata. The proposed approach makes it possible, using signature analysis, to solve the problem of identifying not only simple defects but also multiple defects. This approach is based essentially on the characteristic times of a system. Nevertheless, some improvements can be made: in order to avoid the uncertainty of the diagnosis, the choice of the system variables used for the diagnosis could be studied. By adding some sensors to a specific location on the system, it might be possible to add diagnostic variables and analyze the results.

References

[1] A. PHILIPPOT, M. SAYED-MOUCHAWEH, & V. CARRÉ-MÉNÉTRIER. Unconditional decentralized structure for the fault diagnosis of discrete event systems. IFAC Proceedings Volumes, vol. 40, no 6, p. 67-72, 2007.

[2] E.Gascard, & Z. Simeu-Abazi. Modular modeling for the diagnostic of complex discrete-event systems. IEEE Transactions on Automation Science and Engineering, 10(4), 1101-1123, 2013.

[3] P. Marangé, A. Philippot, J.-F. Pétin, and F. Gellot, "VERIFICATION DE LA DIAGNOSTICABILITE PAR MODEL-CHECKING," in MOSIM, 2014.

[4] M. Sayed-Mouchaweh and E. Lughofer, "Decentralized fault diagnosis approach without a global model for fault diagnosis of discrete event systems," International Journal of Control, pp. 1-14, 2015.

[5] J. Zaytoon and S. Lafortune, "Overview of fault diagnosis methods for Discrete Event Systems," Annual Reviews in Control, vol. 37, pp. 308-320, 2013.

[6] S. Tripakis. "Fault diagnosis for timed automata." FTRTFT. Vol. 2469. 2002.

[7] J. Bengtsson, et W. 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.

ISSN 2356-5608