Research Article

Detection and isolation of temporal drifts in manufacturing systems with observers and control charts

Dimitri Lefebvre[1] · Edouard Leclercq[1]

Received: 5 December 2019 / Accepted: 29 May 2020 / Published online: 13 June 2020

Abstract

Faults detection and isolation is a major issue for a large variety of manufacturing systems because such systems combine operations performed by several machines and use some common resources. The failure of a single machine slows down or even stops the whole system. This paper is about some specific insidious faults—namely temporal drifts—that affect manufacturing systems with mass production. Such temporal drifts are studied in a probabilistic setting by modeling the operation durations with random variables and by considering significant variations of the probability density function of these variable as the faults to be detected and isolated. Labeled stochastic timed Petri nets are used to model the considered systems and faults. A new class of observers—namely elementary trajectory observers—that estimate characteristic segments of the state trajectory is developed. Then, moving average control charts and observers are combined in order to detect the variations of the firing durations mean values. The proposed analysis is suitable to detect variations in the mean value of the operation durations when parameters vary gradually. It is also suitable to isolate the faults thanks to a set of minimal-size elementary trajectories consistent with the successive measurements. Detection and isolation decision functions are proposed, and the approach is illustrated on an example.

Keywords Discrete event systems · Temporal faults · Labeled timed Petri nets · Observers · Control charts

1 Introduction

Fault detection and diagnosis is an important challenge for many systems, in particular manufacturing systems. On the one hand, fault detection consists in generating an alarm once a fault is detected based on the available measurements. On the other hand, fault diagnosis consists in isolating the detected fault among a set of fault candidates [1]. Fault detection and diagnosis methods may be separated into model-based and data-based approaches. Model-based approaches have been intensively studied for continuous time systems [2] and also for discrete event systems [3, 4]. Among model-based approaches, numerous methods are based on the design of an observer. The aim of such an observer is to estimate the system state. In the context of faults diagnosis, the state estimation is compared to the measurements to track the occurrence of faults. With continuous time systems, observers serve to generate residual signals that reveal the occurrence of faults when these signals become significantly different from zero [2]. With discrete event systems, the faulty states are mainly incorporated in the model [1] and the observer is transformed into a diagnoser that tags each possible state as normal or faulty. Consequently, state estimation leads directly to fault diagnosis. Diagnosis methods based on observers have been first developed for finite state automata [5], stochastic automata [6], and timed automata [7]. Then, methods have been extended to Petri nets [8, 9]. The previous methods are mainly devoted to logical faults (i.e., some unexpected events). Among data-based

✉ Dimitri Lefebvre, dimitri.lefebvre@univ-lehavre.fr; Edouard Leclercq, edouard.leclercq@univ-lehavre.fr | [1]GREAH, Univ. Le Havre Normandie, 75, Rue Bellot, 76600 Le Havre, France.

SN Applied Sciences (2020) 2:1211 | https://doi.org/10.1007/s42452-020-2981-z

approaches, the use of statistical control charts has been successfully used in many application domains and in particular to detect and diagnose operation faults in manufacturing systems [10–12]. One limitation of such approaches is that they consider the systems according to an atomic point of view and consequently focus on specific operations. But, the advantages of such methods are efficient for temporal faults (i.e., unexpected changes in the occurrence frequency of some events). One contribution of this work is to combine a discrete event observer with control charts in order to reveal timed drift (due, for example, to unexpected delays) in timed discrete event systems.

From a high-level point of view, manufacturing systems are basically modeled as discrete event systems (DES) that are pure logical models [2, 3]. In order to incorporate temporal specifications [13], DES can be enriched by adding time stamps or some information that represent the temporal evolution of the systems. Petri nets (PN) have demonstrated their ability to model such specifications and present numerous advantages. They are ready to perform qualitative and quantitative analysis of the systems thanks to their underlying mathematical structure; they can be directly converted into simulation models; in addition, they are graphical, easy to develop, extend, and offer a good understanding of the dynamic behavior of the systems. In particular, activities, resources, and constraints of a manufacturing system can be represented in a single consistent formulation. In the context of manufacturing systems, a large variety of PN subclasses have been listed and discussed, in particular to solve scheduling problems [14]. Some examples of these subclasses of models are systems of simple sequential processes with resources (S3PR) [15], systems of sequential systems with shared resources (S4R) [16] proposed for job-shop problems (that consist of several operations with total precedence constraints), and sets of simple open processes with resources (S2OPR) [17] proposed for open-shop problems (that consist of several jobs with full routing flexibility within the operations). In order to add flexibility at the operation level, colored Petri nets have been also used to model manufacturing systems [18], in particular for control applications [19]. A few applications are also devoted to diagnosis issues [20], but the extension of colored nets applied to temporal faults remains to the best of our knowledge an open question.

In this article, we aim to apply the principles of statistical control charts to timed DES in order to extract significant information about potential faults from heterogeneous measurements collected in different parts of the system. In particular, we propose to use a moving average (MA) control chart to detect and diagnose faults of manufacturing systems that are modeled as stochastic DES with labeled timed Petri nets (LTPN) [8]. LTPN are characterized by deterministic timed transitions that fire after constant durations and also stochastic timed transitions that fire according to probability density functions (PDF) of finite supports. LTPN encode the system sensors as a measurement function that define the measurements collected when observable events occur. The considered faults correspond to significant variations of the PDF supports that define the stochastic firing durations. For detection and isolation purposes, MA control charts [21] are combined with observers that compute the set of trajectories consistent with the observation collected thus far. For this purpose, the MA control charts smooth the successive measurements and compare these measurements with some thresholds. In particular, the proposed observers provide a set of minimal-size elementary trajectories (MSET) consistent with each new measurement. The MSET are also used for fault isolation (or diagnosis) by identifying the suspicious transitions for which the mean firing durations are outdated. Detection and isolation functions are proposed as a result of the approach. This approach is applicable when large sets of transition firing durations are collected. In contrast with many other diagnosis approaches, main advantages of the proposed approach are: (1) to detect and isolate temporal faults that only affect the mean duration of some activities (such faults are more or less undetectable to the main existing diagnosis methods as in [5–8, 22]); (2) to be applicable with a large variety of time processes; (3) to be suitable also for slow deviations and gradual drifts; (4) to reduce the number of sensors and consequently the cost of the sensoring (compared to approaches used at the level of the workstations): For an appropriated sensor configuration, it becomes possible to detect various faults; (5) to avoid the computation of the trajectory probabilities that is very expensive in time and space [23]. The current work continues our preliminary study on slow deviations of firing durations [24]. More precisely, in [24] we have proposed a simple method to use control charts for manufacturing systems based on the decomposition of the system with some specific paths. The main limitation of that former approach is that it is only suitable for 1-bounded nets (i.e., systems where a single operation is performed at each time and where multiples products cannot circulate simultaneously). The main contribution of the present paper is to introduce a new class of observers that track the recent trajectories in the state space for system where several operations and products may be simultaneously considered.

The rest of this document is organized as follows: In Sect. 2, tools and useful definitions are introduced. In Sect. 3, the working assumptions and the model of the temporal faults are detailed. Section 4 provides a detailed explanation of the observer design. Section 5 describes the detection and diagnosis functions used to generate alarms and to isolate the faulty transitions. Section 6 sums up conclusion and future works.

2 Definitions and notations

2.1 Notations

Table 1 explains the notations and acronyms used in the rest of the paper.

2.2 Timed Petri nets

A PN structure is defined as PN $= \langle \boldsymbol{P}, \boldsymbol{T}, W_{pr}, W_{po} \rangle$, where $\boldsymbol{P} = \{P_1, \ldots, P_n\}$ is a set of n places, $\boldsymbol{T} = \{T_1, \ldots, T_q\}$ is a set of q transitions, $W_{pr} \in (\mathbf{N})^{n \times q}$ and $W_{po} \in (\mathbf{N})^{n \times q}$ are the pre- and post-incidence matrices (**N** is the set of nonnegative integer numbers), and $W = W_{po} - W_{pr}$ is the incidence matrix. $\langle$PN, $M_I\rangle$ is a PN system with initial marking M_I, and $M \in (\mathbf{N})^n$ represents the PN marking vector corresponding to the number of tokens in each place. A transition T_j is enabled at M if $M \geq W_{pr}(:, j)$, where W_{pr} (:, j) stands for the column j of matrix W_{pr}. When T_j is enabled at M, we write $M[T_j\rangle$, and then, T_j may fire. When T_j fires once, the marking varies according to $\Delta M = M' - M = W(:, j)$. This is denoted as $M[T_j\rangle M'$, and $T(M, M') \subseteq \boldsymbol{T}$ is defined as the subset of transitions such that $M[T\rangle M'$. If σ is a sequence of several firings, feasible at M, then one can write $M[\sigma\rangle M'$ and $X(\sigma)$ refers to the firing count vector of σ. For systems with a finite number N of states, R and G are, respectively, the set of reachable markings and the generator matrix (i.e., G is a matrix of dimension $N \times N$ such that the entry $g(M, M') \in G$ is the transition T such that $M[T\rangle M'$) of the reachability graph of the net system $\langle$PN, $M_I\rangle$.

As far as time is considered, the basic logical PN model can be extended to include the time stamps. Time is measured with time units (TU) and can be associated with the duration of the transitions firing or with the sojourn of the tokens in the places. In this paper, the time is associated with the transitions and we refer to such extension of PN as timed-transition Petri nets (TPN): The firing of each transition T occurs after a time d that is either deterministic

Table 1 Abbreviations and notations

Abbreviation	Description
DES	Discrete event system
PN	Petri net
TPN	Timed-transition Petri net
LTPN	Labeled timed-transition Petri net
ETO	Elementary trajectory observer
PDF	Probability density function
MSET	Minimal-size elementary trajectories
MA	Moving average
Notation	Description
$\boldsymbol{P}$	Set of Petri net places
$\boldsymbol{T}$	Set of Petri net transitions
$\boldsymbol{E}$	Set of observable labels
W_{po}, W_{pr}, W	Post-, pre- and incidence matrices
M	Marking vector
M_I	Initial marking
Ω	Weight vector with firing probability of each transition
L	Labeling function
σ	A given sequence
(M_O, M_D, σ)	A given trajectory
UNXPL	Set of unexplored trajectories already obtained
R_{obs}	Set of ETO states
G_{obs}	Generator matrix of the ETO
S	A set of trajectories that form an ETO state
$S_O(S)$	Set of the origin markings to state S
$S_D(S)$	Set of the destination markings from state S
N'_{ET}	Number of MSET in all states of the ETO

(d has a constant value) or stochastic (d is a random variable (RV) with a PDF $f(d)$) [25]. When d is deterministic, its value is either strictly positive or can eventually be zero. In that case, the firing is said to be immediate; in the other cases, it is said to be delayed. When d is stochastic, the firing times are distributed according to an arbitrary PDF that is assumed to have a finite support. In addition, this support is known for each transition. In the next, we will consider TPN with stochastic firing durations with two particular types of PDF: uniform PDF on finite support (Fig. 1 left) and symmetrical triangular PDF (Fig. 1 right) defined, respectively, with Eqs. (1) and (2):

$$\begin{aligned} &f(d) = 1/(b-a) && \text{if} \quad d \in [a,b] \\ &f(d) = 0 && \text{otherwise} \end{aligned} \tag{1}$$

$$\begin{aligned} &f(d) = 4/(b-a)^2 \cdot (d-a) && \text{if} \quad d \in [a,(a+b)/2] \\ &f(d) = 4/(b-a)^2 \cdot (b-d) && \text{if} \quad d \in [(a+b)/2,b] \\ &f(d) = 0 && \text{otherwise} \end{aligned} \tag{2}$$

The time semantic of the considered TPN [26, 27] is completed by defining the server, choice, and memory policies:

a. The servers are single server.
b. The choice policy is a preselection policy. Such a policy is used in case of effective conflicts. In such situations, the next transition is randomly chosen from all currently enabled transitions according to a uniform PDF and a set of weights ω_j, $j=1,\ldots,q$ associated with the transitions. The weight vector $\Omega=(\omega_j)\in(\mathbf{R}^{+*})^q$ ($\mathbf{R}^{+*}$ is the set of strictly positive real numbers) determines the firing probability of each transition. For simplicity and without any loss of generality, $\omega_j=1$, $j=1,\ldots,q$, in the rest of the paper and the firing probability of the transitions in conflict is identical.
c. The enabling memory is a memory policy. With such a policy, at each firing, the residual durations associated with still enabled transitions are decremented and the residual durations associated with disabled transitions are reset.

A *timed firing sequence* is defined as $\sigma=T(j_1,t_1)\,T(j_2,t_2)\ldots T(j_h,t_h)$ with $T(j_k,t_k)\in\boldsymbol{T}$, $k=1,\ldots,h$ being the transitions that consecutively fire in the sequence σ. The integers j_k are the indexes of the transitions that successively fire, the times t_k are the time stamps of the successive firings and h is the length of σ. In addition, *a timed trajectory* is defined as $(M(t_0), M(t_h), \sigma)=M(t_0)[T(j_1,t_1)\rangle\, M(t_1)\ldots[T(j_h,t_h)\rangle\, M(t_h)$, from marking $M(t_0)$ at time t_0 to marking $M(t_h)$ at time t_h according to the timed firing sequence σ. Making abstraction of the timing information, a *firing sequence* is defined as $\sigma=T(j_1)\,T(j_2)\ldots T(j_h)$ with $T(j_k)\in\boldsymbol{T}$, $k=1,\ldots,h$ and *a trajectory* is defined as $(M(0), M(h), \sigma)=M(0)[T(j_1,1)\rangle\, M(1)\ldots[T(j_h,h)\rangle\, M(h)$.

Example 1 Figure 2 is an example of TPN with a set of places $\boldsymbol{P}=\{P_1,\ldots,P_8\}$ and a set of transitions $\boldsymbol{T}=\{T_1,\ldots,T_6\}$. Matrices W_{PR}, W_{PO} are both of dimensions 8×6. For example, $W_{PR}(1,1)=W_{PR}(7,1)=W_{PO}(2,1)=1$ describes how transition T_1 is connected to the rest of the net. The initial marking M_I in this example is such that $M_I=\left(n_1\;0\;0\;n_2\;0\;0\;r_1\;r_2\right)^T$ with $n_1, n_2, r_1, r_2\in\mathbf{N}^*$ (the set of strictly positive integer numbers). The transitions T_1 and T_4 are enabled at M_I. The firing durations in this TPN are stochastic and defined with a set of uniform PDF of support $[a_i, b_i]$, $i=1,\ldots,6$. The values of a_i and b_i are reported

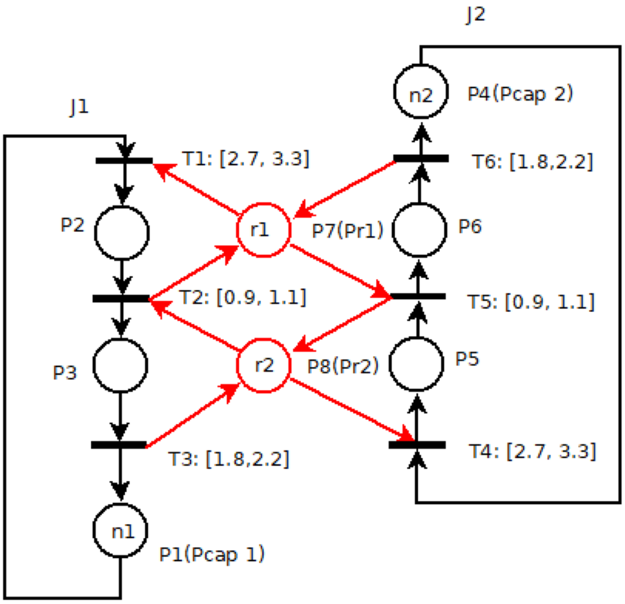


Fig. 2 An example of manufacturing system modeled with a TPN

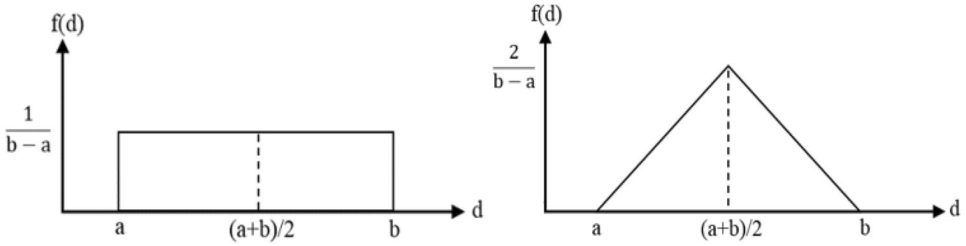


Fig. 1 PDF of the transition firing durations: bounded uniform PDF (left); symmetrical triangular PDF (right)

in Fig. 2 so that "T_1 : [2.7, 3.3]" means that transition T_1 needs a time d_1 that is uniformly distributed within the time interval [2.7, 3.3] before it fires. For the values $n_1 = 1$, $n_2 = 1$, $r_1 = 2$, and $r_2 = 2$, the system owns nine reachable markings and no deadlock. Increasing n_1, n_2, r_1, r_2 increases the number of reachable markings. When $r_1 < n_1 + n_2$ or $r_2 < n_1 + n_2$, deadlocks appear. For $n_1 = 1$, $n_2 = 1$, $r_1 = 2$, and $r_2 = 2$, an example of timed firing sequence enabled at M_I is $\sigma = T(4, 2.8)\ T(1,3.1)\ T(5,3.8)\ T(2,4.2)$. The corresponding timed trajectory is $(M(0), (4.2), \sigma) = M(0)\ [T(4, 2.8)\rangle\ M(2.8)\ [T(1, 3.1)\rangle\ M(3.1)\ [T(5,3.8)\rangle\ M(3.8)\ [T(2,4.2)\rangle\ M(4.2)$ with $M(0) = M_I = (1\ 0\ 0\ 1\ 0\ 0\ 2\ 2)^T$ and $M(4.2) = (0\ 0\ 1\ 0\ 0\ 1\ 1\ 1)^T$.

2.3 Labeled timed Petri nets

The transitions of the net are basically separated into observable transitions that deliver a label and silent ones that do not. L: $\boldsymbol{T} \rightarrow \boldsymbol{E} \cup \{\varepsilon\}$ is a labeling function that assigns a label to each transition where $\boldsymbol{E} = \{\boldsymbol{e_1},\ldots,\boldsymbol{e_{qo}}\}$ is the set of q_O labels that are assigned to observable transitions and $\boldsymbol{\varepsilon}$ is the null label that is assigned to the silent ones.

Labeled timed Petri nets (LTPN) are finally defined as <PN, PDF, Ω, L, M_I> where PN is a Petri net structure, PDF is a set of density probability functions, Ω is a set of weights, L is the measurement function that defines the sensor configuration and M_I is the initial marking. The measurement of any trajectory is obtained according to the selected sensor configuration. The function L collects the K successive dated measurements of a timed trajectory (σ, M_I) over time interval [t_0, t_p]. These measurements are organized in a measured trajectory: TRo $= L(\sigma, M_I) = (e_1, \tau_1) \ldots (e_K, \tau_K)$ that will be analyzed in the next section to detect and isolate temporal faults.

Example 2 Consider again the TPN in Fig. 2 with $n_1 = 1$, $n_2 = 1$, $r_1 = 2$, and $r_2 = 2$. Assume that transition T_3 delivers a label a when T_3 fires and similarly that transition T_6 delivers b when T_6 fires. The other transitions are assumed to be silent. Consequently, the labeling function is defined as $L(T_3) = a$, $L(T_6) = b$, and $L(T_j) = \varepsilon$, for $j = 1, 2, 4, 5$. With the addition of the labeling function, the TPN is now a LTPN. Figure 3 reports the observations captured by the labeling function when the LTPN works during a period of 100 time units (TU). According to the labeling function, only the complete execution of the cycles $\{T_1, T_2, T_3\}$ and $\{T_4, T_5, T_6\}$ is detected. Figure 3 top reports the number of executions of cycle $\{T_1, T_2, T_3\}$ in red light (resp. cycle $\{T_4, T_5, T_6\}$ in blue dark) with respect to (wrt) time (in TU). One can notice that $n_1 = 1$, $n_2 = 1$, $r_1 = 2$, and $r_2 = 2$. Both cycles are executed at maximal speed with an average period of 6 TU (25 executions of each cycle are detected within the time window of width 150 TU). Figure 3 bottom reports the distributions of the durations of the cycles $\{T_1, T_2, T_3\}$ in light gray and $\{T_4, T_5, T_6\}$ in dark gray. One can notice the dispersion of these distributions due to the uniform pdf of the firing durations.

2.4 Models of manufacturing systems with LTPN

From a high-level perspective, a manufacturing system is often considered as a workshop composed of multiple

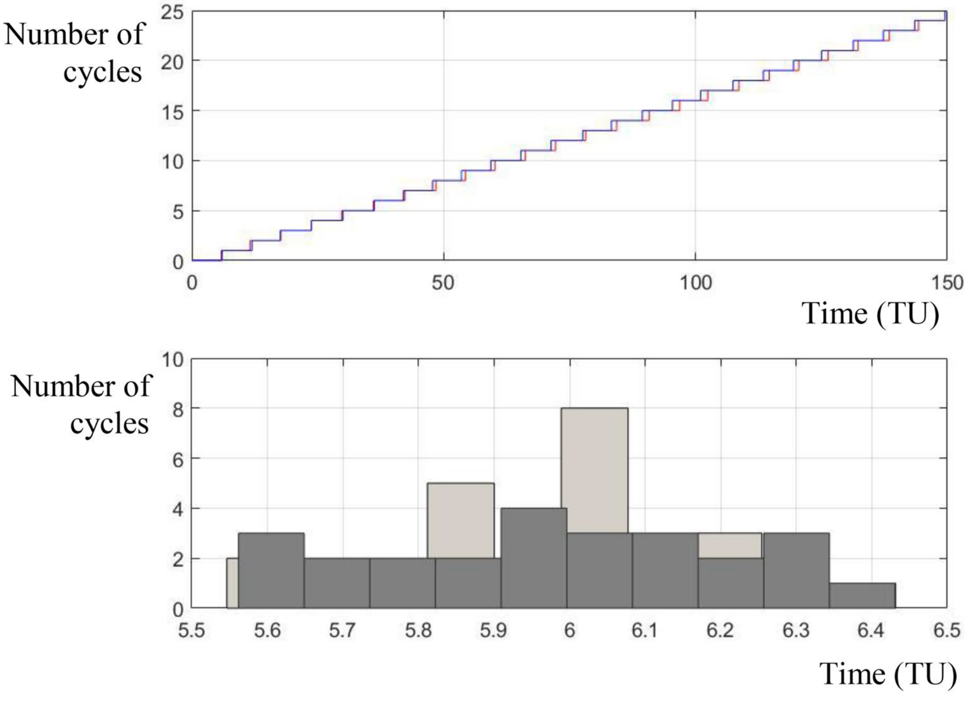


Fig. 3 Number of executions (top) of cycle $\{T_1, T_2, T_3\}$ in red (resp. cycle $\{T_4, T_5, T_6\}$ in blue) with respect to the time; histogram of the durations (bottom) of cycle $\{T_1, T_2, T_3\}$ in light gray (resp. cycle $\{T_4, T_5, T_6\}$ in dark gray)

jobs where each job consists of a set of operations performed on several resources. Sensors are used to detect the achievement of some operations.

- Each operation o_j is modeled by a transition T_j, an input buffer represented by a place and an output buffer represented by another place (that may be in the same time the input buffer of the next operation of the job when intermediate buffers do not exist). The duration of o_j is represented by the firing delay d_j of T_j. The definition of the firing delay as a stochastic distribution is suitable to take into account some uncertainties in the execution of the operations.
- A set of resources R_j is generally required to perform each operation o_j. Each type k of resource is represented by a specific place with an initial marking r_k that indicates how many resources of type k are available to perform the different operations. The resource place is simultaneously in the preset and postset of T_j (and can also belong to the preset and postset of other transitions to model resources that are shared by several operations).
- Sensors are used to measure the activities in the workshop. The labeling function collects particular labels when some transitions fire and these labels give the information that the corresponding operation is achieved. Note that improved strategies for data collections have been investigated (see, for example, cloud infrastructure-based methods which collect data in real time from intelligent devices [28]) and can be used for the same purpose.

In addition, a supervisor may be added to remove some forbidden markings (for example to avoid the deadlock markings).

Example 3 Consider again the TPN in Fig. 2. This net is the model of a manufacturing system with two jobs. The first job J_1 consists in three operations $\{o_1, o_2, o_3\}$ that are modeled with the transitions T_1, T_2, and T_3. The second job J_2 consists in the operations $\{o_4, o_5, o_6\}$ that are modeled with the transitions T_4, T_5, and T_6. The lot size (i.e., the number of products that are simultaneously accepted by the job) for J_1 (resp. J_2) is given by the initial marking of the place P_1 (resp. P_4). Two types of resources are required to perform the operations, and the initial marking (r_1 and r_2) of the places P_7 and P_8 gives the number of resources of each type. The resource r_1 is needed to perform the operation $\{o_1, o_2\}$ and $\{o_5, o_6\}$. Similarly, the resource r_2 is needed to perform the operation $\{o_2, o_3\}$ and $\{o_4, o_5\}$. No supervisor has been considered in this example. The labeling function introduced in Example 2 gives the information that operation o_3 and consequently job J_1 are performed when a label a is detected, whereas it gives the information that operation o_6 and job J_2 are performed when a label b is detected. Consequently, Fig. 3 top can be interpreted as the number of executions of job J_1 in red light (resp. J_2 in blue dark) and Fig. 3 bottom can be interpreted as the distributions of the durations of job J_1 in light gray (resp. J_2 in dark gray).

3 Faults modeling and control charts

3.1 Assumptions

Assumptions 1–8 will be considered in the next:

1. The LTPN are bounded, and consequently, the set R and generator matrix G are of finite dimensions.
2. $T(M, M')$ is at most of cardinality 1, for any pair of markings $M, M' \in R$.
3. The silent part of the considered LTPN is acyclic.
4. The system parameters, the PDF of the stochastic firing durations, and the net initial marking are assumed to be known.
5. The time semantic is defined according to single server, preselection, and enabling memory policies.
6. The temporal faults correspond to significant variations of the PDF support (with respect to the measurement errors).
7. Single faults are considered.
8. The type of the PDF is not affected by the faults.

Assumptions 1–4 are usual assumptions needed to design observers for labeled Petri nets [8, 9]. In particular, Assumptions 1 and 3 ensure that a finite size deterministic observer exists. Assumption 2 is stated for simplicity and may be relaxed as the knowledge of initial marking (Assumption 4). Then, Assumptions 5–8 are required to obtain significant residuals for the temporal faults. Assumption 5 describes the time semantic specifications and is frequently used for timed stochastic discrete event systems [27]. Assumption 7 is also a common assumption for fault isolation [2]. Finally, Assumptions 6 and 8 restrict the class of considered temporal faults in order to make the approach tractable.

3.2 Support variation of the PDF

The firing durations of the transitions are random variables defined by their PDF. The core of the approach is to detect and characterize the PDF support variations according to the variation of the firing durations mean value. Such a variation is captured by a control chart as illustrated in Fig. 4. In particular, Fig. 4 left shows a translation for a

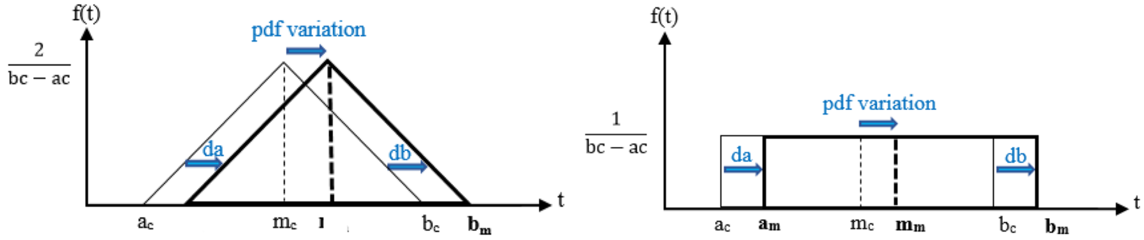


Fig. 4 Support variation of a symmetrical triangular (left) and bounded uniform (right) PDF

bounded uniform PDF and Fig. 4 right shows a translation for a symmetrical triangular PDF. The original PDF has a support [a_c, b_c] and a mean value m_c, whereas the resulting PDF has a support [a_m, b_m] and a mean value m_m.

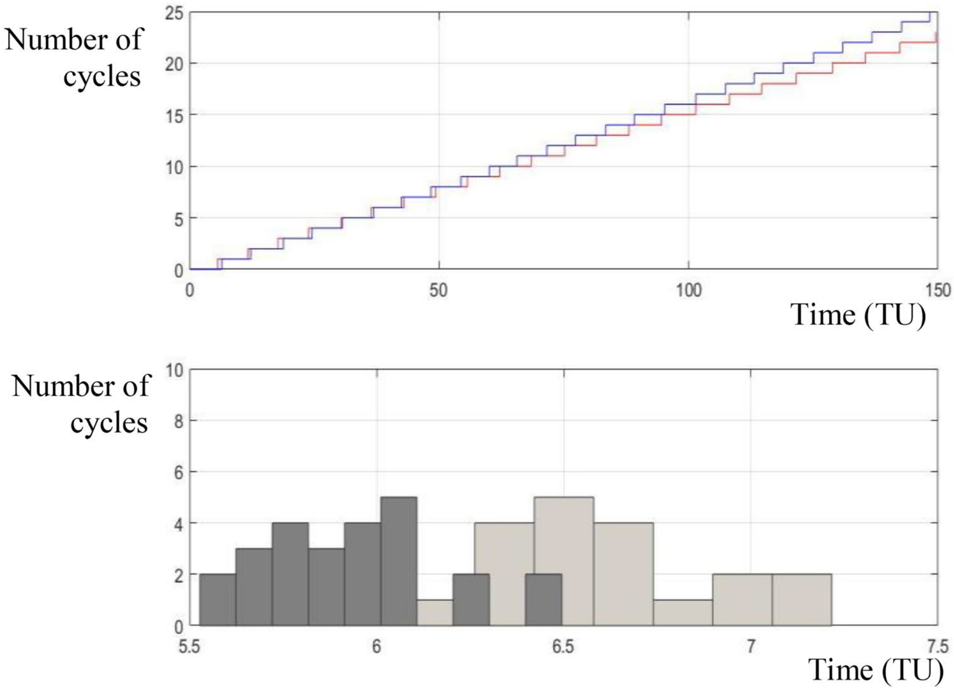
From the perspective of manufacturing systems, the time drifts represented in Fig. 4 concern the operation execution times. In a given workshop, delays may occur due to an accumulation of atomic unexpected behaviors. In many cases, the occurrence and increase in such delays are the symptom of dysfunctions in the system.

Example 4 Consider again the TPN in Fig. 2 as the model of a manufacturing system with two jobs and the labeling function introduced in Example 2. Assume that a time drift of more or less 1% affects the duration of operation o_3. The sensoring of the system remains unchanged. Figure 5 top reports the number of executions of job J_1 in red light (resp. J_2 in blue dark) for this situation. Observe that the job J_1 gradually slows down, whereas the makespan of J_2 is not affected by the dysfunction. In addition, Fig. 5 bottom reports the distributions of the durations of job J_1 in light gray (resp. J_2 in dark gray) and one can notice that the dispersion of job J_1 durations increases with respect to the time. This example illustrates how time drifts may affect the makespan of the manufacturing systems and how such time drifts could be tracked with LTPN models.

3.3 Detection of the PDF support variation with control chart

The firing durations of a given transition T are initially distributed according to the PDF of known support [a_c, b_c]. A variation of this PDF support is deducted by the calculation of the mean and the standard deviation of the durations of N consecutive firings for the same transition. Then, the new PDF support [a, b] can be computed. The support [a, b] of a bounded uniform PDF is computed from the mean m and the standard deviation σ of this PDF by Eq. (3) [24]:

Fig. 5 Number of executions of job J_1 in red (resp. J_2 in blue) with respect to the time (top); histogram of the durations of job J_1 in light gray (resp. J_2 in dark gray) when a temporal drift affects the duration of operation o_3



$$\begin{aligned} a &= m - \sigma \cdot (3)^{1/2} \\ b &= m + \sigma \cdot (3)^{1/2} \end{aligned} \tag{3}$$

In a similar way, the support [*a, b*] of a symmetrical triangular PDF is computed from the mean *m* and the standard deviation *σ* of this PDF by Eq. (4) [24]:

$$\begin{aligned} a &= m - \sigma \cdot (6)^{1/2} \\ b &= m + \sigma \cdot (6)^{1/2} \end{aligned} \tag{4}$$

A MA control chart aims to detect the variations in the average of a data series. For this purpose, *N* consecutive measurements collected by the sensors of the system are saved and some computations are proposed for these histories of data. In particular, lower and upper thresholds δ and Δ are defined according to the desired tolerances of the system. Then, a fault is detected in case of violation of the thresholds. A time window of variable size that contains *N* consecutive values of the firing duration associated with a given transition *T* is considered, and the mean value MA(d_n) of the *N* firing durations $d_{n-N+1},\ldots,d_n$ is computed. For $n > N$, an updating of MA(d_n) is obtained with Eq. (5):

$$\mathrm{MA}(d_n) = \mathrm{MA}(d_{n-1}) + \frac{\left(d_n - d_{(n-N+1)}\right)}{N} \quad \text{if } n > N \tag{5}$$

The detection thresholds δ and Δ define the acceptable variations of the mean value: The system is assumed to have a fault-free behavior as long as MA(d_n) $\in]\delta, \Delta[$; otherwise, it is assumed to have a faulty behavior. Consequently, the thresholds are selected in order to fulfill some desired performance with respect to safety requirements. A usual method is to compute these thresholds according to mean *m* and standard deviation *σ* of the firing durations of *T* measured during nominal (i.e., fault-free) behavior: $\Delta = m + \gamma \cdot \sigma$ and $\delta = m - \gamma \cdot \sigma$ where *γ* is an input parameter.

4 Observer design

4.1 Elementary trajectory observer design

In this section, elementary trajectory observers for LTPN are detailed to track the trajectories consistent with the initial marking and the successive measurements. The design of such observers is motivated by the fact that the temporal fault of a given transition will affect the durations of the sequences in which this transition occurs. An elementary trajectory observer (ETO) that computes all elementary trajectories (i.e., trajectories between two consecutive measurements) consistent with the measurements observed thus far is obtained with Algorithm 1. This algorithm aims to design in an iterated way all trajectories that are feasible at a given marking. Then, it projects the firing sequence of the trajectory within the set of observable labels. Only the trajectories that coincide with the sequence of observations collected thus far are saved for diagnosis issues. As far as the number of consistent trajectories will necessarily grow with respect to the number of successive collected observations, only elementary trajectories between two successive observations are considered. This motivates the design of the ETO. This algorithm uses the labeling function *L* and a list of unexplored states UNXPL. It returns the set R_{TO} of observer states and the generator matrix G_{obs} of the ETO. Each state *S* of ETO is composed by a set of trajectories tr = (M_{O}, M_{D}, *σ*) where M_{O} is an origin marking, M_{D} is a destination (final) marking and *σ* is a logical (i.e., making abstraction of the timing information) feasible sequence from M_{O} to M_{D} (i.e., $M_{\mathrm{O}}\,[\sigma\rangle\, M_{\mathrm{D}}$) that satisfies: $\sigma = \sigma' T$ with $L(\sigma') = \varepsilon$ and $L(T) = e$ (i.e., *σ* is consistent with a given label *e* and has no silent closure). Consequently, by construction, all edges arriving in a given state of the elementary trajectory observer will be tagged with the same label. Note that a given origin marking M_{O} (resp. a given destination marking M_{D}) can appear in several elementary trajectories associated with the same state. From *S*, it is easy to compute the set $S_{\mathrm{O}}(S)$ of origin markings and the set $S_{\mathrm{D}}(S)$ of destination markings. The set $S_{\mathrm{D}}(S)$ represents also the set of the current markings consistent with the measurement. The states are stored in R_{obs}. Each entry $g_{\mathrm{obs}}(S, S')$ of G_{obs} is composed by the label that is measured when the observer state varies from *S* to *S′*.

Algorithm 1: ETO design for LTPN

```
(Inputs : $PN, M_I, L$; Outputs: $G_{obs}, R_{obs}$)
$S \leftarrow \{(M_I, M_I, \varnothing)\}$ // initialization of $S$ //
$R_{obs} \leftarrow S$, $G_{obs} \leftarrow \varnothing$, $UNXPL \leftarrow S$ // initialization of $R_{ob}$, $G_{obs}$ and $UNXPL$ //
for each state $S$ in $UNXPL$,  // for each already computed trajectory //
  for each label $e$ in $\boldsymbol{E}$, // for observable label //
      compute the set $S'$ of elementary trajectories $(M'_O, M'_D, \sigma')$ originated from a marking $M'_O \in S_D(S)$
       and such that $\sigma' = \sigma''T$ with $L(\sigma'') = \varepsilon$ and $L(T) = e$ (Algorithm 2)
      if $S'$ does not already exist in $R_{obs}$
          $R_{obs} \leftarrow R_{obs} \cup \{S'\}$ // Add the new computed trajectory to the set $R_{obs}$ //
          $UNXPL \leftarrow UNXPL \cup \{S'\}$ // Add also the new computed trajectory to the set $UNXPL$ //
      end
      $g_{obs}(S, S') \leftarrow \{e\}$ // Update the generator matrix by adding a $e$-transition from state $S$ to state $S'$ //
end
remove $S$ in set $UNXPL$
end
```

The set S' of elementary trajectories (M'_O, M'_D, σ') originated from a marking $M'_O \in S_D(S)$ and such that $\sigma' = \sigma''T$ with $L(\sigma'') = \varepsilon$ and $L(T) = e$ is computed with Algorithm 2.

Proposition 1 *Consider a LTPN < PN, PDF, Ω, L, M_I > that satisfies Assumptions 1–4. The ETO obtained with Algorithm 1 has a finite number of states that does not exceed*

Algorithm 2: Computation of *S'*

```
(Inputs : $PN, S_D(S), e, L$; Outputs: $S'$)
$S' \leftarrow \varnothing$, $UNXPL \leftarrow \varnothing$ // initialization of $S'$ and $UNXPL$ //
for each marking $M$ in $S_D(S)$,  // for each marking in the set of the destination markings from state $S$ //
    $UNXPL \leftarrow UNXPL \cup \{(M, M, \varnothing)\}$ // Add a new trajectory to the set $UNXPL$ //
end
for each triplet $(M_O, M_D, \sigma)$ in $UNXPL$,
    for each $T$ enabled at $M_D$
        compute $M'_D$ such that $M_D\, [\, T \,\rangle\, M'_D$ and $\sigma' = \sigma\, T$ // Computed new marking and sequence resulting from the firing of
transition $T$ //
        if $((M_O, M'_D, \sigma') \notin S') \wedge (L(T) = e)$ // If the resulting trajectory does not already exist and if the firing of $T$ delivers the label
$e$ //
          $S' \leftarrow S' \cup \{(M_O, M'_D, \sigma')\}$ // Add the new computed trajectory to the set $S'$ //
        end
        if $(L(T) = \varepsilon)$ // If the firing of $T$ is silent //
            $UNXPL \leftarrow UNXPL \cup \{(M_O, M'_D, \sigma')\}$ // Add the new computed trajectory to the set $UNXPL$ //
        end
    end
remove $(M_O, M_D, \sigma)$ from $UNXPL$
end
```

$$\sum_{j=1,\ldots,H+1}\left(2^{(N^{j+1})}-1\right).$$

where $h_{\max}$ is the maximal number of consecutive silent transitions.

Proof

i. A trajectory of length k includes not only a firing sequence of length k but also the origin and destination markings, and because of Assumption 2, a trajectory of length 1 (M_O, M_D, T) is indifferently defined by (M_O, T) or (M_O, M_D). Consequently, the total number of trajectories of length k does not exceed $N_1 = N^{k+1}$ where N is the finite number of system states. Moreover, the number of states with trajectories of length k cannot exceed the sum of the combinations $C^1_{N1} + \cdots + C^{N1}_{N1} = 2^{N_1} - 1$.

ii. According to Assumption 3, the number of consecutive silent transitions is finite and cannot exceed a maximal number referred to as H. Consequently, the trajectories encoded by the observer states have a maximal length of $H+1$ and each state of the observer is a combination of trajectories with length 1 to $H+1$.

The upper bound $\sum_{j=1,\ldots,H+1}\left(2^{(N^{j+1})}-1\right)$ results from (i) and (ii). □

Proposition 2 *Consider a LTPN < PN, PDF, Ω, L, M_I > that satisfies Assumptions 1–4. The ETO obtained with Algorithm 1 is of minimal size.*

Proof First observe that the ETO is designed in an iterated schema: For each state S and label e, a new state S' is created only if the set of trajectories encoded in S' is different from the set of trajectories encoded in S. Now, imagine a reduced observer obtained by merging two different states S and S' in a single state S''. For simplicity and without any loss of generality, assume that S and S' differ only by a single trajectory of length k: $\mathrm{tr} \in S$ and $\mathrm{tr} \notin S'$.

In case S'' is defined so that $\mathrm{tr} \in S''$, all trajectories that results in S' have a postfix different from tr. So, tr is not consistent with some sequences of observation and the reduced ETO is no longer an elementary trajectory observer.

In case S'' is defined so that $\mathrm{tr} \notin S''$, there exist trajectories with a postfix tr that results in S but not in S'' and the reduced ETO is no longer an elementary trajectory observer.□

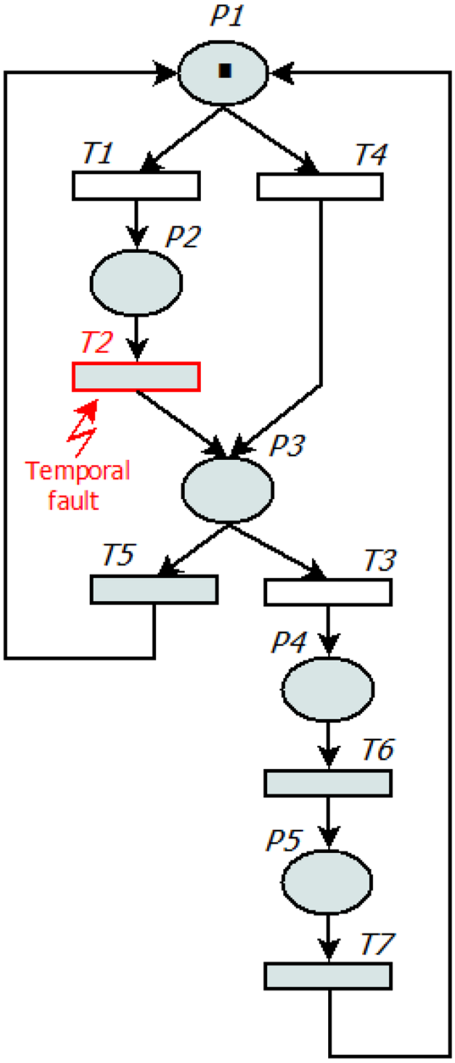


Fig. 6 The LTPN system: < PN_2, PDF, Ω, L, M_I > of Example 5

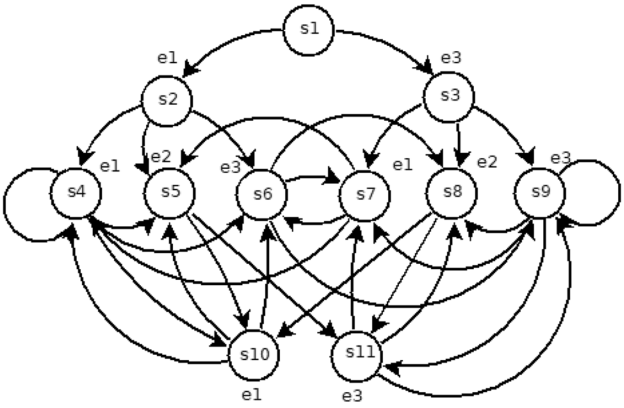


Fig. 7 ETO for < PN_2, PDF, Ω, L, M_I > (for clarity labels are reported near to the states—all edges that reach the same state sharing the same label)

Table 2 Elementary trajectories and mean duration in fault-free behavior

S	Elementary trajectories	Mean duration
S_1	–	–
S_2	$\{M_1\,[T_1\rangle\,M_2\}$	–
S_3	$\{M_1\,[T_4\rangle\,M_3\}$	–
S_4	$\{M_2\,[T_2T_5T_1\rangle\,M_2\}$	7.03
S_5	$\{M_2\,[T_2T_3\rangle\,M_4\}$	3.51
S_6	$\{M_2\,[T_2T_5T_4\rangle\,M_3\}$	8.02
S_7	$\{M_3\,[T_5T_1\rangle\,M_2\}$	5.00
S_8	$\{M_3\,[T_3\rangle\,M_4\}$	1.50
S_9	$\{M_3\,[T_5T_4\rangle\,M_3\}$	6.04
S_{10}	$\{M_4\,[T_6T_7T_1\rangle\,M_2\}$	8.52
S_{11}	$\{M_4\,[T_6T_7T_4\rangle\,M_3\}$	9.46

Table 3 Complexity increases with respect to initial marking

k	1	2	3	…
N	5	15	35	…
N_{obs}	11	53	157	…
N_{ET}	10	664	18992	…
N'_{ET}	10	185	1418	…

Example 5 Consider the marked LTPN system < PN$_2$, PDF, Ω, L, M_I > in Fig. 6 with $M_1 = (1\ 0\ 0\ 0\ 0)^T$ and the sensor configuration defined by $L(T_1) = e_1$, $L(T_3) = e_2$, $L(T_4) = e_3$ and $L(T_j) = \varepsilon$ for $j = 2, 5, 6, 7$. The reachability set of < PN$_2$, PDF, Ω, L, M_I > has five states: $M_1 = (1\ 0\ 0\ 0\ 0)^T$; $M_2 = (0\ 1\ 0\ 0\ 0)^T$; $M_3 = (0\ 0\ 1\ 0\ 0)^T$; $M_4 = (0\ 0\ 0\ 1\ 0)^T$; and $M_5 = (0\ 0\ 0\ 0\ 1)^T$. The observer of < PN$_2$, PDF, Ω, L, M_I > has 11 states and is reported in Fig. 7. Note that this observer is composed of a transient part that corresponds to the set of states $\{S_1, S_2, S_3\}$ and to a steady state part that corresponds to the other states. The list of elementary trajectories consistent with each state of the ETO is reported in the second column of Table 2. Let us also report in column 3 the mean duration in nominal behavior for these trajectories. Such a mean duration is not calculated for the transient states for which only a single measurement is collected at most.

4.2 Minimal-size elementary trajectories

In the previous simple example, each state is associated with a single elementary trajectory. This is no longer the case for more complex systems or when the initial marking increases. To illustrate this difficulty, let us consider the previous example of Fig. 6 and an initial marking $M_1(k) = k\cdot(1\ 0\ 0\ 0\ 0)^T$ that depends on k; Table 3 illustrates how the system size N, the observer size N_{obs}, and the number N_{ET} of elementary trajectories increase with respect to k. In particular, one can notice the rapid increase in N_{ET} with respect to k.

In order to limit this explosion of complexity, minimal-size elementary trajectories (MSET) are introduced. A MSET: $M_O\,[\sigma\rangle\,M_D$ in state S is defined as an elementary trajectory that is matched [22, 29, 30] by all other elementary trajectories in S and originated from the same marking M_O. A trajectory $M_O\,[\sigma'\rangle\,M'_D$ matches another trajectory $M_O\,[\sigma\rangle\,M_D$ if σ' contains all transitions of σ and these transitions respect the same precedence conditions (i.e., fire in the same order). In a more formal way, a sequence σ' matches a sequence $\sigma \neq \varepsilon$ (one write $\sigma \ll \sigma'$) if $\sigma = \sigma_1\ \sigma_2$ with $\sigma_1 \in \boldsymbol{T}^*$ ("*" denotes the Kleene star and $\boldsymbol{T}^*$ is the set of sequences of transitions in $\boldsymbol{T}$) and $\sigma_2 \in \boldsymbol{T}^*$ and there exists $\sigma'_1 \in \boldsymbol{T}^*$ and $\sigma'_2 \in \boldsymbol{T}^*$ such that $\sigma' = \sigma'_1\ \sigma_1\ \sigma'_2$ and $\sigma_2 \ll \sigma'_1$. MSET(S) is defined as the set of MSET in S. MSET(S) is the subset of elementary trajectories of S that is matched by all other elementary trajectories in S that are originated from the same marking M_O. N'_{ET} is defined as the global number of MSET in all states of the ETO (i.e., in R_{obs}). The idea behind the MSET computation is to remove the transitions that fire concurrently in some (but not all) elementary trajectories. Such an elimination is reasonable because when a temporal fault affects such a transition, it will not affect the collected measurements. Algorithm 3 computes the set MSET(S) of MSET for a given state S of the ETO. In this algorithm, $T(h)$ stands for the hth transition of sequence σ.

Algorithm 3: Computation of MSET

(Inputs : S; Outputs: $MSET$)

```
$MSET \leftarrow \emptyset$ // initialization of $MSET$ //
for each triplet ($M_O, M_D, \sigma$) in $S$,
    for each triplet ($M'_O, M'_D, \sigma'$) in $S$, // for each pair of trajectories in set $S$ //
        $new \leftarrow 0$// set flag $new$ to 0 //
        if $(\sigma \neq \sigma') \wedge (M_O = M'_O)$ // if the trajectories are different but with the same origin markings //
            $k \leftarrow 1$
            for $h = 1,\ldots,|\sigma'|$
                while $(k \leq |\sigma|) \wedge (T'(h) \neq T(k))$ // search if $\sigma'$ matches $\sigma$//
                    $k = k + 1$
                end
                if $k > |\sigma|$, $match \leftarrow 0$, else $match \leftarrow 1$, end // set flag $match$ to 0 is the sequence is completely explored //
                $k = k + 1$
            end
            $new \leftarrow \max(new, match)$ // update the flag $match$ according to the result of the matching test //
            if $new = 0$
                $MSET \leftarrow MSET \cup \{(M_O, M_D, \sigma)\}$ // add the trajectory $(M_O, M_D, \sigma)$ to the set $MSET$ //
            end
        end
remove ($M_O, M_D, \sigma$) from $S$
end
```

Example 6 illustrates the computation of MSET.

Example 6 Consider again the marked LTPN system < PN_2, PDF, Ω, L, M_I > in Fig. 6. If $M_1 = (2\ 0\ 0\ 0\ 0)^T$, the ETO has 53 states and each state is composed of a set of elementary trajectories. Consider, for example, a particular state S composed of nine elementary trajectories:

$$\begin{aligned} S = \{ & M_3[T_1\rangle M_5; \\ & M_3[T_5T_1\rangle M_2; \\ & M_5[T_2T_5T_1\rangle M_5; \\ & M_5[T_2T_5T_5T_1\rangle M_2; \\ & M_5[T_5T_1\rangle M_4; \\ & M_5[T_5T_2T_1\rangle M_5; \\ & M_5[T_5T_2T_5T_1\rangle M_2; \\ & M_7[T_5T_1\rangle M_5; \\ & M_7[T_5T_5T_1\rangle M_2 \} \end{aligned}$$

with $M_2 = (1\ 1\ 0\ 0\ 0)^T$; $M_3 = (1\ 0\ 1\ 0\ 0)^T$; $M_4 = (0\ 2\ 0\ 0\ 0)^T$; $M_5 = (0\ 1\ 1\ 0\ 0)^T$; $M_7 = (0\ 0\ 2\ 0\ 0)^T$.

From these elementary trajectories, three MSET are computed:

$$\begin{aligned} & M_3[T_1\rangle M_5; \\ & M_5[T_5T_1\rangle M_4; \\ & M_7[T_5T_1\rangle M_5. \end{aligned}$$

In the elementary trajectory M_3 $[T_5T_1\rangle$ M_2, the transition T_5 fires concurrently with T_1, but the measurement is obtained when T_1 fires. A fault in T_5 will not affect the duration of the sequence M_3 $[T_5T_1\rangle$ M_2, and this trajectory can be removed. The global number N'_{ET} of MSET is reported in Table 2 with respect to k, and one can notice the reduction of complexity, compared to N_{ET}.

5 Detection and diagnosis

In this section, a detection and diagnosis method is proposed for temporal faults. This method uses a moving average control chart and the ETO previously defined. It has three steps: (a) the computation of residuals for each MSET, (b) the computation of the detection function, and (c) the computation of the isolation function. In addition to Assumptions 1–4 that are needed to obtain the ETO, Assumptions 5–8 will be also considered in this section to compute significant residuals.

5.1 Residuals computation

Once the MSET are computed, the durations between two successive measurements are affected to all MSET that are consistent with these measurements. The detection and isolation of temporal faults is then based on the analysis of a set of residuals obtained for the MSET. For each MSET, the mean duration of the nominal behavior is first evaluated and $d_N(mset, S)$ is defined as the mean duration in nominal behavior for the minimal-size elementary trajectory mset in set MSET(S). This evaluation can be obtained from an analytical computation or from the statistical analysis of a fault-free sequence of measurements or finally from an expert knowledge about the system. In order to compute the residuals used to detect and isolate the temporal faults, the proposed approach has the following steps:

1. For each pair of consecutive measurements (e_{k-1}, τ_{k-1}) and (e_k, τ_k) of TR_O, the state S consistent with the observed trajectory $TRo(k) = (e_1,\tau_1) \ldots (e_{k-1},\tau_{k-1})\ (e_k,\tau_k)$ is computed thanks to the ETO.
2. The duration $d_k = \tau_k - \tau_{k-1}$, measured at time τ_k, is filtered with the MA control charts to smooth the variations.
3. The filtered duration MA(d_k) is associated with each minimal-size elementary trajectory mset in set MSET(S). The series of measurements collected for the minimal-size elementary trajectory mset in state S at time τ_k is consequently defined as $D(mset, S, \tau_k)$:

$$D(mset, S, \tau_k) = \{(\mathrm{MA}(d_k), \tau_k) \text{ such that } mset \in \mathrm{MSET}(S) \text{ and } S \text{ is consistent with } \mathrm{TRo}(k)\} \tag{6}$$

4. One difficulty is that the series of measurements $D(mset,S,\tau_k)$ are generated at specific values of times τ_k that only depend on the occurrence of the observable events. On the contrary, the detection and diagnosis decision are expected to be computed periodically. To solve this issue, the series of measurements $D(mset,S,\tau_k)$ are resampled with a given sampling time dt according to Eq. (7):

$$D'(mset, S, h.dt) = \{(\mathrm{MA}(d_h), h.dt), h = 0, \ldots, \lfloor \tau_k/dt \rfloor + 1, \text{ and } (\mathrm{MA}(d_h), h.dt) \in D(mset, S, \tau_k) \text{ is the measurement such that } \tau_k \text{ is the larger measurement time that satisfies } \tau_k \le h.dt\} \tag{7}$$

   A consequence of the resampling operation is that the values of the firing durations are maintained constant and equal to the last collected value as long as no new measurement is collected in series $D(mset,S,\tau_k)$.
5. The series of residuals are computed from the series of durations. The filtered resampled durations MA(d_h) are used to compute the residuals δ(MA(d_h), $mset$, S) for all $mset \in$ MSET(S):

$$\delta(\mathrm{MA}(d_h), mset, S) = \mathrm{MA}(d_h) - d_N(mset, S) \tag{8}$$

5.2 Detection and isolation of temporal faults

For each transition $T_j \in \boldsymbol{T}$, let us first define the set MSET(T_j) of MSET in which T_j fires. The two following functions are then defined for detection and diagnosis issues:

$$\mathrm{diag}^+(T_j, h.dt) = \max\{\delta(\mathrm{MA}(d_h), mset, S) \text{ for all } mset \in \mathrm{MSET}(T_j) \text{ and } S \in R_{\mathrm{obs}}\} \tag{9}$$

$$\mathrm{diag}^-(T_j, h.dt) = \min\{|\delta(\mathrm{MA}(d_h), mset, S)| \text{ for all } mset \in \mathrm{MSET}(T_j) \text{ and } S \in R_{\mathrm{obs}}\} \tag{10}$$

The function detect($h.dt$) is proposed as a detection function that captures the variation of the collected durations and evaluates whether a temporal fault has occurred at sampled time $h.dt$. This function is sensitive to the sum of the maximal residuals computed for each transition:

$$\mathrm{detect}(h.dt) = \frac{1}{q} \sum_{T_j \in T} \mathrm{diag}^+\left(T_j, h.dt\right) \tag{11}$$

This function will be compared with a detection threshold Δ_D in order to generate an alarm when the threshold is excessed.

Table 4 Support of the transition PDF for <SPN_2, L, M_I>

	T_1	T_2	T_3	T_4	T_5	T_6	T_7
a_c	1.8	1.8	1.3	2.8	2.8	3.3	2.8
b_c	2.8	2.2	1.7	3.2	3.2	3.7	3.2

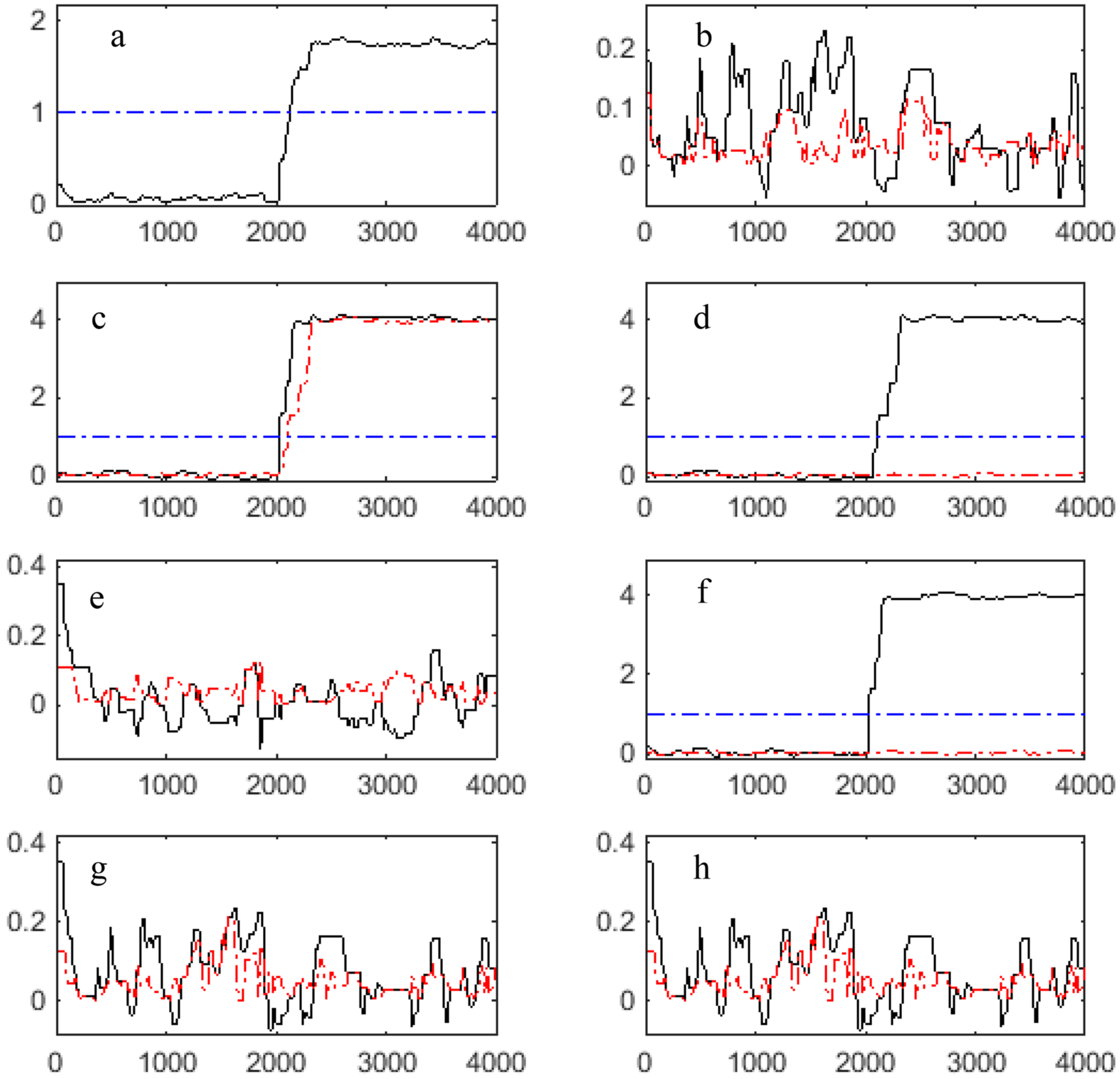


Fig. 8 Diagnosis of < PN_2, *PDF*, Ω, *L*, M_I > with $M_I = (1\ 0\ 0\ 0\ 0)^T$: **a** detection function wrt time (TU); isolation functions wrt time (TU); **b** T_1; **c** T_2; **d** T_3; **e** T_4; **f** T_5; **g** T_6; **h** T_7 ($\text{diag}^+(T_j, h.dt)$ in full line and $\text{diag}^-(T_j, h.dt)$ in dashed line)

The fault isolation results from the combine use of the functions $\text{diag}^+(T_j, h.dt)$ and $\text{diag}^-(T_j, h.dt)$. These functions evaluate how the firing of a given transition T_j is affected by the variation of the collected durations. On the one hand, the function $\text{diag}^+(T_j, h.dt)$ is sensitive to the maximal value of the residuals computed for T_j. When this function increases significantly from zero, it means that a variation of the durations affects at least one of the MSET where T_j appears. Note that the use of $\text{diag}^+(T_j, h.dt)$ without $\text{diag}^-(T_j, h.dt)$ for isolation may lead to diagnosis errors by overestimating the risk that the temporal fault concerns T_j. On the other hand, the function $\text{diag}^-(T_j, h.dt)$ is sensitive to the minimal absolute value of the residuals computed for T_j. When this function increases significantly from zero, it means that a variation of the durations affects necessarily all MSET where T_j occurs. The use of $\text{diag}^-(T_j, h.dt)$ without $\text{diag}^+(T_j, h.dt)$ may also lead to diagnosis errors by underestimating the risk that the temporal fault concerns T_j. Consequently, both functions are used together to compute the probability $\text{prob}(T_j, h.dt)$ that the detected fault has affected the transition T_j. Let us introduce a normalization parameter $N(h.dt)$ at time $h.dt$ with Eq. (12):

$$N(h.dt) = \sum_{\substack{T_j \text{ such that} \\ (\text{detect}(h \cdot dt) > \Delta_D) \\ \wedge(\text{diag}^+(T_j, h \cdot dt) > \Delta_D)}} \text{diag}^-(T_j, h.dt) \tag{12}$$

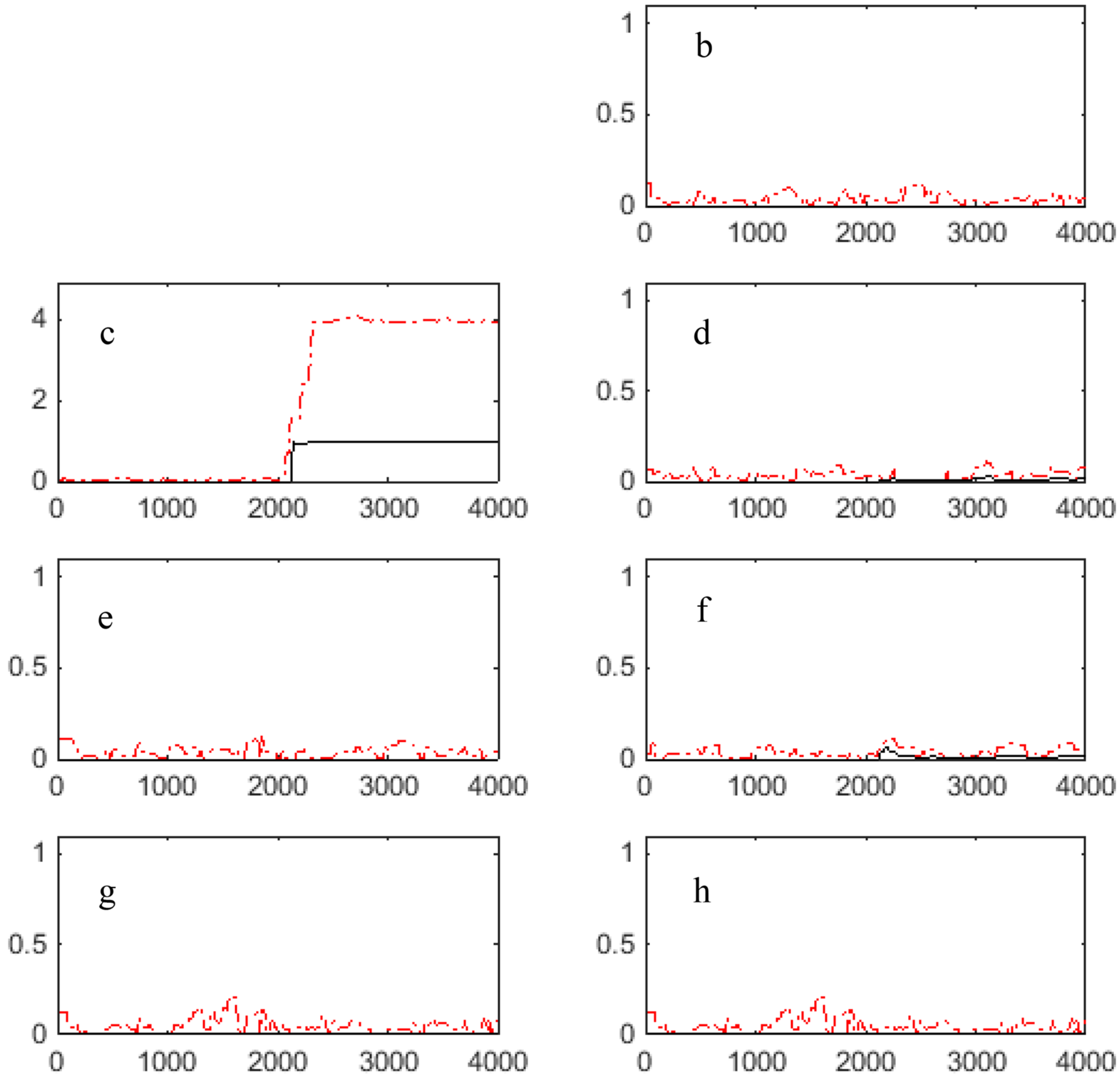


Fig. 9 Fault probabilities for < PN_2, PDF, Ω, *L*, M_I > with $M_I = (1\ 0\ 0\ 0\ 0)^T$: **b** T_1; **c** T_2; **d** T_3; **e** T_4; **f** T_5; **g** T_6; **h** T_7 (prob(T_j, *h.dt*) in full line and diag−(T_j, *h.dt*) in dashed line)

The probability prob(T_j, *h.dt*) is computed for each *h* and T_j with Eq. (13):

$$\begin{aligned} \mathrm{prob}(T_j, h.dt) &= \mathrm{diag}^-(T_j, h.dt)/N(h.dt),\ \text{if}\,(\mathrm{detect}(h.dt) > \Delta_D) \wedge (\mathrm{diag}^+(T_j, h.dt) > \Delta_D) \\ \mathrm{prob}(T_j, h.dt) &= \mathrm{diag}^-(T_j, h.dt)/N(h.dt),\ \text{otherwise} \end{aligned} \quad (13)$$

Note that the detection delay depends on the time drift and on the number of sensors used by the labeling function. For poor sensoring systems, this delay may increase in a critical way. Improving the detection delay is one of the perspectives of this work.

Example 7 Consider again the marked LTPN system < PN_2, PDF, Ω, *L*, M_I > in Fig. 6 with $M_I = (1\ 0\ 0\ 0\ 0)^T$ and $L(T_1) = e_1$, $L(T_3) = e_2$, $L(T_4) = e_3$. Uniform bounded PDF are considered for all transitions with supports provided in Table 4.

The mean values resulting from a fault-free trajectory of duration 4000 TU are reported in the last column of Table 1. Note that mean values are not computed for the transient states S_1, S_2, and S_3 because these transitions fire at most once.

It is now assumed that transition T_2 experiences a PDF support variation from [1.8, 2.2] to [5.8, 6.2] between 2000 and 4000 TU (for the clarity of the presentation, an abrupt

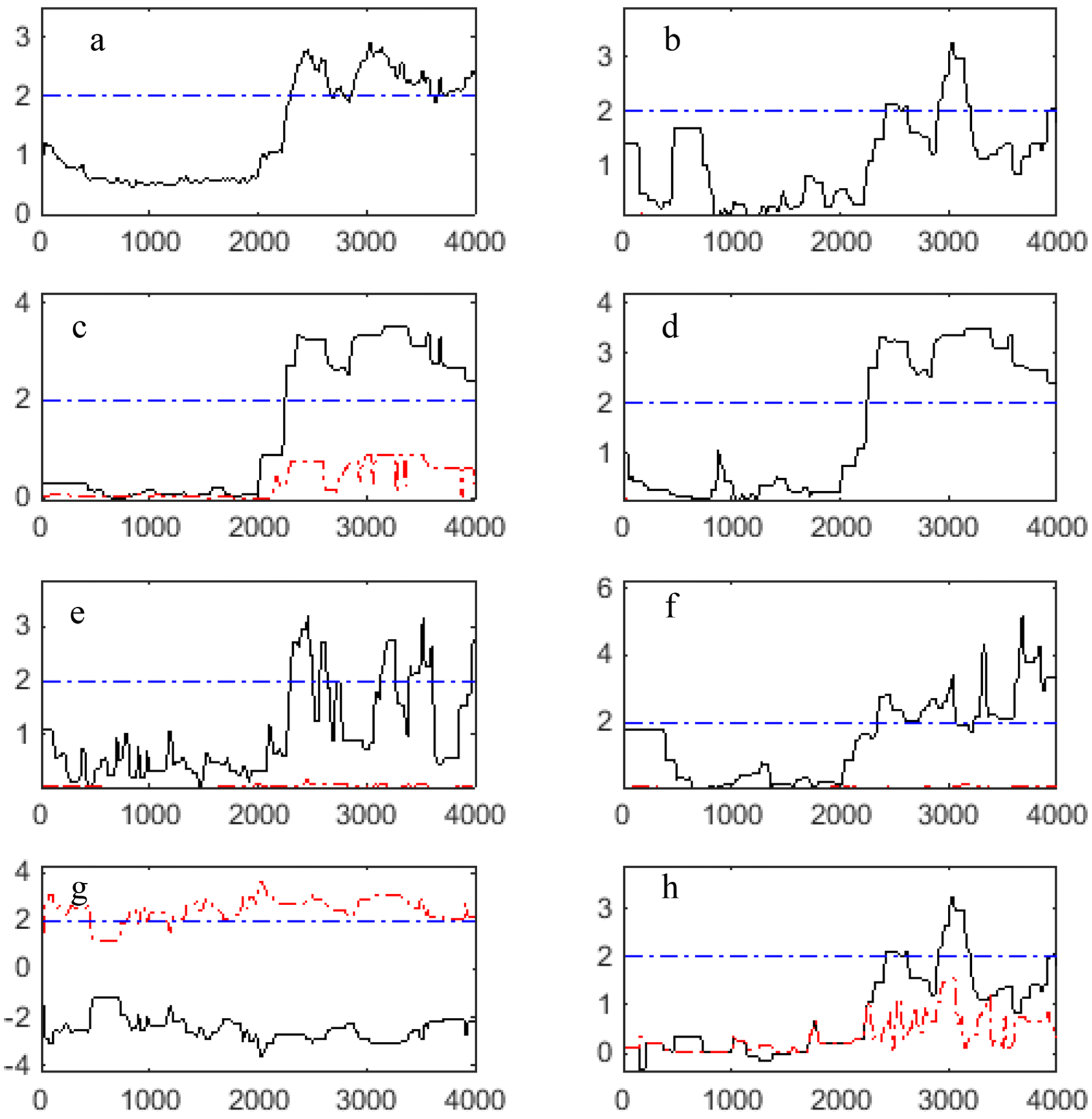


Fig. 10 Diagnosis of < PN_2, PDF, Ω, *L*, M_I > with $M_I = (2\ 0\ 0\ 0\ 0)^T$: **a** detection function wrt time (TU); isolation functions wrt time (TU); **b** T_1; **c** T_2; **d** T_3; **e** T_4; **f** T_5; **g** T_6; **h** T_7 (diag$^+$(T_j, *h.dt*) in full line and diag$^-$(T_j, *h.dt*) in dashed line)

change of the support was preferred for this example instead of a slow timed drift). The detection threshold is defined as $\Delta_D = 1$.

The functions detect(*h.dt*), diag$^+$(T_j, *h.dt*) (in full line), and diag$^-$(T_j, *h.dt*) (in dashed line) resulting from this simulation are reported in Fig. 8. Temporal faults are detected once the detection function detect(*h.dt*) exceeds the detection threshold (Fig. 8a). For the considered series, the detection time is 2020 TU and the delay is 20 TU that corresponds more or less to ten successive transition firings. One can notice that the function diag$^+$(T_j, *h.dt*) is not enough to isolate the fault. This is because T_2 occurs in MSET that also include the transitions T_1, T_3, T_4, and T_5 (Table 2). For this reason, these transitions are also suspicious. The use in addition of the function diag$^-$(T_j, *h.dt*) confirms that the fault has affected T_2. Note that in the present case, the function diag$^-$(T_j, *h.dt*) by itself clearly indicates the faulty transition, but for more complicated cases, both functions are required. This is confirmed with the probabilities prob(T_j, *h.dt*) reported in Fig. 9 in addition to the residuals diag$^-$(T_j, *h.dt*). In particular, one can notice that the probability prob(T_2, *h.dt*) that isolates transition T_2 is more or less equal to 1 from the detection time at 2020 TU.

Consider again the marked LTPN system < PN_2, PDF, Ω, *L*, M_I > in Fig. 6 with $M_I = (2\ 0\ 0\ 0\ 0)^T$, $L(T_1) = e_1$, $L(T_3) = e_2$, $L(T_4) = e_3$ and the supports provided in Table 4. The transition T_2 experiences again a PDF support variation from

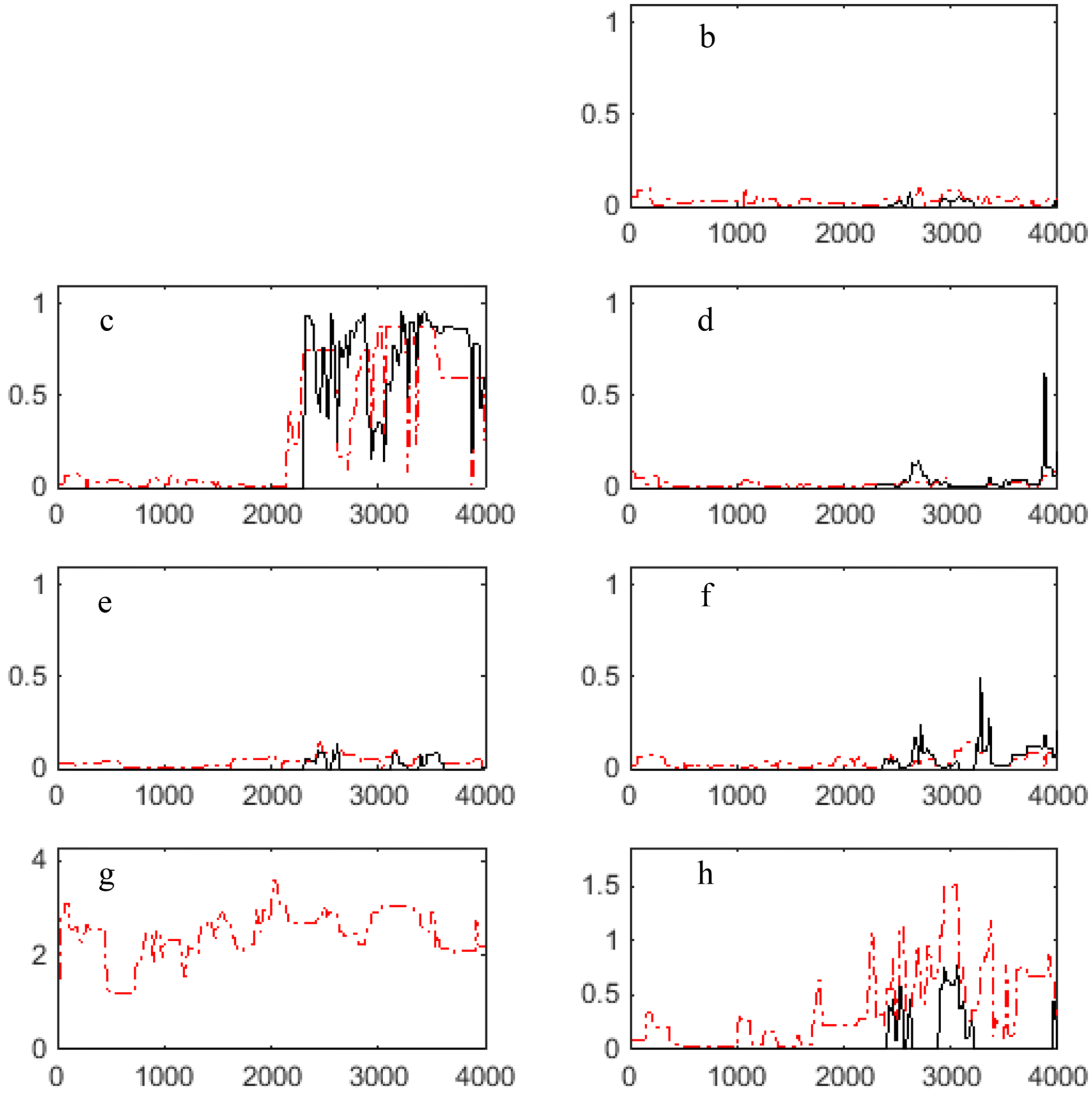


Fig. 11 Fault probabilities for < PN_2, PDF, Ω, *L*, M_I > with $M_I = (2\ 0\ 0\ 0\ 0)^T$: **b** T_1; **c** T_2; **d** T_3; **e** T_4; **f** T_5; **g** T_6; **h** T_7 (prob(T_j, *h.dt*) in full line and diag–(T_j, *h.dt*) in dashed line)

[1.8, 2.2] to [5.8, 6.2] between 2000 and 4000 TU. The detection threshold is defined as $\Delta_D = 2$.

The detection and isolation results are reported in Figs. 10 and 11. The isolation of transition T_2 that should be preferred compared to the other transitions results from the comparison of the probabilities prob(T_j, *h.dt*) after detection at date 2100 TU. In that case, the detection delay increases to 100 TU. One can notice that the diagnosis decisions (and even the detection one) have less confidence in the present case due to the large number of possible behaviors that increases the risk of error.

6 Conclusion

This article has proposed an approach that can be used to detect and isolate temporal drifts in timed manufacturing systems that are characterized by cyclic behaviors and repetitive operations. Such systems were modeled as timed Petri nets where operations are represented by the net transitions and their durations correspond to the transition firing times considered as random variables with arbitrary PDF. In this context, the temporal drifts were characterized by the variations of the mean firing durations. The detection and isolation problem has been solved by combining moving average control charts with a new class of observers that estimate the recent elementary trajectories consistent with the last measurements. On the one hand, fault detection has been obtained by comparing residual signal with some thresholds. On the other hand, the isolation of the faulty operation among the set of fault candidates has been performed thanks to the analysis of the measurements with respect to the set of minimal-size elementary trajectories generated by the observer. The following concluding comments hold that lead to some interesting perspectives:

- The proposed method can also be used to test and compare several sensor configurations in order to select the most appropriated one. One interesting issue is to reverse the problem and to propose a method that aims to search for the best selection and positioning of the sensors in order to improve the performance of the detection.
- When competition is used as a choice policy instead of preselection, one can compute, in a similar way, additional residuals based on the frequency of the MSET occurrences. With competition, conflicts are solved according to the duration of the firings. When a transition experiences a temporal fault, the conditions of the competition change and this modifies the MSET frequency.

The main limitation of the proposed approach is due to the rapid increase in the complexity in the design of the observers. In particular, the size of the elementary trajectory observer grows rapidly with respect to the initial marking of the LTPN and to the labeling function that models the sensors, resulting in large nets that may prevent to use the method for more complicated systems. Consequently, the first objective in our future work will be to consider the complexity and scalability issues. Another perspective is to pay more attention to the detection delay and to improve the approach in order to decrease this delay. The possible combined use of time and colors in Petri nets models lies also under the perspectives of that work.

References

1. Zaytoon J, Lafortune S (2013) Overview of fault diagnosis methods for discrete event systems. Ann Rev Control 37(2):308–320
2. Blanke M, Kinnaert M, Lunze J et al (2003) Diagnosis and fault-tolerant control. Springer, Berlin
3. Cassandras C, Lafortune S (2008) Introduction to discrete event systems, 2nd edn. Springer, New York
4. Hadjicostis CN (2020) Estimation and inference in discrete event systems: a model-based approach with finite automata. Springer, Berlin
5. Sampath M, Sengupta R, Lafortune S et al (1995) Diagnosability of discrete-event systems. IEEE Trans Autom Control 40(9):1555–1575
6. Lunze J, Schröder J (2001) State observation and diagnosis of discrete-event systems described by stochastic automata. Discret Event Dyn Syst 11(4):319–369
7. Tripakis S (2002) Fault diagnosis for timed automata. In formal techniques in real-time and fault-tolerant systems. Springer, Berlin, pp 205–221
8. Ru Y, Hadjiscotis H (2009) Fault diagnosis in discrete event systems modeled by partially observed Petri nets. Discret Event Dyn Syst 19:551–575
9. Lefebvre D (2014) On-line fault diagnosis with partially observed Petri nets. IEEE Trans Autom Control 59(7):1919–1924
10. Saporta G (2006) Probabilités, analyse des données et statistique. Editions TECHNIP, p 284
11. Benneyan JC (2001) Design, use, and performance of statistical control charts for clinical process improvement
12. Ali S, Pievatolo A, Gob R (2016) An overview of control charts for high-quality processes, quality and reliability engineering international
13. Rachidi S, Leclercq E, Pigné Y, Lefebvre D (2017) PN modeling of discrete event systems with temporal constraints. ICSTCC, Romania
14. Liu G, Barkaoui K (2016) A survey of siphons in Petri nets. Inf Sci 363:198–220
15. Ezpeleta J, Colom JM, Martinez J (1995) A Petri net based deadlock prevention policy for flexible manufacturing systems. IEEE Trans Robot Autom 11:173–184
16. Barkaoui K, Ben Abdallah I (1996) Analysis of a resource allocation problem in FMS using structure theory of Petri nets. In: Proceedings, first international workshop on manufacturing and petri nets, Japan, pp 1–15
17. Mejía G, Caballero-Villalobos JP, Montoya C (2018) Petri nets and deadlock-free scheduling of open shop manufacturing systems. IEEE Trans Syst Man Cybern Syst 48(6):1017–1028
18. Viswanadham N, Narahari Y (1987) Coloured Petri net models for automated manufacturing systems. In: Proceedings of IEEE international conference on robotics and automation, Raleigh, NC, USA, pp 1985–1990
19. Ezpeleta J, Colom JM (1997) Automatic synthesis of colored Petri nets for the control of FMS. IEEE Trans Robot Autom 13(3):327–337
20. Kuo Chung-Hsien, Huang Han-Pang (2000) Failure modeling and process monitoring for flexible manufacturing systems using colored timed Petri nets. IEEE Trans Robot Autom 16(3):301–312
21. Zhou ZG, Tang P (2016) Improving time series anomaly detection based on exponentially weighted moving average (EWMA) of season-trend model residuals, IGARSS
22. Jiang S, Kumar R (2004) Failure diagnosis of discrete-event systems with linear-time temporal logic specifications. IEEE Trans Autom Control 49(6):934–945
23. Lefebvre D, Rachidi S, Leclercq E, Pigné Y (2019) Diagnosis of structural and temporal faults for k-bounded non-Markovian stochastic Petri nets. Trans IEEE SMCA. https://doi.org/10.1109/TSMC.2018.2875726
24. Rachidi S, Leclercq E, Pigné Y, Lefebvre D (2018) Moving average control chart for the detection and isolation of temporal faults in stochastic Petri nets. In: Proceedings of IEEE ETFA, Torino, Italy
25. Brinksma E, Hermanns H, Katoen J-P (2001) Introduction to stochastic Petri nets. FMPA 2000, LNCS 2090, pp 84–155
26. Marsan MA, Balbo G, Conte G, Donatelli S, Franceschinis G (1994) Modelling with generalized stochastic petri nets. Wiley series in parallel computing. Wiley, Hoboken

27. Haddad S, Moreaux P (2009) Stochastic Petri nets (chapter 7). In: Petri nets: fundamental models and applications. Wiley, Hoboken
28. Răileanu S, Anton F, Borangiu T, Anton S, Nicolae M (2018) A cloud-based manufacturing control system with data integration from multiple autonomous agents. Comput Ind 102:50–61
29. Jéron T, Marchand H, Pinchinat S, Cordier MO (2006) Supervision patterns in discrete event systems diagnosis. In: 8th International workshop on discrete event systems, Ann Arbor, MI, United States, pp 262–268
30. Gougam H-E, Subias A, Pencolé Y (2013) Supervision patterns: formal diagnosability checking by Petri net unfolding. In: 4th IFAC workshop on dependable control of discrete systems, York, United Kingdom, pp 73–78

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.