Weak Time Petri Nets strike back!

Pierre-Alain Reynier, Arnaud Sangnier

Weak Time Petri Nets strike back!

Pierre-Alain Reynier[1] and Arnaud Sangnier[2,⋆]

[1] LIF, Université Aix-Marseille & CNRS, France
pierre-alain.reynier@lif.univ-mrs.fr
[2] Dipartimento di Informatica, Università di Torino, Italy
sangnier@di.unito.it

Abstract. We consider the model of Time Petri Nets where time is associated with transitions. Two semantics for time elapsing can be considered: the strong one, for which all transitions are urgent, and the weak one, for which time can elapse arbitrarily. It is well known that many verification problems such as the marking reachability are undecidable with the strong semantics. In this paper, we focus on Time Petri Nets with weak semantics equipped with three different memory policies for the firing of transitions. We prove that the reachability problem is decidable for the most common memory policy (intermediate) and becomes undecidable otherwise. Moreover, we study the relative expressiveness of these memory policies and obtain partial and surprising results.

1 Introduction

Verification of timed and distributed systems. For verification purpose, *e.g.* in the development of embedded platforms, there is an obvious need for considering time features and the study of timed models has thus become increasingly important. For distributed systems, different timed extensions of Petri nets have been proposed which allow the combination of an unbounded discrete structure with dense-time variables.

About time-elapsing policies. There are several ways to express urgency in timed systems, as discussed in [15]. In timed extensions of Petri nets, two types of semantics are considered for time elapsing. In the *weak* semantics, all time delays are allowed whereas in the *strong* one, all transitions are urgent, *i.e.* time delays can not disable transitions. While for models with finite discrete structure (such as timed extensions of bounded Petri nets or timed automata [3]), standard verification problems are decidable for both semantics, for models with infinite discrete structure, the choice of the semantics has a deep influence on decidability issues. For instance, in the model of timed-arc Petri nets [5], where clocks are associated with tokens and which is equipped with a weak semantics, many verification problems are decidable (coverability, boundedness...). Indeed, this semantics entails for this model monotonicity properties which allow the application of well-quasi-ordering techniques, see [7, 2, 1]. Note however that the reachability of a discrete marking is undecidable, as proven in [16]. On the other side, in the model of Time Petri Nets [13] (TPN), where clocks are associated with transitions, and which is commonly considered under a strong semantics, all the standard verification problems are known to be undecidable [9]. A natural question, which

⋆ Supported by a post-doctoral scolarship from DGA/ENS Cachan

had surprisingly no answer until now, as mentioned in a recent survey on the topic [6], is thus to study TPN under a weak semantics of time elapsing. Note that due to the semantics of TPN, there is no hope for monotonicity properties in this model.

Memory policies in TPN. The time-elapsing policy states which delays are allowed in a configuration. The memory policy is concerned with the resets of clocks, and intuitively specifies, when firing a transition, which timing informations are preserved. In the original model of Merlin [13], an *intermediate* semantics was considered, meaning that when firing a transition t, any transition disabled by the consumption of tokens by t is reinitialized. Two others memory policies have been considered in [4]: the *atomic* and the *persistent atomic*; for both of them the firings of transitions are performed atomically and the difference between them lies in considering or not a transition as newly enabled by its own firing. While these policies can be thought as cosmetic for the model of TPN, the results we obtain show this is not the case.

Our contributions. We are interested in the impact of the weak semantics on TPN. We first study the decidability issues, and therefore have to distinguish the memory policy considered. We prove that for TPN with weak intermediate semantics, a discrete marking is reachable if and only if it is reachable in the underlying untimed Petri net. As a corollary, the problem of the marking reachability (and also coverability, boundedness) is decidable for this model. More surprisingly, we also prove that when changing the memory policy this result does not hold anymore and the verification problems become undecidable (we encode a Minsky machine). We then compare w.r.t. weak time bisimilarity (weak stands here for silent transitions) the expressive power of weak TPN looking at the different memory policies. We first prove that the persistent atomic semantics is strictly more expressive that the atomic semantics. Then, concerning the comparison of atomic and intermediate memory policies, we prove that the translation proposed in [4] for the strong semantics is not correct (for weak and strong semantics). Finally, we provide a TPN which allows to prove that the atomic semantics is not included in the intermediate one.

Related works. As mentioned above, there are, up to our knowledge, only very few works considering TPN under a weak semantics. In [6] the authors have proven that the weak intermediate semantics and the strong intermediate semantics are incomparable. In another line of work, [8] consider TPN under a semantics which is a kind of compromise between the standard strong and weak semantics. They provide translations between this model and timed state machines.

2 Definitions

Let Σ be a finite alphabet, Σ^* is the set of finite words over Σ. We note $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$ with ε (the empty word) not in Σ. The sets $\mathbb{N}$, $\mathbb{Q}$, $\mathbb{Q}_{\geq 0}$, $\mathbb{R}$ and $\mathbb{R}_{\geq 0}$ are respectively the sets of natural, rational, non-negative rational, real and non-negative real numbers. A *valuation* v over a finite set X is a mapping in $\mathbb{R}_{\geq 0}^X$. For $v \in \mathbb{R}_{\geq 0}^X$ and $d \in \mathbb{R}_{\geq 0}$, $v + d$ denotes the valuation defined by $(v+d)(x) = v(x) + d$. We note $\mathbf{0}$ the valuation which assigns to every clock $x \in X$ the value 0.

We introduce a notation to define intervals of reals with rational bounds. We denote by $\mathcal{B}$ the set $\{<, \leq\} \times (\mathbb{Q} \cup \{\infty\})$ we will use to characterize the bounds of intervals. We

define the relation of satisfaction $\models$ of a bound $(\prec, a)$ by a real number y, by $y \models (\prec, a)$ if and only if $y \prec a$. We define the order $\sqsubset$ on $\mathcal{B}$ by, for $b_i = (\prec_i, a_i)$ with $i \in \{1, 2\}$:

$$b_1 \sqsubset b_2 \iff \begin{cases} a_1 < a_2, \text{ or,} \\ a_1 = a_2 \text{ and } \prec_1 \in \{<\} \text{ and } \prec_2 \in \{\leq\} \end{cases}$$

Furthermore, we denote by $b_1 \sqsubseteq b_2$ the fact that $b_1 \sqsubset b_2$ or $b_1 = b_2$. Intuitively, a larger bound for this order means that it is less restrictive. Formally, given two bounds $b_1 \sqsubseteq b_2$, we have $\forall y \in \mathbb{R}, y \models b_1 \Rightarrow y \models b_2$. Given a finite set of bounds $F \subset \mathcal{B}$, we denote by $\min(F)$ the bound b of F minimal for the order $\sqsubseteq$. We also define an addition between a bound $b = (\prec, a)$ and a rational $q \in \mathbb{Q}$ as $b + q = (\prec, a + q)$.

We denote by $\mathcal{B}_{\geq 0}$ the restriction of $\mathcal{B}$ to $\{<, \leq\} \times (\mathbb{Q}_{\geq 0} \cup \{\infty\})$. We say that a pair $I \in \mathcal{B}_{\geq 0} \times \mathcal{B}_{\geq 0}$ is a non-empty $\mathbb{Q}_{\geq 0}$-*interval* if $I = (b_1, b_2)$ with $b_1 \sqsubset b_2$ and $b_2 \neq (\leq, \infty)$. We denote by $\mathcal{I}(\mathbb{Q}_{\geq 0})$ the set of non-empty $\mathbb{Q}_{\geq 0}$-intervals of $\mathbb{R}_{\geq 0}$. A real number y belongs to an interval $I = (b_1, b_2)$, denoted by $y \in I$, iff $y \not\models b_1$ and $y \models b_2$. For instance, the interval $]1, 2]$ is encoded as the pair of bounds $(\leq, 1)$ and $(\leq, 2)$.

2.1 Petri nets

Definition 1 (Labeled Petri Net (PN)). *A Labeled Petri Net over the alphabet Σ_ε is a tuple $(P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda)$ where:*

- *P is a finite set of* places,
- *T is a finite set of* transitions *with $P \cap T = \emptyset$,*
- *${}^\bullet(.) \in (\mathbb{N}^P)^T$ is the* backward *incidence mapping,*
- *$(.)^\bullet \in (\mathbb{N}^P)^T$ is the* forward *incidence mapping,*
- *$M_0 \in \mathbb{N}^P$ is the* initial *marking,*
- *$\Lambda : T \to \Sigma_\varepsilon$ is the* labeling function

As commonly in use in the literature, we write ${}^\bullet t$ (resp. $t^\bullet$) to denote the set of places ${}^\bullet t = \{p \in P \mid {}^\bullet t(p) > 0\}$ (resp. $t^\bullet = \{p \in P \mid t^\bullet(p) > 0\}$). The semantics of a PN $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda)$ is given by its associated labeled transition system $[\![\mathcal{N}]\!] = (\mathbb{N}^P, M_0, \Sigma_\varepsilon, \Rightarrow)$ where $\Rightarrow \subseteq Q \times \Sigma_\varepsilon \times Q$ is the transition relation defined by $M \stackrel{a}{\Longrightarrow} M'$ iff $\exists t \in T$ s.t. $\Lambda(t) = a \wedge M \geq {}^\bullet t \wedge M' = M - {}^\bullet t + t^\bullet$. For convenience we will sometimes also write, for $t \in T$, $M \stackrel{t}{\Longrightarrow} M'$ if $M \geq {}^\bullet t$ and $M' = M - {}^\bullet t + t^\bullet$. We also write $M \Rightarrow M'$ if there exists $a \in \Sigma_\varepsilon$ such that $M \stackrel{a}{\Longrightarrow} M'$. The relation $\Rightarrow^*$ represents the reflexive and transitive closure of $\Rightarrow$. We denote by $\texttt{Reach}(\mathcal{N})$ the set of reachable markings defined by $\{M \in \mathbb{N}^P \mid M_0 \Rightarrow^* M\}$.

It is well known that for PN the reachability problem which consists in determining whether a given marking M belongs to $\texttt{Reach}(\mathcal{N})$ is decidable; it has in fact been proved independently in [12] and [11].

We introduce a last notation concerning Labeled Petri Nets. Given a PN $\mathcal{N}$, a marking M of $\mathcal{N}$ and a multi-set $\Delta = \langle t_1, \ldots, t_n \rangle$ of transitions of $\mathcal{N}$, we write $M \stackrel{\Delta}{\Longrightarrow} M'$ if and only if the multi-set Δ can be fired from M, meaning that there exists an ordering of transitions in Δ, represented as a permutation φ of $\{1, \ldots, n\}$, such that the sequence of firings $M \stackrel{t_{\varphi(1)}}{\Longrightarrow} M_1 \stackrel{t_{\varphi(2)}}{\Longrightarrow} M_2 \ldots \stackrel{t_{\varphi(n)}}{\Longrightarrow} M'$ exists in $[\![\mathcal{N}]\!]$.

2.2 Timed Transition Systems

Timed transition systems describe systems which combine discrete and continuous evolutions. They are used to define the behavior of timed systems such as Time Petri Nets [13] or Timed Automata [3].

Definition 2 (Timed Transition System (TTS)). *A* timed transition system *over the alphabet Σ_ε is a transition system $S = (Q, q_0, \Sigma_\varepsilon, \rightarrow)$, where the transition relation $\rightarrow \subseteq Q \times (\Sigma_\varepsilon \cup \mathbb{R}_{\geq 0}) \times Q$ consists of continuous transitions $q \xrightarrow{d} q'$ (with $d \in \mathbb{R}_{\geq 0}$), and discrete transitions $q \xrightarrow{a} q'$ (with $a \in \Sigma_\varepsilon$).*

Moreover, we require the following standard properties for TTS :

- TIME-DETERMINISM : if $q \xrightarrow{d} q'$ and $q \xrightarrow{d} q''$ with $d \in \mathbb{R}_{\geq 0}$, then $q' = q''$,
- 0-DELAY : $q \xrightarrow{0} q$,
- ADDITIVITY : if $q \xrightarrow{d} q'$ and $q' \xrightarrow{d'} q''$ with $d,\ d' \in \mathbb{R}_{\geq 0}$, then $q \xrightarrow{d+d'} q''$,
- CONTINUITY : if $q \xrightarrow{d} q'$, then for every d' and d'' in $\mathbb{R}_{\geq 0}$ such that $d = d' + d''$, there exists q'' such that $q \xrightarrow{d'} q'' \xrightarrow{d''} q'$.

With these properties, a *run* of S can be defined as a finite sequence of moves $\rho = q_0 \xrightarrow{d_0} q'_0 \xrightarrow{a_0} q_1 \xrightarrow{d_1} q'_1 \xrightarrow{a_1} q_2 \ldots \xrightarrow{a_n} q_{n+1}$ where discrete and continuous transitions alternate. To such a run corresponds the timed word $w = (a_i, \tau_i)_{0 \leq i \leq n}$ over Σ_ε where $\tau_i = \sum_{j=0}^{i} d_j$ is the time at which a_i happens. We then denote by *Untimed*(w) the word $a_0 a_1 \ldots a_n$ over the alphabet Σ_ε and by *Duration*(w) the duration τ_n. We will sometimes apply, without possible ambiguities, these notations to runs writing *Untimed*(ρ) and *Duration*(ρ). We might also describe the run writing directly $q_0 \xrightarrow{w} q_{n+1}$. Finally, because of the characteristics of the empty word ε, we will consider in the sequel that for all timed words w, *Untimed*(w) is either a word over Σ (which means without ε) or the empty word ε.

2.3 Time Petri Nets

Syntax. Introduced in [13], Time Petri nets (TPNs) associate a time interval to each transition of a Petri net.

Definition 3 (TPN). *A Labeled Time Petri Net over the alphabet Σ_ε is a tuple $(P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$ where:*

- $(P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda)$ *is a PN,*
- $I : T \mapsto \mathcal{I}(\mathbb{Q}_{\geq 0})$ *associates with each transition a* firing interval.

In the sequel, we associate to an interval its left bound and its right bound. More generally, given a transition t of a TPN, we will denote by $eft(t)$ (resp. $lft(t)$) the left bound of $I(t)$ (resp. the right bound of $I(t)$), standing for earliest firing time (resp. latest firing time). We have hence $I(t) = (eft(t), lft(t))$. We also use the notion of granularity of a TPN, which intuitively corresponds to the precision of the TPN, and is defined as the least common multiple of denominators of (rational) bounds of intervals.

Semantics. A *configuration* of a TPN is a pair (M, ν), where M is a *marking* over P, *i.e.* a mapping in $\mathbb{N}^P$, with $M(p)$ the number of tokens in place p. A transition t is *enabled* in a marking M if $M \geq {}^\bullet t$. We denote by $\mathit{En}(M)$ the set of enabled transitions in M. The second component of the pair (M, ν) is a valuation over T, *i.e.* a mapping in $\mathbb{R}^T_{\geq 0}$. Intuitively, for any enabled transition t, $\nu(t)$ represents the amount of time that has elapsed since t is enabled. An enabled transition t can be fired if $\nu(t)$ belongs to the interval $I(t)$. The marking obtained after this firing is as usual the new marking $M' = M - {}^\bullet t + t^\bullet$. Moreover, some valuations are reset and we say that the corresponding transitions are newly enabled.

Different semantics can be chosen in order to realize these resets. This choice depends of what is called the *memory policy*. For $M \in \mathbb{N}^P$ and $t, t' \in T$ such that $t \in \mathit{En}(M)$ we define in different matters a predicate $\uparrow\mathit{enabled}(t', M, t)$ which is true if t' is *newly enabled* by the firing of transition t from marking M, and false otherwise:

I: The *intermediate semantics* considers that the firing of a transition is performed in two steps : consuming the tokens in ${}^\bullet t$, and then producing the tokens in $t^\bullet$. In this case, the newly enabled predicate, denoted $\uparrow\mathit{enabled}_I(t', M, t)$, is defined by:

$$\uparrow\mathit{enabled}_I(t', M, t) = \big(t' \in \mathit{En}(M - {}^\bullet t + t^\bullet) \wedge (t' \notin \mathit{En}(M - {}^\bullet t) \vee t = t')\big)$$

A: The *atomic semantics* considers that the firing of a transition is obtained by an atomic step. The corresponding predicate $\uparrow\mathit{enabled}_A(t', M, t)$ is defined by:

$$\uparrow\mathit{enabled}_A(t', M, t) = \big(t' \in \mathit{En}(M - {}^\bullet t + t^\bullet) \wedge (t' \notin \mathit{En}(M) \vee t = t')\big)$$

PA: The *persistent atomic semantics* considers that the firing of a transition is also obtained by an atomic step. The difference with the previous semantics appears only when $t = t'$; in fact here the transition just being fired is not always newly enabled.

$$\uparrow\mathit{enabled}_{PA}(t', M, t) = \big(t' \in \mathit{En}(M - {}^\bullet t + t^\bullet) \wedge t' \notin \mathit{En}(M)\big)$$

Finally, as recalled in the introduction, there are two ways of letting the time elapse in TPN. The first way, known as the *strong semantics*, is defined in such a matter that time elapsing cannot disable a transition. Hence, when the upper bound of a firing interval is reached then the transition must be fired or disabled. In contrast to that the *weak semantics* does not make any restriction on the elapsing of time. In this work, we focus on the weak semantics of TPN.

Definition 4 (Weak semantics of a TPN). *Let $s \in \{I, A, PA\}$. The weak s-semantics of a TPN $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$ is a timed transition system $[\![\mathcal{N}]\!]_s = (Q, q_0, \Sigma_\varepsilon, \rightarrow_s)$ where $Q = \mathbb{N}^P \times \mathbb{R}^T_{\geq 0}$, $q_0 = (M_0, \mathbf{0})$ and $\rightarrow_s$ consists of discrete and continuous moves :*

– *the discrete transition relation is defined $\forall a \in \Sigma_\varepsilon$ by :*

$$(M, \nu) \xrightarrow{a}_s (M', \nu') \text{ iff } \exists t \in T \text{ s.t. } \begin{cases} \Lambda(t) = a, \text{ and,} \\ t \in \mathit{En}(M) \wedge M' = M - {}^\bullet t + t^\bullet, \text{ and,} \\ \nu(t) \in I(t), \text{ and,} \\ \forall t' \in T, \\ \quad \nu'(t') = \begin{cases} 0 & \text{if } \uparrow\mathit{enabled}_s(t', M, t) \\ \nu(t') & \text{otherwise} \end{cases} \end{cases}$$

– *the continuous transition relation is defined* $\forall d \in \mathbb{R}_{\geq 0}$ *by :*

$$(M, \nu) \xrightarrow{d}_s (M, \nu') \textit{ iff } \nu' = \nu + d$$

We also write a discrete transition $(M, \nu) \xrightarrow{t}_s (M', \nu')$ to characterize the transition $t \in T$ which allows the firing $(M, \nu) \xrightarrow{\Lambda(t)}_s (M', \nu')$. We extend this notation to words $\theta \in (T \cup \mathbb{R}_{\geq 0})^*$, which correspond to sequences of delays and transitions and lead to a unique (if it exists) run ρ. We may write this run $\rho : (M, \nu) \xrightarrow{\theta}_s (M', \nu')$ and use *Untimed*(θ) (resp. *Duration*(θ)) to denote the word *Untimed*(ρ) (resp. to represent the delay *Duration*(ρ)). Finally, for $s \in \{I, A, PA\}$, we write $(M, \nu) \rightarrow_s (M', \nu')$ if there exists $a \in \mathbb{R}_{\geq 0} \cup \Sigma_\varepsilon$ such that $(M, \nu) \xrightarrow{a}_s (M', \nu')$. The relation $\rightarrow_s^*$ denotes the reflexive and transitive closure of $\rightarrow_s$. For a TPN $\mathcal{N}$ with an initial marking M_0 we define the following reachability sets according to the considered semantics : $\texttt{Reach}(\mathcal{N})_s = \{(M, v) \mid (M_0, \mathbf{0}) \rightarrow_s^* (M, v)\}$.

3 Decidability

3.1 Considered problems and known results

Assume $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$ is a TPN. In this section, we will consider the following problems for $s \in \{I, A, PA\}$:

(1) The *marking reachability problem* : given $M \in \mathbb{N}^P$, does there exist $\nu \in \mathbb{R}_{\geq 0}^T$ such that $(M, \nu) \in \texttt{Reach}(\mathcal{N})_s$?
(2) The *marking coverability problem* : given $M \in \mathbb{N}^P$, does there exist $M' \in \mathbb{N}^P$ and $\nu \in \mathbb{R}_{\geq 0}^T$ such that $M' \geq M$ and $(M', \nu) \in \texttt{Reach}(\mathcal{N})_s$?
(3) The *boundedness problem* : does there exist $b \in \mathbb{N}$ such that for all $(M, \nu) \in \texttt{Reach}(\mathcal{N})_s$ and for all $p \in P$, $M(p) \leq b$?

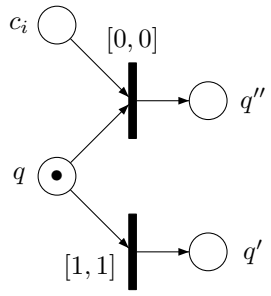

Fig. 1. Encoding decrementation with strong semantics.

It is well known that the "untimed" versions of these problems are decidable in the case of Petri nets. In fact, as mentioned before the marking reachability problem is decidable for Petri nets [11, 12] and the two other problems can be solved using the Karp and Miller tree whose construction is given in [10]. Furthermore, from [9], we know that these problems are all undecidable when considering TPN with strong semantics no matter whether the semantics is intermediate, atomic or persistent atomic. In fact a TPN with strong semantics can simulate a Minsky machine. A Minsky machine manipulates two integer variables c_1 and c_2 and is composed of a finite number of instructions, each of these instructions being either an incrementation ($q : c_i := c_i + 1$) or a decrementation with a test to zero (q : if $c_i = 0$ goto q' else $c_i := c_i - 1$; goto q''), where $i \in \{1, 2\}$ and q, q', q'' are some labels preceding each instruction. There is also a special label q_f from which the

machine cannot do anything. In [14], Minsky proved that the halting problem, which consists in determining if the instruction labeled with q_f is reachable, is undecidable.

It is easy to encode an incrementation using a TPN (or even a PN), with a transition consuming a token in a place characterizing the current control state and producing a token in the next control state and in a place representing the incremented counter. When encoding the decrementation with the test to zero, the strong semantics plays a crucial role. This encoding is represented on Figure 1. If there is a token in the place c_i, there is no way for the TPN to produce a token in the place q' because time cannot elapse since the transition labeled with the interval $[0, 0]$ is firable. This construction obviously fails with the weak semantics.

3.2 The peculiar case of TPN with weak intermediate semantics

We prove here that the undecidability results we had before in the case of TPN with strong semantics do not hold anymore when considering the weak intermediate semantics. Before proving this we introduce some notations. For a TPN $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$, we denote $\mathcal{N}^U$ the untimed PN obtained by removing from $\mathcal{N}$ the component I. Furthermore given a set of configurations $C \subseteq \mathbb{N}^P \times \mathbb{R}^T_{\geq 0}$ of $\mathcal{N}$, we denote by $\texttt{Untime}(C)$ the projection of C over the set $\mathbb{N}^P$. For $s \in \{I, A, PA\}$, we have by definition of the different semantics that $\texttt{Untime}(\texttt{Reach}(\mathcal{N})_s) \subseteq \texttt{Reach}(\mathcal{N}^U)$ and as shown by the example given in Figure 1 this inclusion is strict in the case of the strong semantics. When considering the **weak intermediate** semantics, we prove that from any sequence of transitions Δ firable in $[\![\mathcal{N}^U]\!]$, we can effectively compute a reordering of Δ, and associated timestamps, leading to a correct run of $[\![\mathcal{N}]\!]_I$.

Theorem 5. *For all TPN $\mathcal{N}$,* $\texttt{Untime}(\texttt{Reach}(\mathcal{N})_I) = \texttt{Reach}(\mathcal{N}^U)$.

Proof. Let $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$ be a TPN. According to the previous remark, we only have to prove that $\texttt{Reach}(\mathcal{N}^U) \subseteq \texttt{Untime}(\texttt{Reach}(\mathcal{N})_I)$. We consider the transition system $[\![\mathcal{N}^U]\!] = (\mathbb{N}^P, M_0, \Sigma_\varepsilon, \Rightarrow)$ associated to $\mathcal{N}^U$ and the TTS $[\![\mathcal{N}]\!]_I = (Q, q_0, \Sigma_\varepsilon, \rightarrow_I)$ associated to $\mathcal{N}$ under the weak intermediate semantics.

Let $M, M_f \in \mathbb{N}^P$ be two markings of $\mathcal{N}^U$ such that $M \Rightarrow^* M_f$. Consider a multiset Δ of transitions of $\mathcal{N}$ such that $M \overset{\Delta}{\Longrightarrow} M_f$. We will show that there exists a valuation $\nu_f \in \mathbb{R}^T_{\geq 0}$ such that $(M, \mathbf{0}) \rightarrow^*_I (M_f, \nu_f)$. The effective construction of this run in $[\![\mathcal{N}]\!]_I$ is given by Algorithm 1. We will prove that this Algorithm computes a correct execution and that it finishes without returning "Error".

First, analyzing the different steps of the Algorithm 1, we can deduce that:

Proposition 6. *For all $i \in \mathbb{N}$, if* $\textsf{Timed-En}_i \neq \emptyset$ *then we have:*

(i) $(M_i, \nu_i) \xrightarrow{d_{i+1}}_I (M_i, \nu_i + d_{i+1}) \xrightarrow{t_{i+1}}_I (M_{i+1}, \nu_{i+1})$,
(ii) $\forall t' \in \textsf{Timed-En}_i, \nu_{i+1}(t') \models \mathit{lft}(t')$.

Intuitively, when the set $\textsf{Timed-En}_i$ is not empty, at Line 7, the algorithm begins by choosing transition with the closest relative earliest firing time (which can eventually be negative if some transitions are already enabled). Then, it chooses a delay to fire this transition that preserves the firability of all others transitions in $\textsf{Timed-En}_i$. Due to the weak semantics, from any configuration, the system can wait any amount of time.

Proof. We assume that $\mathsf{Timed\text{-}En}_i$ is not empty and we distinguish three cases according to the **if** of Line 10 of the Algorithm 1:

- *First case:* $e_{min} < 0$. This corresponds to a transition which is already enabled. In this case, the value of d_{i+1} is equal to 0. Since t belongs to $\mathsf{Timed\text{-}En}_i$, this ensures that t is immediately firable, thus proving point (i). Property (ii) immediately follows from the definition of $\mathsf{Timed\text{-}En}_i$ since $d = 0$ and since for all $t' \in T$, we have $\nu_{i+1}(t') = \nu_i(t') + d$ or $\nu_{i+1}(t') = 0$ (if $\uparrow \mathit{enabled}_I(t', M, t) = true$).
- *Second case:* $e_{min} \geq 0$ and $\prec_{min} \in \{<\}$. This means that the left bound of $I(t)$ is closed. Then there exists a minimal delay $\delta \geq 0$ such that $\nu_i(t) + \delta \in I(t)$. This delay is in fact computed at line 11, thus with $d_{i+1} = \delta = e_{min}$ and the fact that we are considering the weak semantics, we have immediately (i). In addition, using the minimization criterion of Line 7 we have $\forall t' \in \mathsf{Timed\text{-}En}_i$, $\mathit{eft}(t) - \nu_i(t) \sqsubseteq \mathit{eft}(t') - \nu_i(t')$. Since for any transition t', $\mathit{eft}(t') \sqsubseteq \mathit{lft}(t')$, we deduce that we have for any $t' \in \mathsf{Timed\text{-}En}_i$, $(<, d_{i+1}) \sqsubseteq \mathit{eft}(t') - \nu_i(t') \sqsubseteq \mathit{lft}(t') - \nu_i(t')$ and applying the definition of $\sqsubseteq$ we have $\nu_i(t') + d \models \mathit{lft}(t')$. Since for all $t' \in T$, we have $\nu_{i+1}(t') = \nu_i(t') + d$ or $\nu_{i+1}(t') = 0$, we obtain the point (ii).
- *Third case:* $e_{min} \geq 0$ and $\prec_{min} \in \{\leq\}$. In this case the valuation $\nu_i(t)$ does not belong to $I(t)$, and the left bound of $I(t)$ is strict. Intuitively, we have to delay a bit more than the value e_{min}. By construction of $(\prec_{min}, e_{min})$ and $(\prec, l_{min})$ we have directly that $\nu_i(t) \in I(t)$ which allows us to deduce the point (i) since we can let time elapse without constraint. We will now focus on proving the point (ii). By definition of $(\prec, l_{min})$ (line 13), we have that for all $0 \leq \delta < l_{min}$, and for any $t' \in \mathsf{Timed\text{-}En}_i$, $\nu_i(t') + \delta \models \mathit{lft}(t')$. It remains to prove that $0 \leq d_{i+1} < l_{min}$. For any transitions t', we have $\mathit{eft}(t') \sqsubseteq \mathit{lft}(t')$, consequently $(\prec_{min}, e_{min}) \sqsubseteq \mathit{lft}(t')$ and, by defintion of $(\prec, l_{min})$, $(\prec_{min}, l_{min}) \sqsubseteq (l_{min}, \prec)$. Furthermore, since $\prec_{min} \in \{\leq\}$, we deduce that $(<, e_{min}) \neq (\prec, l_{min})$. Using the definition of $\sqsubseteq$, we obtain $e_{min} < l_{min}$. Consequently $(e_{min} + l_{min})/2 < l_{min}$, which allows us to deduce that for all $t' \in \mathsf{Timed\text{-}En}_i$, $\nu_i(t') + d_{i+1} \models \mathit{lft}(t')$. And since for all $t' \in T$, we have $\nu_{i+1}(t') = \nu_i(t') + d$ or $\nu_{i+1}(t') = 0$, we obtain the point (ii).

□

Second, we deduce from Proposition 6 the following property:

$$\forall i \in \mathbb{N}, \forall t \in \Delta_i, \text{ if } M_i \xRightarrow{t} M' \xRightarrow{\Delta_i \setminus \{t\}} \text{ then } \nu_i(t) \models \mathit{lft}(t) \tag{1}$$

The proof proceeds by induction on i. Initially, the property holds because $\nu_0 = \mathbf{0}$. For $i > 0$, pick $t \in \Delta_i$ such that $M_i \xRightarrow{t} M' \xRightarrow{\Delta_i \setminus \{t\}}$. If $\uparrow \mathit{enabled}_I(t, M_{i-1}, t_i) = true$, then $\nu_i(t) = 0$ and the property holds. Otherwise, we have $\uparrow \mathit{enabled}_I(t, M_{i-1}, t_i) = false$ and since $t \in \mathit{En}(M_i)$, we get by definition that $t \in \mathit{En}(M_{i-1} - {}^\bullet t)$. In other words, we have the inequality $M_{i-1} \geq {}^\bullet t + {}^\bullet t_i$ which allows us to swap the order of t and t_i and then the following execution exists in $[\![\mathcal{N}^U]\!]$: $M_{i-1} \xRightarrow{t} M'_{i-1} \xRightarrow{t_i} M' \xRightarrow{\Delta_i \setminus \{t\}}$. Since $\Delta_i = \Delta_{i-1} \setminus \{t_i\}$, we can apply the induction hypothesis for $i - 1$ and get $\nu_{i-1}(t) \models \mathit{lft}(t)$. Thus, $t \in \mathsf{Timed\text{-}En}_{i-1}$ and point (ii) of Proposition 6 entails that $\nu_i(t) \models \mathit{lft}(t)$, as desired.

Algorithm 1 Algorithm constructing a feasible timed ordering

Require: A marking $M \in \mathbb{N}^P$ and a finite multi-set Δ of transitions s.t. $M \overset{\Delta}{\Longrightarrow}$
Ensure: A run $(M, \mathbf{0}) \xrightarrow{d_1}_I \xrightarrow{t_1}_I (M_1, \nu_1) \xrightarrow{d_2}_I \xrightarrow{t_2}_I \dots \xrightarrow{d_n}_I \xrightarrow{t_n}_I (M_n, \nu_n)$
s.t. $\Delta = \langle t_1, \dots, t_n \rangle$

$i := 0$; $M_0 := M$; $\nu_0 := \mathbf{0}$; $\Delta_0 = \Delta$
while $\Delta_i \neq \emptyset$ **do**
  $\mathsf{Timed\text{-}En}_i := \{t \in \Delta_i \mid M_i \overset{t}{\Longrightarrow} M' \overset{\Delta_i \setminus \{t\}}{\Longrightarrow}$ and $\nu_i(t) \models \mathit{lft}(t)\}$
  if $\mathsf{Timed\text{-}En}_i = \emptyset$ **then**
    Return "Error"
  else
    $(\prec_{min}, e_{min}) := \min\{\mathit{eft}(t') - \nu_i(t') \mid t' \in \mathsf{Timed\text{-}En}_i\}$
    Pick $t \in \mathsf{Timed\text{-}En}_i$ such that $\mathit{eft}(t) - \nu_i(t) = (d_{min}, \prec_{min})$
    $t_{i+1} := t$
    if $e_{min} < 0$ or $\prec_{min} \in \{<\}$ **then**
      $d_{i+1} := \max(0, e_{min})$
    else
      $(\prec, l_{min}) := \min\{\mathit{lft}(t') - \nu_i(t') \mid t' \in \mathsf{Timed\text{-}En}_i\}$
      $d_{i+1} := (e_{min} + l_{min})/2$
    end if
    $\Delta_{i+1} := \Delta_i \setminus \{t\}$
    (M_{i+1}, ν_{i+1}) is s.t $(M_i, \nu_i) \xrightarrow{d_{i+1}}_I \xrightarrow{t_{i+1}}_I (M_{i+1}, \nu_{i+1})$
    $i := i + 1$
  end if
end while

We can now prove that Algorithm 1 never returns "Error". For this, we only have to prove that for all $i \in \mathbb{N}$, $\Delta_i \neq \emptyset$ implies $\mathsf{Timed\text{-}En}_i \neq \emptyset$. We proceed again by induction on $i \in \mathbb{N}$. The property for $i = 0$ is trivial since $\nu_0 = \mathbf{0}$. Let $i > 1$. Since t_i has been choosen in the set $\mathsf{Timed\text{-}En}_{i-1}$ and $\Delta_i = \Delta_{i-1} \setminus \{t_i\}$, we have $M_{i-1} \overset{t_i}{\Longrightarrow} M_i \overset{\Delta_i}{\Longrightarrow}$. Then there exists $t \in \Delta_i$ such that $M_i \overset{t}{\Longrightarrow} M' \overset{\Delta_i \setminus \{t\}}{\Longrightarrow}$ and the result follows from property (1).

Finally, we have proven that $\Delta_i \neq \emptyset$ implies $\mathsf{Timed\text{-}En}_i \neq \emptyset$, thus the Algorithm never returns "Error" and furthermore it always terminates because Δ_0 is finite and at each step $|\Delta_i| < |\Delta_{i+1}|$. Moreover, property (i) establishes the correction of the constructed run of $[\![N]\!]_I$. Hence if $M_f \in \mathtt{Reach}(\mathcal{N}^U)$, there exists a multiset $\Delta = \langle t_1, \dots, t_n \rangle$ such that $M_0 \overset{\Delta}{\Longrightarrow} M_f$, using the Algorithm 1, we can build a run $(M_0, \mathbf{0}) \xrightarrow{d_1}_I \xrightarrow{t_1}_I (M_1, \nu_1) \xrightarrow{d_2}_I \xrightarrow{t_2}_I \dots \xrightarrow{d_n}_I \xrightarrow{t_n}_I (M_n, \nu_n)$ and by definition of the different semantics, we have $M_n = M_f$. □

Using the decidability results in the case of PN, we obtain the following corollary:

Corollary 7. *The marking reachability, marking coverability and boundedness problems are decidable in the case of TPN with weak intermediate semantics.*

3.3 Undecidability for weak atomic and weak persistent atomic semantics

We consider now the case of the weak atomic and weak persistent atomic semantics. As for the strong semantics, but with a more involved construction, we will show that it is possible to encode the behavior of a Minsky machine into a TPN with weak (persistent) atomic semantics from which we will deduce the undecidability results.

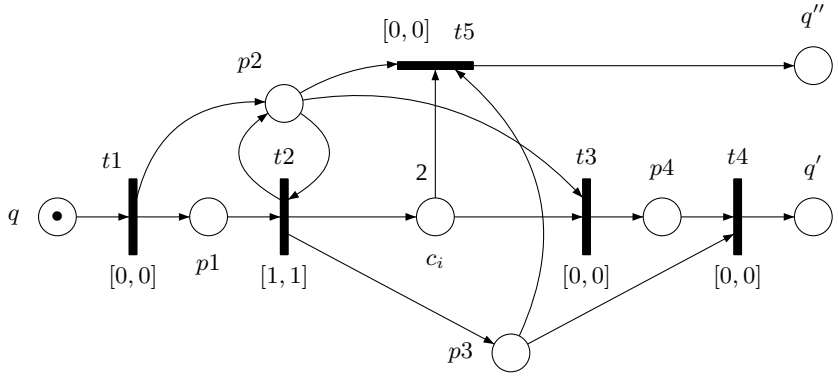

Fig. 2. Encoding decrementation with weak atomic or persistent atomic semantics.

Encoding an incrementation can be done as in the strong semantics. Figure 2 shows how to encode the instruction (q : if $c_i = 0$ goto q' else $c_i := c_i - 1$; goto q'') using a TPN with weak atomic or persistent atomic semantics. We now explain the idea of this encoding. We consider the two following cases for the net shown in Figure 2:

1. Assume that the only place which contains a token is the place q, which means we are in the case where the value of c_i is equal to 0 (no token in place c_i). We have the following sequence of transitions:
   - the only transition which is firable is $t1$, so the net fires it;
   - then only $t2$ can be fired, adding one token in place c_i and one in place $p3$;
   - since there is only one token in c_i the net can then only fire $t3$ and then $t4$ to end up in a marking with a unique token in place q'.
2. Assume now that there is a token in place q and that there is at least one token in place c_i. We are in the case where the value of c_i is different of 0. We have the following sequence of transitions:
   - only the transition $t1$ is firable, so the net fires it;
   - afterwards the transition $t2$ and the transition $t3$ are firable, but note that if the net chooses to fire $t3$, it will reach a deadlock state where no more transitions can be fired without having put a token in the place q' or q'', therefore we assume that the transition $t2$ is first fired. It is possible not to fire $t3$ because we are considering the weak semantics (in the strong semantics, the net would have to fire $t3$ first);
   - after having fired $t2$, the only transition which can be fired is $t5$. In fact since we are considering atomic (or persistent atomic) semantics, the firing of $t2$ does not make newly enabled $t3$ and because of the time constraints linked to $t2$ and

$t3$, $t3$ is not anymore firable. So the net fires $t5$ consuming the token in $p2$, $p3$ and two tokens in c_i (at least one was present from the initial configuration and the first firing of $t2$ added another one);
- finally the net ends in a configuration with one token in q'' and the place c_i contains one token less than in the initial configuration.

Since we can simulate a Minsky machine using a TPN with atomic or persistent atomic weak semantics, we deduce the following result [3]:

Theorem 8. *The marking reachability, marking coverability and boundedness problems are undecidable for TPN with weak atomic or persistent atomic semantics.*

In comparison with what occurs in the case of the strong semantics, this result is surprising, and it reveals the important role played by the memory policy when considering the weak semantics. Recall that as we have seen earlier, with the strong semantics, these problems are undecidable no matter which memory policy is chosen.

4 Expressiveness

4.1 Preliminaries

Let $S = (Q, q_0, \Sigma_\varepsilon, \rightarrow)$ be a TTS. We define the relation $\hookrightarrow \subseteq Q \times (\Sigma \cup \mathbb{R}_{\geq 0}) \times Q$ by:

- for $d \in \mathbb{R}_{\geq 0}$, $q \overset{d}{\hookrightarrow} q'$ iff there is a run ρ in S such that $\rho = q \xrightarrow{w} q'$ and $Untimed(w) = \varepsilon$ and $Duration(w) = d$,
- for $a \in \Sigma$, $q \overset{a}{\hookrightarrow} q'$ iff there is a run ρ in S such that $\rho = q \xrightarrow{w} q'$ and $Untimed(w) = a$ and $Duration(w) = 0$.

This allows us to define the following notion:

Definition 9 (Weak Timed Bisimulation). *Let $S_1 = (Q_1, q_0^1, \Sigma_\varepsilon, \rightarrow_1)$ and $S_2 = (Q_2, q_0^2, \Sigma_\varepsilon, \rightarrow_2)$ be two TTS and $\approx$ be a binary relation over $Q_1 \times Q_2$. $\approx$ is a weak timed bisimulation between S_1 and S_2 if and only if:*

- *$q_0^1 \approx q_0^2$, and,*
- *for $a \in \Sigma \cup \mathbb{R}_{\geq 0}$, if $q_1 \overset{a}{\hookrightarrow}_1 q_1'$ and if $q_1 \approx q_2$ then there exists $q_2' \in Q_2$ such that $q_2 \overset{a}{\hookrightarrow}_2 q_2'$ and $q_1' \approx q_2'$; conversely if $q_2 \overset{a}{\hookrightarrow}_2 q_2'$ and if $q_1 \approx q_2$ then there exists $q_1' \in Q_1$ such that $q_1 \overset{a}{\hookrightarrow}_1 q_1'$ and $q_1' \approx q_2'$.*

Two TTS S_1 and S_2 are *weak timed bisimilar* if there exists a weak timed bisimulation between S_1 and S_2. We then write $S_1 \approx_{\mathcal{W}} S_2$.

Definition 10 (Expressiveness w.r.t. Weak Timed Bisimilarity). *The class $\mathcal{C}$ of TTS is more expressive than $\mathcal{C}'$ w.r.t. weak timed bisimilarity if for all TTS $S' \in \mathcal{C}'$ there is a TTS $S \in \mathcal{C}$ such that $S \approx_{\mathcal{W}} S'$. We write $\mathcal{C} \leq_{\mathcal{W}} \mathcal{C}'$. If moreover there is a $S \in \mathcal{C}$ such that there is no $S' \in \mathcal{C}'$ with $S \approx_{\mathcal{W}} S'$, then $\mathcal{C}$ is strictly more expressive than $\mathcal{C}'$. We then write $\mathcal{C} <_{\mathcal{W}} \mathcal{C}'$.*

For $s \in \{I, A, PA\}$, we will denote by $\mathcal{TPN}_s$ the class of TTS induced by TPN with s-semantics.

[3] Note that one can also easily prove that weak intermediate TPN with read arcs are undecidable.

4.2 Atomic versus Persistent Atomic semantics

In [4], the authors prove that for TPN with strong semantics, the persistent atomic semantics is more expressive than the atomic semantics. We prove here that this result still holds in the case of the weak semantics.

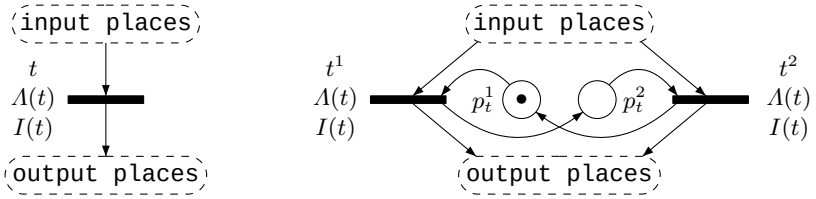

Fig. 3. From atomic to persistent atomic semantics.

Proposition 11. *For all TPN $\mathcal{N}$, we can build a TPN $\mathcal{N}'$ such that $[\![\mathcal{N}]\!]_A \approx_{\mathcal{W}} [\![\mathcal{N}']\!]_{PA}$.*

Proof. Let $\mathcal{N} = (P, T, \Sigma_\varepsilon, {}^\bullet(.), (.)^\bullet, M_0, \Lambda, I)$ be a TPN over Σ_ε. Figure 3 represents the construction of the TPN $\mathcal{N}'$. Formally, its set of places P' is equal to $P \cup \{p_t^1, p_t^2 \mid t \in T\}$ and its set of transitions T' contains two copies t^1 and t^2 of each transition $t \in T$. These copies are connected in the same way than the transition t is in $\mathcal{N}$, plus additional edges to the places p_t^1 and p_t^2, as depicted on Figure 3. Finally the initial marking of $\mathcal{N}'$ is M_0' such that for all $p \in P$, $M_0'(p) = M_0(p)$ and for all $t \in T$, $M_0'(p_t^1) = 1$ and $M_0'(p_t^2) = 0$.

We now consider the relation $\mathcal{R} \subseteq (\mathbb{N}^P \times \mathbb{R}_{\geq 0}^T) \times (\mathbb{N}^{P'} \times \mathbb{R}_{\geq 0}^{T'})$ between the configurations of $[\![\mathcal{N}]\!]_A$ and the ones of $[\![\mathcal{N}']\!]_{PA}$ defined by $(M, \nu)\mathcal{R}(M', \nu')$ iff:

- for all $p \in P$, $M(p) = M'(p)$ and for all $t \in T$, $M'(p_t^1) + M'(p_t^2) = 1$,
- for all $t \in T$, for all $i \in \{1, 2\}$ if $t \in \mathit{En}(M)$ and $t^i \in \mathit{En}(M')$ then $\nu(t) = \nu'(t^i)$.

It is then easy to verify that the relation $\mathcal{R}$ is a weak timed bisimulation. □

We will now prove that the inclusion we obtain in the above proposition is strict. Therefore, we consider the TPN $\mathcal{N}_1$ represented on Figure 4. Equipped with persistent atomic semantics, it accepts the set of timed words composed of letters a occurring before time 1. Intuitively, our proof will rely on the fact that a TPN has a finite memory, more or less one date for each transition. Before to prove this, we need a technical lemma which intuitively states that, for any memory policy, once the memories of transitions are controlled, executions can be delayed. This constitutes a deep difference with the strong semantics, for which time elapsing is limited in a global way.

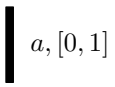

Fig. 4. The TPN $\mathcal{N}_1$.

Lemma 12. *Let $s \in \{I, A, PA\}$ and consider a TPN $\mathcal{N}$ whose granularity is d. Let ρ be a run in $[\![\mathcal{N}]\!]_s$ of the form $\rho : (M, \nu) \xrightarrow{\delta>0}_s (M, \nu + \delta) \xrightarrow{t_1}_s \cdots \xrightarrow{t_n}_s$, such that there exists a value $\tau \geq 0$ verifying:*

(i) $\forall i \in \{1,\ldots,n\},\ t_i \in \mathit{En}(M) \Rightarrow \nu(t_i) \leq \tau$,
(ii) $\tau + \delta < \frac{1}{2d}$
Then the sequence $\rho' : (M,\nu) \xrightarrow{\delta+\frac{1}{2d}}_s (M,\nu') \xrightarrow{t_1}_s \cdots \xrightarrow{t_n}_s$ *is firable in* $[\![\mathcal{N}]\!]_s$.

Proof. We introduce the following notations describing the run ρ:
$(M,\nu) \xrightarrow{\delta>0}_s (M,\nu+\delta) = (M_1,\nu_1) \xrightarrow{t_1}_s (M_2,\nu_2) \xrightarrow{t_2}_s (M_3,\nu_3) \cdots (M_n,\nu_n) \xrightarrow{t_n}_s (M_{n+1},\nu_{n+1})$.

We split the set of indices $\{1,\ldots,n\}$ into two disjoint subsets, $I_{=0} = \{i \in \{1,\ldots,n\} \mid \nu_i(t_i) = 0\}$, and $I_{\neq 0} = \{i \in \{1,\ldots,n\} \mid \nu_i(t_i) \neq 0\}$. These two sets contain respectively the index of the transitions fired with a null (respectively non null) valuation.

We first show the following property:

$$\forall i \in I_{\neq 0}, 0 < \nu_i(t_i) < \frac{1}{2d} \text{ and }]0, \frac{1}{d}[\subseteq I(t_i) \tag{2}$$

Consider the first part of this property. Note that these valuations are strictly positive by definition of $I_{\neq 0}$, thus proving the left inequality. Consider now the right inequality, and let $i \in I_{\neq 0}$. Since the sequence $t_1 \ldots t_i$ is instantaneous and follows a non null delay step, t_i is never newly enabled during $t_1 \ldots t_{i-1}$ (otherwise $\nu_i(t_i) = 0$). As a consequence, we have $\nu_i(t_i) = \nu(t_i) + \delta$, and by properties (i) and (ii) of ρ, we obtain $\nu_i(t_i) \leq \tau + \delta < \frac{1}{2d}$, as desired.

By definition of the granularity d, and since t_i is firable from (M_i,ν_i), the inclusion $]0, \frac{1}{d}[\subseteq I(t_i)$ holds for any $i \in I_{\neq 0}$.

We consider now the run $\rho' : (M,\nu) \xrightarrow{\delta+\frac{1}{2d}}_s (M,\nu+\delta+\frac{1}{2d}) = (M_1,\nu'_1) \xrightarrow{t_1}_s (M_2,\nu'_2) \xrightarrow{t_2}_s (M_3,\nu'_3) \cdots (M_n,\nu'_n) \xrightarrow{t_n}_s (M_{n+1},\nu'_{n+1})$. Note that the increasing of the delay is possible because of the weak semantics: in the strong one, the modification of the delay step could be impossible. To prove that this sequence is firable in $[\![\mathcal{N}]\!]_s$, we proceed by contradiction. Assume there exists a position $i \in \{1,\ldots,n\}$ such that t_i is not firable from (M_i,ν'_i), and pick the smallest position verifying this property. We distinguish two cases:

1. **If** $i \in I_{=0}$**.** Then we have $\nu_i(t_i) = 0$. Since the instantaneous sequence $t_1 \ldots t_i$ immediately follows in ϱ the delay step $\delta > 0$, t_i is newly enabled by the firing of $t_1 \ldots t_{i-1}$. Since this property only depends on discrete markings, which are preserved in ρ', t_i is also newly enabled by the firing of $t_1 \ldots t_{i-1}$ in ρ'. As a consequence, we have $\nu'_i(t_i) = 0 = \nu_i(t_i)$, thus proving that t_i is firable in ρ', yielding a contradiction.
2. **If** $i \in I_{\neq 0}$**.** Then we have $\nu_i(t_i) \neq 0$. As already mentionned above, we have in this case that t_i is never newly enabled during $t_1 \ldots t_{i-1}$ in ϱ. Since the discrete markings are preserved, this conclusion holds also in ρ'. As a consequence, we have $\nu_i(t_i) = \nu(t_i) + \delta$ and $\nu'_i(t_i) = \nu(t_i) + \delta + \frac{1}{2d}$, what yields $\nu'_i(t_i) = \nu(t_i) + \frac{1}{2d}$. Using Property (2), we obtain $\nu'_i(t_i) \in I(t_i)$, and then t_i is firable from (M_i,ν'_i), contradicting the assumption.

Finally, we have obtained a contradiction in both cases, thus proving that ρ' is firable in $[\![\mathcal{N}]\!]_s$. □

We then have the following proposition:

Proposition 13. *There exists no TPN $\mathcal{N}$ (even unbounded) s.t. $[\![\mathcal{N}]\!]_A \approx_{\mathcal{W}} [\![\mathcal{N}_1]\!]_{PA}$.*

Proof. We consider the TPN $\mathcal{N}_1$ represented on Figure 4. Assume there exists a TPN $\mathcal{N}$ such that $[\![\mathcal{N}]\!]_A \approx_{\mathcal{W}} [\![\mathcal{N}_1]\!]_{PA}$. Denote by N the number of transitions of $\mathcal{N}$, by d its granularity, and consider a timed word $w = (a,\tau_1)(a,\tau_2)\dots(a,\tau_k)$ such that $\forall i, 1-\frac{1}{2d} < \tau_i < \tau_{i+1} < 1$, and $k \geq N+1$.

This timed word w is recognized by $[\![\mathcal{N}_1]\!]_{PA}$ and there exists thus a run of $[\![\mathcal{N}]\!]_A$ along w. We denote it by ρ and decompose it as follows :

$$\rho : \xrightarrow{\theta_0}_A \xrightarrow{d_1}_A \xrightarrow{\theta_1}_A \xrightarrow{t_a^1}_A \xrightarrow{\theta'_1}_A \cdots \xrightarrow{d_i}_A \xrightarrow{\theta_i}_A \xrightarrow{t_a^i}_A \xrightarrow{\theta'_i}_A \cdots \xrightarrow{d_k}_A \xrightarrow{\theta_k}_A \xrightarrow{t_a^k}_A$$

To obtain this decomposition we proceed as follows. We denote by t_a^i the i-th transition labelled by a. Then for each position i, we isolate the last delay step occuring before the transition t_a^i (it exists since $\tau_i > \tau_{i-1}$) and denote it by d_i. Then we gather all the silent transitions occuring between this delay step and the transition t_a^i, and denote this sequence by θ_i. The transitions between t_a^{i-1} and the delay step constitute the sequence θ'_{i-1}. In particular, the following properties hold for any position i : $\Lambda(t_a^i) = a$, *Untimed*(θ_i) = *Untimed*$(\theta'_i) = \varepsilon$, $d_i > 0$, *Duration*$(\theta_i) = 0$, and t_a^i occurs at time τ_i.

We claim there exists an index $i \in \{1,\dots,k\}$ such that any transition t appearing in $\theta_i\ t_a^i$ has already appeared since θ_0, that is t also fired in $\theta_1\ t_a^1\ \theta'_1 \cdots \theta_{i-1}\ t_a^{i-1}\ \theta'_{i-1}$. By contradiction, if it is not the case, then we can find, for any index $i \in \{1,\dots,k\}$, a transition, denoted t_i, that never appears before. The choice of k verifying $k \geq N+1$ then implies that there exist two positions $i \neq j$ such that $t_i = t_j$, thus yielding a contradiction. We can now fix an index i verifying above described property.

We now show that Lemma 12 can be applied to the part of ρ associated with the sequence $d_i\ \theta_i\ t_a^i$. More precisely, (M,ν) is the configuration reached after firing $\theta_0 \cdots t_a^{i-1}\ \theta'_{i-1}$, the delay δ is equal to d_i, the sequence $t_1 \cdots t_n$ corresponds to $\theta_i\ t_a^i$, and τ is defined as $(\tau_i - d_i) - (1-\frac{1}{2d})$. In the atomic semantics, when a transition is fired, its clock is reset if it is still enabled. This property allows, together with timing constraints on the word w, to verify hypotheses (i) and (ii) of the Lemma 12. Indeed, since each transition in $\theta_i\ t_a^i$ has been reset along $\theta_1\ t_a^1\ \theta'_1 \cdots \theta'_{i-1}$, it has been reset since time τ_1. Since the global time associated with (M,ν) is equal to $\tau_i - d_i$, these valuations are bounded by above by the value $(\tau_i - d_i) - \tau_1 \leq (\tau_i - d_i) - (1-\frac{1}{2d}) = \tau$. Second, we have $\tau + \delta = \tau_i - (1-\frac{1}{2d}) < \frac{1}{2d}$, as desired (this follows from the inequalities $1-\frac{1}{2d} < \tau_i < 1$).

Finally, Lemma 12 thus allows to delay of $\frac{1}{2d}$ the firing of the sequence $\theta_i\ t_a^i$. In particular, this will produce a letter a at date $\tau_i + \frac{1}{2d} > 1$. The TTS $[\![\mathcal{N}]\!]_A$ thus accepts a timed word not recognized by $[\![\mathcal{N}_1]\!]_{PA}$, providing a contradiction. □

Using the results of Propositions 11 and 13, we deduce the:

Theorem 14. $\mathcal{TPN}_A <_{\mathcal{W}} \mathcal{TPN}_{PA}$.

4.3 About Atomic and Intermediate policies in weak and strong semantics

In this subsection, we discuss the comparison of the intermediate and atomic policies. As we will see, the situation is more complex than in the previous comparison.

On the inclusion of $\mathcal{TPN}_I$ into $\mathcal{TPN}_A$. For the strong semantics, a construction has been proposed in [4] to transform any TPN with intermediate policy into an equivalent (w.r.t. weak timed bisimilarity) TPN with atomic semantics. A first attempt was thus to adapt this construction for the weak semantics. But studying this construction, we noticed that it is erroneous (even for the strong semantics). We present below an example exhibiting the error.

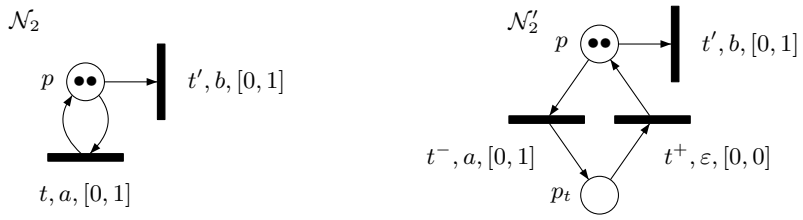

Fig. 5. A counter example to the construction of [4].

Example 1. Consider the net $\mathcal{N}_2$ depicted on the left of Figure 5. The application of the construction proposed in [4] leads to the net $\mathcal{N}_2'$ depicted on the right of Figure 5. According to [4], we should have, under the **strong semantics**, the relation $[\![\mathcal{N}_2]\!]_I \approx_{\mathcal{W}} [\![\mathcal{N}_2']\!]_A$. However, it is easy to verify that in the TTS $[\![\mathcal{N}_2']\!]_A$ the timed word $w = (a,1)(a,1)(b,2)$ is accepted, whereas it is not in $[\![\mathcal{N}_2]\!]_I$, thus proving that the construction proposed in [4] is erroneous.

This example leaves open the question of the inclusion of $\mathcal{TPN}_I$ into $\mathcal{TPN}_A$ for the strong semantics, and then for this semantics both inclusions are left open. For weak semantics, this inclusion is also open, but we show below that the converse inclusion is false.

Non inclusion of $\mathcal{TPN}_A$ into $\mathcal{TPN}_I$. We exhibit a TPN with atomic semantics which can not be expressed in an equivalent way by any TPN with intermediate semantics (with weak elapsing of time). This is formally stated in the Proposition below. We consider the TPN represented on Figure 6. Interpreted in weak atomic semantics, the firing of the a-labelled transition does not newly enable transition labelled by b. This transition thus shares a token with transition a while preserving a time reference to the origin of global time, what is impossible in intermediate semantics.

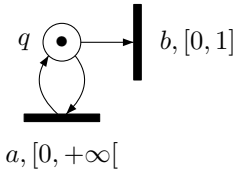

Fig. 6. The TPN $\mathcal{N}_3$.

Proposition 15. *There exists no TPN $\mathcal{N}$ (even unbounded) such that $[\![\mathcal{N}]\!]_I \approx_{\mathcal{W}} [\![\mathcal{N}_3]\!]_A$.*

Proof. We proceed by contradiction and assume there exists such a TPN $\mathcal{N}$, and denote by N its number of transitions, and by d its granularity. As in the proof of Proposition 13, we first exhibit a particular execution ρ of $[\![\mathcal{N}]\!]_I$:

Lemma 16. *Let $(\tau_i)_{1\leq i\leq k}$ be a set of timestamps such that for any $1 \leq i \leq k$, $1-\frac{1}{2d} < \tau_i < \tau_{i+1} < 1$ and $k \geq N+1$. There exists a run ρ in $[\![\mathcal{N}]\!]_I$ of the following form:*

$$\rho : \xrightarrow{1-\frac{1}{2d}}_I \xrightarrow{\theta_1}_I \xrightarrow{d_1}_I \xrightarrow{\theta_1'}_I \xrightarrow{t_a^1}_I \xrightarrow{\theta_1''}_I \cdots \xrightarrow{\theta_i}_I \xrightarrow{d_i}_I \xrightarrow{\theta_i'}_I \xrightarrow{t_a^i}_I \xrightarrow{\theta_i''}_I \cdots \xrightarrow{\theta_n}_I \xrightarrow{d_n}_I \xrightarrow{\theta_n'}_I \xrightarrow{t_a^n}_I \xrightarrow{\theta_n''}_I$$

such that for any position i, $\Lambda(t_a^i) = a$, the transition t_a^i occurs at time τ_i, $d_i > 0$, Untimed$(\theta_i) =$ Untimed$(\theta_i') =$ Untimed$(\theta_i'') = \varepsilon$, Duration$(\theta_i') =$ Duration$(\theta_i'') = 0$, and there exists a transition t_b^i, labelled by b, newly enabled by the last transition of t_a^i θ_i'' and (immediately) firable from the configuration reached after θ_i''.

Proof. We present here the details of the construction of the run ρ. This construction proceeds in three steps.

First step : construction of the structure of ρ. This step is the most involved one. We deeply use the bisimulation property between $[\![\mathcal{N}_3]\!]_A$ and $[\![\mathcal{N}]\!]_I$ to build a first sequence. Figure 7 illustrates this construction. A dashed arrow between two configurations meens that these configurations are bisimilar. The direction of this arrow indicates which implication is used to obtain the bisimilarity (from $[\![\mathcal{N}_3]\!]_A$ to $[\![\mathcal{N}]\!]_I$, or conversely). In this figure, we omit the index I and A which should be associated to each of the step. We now detail step by step how this Figure reads. Initially, due to weak semantics, the net $\mathcal{N}$ in intermediate semantics can choose to delay $1 - \frac{1}{2d}$ time units. By bisimulation (and because there are no silent transitions in $\mathcal{N}_3$), the same delay leads to a bisimilar configuration in $[\![\mathcal{N}_3]\!]_A$, thus indicated by a bottom-up dashed arrow. From this configuration one can in $[\![\mathcal{N}_3]\!]_A$ delay $\tau_1 - (1 - \frac{1}{2d})$ t.u. [4] and then fire the a transition. By bisimulation, there exists a path in $[\![\mathcal{N}]\!]_I$, written as $u_1 \cdot t_a^1 \cdot u_1'$, leading to a bisimilar configuration and such that u_1 only contains silent transitions and is of duration exactly $\tau_1 - \tau_0$, t_a^1 is labelled by a, and u_1' is labelled by ε and has a null duration. The bisimilarity is indicated by a top-down dashed arrow since it is the existence of a path in $[\![\mathcal{N}_3]\!]_A$ that implies the existence of a path in $[\![\mathcal{N}]\!]_I$. Next, one can fire instantaneously b in $[\![\mathcal{N}_3]\!]_A$ what implies the existence of a path $\sigma_1 \cdot t_b^1 \cdot \sigma_1'$, leading to a bisimilar configuration, and such that σ_1 and σ_1' are labelled by ε and of null duration, and t_b^i is labelled by b. Then, we use the bisimulation property in the converse direction : the configuration reached after the prefix σ_1, which is labelled by ε and of null duration, is still bisimilar with the configruation of $[\![\mathcal{N}_3]\!]_A$ reached after the a transition, as indicated by the bottom-up dashed arrow (the only path in $[\![\mathcal{N}_3]\!]_A$ of null duration and labelled by ε is the empty path). Then the same reasoning is applied from these two bisimilar configurations, and can be repeated arbitrarily many times.

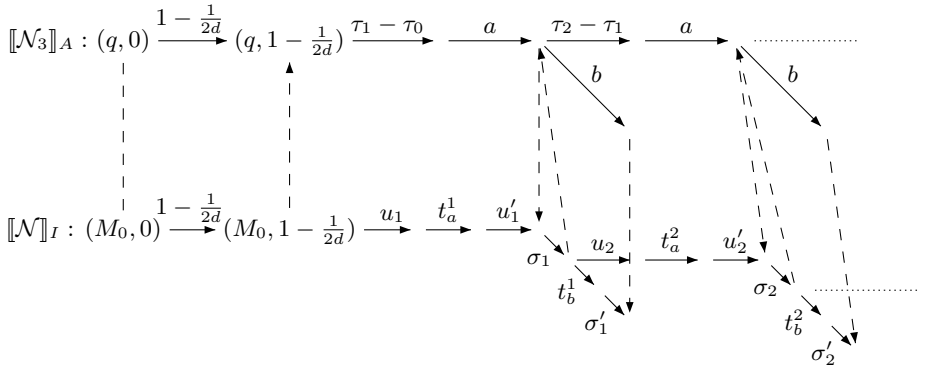

Fig. 7. Using bisimulation to build ρ.

[4] We denote by τ_0 the value $1 - \frac{1}{2d}$.

Second step : definition of θ_i, θ'_i and d_i. This step is much simpler. We simply split the paths u_i in three (eventually empty) parts. The duration of u_i is equal to $\tau_i - \tau_{i-1} > 0$, thus we can consider the last positive delay step appearing in u_i, and denote it by d_i. Then we simply write $u_i = \theta_i \cdot d_i \cdot \theta'_i$.

Third step : definition of θ''_i. In this last step, we will define the path θ''_i as a prefix of the path $u'_i \cdot \sigma_i$. This path possesses the following properties : its duration is null, it is labelled by ε, and there exists a b labelled transition t^i_b immediately firable after it. The only property missing is that its last transition newly enables transition t^i_b. To obtain this property, we will show that we can find a prefix of $u'_i \cdot \sigma_i$ having this additional property. Therefore we prove the following lemma, specific to the intermediate semantics:

Lemma 17. *Let $\mathcal{N}$ be a TPN, and consider an execution of $[\![\mathcal{N}]\!]_I$, denoted by $\rho : (M_1, v_1) \xrightarrow{t_1}_I (M_2, v_2) \xrightarrow{t_2}_I \cdots (M_n, v_n) \xrightarrow{t_n}_I (M_{n+1}, v_{n+1}) \xrightarrow{t}_I$, such that for any $i \in \{1, \ldots, n\}$, $\uparrow$enabled$_I(t, M_i, t_i) = false$.*
Then, for any $i \in \{1, \ldots, n\}$, the execution $\rho_i : (M_i, v_i) \xrightarrow{t}_I \xrightarrow{t_i}_I$ exists in $[\![\mathcal{N}]\!]_I$.

Proof (of Lemma 17). By definition of the predicate $\uparrow$*enabled*$_I$, and since t is newly enabled by no t_i and firable from (M_{n+1}, v_{n+1}), we have that t is enabled in every M_i and, because the sequence $t_1 \ldots t_n$ is instantaneous (no delay step), $v_{n+1}(t) = v_i(t)$ for any i. Thus, t is firable from the configuration (M_i, v_i). Moreover, we can notice, as t_i is enabled in M_i, that $\uparrow$*enabled*$_I(t, M_i, t_i) = false$ implies that t_i is still enabled after the firing of t from M_i, and has not been newly enabled: $\uparrow$*enabled*$_I(t_i, M_i, t) = false$. As a consequence, t_i is firable from the configuration reached after firing t, thus proving the Lemma. □

Lemma 17 entails that there exists a transition in the sequence $t^i_a \cdot u'_i \cdot \sigma_i$ which newly enables the transition t^i_b. Indeed, if it is not the case, the result of Lemma 17 implies that after the firing of u_i, one can first fire the transition t^i_b, and immediately after fire the transition t^i_a. This leads to the production of a letter a after the letter b, which in impossible in $[\![\mathcal{N}_3]\!]_A$, thus leading to a contradiction. Finally, we define θ''_i as the (eventually empty if t^i_b is newly enabled by t^i_a) prefix of $u'_i \cdot \sigma_i$ up to the first transition that newly enables t^i_b. This concludes the proof of the existence of the sequence ρ. □

We now return on the proof of Proposition 15. First, we claim that there exists an index $i \in \{1, \ldots, k\}$ such that any transition t appearing in $\theta'_i\ t^i_a\ \theta''_i$ has already been fired since time τ_1 that is t also appears in $\theta'_1\ t^1_a\ \theta''_1 \cdots \theta_{i-1}\ \theta'_{i-1} t^{i-1}_a\ \theta''_{i-1}\theta_i$. The reasonning is similar to the one of the proof of Proposition 13: by contradiction, if there exists no such position, then we can find, for any index $i \in \{1, \ldots, k\}$, a transition, denoted t_i, that never appears before. The choice of k verifying $k \geq N + 1$ then implies that there exist two positions $i \neq j$ such that $t_i = t_j$, thus yielding a contradiction. We can now fix an index i verifying the above described property.

We now show that Lemma 12 can be applied to the portion of ρ associated with the sequence $\xrightarrow{d_i} \xrightarrow{\theta'_i} \xrightarrow{t^i_a} \xrightarrow{\theta''_i}$. More precisely, let (M, ν) be the configuration reached after firing $(1 - \frac{1}{2d})\theta_1 \ldots t^{i-1}_a \theta''_{i-1}\theta_i$, the delay δ is equal to d_i, the sequence $t_1 \cdots t_n$ corresponds to $\theta'_i t^i_a \theta''_i$, and τ is defined as $(\tau_i - d_i) - (1 - \frac{1}{2d})$. In the intermediate semantics, when a transition is fired, its clock is reset if it is still enabled. This property

allows, together with timing constraints on the run ρ, to verify hypotheses (i) and (ii) of the Lemma 12. Indeed, we first have that each transition in $\theta'_i\ t^i_a\ \theta''_i$ has been reset since time τ_1. Since the global time associated with (M, ν) is equal to $\tau_i - d_i$, these valuations are bounded by above by the value $(\tau_i - d_i) - \tau_1 \leq (\tau_i - d_i) - (1 - \frac{1}{2d}) = \tau$. Second, we have $\tau + \delta = \tau_i - (1 - \frac{1}{2d}) < \frac{1}{2d}$, as desired (this follows from the inequalities $1 - \frac{1}{2d} < \tau_i < 1$).

Lemma 12 thus allows to delay of $\frac{1}{2d}$ time units the firing of thesequence $\theta'_i\ t^i_a\ \theta''_i$. Moreover, as the transition t^i_b is newly enabled by the last transition of $t^i_a\ \theta''_i$, and can be immediately firable after θ''_i in ρ, we deduce that this immediate firing can also be performed in the delayed execution. We thus obtain an execution in $[\![\mathcal{N}]\!]_{\mathcal{I}}$ with a b action following an a action after time 1, which is impossible in $[\![\mathcal{N}_3]\!]_A$, thus yielding a contradiction. □

5 Conclusion

We have studied in this paper the model of Time Petri Nets under a weak semantics of time elapsing, allowing any delay transition. We have first proven that for the intermediate memory policy, the set of reachable markings coincides with the reachability set of the underlying untimed Petri net. As a consequence, many verification problems are decidable for weak intermediate TPN. On the other hand, we have proven that the two others memory policies, namely atomic and persistent atomic, allow to simulate Minsky machines and thus are undecidable. Finally, we have studied expressiveness and have proven that (i) the atomic semantics is strictly less expressive than the persistent atomic one and (ii) the atomic semantics is not included in the intermediate one.

In further work, we plan to investigate properties concerning executions of weak intermediate TPN; such as time-optimal reachability, or LTL model checking. Indeed, while discrete markings are the same, the executions are different from those accepted by the underlying Petri net. Concerning expressiveness, we conjecture that intermediate and atomic semantics are uncomparable in general, and that bounded weak TPN are strictly less expressive than timed automata (without invariants).

Aknowledgments. We would like to thank Fabrice Chevalier for fruitful discussions.

References

1. P. A. Abdulla, P. Mahata, and R. Mayr. Dense-timed Petri nets: Checking zenoness, token liveness and boundedness. *Logical Methods in Computer Science*, 3(1):1–61, 2007.
2. P. A. Abdulla and A. Nylén. Timed Petri nets and bqos. In *ICATPN'01*, vol. 2075 of *LNCS*, pages 53–70. Springer, 2001.
3. R. Alur and D. L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126(2):183–235, 1994.
4. B. Bérard, F. Cassez, S. Haddad, D. Lime, and O. H. Roux. Comparison of different semantics for time Petri nets. In *ATVA'05*, vol. 3707 of *LNCS*, pages 293–307. Springer, 2005.
5. T. Bolognesi, F. Lucidi, and S. Trigila. From timed Petri nets to timed LOTOS. In *PSTV'90*, pages 395–408. North-Holland, 1990.
6. M. Boyer and O. H. Roux. On the compared expressiveness of arc, place and transition time Petri nets. *Fundamenta Informaticae*, 88(3):225–249, 2008.

7. D. de Frutos-Escrig, V. Valero Ruiz, and O. M. Alonso. Decidability of properties of timed-arc Petri nets. In *ICATPN'00*, vol. 1825 of *LNCS*, pages 187–206. Springer, 2000.
8. S. Haar, F. Simonot-Lion, L. Kaiser, and J. Toussaint. Equivalence of timed state machines and safe time Petri nets. In *WoDES'02*, pages 119–126, 2002.
9. N. D. Jones, L. H. Landweber, and Y. E. Lien. Complexity of some problems in Petri nets. *Theoretical Computer Science*, 4(3):277–299, 1977.
10. R. M. Karp and R. E. Miller. Parallel program schemata. *Journal of Computer System Sciences*, 3(2):147–195, 1969.
11. S. R. Kosaraju. Decidability of reachability in vector addition systems (preliminary version). In *STOC'82*, pages 267–281. ACM, 1982.
12. E. W. Mayr. An algorithm for the general Petri net reachability problem. *SIAM Journal on Computing*, 13(3):441–460, 1984.
13. P. M. Merlin. *A Study of the Recoverability of Computing Systems*. PhD thesis, University of California, Irvine, CA, USA, 1974.
14. M. L. Minsky. *Computation: finite and infinite machines*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1967.
15. J. Sifakis and S. Yovine. Compositional specification of timed systems. In *STACS'96*, vol. 1046 of *LNCS*, pages 347–359. Springer, 1996.
16. V. Valero Ruiz, D. de Frutos-Escrig, and F. Cuartero. On non-decidability of reachability for timed-arc Petri nets. In *PNPM'99*, pages 188–196. IEEE Computer Society Press, 1999.