Time Supervision of Concurrent Systems using Symbolic Unfoldings of Time Petri Nets [⋆]

Thomas Chatain and Claude Jard

IRISA/ENS Cachan-Bretagne,
Campus de Beaulieu, F-35042 Rennes cedex, France
Thomas.Chatain@irisa.fr, Claude.Jard@bretagne.ens-cachan.fr

Abstract. Monitoring real-time concurrent systems is a challenging task. In this paper we formulate (model-based) supervision by means of hidden state history reconstruction, from event (e.g. alarm) observations. We follow a so-called true concurrency approach using time Petri nets: the model defines explicitly the causal and concurrency relations between the observable events, produced by the system under supervision on different points of observation, and constrained by time aspects. The problem is to compute on-the-fly the different partial order histories, which are the possible explanations of the observable events. We do not impose that time is observable: the aim of supervision is to infer the partial ordering of the events and their possible firing dates. This is achieved by considering a model of the system under supervision, given as a time Petri net, and the on-the-fly construction of an unfolding, guided by the observations. Using a symbolic representation, this paper presents a new definition of the unfolding of time Petri nets with dense time.

1 Introduction and related work

Monitoring real-time concurrent systems is a challenging task. In this paper we formulate model-based supervision by means of hidden state history reconstruction, from event (e.g. alarm) observations. We follow a so-called true concurrency approach using time Petri nets: the model defines explicitly the causal and concurrency relations between the observable events, produced by the system under supervision on different points of observation, and constrained by time aspects. The problem is to compute on-the-fly the different partial order histories, which are the possible explanations of the observable events. An important application is the supervision of telecommunications networks, which motivated this work.

Without considering time, a natural candidate to formalize the problem are safe Petri nets with branching processes and unfoldings. The previous work of our group used this framework to define the histories and a distributed algorithm to build them as a collection of consistent local views[2]. The approach defines

[⋆] This work was supported by the french RNRT projects Swan and Persiform, funded by the Ministère de la Recherche ; partners of the Swan project are Inria, France Telecom R&D, Alcatel, QosMetrics, and University of Bordeaux ; partners of the Persiform project are Inria, France Telecom R&D, INT, Orpheus, and University of Grenoble.

the possible explanations as the underlying event structure of the unfolding of the product of the Petri net model and of an acyclic Petri net representing the partial order of the observed alarms.

In this paper we extend our method to time Petri nets, allowing the designer to model time constraints, restricting by this way the set of possible explanations, We do not impose that time is observable: the aim of supervision is to infer the partial ordering of the events and their possible firing dates. Using a symbolic representation, this paper presents a new definition of the unfolding of time Petri nets with dense time.

Model-based diagnosis using time Petri nets and partial orders has already been addressed in [12]. In this work, temporal reasoning is based on (linear) logic. The first reference to time Petri net unfolding seems to be in 1996, by A. Semenov, A. Yakovlev and A. Koelmans [13] in the context of hardware verification. They deal only with a quite restricted class of nets, called *time independent choice time Petri net*, in which any choice is resolved independently of time. In [1], T. Aura and J. Lilius give a partial order semantics to time Petri nets, based on the nonsequential processes semantics for untimed net systems. A time process of a time Petri net is defined as a traditionally constructed causal process that has a valid timing. An algorithm for checking validness of a given timing is presented. It is proved that the interleavings of the time processes are in bijection with the firing schedules. But unfortunately, they do not provide a way to represent all the valid processes using the notion of unfolding of time Petri net, as usual in the untimed case. A few years later (in 2002), H. Fleischhack and C. Stehno in [10] give the first notion of a finite prefix of the unfolding of a time Petri net. Their method relies on a translation towards an ordinary place/transition net. This requires to consider only discrete time and to enumerate all the situations. This also relies on the introduction of new transitions, which represent the clock ticks. Although relevant for model-checking, it is not clear that it allows us to recover causalities and concurrencies, as required in the diagnosis application. Furthermore, we are convinced that time constraints must be treated in a symbolic way, using the analog of state class constructions of B. Berthomieu [3,4].

The rest of the paper is organized as follows. Section 2 defines the different ingredients of our model-based supervision, namely the diagnosis setup, the time Petri net model and its partial order semantics. Section 3 describes the symbolic unfolding technique used to compute the symbolic processes, which serve as explanations. Before entering the general case, we consider the simplest case of extended free-choice time Petri nets [5]. We conclude in Section 4. The proofs of the theorems are available in the research report [7].

2 Diagnosis, time Petri nets and partial order semantics

2.1 Diagnosis: problem

Let us consider a real distributed system, which produces on a given set of sensors some events (or alarms) during its life. We consider the following setup for diagnosis, assuming that alarms are not lost. Each sensor records its local alarms in sequence, while respecting causality (i.e. the observed sequences cannot contradict the causal and temporal ordering defined in the model). The different sensors perform independently and asynchronously, and a single supervisor collects the records from the different sensors. Thus any interleaving of the records from different sensors is possible, and causalities and temporal ordering among alarms from different sensors are lost. This architecture is illustrated in Figure 1.

For the development of the example, we consider that the system under supervision produces the sequences $\alpha\gamma\alpha\gamma$ on sensor A, and $\beta\beta$ on sensor B. Given the time Petri net model of Figure 1 (left), the goal of the diagnoser is to compute all the possible explanations shown in Figure 1. Explanations are labelled partial orders. Each node is labeled by a transition of the Petri net model and a possible date given by a symbolic expression. Notice that the diagnoser infers the possible causalities between alarms, as well as the possible dates for each of them. The first alarms $\alpha\gamma\alpha$ and $\beta\beta$ imply that transitions t_1 and t_2 are fired twice and concurrently. The last γ can be explained by two different transitions in conflict (t_3 and t_4).

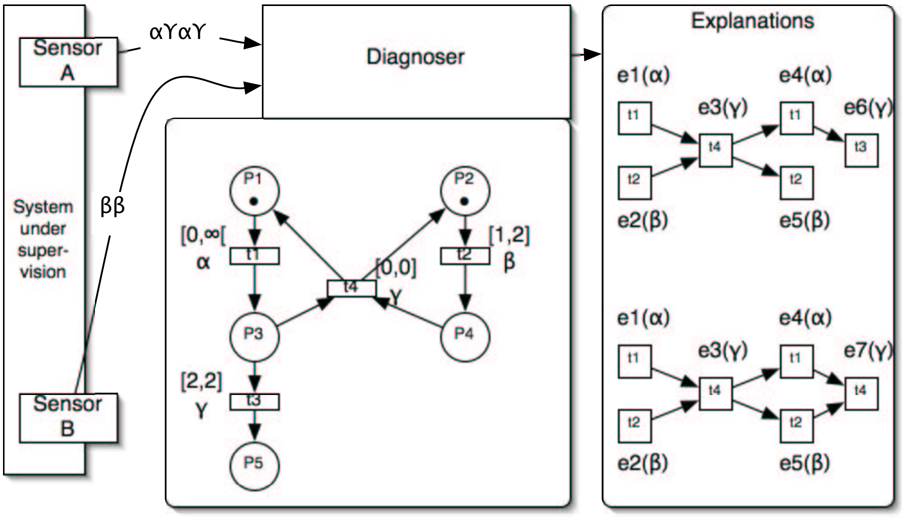

Fig. 1. Model-based diagnosis of distributed systems using time Petri nets.

2.2 Time Petri net: definition

Notations. We denote f^{-1} the inverse of a bijection f. We denote $f_{|A}$ the restriction of a mapping f to a set A. The restriction has higher priority than the inverse: $f_{|A}^{-1} = (f_{|A})^{-1}$. We denote $\circ$ the usual composition of functions. Q denotes the set of nonnegative rational numbers.

Time Petri nets were introduced in [11].

A *time Petri net* (TPN) is a tuple $N = \langle P, T, pre, post, efd, lfd \rangle$ where P and T are finite sets of *places* and *transitions* respectively, *pre* and *post* map each transition $t \in T$ to its *preset* often denoted ${}^\bullet t \overset{\text{def}}{=} pre(t) \subseteq P$ (${}^\bullet t \neq \emptyset$) and its *postset* often denoted $t^\bullet \overset{\text{def}}{=} post(t) \subseteq P$; $efd : T \longrightarrow Q$ and $lfd : T \longrightarrow Q \cup \{\infty\}$ associate the *earliest firing delay* $efd(t)$ and *latest firing delay* $lfd(t)$ with each transition t. A TPN is represented as a graph with two types of nodes: places (circles) and transitions (bars). The closed interval $[efd(t), lfd(t)]$ is written near each transition. For the purpose of supervision, we consider *labelled time Petri nets* $\langle N, \Lambda, \lambda \rangle$ where Λ is a set of event types (or alarms), and λ the typing of transitions (α, β, γ in Figure 1).

A *state* of a time Petri net is given by a triple $\langle M, dob, \theta \rangle$, where $M \subseteq P$ is a *marking* denoted with tokens (thick dots), $\theta \in Q$ is its date and $dob : M \longrightarrow Q$ associates a *date of birth* $dob(p) \leq \theta$ with each token (marked place) $p \in M$. A transition $t \in T$ is *enabled* in the state $\langle M, dob, \theta \rangle$ if all of its input places are marked: ${}^\bullet t \subseteq M$. Its *date of enabling* $doe(t)$ is the date of birth of the youngest token in its input places: $doe(t) \overset{\text{def}}{=} \max_{p \in {}^\bullet t} dob(p)$. All the time Petri nets we consider in this article are *safe*, i.e. in each reachable state $\langle M, dob, \theta \rangle$, if a transition t is enabled in $\langle M, dob, \theta \rangle$, then $t^\bullet \cap (M \setminus {}^\bullet t) = \emptyset$.

A process of a TPN starts in an *initial state* $\langle M_0, dob_0, \theta_0 \rangle$, which is given by the *initial marking* M_0 and the initial date θ_0. Initially, all the tokens carry the date θ_0 as date of birth: $\forall p \in M_0 \quad dob_0(p) \overset{\text{def}}{=} \theta_0$.

The transition t can fire at date θ' from state $\langle M, dob, \theta \rangle$, if:

- t is enabled: ${}^\bullet t \subseteq M$;
- the minimum delay is reached: $\theta' \geq doe(t) + efd(t)$;
- time progresses: $\theta' \geq \theta$;
- the enabled transitions do not overtake the maximum delays:
  $\forall t' \in T \quad {}^\bullet t' \subseteq M \implies \theta' \leq doe(t') + lfd(t')$.

The firing of t at date θ' leads to the state $\langle (M \setminus {}^\bullet t) \cup t^\bullet, dob', \theta' \rangle$, where $dob'(p) \overset{\text{def}}{=} dob(p)$ if $p \in M \setminus {}^\bullet t$ and $dob'(p) \overset{\text{def}}{=} \theta'$ if $p \in t^\bullet$.

Finally we assume that time *diverges*: when infinitely many transitions fire, time necessarily diverges to infinity.

In the initial state of the net of Figure 1, p_1 and p_2 are marked and their date of birth is 0. t_1 and t_2 are enabled and their date of enabling is the initial date 0. t_2 can fire in the initial state at any time between 1 and 2. Choose time 1. After this firing p_1 and p_4 are marked, t_1 is the only enabled transition and it has already waited 1 time unit. t_1 can fire at any time θ, provided it is greater than 1. Consider t_1 fires at time 3. p_3 and p_4 are marked in the new state, and transitions t_3 and t_4 are enabled, and their date of enabling is 3 because they have just been enabled by the firing of t_1. To fire, t_3 would have to wait 2 time units. But transition t_4 cannot wait at all. So t_4 will necessarily fire (at time 3), and t_3 cannot fire.

Remark. The semantics of time Petri nets are often defined in a slightly different way: the state of the net is given as a pair $\langle M, I\rangle$, where M is the marking, and I maps each enabled transition t to the delay that has elapsed since it was enabled, that is $\theta - doe(t)$ with our notations. It is more convenient for us to attach time information on the tokens of the marking than on the enabled transitions. We have chosen the date of birth of the tokens rather than their age, because we want to make the impact of the firing of transitions as local as possible. And the age of each token in the marking must be updated each time a transition t fires, whereas the date of birth has to be set only for the tokens that are created by t. Furthermore, usual semantics often deal with the delay between the firing of two consecutive transitions. In this paper we use the absolute firing date of the transitions instead. This fits better to our approach in which we are not interested in the total ordering of the events.

2.3 Partial order semantics

Processes. We will define the set X of (finite) processes of a safe time Petri net starting at date θ_0 in the initial marking M_0. These processes are those described in [1]. We define them inductively and use a canonical coding like in [8]. The processes provide a partial order representation of the executions.

Each process will be a pair $x \stackrel{\text{def}}{=} \langle E, \Theta\rangle$, where E is a set of *events*, and $\Theta : E \longrightarrow Q$ maps each event to its firing date. Θ is sometimes represented as a set of pairs $(e, \Theta(e))$. Each event e is a pair $({}^\bullet e, \tau(e))$ that codes an occurrence of the transition $\tau(e)$ in the process. ${}^\bullet e$ is a set of pairs $b \stackrel{\text{def}}{=} ({}^\bullet b, \mathit{place}(b)) \in E \times P$. Such a pair is called a *condition* and refers to the token that has been created by the event ${}^\bullet b$ in the place $\mathit{place}(b)$. We say that the event $e \stackrel{\text{def}}{=} ({}^\bullet e, \tau(e))$ *consumes* the conditions in ${}^\bullet e$. Symmetrically the set $\{(e, p) \mid p \in \tau(e)^\bullet\}$ of conditions that are *created* by e is denoted $e^\bullet$.

For all set B of conditions, we denote $\mathit{Place}(B) \stackrel{\text{def}}{=} \{\mathit{place}(b) \mid b \in B\}$, and when the restriction of place to B is injective, we denote $\mathit{place}^{-1}_{|B}$ its inverse, and for all $P \subseteq \mathit{Place}(B)$, $\mathit{Place}^{-1}_{|B}(P) \stackrel{\text{def}}{=} \{\mathit{place}^{-1}_{|B}(p) \mid b \in P\}$.

The set of conditions that remain at the end of the process $\langle E, \Theta\rangle$ (meaning that they are created by an event of E, and no event of E consumes them) is $\uparrow(E) \stackrel{\text{def}}{=} \bigcup_{e\in E} e^\bullet \setminus \bigcup_{e\in E} {}^\bullet e$ (it does not depend on Θ). The state that is reached after the process $\langle E, \Theta\rangle$ is $\langle \mathit{Place}(\uparrow(E)), \mathit{dob}, \max_{e\in E} \Theta(e)\rangle$, where for all $p \in \mathit{Place}(\uparrow(E))$, $\mathit{dob}(p) \stackrel{\text{def}}{=} \Theta({}^\bullet b)$, with $b \stackrel{\text{def}}{=} \mathit{place}^{-1}_{|\uparrow(E)}(p)$.

We define inductively the set X of (finite) processes of a time Petri net starting at date θ_0 in the initial marking M_0 as follows:

- $\langle\{\bot\}, \{(\bot, \theta_0)\}\rangle \in X$, where $\bot \stackrel{\text{def}}{=} (\emptyset, \epsilon)$ represents the initial event. Notice that the initial event does not actually represent the firing of a transition, which explains the use of the special value $\epsilon \notin T$. For the same reason, the set of conditions that are created by $\bot$ is defined in a special way: $\bot^\bullet \stackrel{\text{def}}{=} \{(\bot, p) \mid p \in M_0\}$.

- For all process $\langle E, \Theta \rangle \in X$ leading to state $\langle M, dob, \theta \rangle$, if a transition t can fire at date θ' from state $\langle M, dob, \theta \rangle$, then $\langle E \cup \{e\}, \Theta \cup \{(e, \theta')\} \rangle \in X$, where the event $e \overset{\text{def}}{=} (Place^{-1}_{|\uparrow(E)}({}^\bullet t), t)$ represents this firing of t.

We define the relation $\rightarrow$ on the events as: $e \rightarrow e'$ iff $e^\bullet \cap {}^\bullet e' \neq \emptyset$. The reflexive transitive closure $\rightarrow^*$ of $\rightarrow$ is called the *causality* relation. For all event e, we denote $\lceil e \rceil \overset{\text{def}}{=} \{f \in E \mid f \rightarrow^* e\}$, and for all set E of events, $\lceil E \rceil \overset{\text{def}}{=} \bigcup_{e \in E} \lceil e \rceil$. We also define $cnds(E) \overset{\text{def}}{=} \bigcup_{e \in E} e^\bullet$ the set of *conditions* created by the events of E.

Two events of a process that are not causally related are said to be *concurrent.*

Symbolic processes. We choose to group the processes that differ only by their firing dates to obtain what we call a *symbolic process.*

A symbolic process of a time Petri net is a pair $\langle E, pred \rangle$ with $pred : (E \longrightarrow Q) \longrightarrow \mathbf{bool}$, such that for all mapping $\Theta : E \longrightarrow Q$, if $pred(\Theta)$, then $\langle E, \Theta \rangle \in X$.

In practice, *pred* is described by linear inequalities. Examples of symbolic processes are given in Figure 1. The first explanation groups all the processes formally defined as $\langle E, \Theta \rangle$ where E contains the six following events, with the associated firing dates (the initial event $\bot$ is not represented):

$$\begin{array}{ll}
1 = (\{(\bot, P_1)\}, t_1) & \Theta(1) \geq \Theta(\bot) \\
2 = (\{(\bot, P_2)\}, t_2) & 1 \leq \Theta(2) - \Theta(\bot) \leq 2 \\
3 = (\{(1, P_3), (2, P_4)\}, t_4) & \Theta(3) = \max\{\Theta(1), \Theta(2)\} \\
4 = (\{(3, P_1)\}, t_1) & \Theta(4) = \Theta(3) \\
5 = (\{(3, P_2)\}, t_2) & \Theta(5) = \Theta(3) + 2 \\
6 = (\{(4, P_3)\}, t_3) & \Theta(6) = \Theta(4) + 2
\end{array}$$

2.4 Diagnosis: formal problem setting

Consider a net N modeling a system and an observation O of this system, which associates a finite sequence of observed alarms $(\lambda_{s,1}, \ldots, \lambda_{s,n_s})$ with each sensor s. The set of sensors is denoted S. For each sensor s, Λ_s indicates which alarms the sensor observes.

To compute a diagnosis, we propose to build a net $\mathcal{D}(N, O)$ whose processes correspond to the processes of N which satisfy the observation O. The idea is to constrain the model by adding new places and transitions so that each transition of the model that sends an alarm to a sensor s is not allowed to fire until all the previous alarms sent to s have been observed.

To achieve this we create a place s_λ for each alarm λ that may be sent to the sensor s, plus one place $\bar{s}$. For each transition t that sends an alarm λ to the sensor s, we add s_λ to the postset of t. After the i^{th} alarm is sent to s, a new transition $t_{s,i}$ which models the observation of this alarm by s, removes the token from s_λ and creates a token in the place $\bar{s}$, meaning that the alarm has been observed. $\bar{s}$ is added to the preset of each transition that sends an alarm to s, so that it cannot fire before the previous alarm has been observed. The transitions $t_{s,i}$ are connected through places $p_{s,i}$ so that they must fire one after another.

Formally, for a net $N \overset{\text{def}}{=} \langle P, T, pre, post, efd, lfd\rangle$ and an observation O from a set S of sensors, we define a net $\mathcal{D}(N,O) \overset{\text{def}}{=} \langle P', T', wpre', pre', post', efd', lfd'\rangle$. This net is almost a time Petri net: a *weak preset* $wpre'(t) \subseteq pre'(t)$, denoted ${}^\circ t$ has been added for each transition $t \in T'$; only the date of birth of the tokens in the weak preset participate in the definition of the *date of weak enabling* of t, which replaces the date of enabling in the semantics: $dowe(t) \overset{\text{def}}{=} \max_{p \in {}^\circ t} dob(p)$. In the processes, for each event e, we denote ${}^\circ e \overset{\text{def}}{=} Place^{-1}_{|{}^\bullet e}({}^\circ \tau(e))$.

$\mathcal{D}(N,O)$ is defined as follows (where $\uplus$ denotes the disjoint union):

- $P' \overset{\text{def}}{=} P \uplus \{\bar{s} \mid s \in S\} \uplus \{s_\lambda \mid s \in S \wedge \lambda \in \Lambda_s\} \uplus \{p_{s,i} \mid s \in S,\ i = 0, \dots, n_s\}$;
- $T' \overset{\text{def}}{=} T \uplus \{t_{s,i} \mid s \in S,\ i = 1, \dots, n_s\}$;
- for all $t \in T$, $\quad wpre'(t) \overset{\text{def}}{=} pre(t), \quad pre'(t) \overset{\text{def}}{=} wpre'(t) \uplus \{\bar{s} \mid \lambda(t) \in \Lambda_s\}$, $post'(t) \overset{\text{def}}{=} post(t) \uplus \{s_{\lambda(t)} \mid \lambda(t) \in \Lambda_s\}$, $efd'(t) \overset{\text{def}}{=} efd(t)$ and $lfd'(t) \overset{\text{def}}{=} lfd(t)$;
- $wpre'(t_{s,i}) = pre'(t_{s,i}) \overset{\text{def}}{=} \{p_{s,i-1}, s_{\lambda_{s,i}}\}, \quad post'(t_{s,i}) \overset{\text{def}}{=} \{p_{s,i}, \bar{s}\}$ and $efd'(t_{s,i}) = lfd'(t_{s,i}) \overset{\text{def}}{=} 0$.

Figure 2 shows the net of Figure 1 constrained by the observation $\alpha\gamma\alpha\gamma$ from sensor A and $\beta\beta$ from sensor B.

We call *diagnosis of observation O on net N* any set of symbolic processes of $\mathcal{D}(N,O)$, which contain all the processes $\langle E, \Theta\rangle$ of $\mathcal{D}(N,O)$ such that: $\{p_{s,n_s} \mid s \in S\} \uplus \{\overline{s_\lambda} \mid s \in S \wedge \lambda \in \Lambda_s\} \subseteq Place(\uparrow(E))$. Unless the model contains loops of non observable events, these processes can be described by a finite set of symbolic processes. These processes can be projected to keep only the conditions and events which correspond to places and transitions of the model. Then we obtain all the processes of N that are compatible with the observation O, as shown in Figure 1. The construction of the explanations is based on the unfolding of $\mathcal{D}(N,O)$. The notion of unfolding allows us to use a compact representation of the processes by sharing the common prefixes. The temporal framework leads naturally to consider the new notion of symbolic unfolding that we detail in the following section.

3 Symbolic unfoldings of time Petri nets

Symbolic unfoldings have already been addressed in the context of high-level Petri nets [6]. In this section we define the symbolic unfolding of time Petri nets, i.e. a quite compact structure that contains all the possible processes and exhibits concurrency. Actually the time Petri nets are extended with weak presets, as required by our diagnosis approach (see Section 2.4). For symbolic unfoldings of classical time Petri nets (such that the underlying untimed Petri net is safe), consider that the weak preset ${}^\circ t$ of any transition $t \in T$ is equal to its preset ${}^\bullet t$.

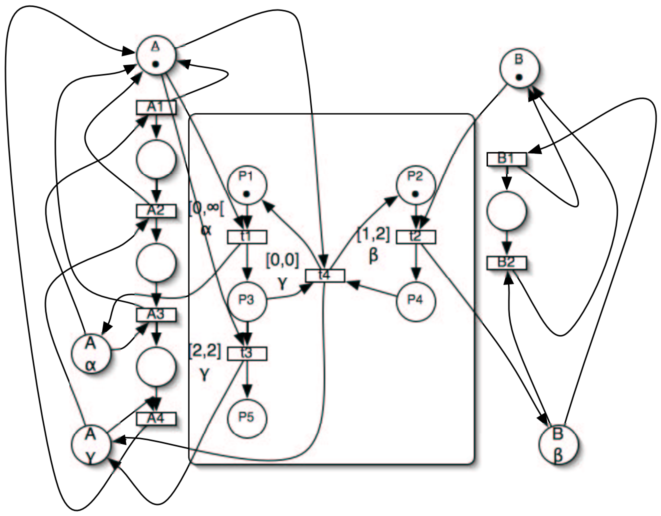

Fig. 2. Our example of TPN, constrained by the observation $\alpha\gamma\alpha\gamma$ from sensor A and $\beta\beta$ from sensor B.

3.1 Pre-processes

For the construction of symbolic unfoldings of time Petri nets, we need the notion of *pre-process*, that extends the notion of process.

For all process $\langle E, \Theta \rangle$, and for all nonempty, causally closed set of events $E' \subseteq E$ ($\bot \in E'$ and $\lceil E' \rceil = E'$), $\langle E', \Theta_{|E'} \rangle$ is called a *pre-process*. The definition of the state that is reached after a process is also used for pre-processes. We define the *prefix* relation $\leq$ on pre-processes as follows:

$$\langle E, \Theta \rangle \leq \langle E', \Theta' \rangle \quad \text{iff} \quad E \subseteq E' \ \wedge \ \Theta = \Theta'_{|E}$$

3.2 Symbolic unfoldings of extended free choice time Petri nets

An *extended free choice* time Petri net is a time Petri net such that:

$$\forall t, t' \in T \quad {}^{\bullet}t \cap {}^{\bullet}t' \neq \emptyset \implies {}^{\bullet}t = {}^{\bullet}t'.$$

We define the *symbolic unfolding* U of an extended free choice time Petri net by collecting all the events that appear in its processes: $U \overset{\text{def}}{=} \bigcup_{\langle E, \Theta \rangle \in X} E$.

This unfolding has two important properties in the case of extended free choice time Petri nets:

Theorem 1. *Let $E \subseteq U$ be a nonempty finite set of events and $\Theta : E \longrightarrow Q$ associate a firing date with each event of E. $\langle E, \Theta \rangle$ is a pre-process iff:*

$$\begin{cases} \lceil E \rceil = E & \textit{(E is causally closed)} \\ \nexists e, e' \in E \quad e \neq e' \ \wedge \ {}^{\bullet}e \cap {}^{\bullet}e' \neq \emptyset & \textit{(E is conflict free)} \\ \forall e \in E \setminus \{\bot\} \quad \mathit{lpred}(e, \Theta) & \textit{(all the events respect the firing delays)} \end{cases}$$

where

$$\mathit{lpred}(e,\Theta) \overset{\text{def}}{=} \begin{cases} \Theta(e) \geq \max_{b \in {}^\bullet e} \Theta({}^\bullet b) & \textit{(t is strongly enabled when e fires)} \\ \Theta(e) \geq \mathit{dowe}(t) + \mathit{efd}(t) & \textit{(the earliest firing delay is reached)} \\ \forall t' \in T \quad {}^\bullet t' = {}^\bullet t \implies \Theta(e) \leq \mathit{dowe}(t') + \mathit{lfd}(t') & \\ & \textit{(the latest firing delays are respected)} \end{cases}$$

with $t \overset{\text{def}}{=} \tau(e)$ *and*
for all $t' \in T$ *such that* ${}^\bullet t' = {}^\bullet t$, $\mathit{dowe}(t') \overset{\text{def}}{=} \max_{b \in \mathit{Place}^{-1}_{|{}^\bullet e}({}^\circ t')} \Theta({}^\bullet b)$.

Theorem 2. *For all* $e \overset{\text{def}}{=} (B,t) \in \mathit{cnds}(U) \times T$,

$$e \in U \ \textit{iff} \ \begin{cases} \mathit{Place}(B) = {}^\bullet t \\ \nexists f, f' \in \lceil e \rceil \quad f \neq f' \ \wedge \ {}^\bullet f \cap {}^\bullet f' \neq \emptyset \\ \exists \Theta : \lceil e \rceil \longrightarrow Q \quad \forall f \in \lceil e \rceil \setminus \{\bot\} \quad \mathit{lpred}(f,\Theta) \end{cases}$$

The first theorem gives a way to extract processes from the unfolding, while the second theorem gives a direct construction of the unfolding: adding a new event e just requires solving linear constraints on the $\Theta(f)$, $f \in \lceil e \rceil$. This also happens with symbolic unfoldings of high-level Petri nets introduced in [6].

3.3 Symbolic unfoldings of time Petri nets: general case

If we define the symbolic unfolding of a time Petri net in the general case as we have done for extended free choice time Petri nets, none of the two previous theorems hold: extracting a process from the unfolding becomes complex (see [1]); and especially we do not know any direct way to build the unfolding. It is also interesting to notice that the union of two pre-processes $\langle E, \Theta \rangle$ and $\langle E', \Theta' \rangle$ is not necessarily a pre-process, even if $\Theta_{|E \cap E'} = \Theta'_{|E \cap E'}$ and $E \cup E'$ is conflict free. In the example of Figure 1, we observe this if $\langle E, \Theta \rangle$ is the process which contains a firing of $t1$ at time 0 and a firing of $t2$ at time 1, and $\langle E', \Theta' \rangle$ is the pre-process that we obtain by removing the firing of $t2$ from the process made of $t1$ at time 0, $t2$ at time 2 and $t3$ at time 2. These difficulties come from the fact that the condition that allows us to extend a process $x \overset{\text{def}}{=} \langle E, \Theta \rangle$ with a new event e concerns all the state reached after the process x, and however the conditions in ${}^\bullet e$ refer only to the tokens in the input places of $\tau(e)$.

From now on we assume that we know a partition of the set P of places of the net in sets $P_i \subseteq P$ of mutually exclusive places[1]; more precisely we demand that for all reachable marking M, $P_i \cap M$ is a singleton. For all place $p \in P_i$, we denote $\bar{p} \overset{\text{def}}{=} P_i \setminus \{p\}$. In the example of Figure 1, we will use the partition $\{p_1, p_3, p_5\}$, $\{p_2, p_4\}$.

[1] If we do not know any such partition, a solution is to extend the structure of the net with one complementary place for each place of the net and to add these new places in the preset (but not in the weak preset) and in the postset of the transitions such that in any reachable marking each place $p \in P$ is marked iff its complementary place is not. This operation does not change the behaviour of the time Petri net: since the weak presets do not change, the tokens in the complementary places do not participate in the definition of the date of enabling.

Notion of partial state. A *partial state* of a time Petri net is a triple $\langle L, dob, \theta\rangle$ with $L \subseteq P$, $\theta \in Q$ is a date and $dob : L \longrightarrow Q$ associates a *date of birth* $dob(p) \leq \theta$ with each token (marked place) $p \in L$.

We define the relation $\sqsubseteq$ on partial states as follows:

$$\langle L, dob, \theta\rangle \sqsubseteq \langle L', dob', \theta'\rangle \text{ iff } L \subseteq L' \ \wedge \ dob = dob'_{|L} \ \wedge \ \theta \leq \theta'$$

Firing a transition from a partial state. Although the semantics of time Petri nets requires to check time conditions for all the enabled transitions in the net, before firing a transition, there are cases when we know that a transition can fire at a given date θ', even if other transitions will fire before θ' in other parts of the net. As an example consider the net of Figure 1 starting at date 0 with the marking $\{p_1, p_2\}$. Although the semantics forbids to fire t_1 at date 10 before firing t_2, we feel that nothing can prevent t_1 from firing at date 10, because only t_1 can remove the token in place p_1. By contrast, the firing of t_3 highly depends on the firing date of t_2 because when t_4 is enabled it fires immediately and disables t_3. So if we want to fire t_3 we have to check whether p_2 or p_4 is marked.

A *partial state firing* is a triple (S, t, θ') where $S \overset{\text{def}}{=} \langle L, dob_L, \theta_L\rangle$ is a partial state, t is a transition such that ${}^{\bullet}t \subseteq L$, and $\theta' \geq \theta_L$ is a date.

The idea in partial state firings is that the partial state S gives enough information to be sure that t can fire at date θ'.

It will be crucial in the following to know how to select partial state firings. However several choices are possible. If we are given a predicate *PSF* on partial state firings, we can build extended processes by using only the extended processes that satisfy *PSF*. Then we will try to map these extended processes into pre-processes. If *PSF* is valid, then all the pre-processes we obtain are correct.

Extended processes. Let *PSF* be a predicate on partial state firings. We will define a notion of *extended process* (parameterized by *PSF*), which is close to the notion of process, but the events are replaced by *extended events* which represent firings from partial states and keep track of all the conditions corresponding to the partial state, not only those that are consumed by the transition.

For all extended event $\dot{e} \overset{\text{def}}{=} \langle B, t\rangle$, we use the same notations as for events: ${}^{\bullet}\dot{e} \overset{\text{def}}{=} B$ and $\tau(\dot{e}) \overset{\text{def}}{=} t$, and we define $\dot{e}^{\bullet} \overset{\text{def}}{=} \{(\dot{e}, p) \mid p \in (\mathit{Place}(B) \setminus {}^{\bullet}t) \cup t^{\bullet}\}$. For all place $p \in \mathit{Place}(\dot{e}^{\bullet})$, we define the extended event $\mathit{origin}(p, \dot{e})$ that actually created the token in the place p:

$$\mathit{origin}(p, \dot{e}) \overset{\text{def}}{=} \begin{cases} \dot{e} & \text{if } p \in t^{\bullet} \text{ or } \dot{e} = \bot \\ \mathit{origin}(p, {}^{\bullet}b) \text{ with } b \overset{\text{def}}{=} \mathit{place}_{|B}^{-1}(p) & \text{otherwise} \end{cases}$$

Like for processes, we define the set of conditions that remain at the end of the extended process $\langle \dot{E}, \Theta\rangle$ as $\uparrow(\dot{E}) \overset{\text{def}}{=} \bigcup_{\dot{e} \in \dot{E}} \dot{e}^{\bullet} \setminus \bigcup_{\dot{e} \in \dot{E}} {}^{\bullet}\dot{e}$. But for extended processes we define not only the global state that is reached after $\langle \dot{E}, \Theta\rangle$, but a partial state associated with each set of conditions $B \subseteq \uparrow(\dot{E})$. The partial state associated with B is $\langle L, dob_L, \theta_L\rangle$, where:

- $L \overset{\text{def}}{=} \mathit{Place}(B)$,
- $dob_L(p) \overset{\text{def}}{=} \Theta(\mathit{origin}(p, {}^\bullet b))$ with $b \overset{\text{def}}{=} \mathit{place}^{-1}_{|B}(p)$,
- $\theta_L \overset{\text{def}}{=} \max_{b \in B} \Theta({}^\bullet b)$.

We define the set $\dot{X}$ of *extended processes* of a time Petri net starting at date θ_0 in the initial marking M_0 as follows.

- Like for processes, $\langle \{\bot\}, \{(\bot, \theta_0)\} \rangle \in \dot{X}$, where $\bot \overset{\text{def}}{=} (\emptyset, \epsilon)$ represents the initial event. The set of conditions that are created by $\bot$ is defined as: $\bot^\bullet \overset{\text{def}}{=} \{(\bot, p) \mid p \in M_0\}$.
- For all extended process $\langle \dot{E}, \Theta \rangle \in \dot{X}$, for all $B \subseteq \uparrow(\dot{E})$ leading to the partial state S, for all t, θ', if $\mathit{PSF}(S, t, \theta')$, then $\langle \dot{E} \cup \{\dot{e}\}, \Theta \cup \{(\dot{e}, \theta')\} \rangle \in \dot{X}$, where the extended event $\dot{e} \overset{\text{def}}{=} (B, t)$ represents this firing of t.

Each extended event $\dot{e} \overset{\text{def}}{=} (B, t)$ can be mapped to the corresponding event $h(\dot{e}) \overset{\text{def}}{=} (B', t)$ with $B' \overset{\text{def}}{=} \Big\{ \big(h(\mathit{origin}(p, \dot{f})), p\big) \mid (\dot{f}, p) \in \mathit{Place}^{-1}_{|B}({}^\bullet t) \Big\}$.

Corectness of *PSF*. We say that PSF is a *valid predicate on partial state firings* iff for all extended process $\langle \dot{E}, \Theta \rangle \in \dot{X}$, $\langle h(\dot{E}), \Theta \circ h^{-1}_{|\dot{E}} \rangle$ is a pre-process (notice that $h_{|\dot{E}}$ is injective). In other terms there exists a process $\langle E', \Theta' \rangle \in X$ such that $\langle h(\dot{E}), \Theta \circ h^{-1}_{\dot{E}} \rangle \leq \langle E', \Theta' \rangle$.

Symbolic unfolding. As we did for extended free choice time Petri nets with events in Section 3.2, we define the *symbolic unfolding* U of a time Petri net by collecting all the extended events that appear in its extended processes: $U \overset{\text{def}}{=} \bigcup_{\langle \dot{E}, \Theta \rangle \in \dot{X}} \dot{E}$.

We have equivalents of the two theorems we had with symbolic unfoldings of extended free choice time Petri nets.

Theorem 3. *Let $\dot{E} \subseteq U$ be a nonempty finite set of extended events and $\Theta : \dot{E} \longrightarrow Q$ associate a firing date with each extended event of $\dot{E}$. $\langle \dot{E}, \Theta \rangle$ is an extended process iff:*

$$\begin{cases} \lceil \dot{E} \rceil = \dot{E} & (\dot{E} \textit{ is causally closed}) \\ \nexists \dot{e}, \dot{e}' \in \dot{E} \quad \dot{e} \neq \dot{e}' \ \wedge \ {}^\bullet \dot{e} \cap {}^\bullet \dot{e}' \neq \emptyset & (\dot{E} \textit{ is conflict free}) \\ \forall \dot{e} \in \dot{E} \setminus \{\bot\} \quad \mathit{PSF}(S, \tau(\dot{e}), \Theta(\dot{e})) & (e \textit{ corresponds to a partial state firing}) \\ \qquad \textit{where } S \textit{ is the partial state associated with } {}^\bullet \dot{e}. & \end{cases}$$

Theorem 4. *For all $\dot{e} \overset{\text{def}}{=} (B, t) \in \mathit{cnds}(U) \times T$,*

$$\dot{e} \in U \textit{ iff } \begin{cases} \nexists \dot{f}, \dot{f}' \in \lceil \dot{e} \rceil \quad \dot{f} \neq \dot{f}' \ \wedge \ {}^\bullet \dot{f} \cap {}^\bullet \dot{f}' \neq \emptyset \\ \exists \Theta : \lceil \dot{e} \rceil \longrightarrow Q \quad \forall \dot{f} \in \lceil \dot{e} \rceil \setminus \{\bot\} \quad \mathit{PSF}(S, \tau(\dot{f}), \Theta(\dot{f})) \\ \qquad \textit{where } S \textit{ is the partial state associated with } {}^\bullet \dot{f}. \end{cases}$$

Selecting partial state firings. The definition of extended processes is parameterized by a predicate *PSF* on partial state firings: each extended event must correspond to a partial firing that satisfies *PSF*, the others are forbidden. A good choice for *PSF* takes three notions into account: completeness, redundancy and preservation of concurrency.

Completeness. A predicate *PSF* on partial state firings is *complete* if for all process $\langle E, \Theta \rangle \in X$, there exists an extended process $\langle \dot{E}, \Theta' \rangle \in \dot{X}$ (with partial state firings in *PSF*) such that $\langle h(\dot{E}), \Theta' \circ h_{\dot{E}}^{-1} \rangle = \langle E, \Theta \rangle$.

Redundancy. Given a predicate *PSF* on partial state firings and a process $\langle E, \Theta \rangle \in X$, there may exist several extended processes $\langle \dot{E}, \Theta' \rangle \in \dot{X}$ (with partial state firings in *PSF*) such that $\langle h(\dot{E}), \Theta' \circ h_{\dot{E}}^{-1} \rangle = \langle E, \Theta \rangle$. This is called *redundancy.* In particular, if *PSF* contains two partial state firings $(\langle L, dob, \theta \rangle, t, \theta')$ and $(\langle L', dob', \theta \rangle, t, \theta')$ where $L' \subsetneq L$ and $dob' = dob_{|L'}$, then all the extended processes involving $(\langle L, dob, \theta \rangle, t, \theta')$ are redundant.

A trivial choice for PSF which does not preserve any concurrency. A trivial complete predicate *PSF* is the predicate that demands that the state S is a global state, and then check that t can fire at date θ' from S. In addition, this choice gives little redundancy. But the extended events of the extended processes that we obtain in this case are totally ordered by causality. In other words, these processes do not exhibit any concurrency at all. Actually we get what we call firing sequences in interleaving semantics.

A proposition for PSF. What we want is a complete predicate on partial state firings that generates as little redundancy as possible and that exhibits as much concurrency as possible.

We first define a predicate *PSF*′ on partial state firings as follows:
$PSF'(\langle L, dob_L, \theta \rangle, t, \theta')$ iff

- t is enabled: ${}^\bullet t \subseteq L$;
- the minimum delay is reached: $\theta' \geq doe(t) + efd(t)$;
- time progresses: $\theta' \geq \theta$;
- the transitions that may consume tokens of L are disabled or do not overtake the maximum delays:
$$\forall t' \in T \quad {}^\bullet t' \cap L \neq \emptyset \implies \begin{cases} \quad \exists p \in {}^\bullet t' \quad \bar{p} \cap L \neq \emptyset \\ \vee\ \theta' \leq \max\limits_{p \in {}^\circ t' \cap L} dob(p) + lfd(t') \end{cases}$$

Now we define *PSF* by eliminating some redundancy in *PSF*′:
$PSF(\langle L, dob, \theta \rangle, t, \theta')$ iff $PSF'(\langle L, dob, \theta \rangle, t, \theta')$ and there exists no $L' \subsetneq L$ such that $PSF'(\langle L', dob_{|L'}, \theta \rangle, t, \theta')$.

It is important that the constraints solving (see Theorems 3 and 4) can be done automatically: with the definition of *PSF* we have proposed here, the quantifiers ($\forall$ and $\exists$) on places and transitions expand into disjunctions and conjunctions. The result is a disjunction of conjunctions of linear inequalities on the $\Theta(\dot{e})$. When a "max" appears in an inequality, this inequality can be rewritten into the desired form. These systems are shown near the events in Figure 3.

Theorem 5. *PSF is a valid, complete predicate on partial state firings.*

3.4 Example of unfolding

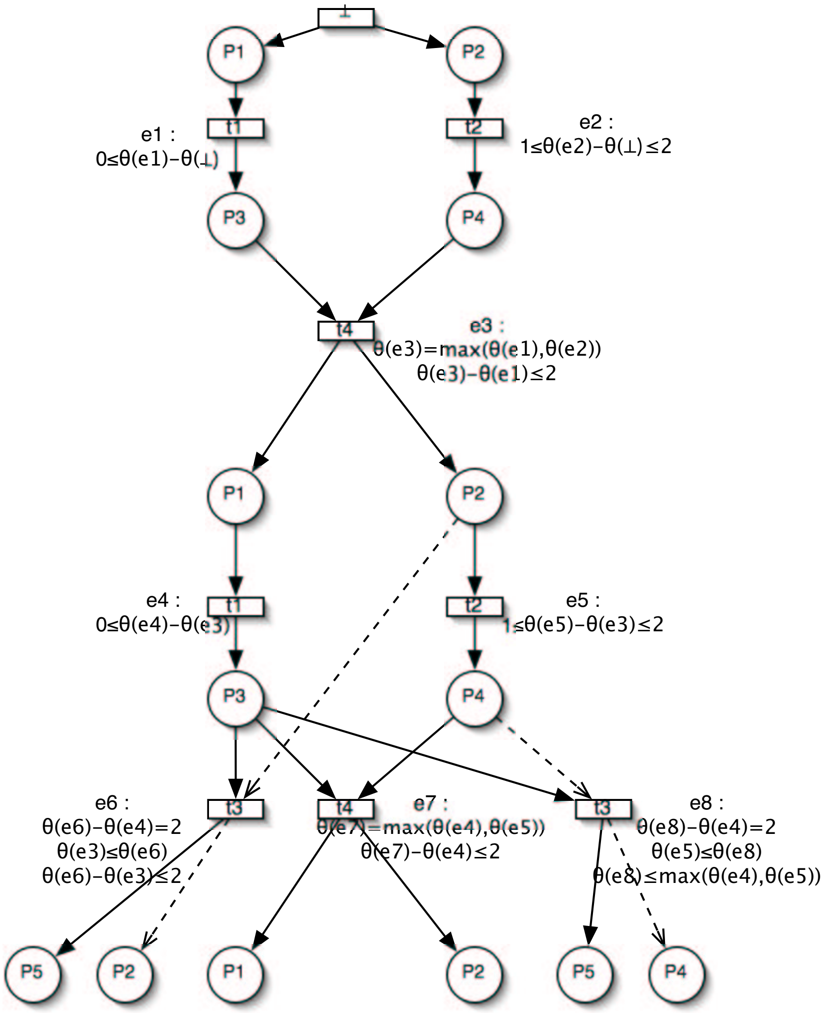

Fig. 3. A prefix of the symbolic unfolding of the time Petri net of Figure 1.

We come back to our simple example of time Petri net given in Figure 1. Figure 3 shows a prefix of its symbolic unfolding. We have kept all the events concerned with the observations (this filtering is done by considering the time net $\mathcal{D}(N, O)$ defined in subsection 2.4). To keep the figure readable, we do not show in the unfolding the supplementary places and transitions induced by the observations).

In this unfolding we see three explanations as extended processes. In contrast, the explanations of Figure 1 are the symbolic processes that have been computed from the extended processes. The linear constraints that appear near the events of Figure 3 can be solved in order to find all the possible values for the dates of the events. The three maximal extended processes of Figure 3 share the prefix $\{e1, e2, e3, e4\}$. The first extended process contains also $e5$ and $e7$. It corresponds to the second explanation of Figure 1. The second extended process contains the prefix, plus $e5$ and $e8$ and the third contains the prefix, plus $e6$ and $e9$. These two extended processes correspond to the same explanation: the first of Figure 1. This is what we have called redundancy. After solving the linear constraints we see that the second occurrence of t_1 must have occured immediately after $t4$ has fired and the second occurrence of t_2 must have fired 2 time units later. Actually the extended process with $e5$ and $e8$ and the one with $e6$ and $e9$ only differ by the fact that transition t_2 has fired *before* t_3 in the first one, whereas t_2 has fired *after* t_3 in the second one. Indeed, because of transition t_4, the firing of t_2 has a strong influence on the firing of t_3. This is the reason why there are too distinct cases in the unfolding.

4 Conclusion

We have presented a possible approach to the supervision/diagnosis of timed systems, using safe time Petri nets. In such nets, time constraints are given by interval of nonnegative rationals and are used to restrict the set of behaviours. The diagnosis problem is to recover the possible behaviours from a set of observations. We consider that the observations are given as a partial order (without any timing information) from the activity of several sensors. The goal of the supervisor is to select the possible timed behaviours of the model, which do not contradict the observations: i.e. presents the same set of events labelled by the alarms and orders the events in the same direction that the sensors do. This goal is achevied by considering a symbolic unfolding of time Petri nets, which is restricted by the observations. The result is a set of explanations, which explicit the causalities (both structural and temporal) between the observations. At the same time, our algorithm infers the possible delays before the firing of the transitions associated with them. Up to our knowledge, our symbolic unfolding for safe time Petri nets is original, and its application to compute symbolic explanations too.

A prototype implementation exists (a few thousands lines of Lisp code) and we plan to use it on real case studies. Another project is to define an algorithm to produce a complete finite prefix of the unfolding [9], which could be used for other applications than diagnosis (for which we do not need this notion since observations are finite sets).

At longer term, the notion of temporal diagnosis could be refined and revisited when considering timed distributed systems, in which alarms could bring a time information.

References

1. Tuomas Aura and Johan Lilius. Time processes for time Petri nets. In *ICATPN*, pages 136–155, 1997.
2. A. Benveniste, E. Fabre, C. Jard, and S. Haar. Diagnosis of asynchronous discrete event systems, a net unfolding approach. *IEEE Transactions on Automatic Control*, 48(5):714–727, May 2003.
3. Bernard Berthomieu and Michel Diaz. Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans. Software Eng.*, 17(3):259–273, 1991.
4. Bernard Berthomieu and François Vernadat. State class constructions for branching analysis of time Petri nets. In *TACAS*, pages 442–457, 2003.
5. Eike Best. Structure theory of Petri nets: the free choice hiatus. In *Proceedings of an Advanced Course on Petri Nets: Central Models and Their Properties, Advances in Petri Nets 1986-Part I*, pages 168–205, London, UK, 1987. Springer-Verlag.
6. Thomas Chatain and Claude Jard. Symbolic diagnosis of partially observable concurrent systems. In *FORTE*, pages 326–342, 2004.
7. Thomas Chatain and Claude Jard. Time supervision of concurrent systems using symbolic unfoldings of time petri nets. Technical Report RR-1740, Institut National de Recherche en Informatique et en Automatique (INRIA), 2005.
8. Joost Engelfriet. Branching processes of Petri nets. *Acta Inf.*, 28(6):575–591, 1991.
9. Javier Esparza, Stefan Römer, and Walter Vogler. An improvement of McMillan's unfolding algorithm. In *TACAS*, pages 87–106, 1996.
10. Hans Fleischhack and Christian Stehno. Computing a finite prefix of a time Petri net. In *ICATPN*, pages 163–181, 2002.
11. P.M. Merlin and D.J. Farber. Recoverability of communication protocols – implications of a theorical study. *IEEE Transactions on Communications*, 24, 1976.
12. B. Pradin-Chézalviel, R. Valette, and L.A. Künzle. Scenario duration characterization of t-timed Petri nets using linear logic. In *IEEE PNPM*, pages 208–217, 1999.
13. Alexei Semenov and Alexandre Yakovlev. Verification of asynchronous circuits using time Petri net unfolding. In *DAC'96: Proceedings of the 33rd annual conference on Design automation*, pages 59–62, New York, NY, USA, 1996. ACM Press.