Decidability of Properties of Timed-Arc Petri Nets

D. de Frutos Escrig[1], V. Valero Ruiz[2], and O. Marroquín Alonso[1]

[1] Dpto. Sistemas Informáticos y Programación, Fac. Matemáticas
Universidad Complutense. Madrid, SPAIN 28040
{defrutos,alonso}@sip.ucm.es
[2] Dpto. Informática, Esc. Politécnica
Univ. Castilla-La Mancha. Albacete, SPAIN 02071
valentin@info-ab.uclm.es

Abstract. Timed-arc Petri nets (TAPN's) are not Turing powerful, because, in particular, they cannot simulate a counter with zero testing. Thus, we could think that this model does not increase significantly the expressiveness of untimed Petri nets. But this is not true; in a previous paper we have shown that the differences between them are big enough to make the reachability problem undecidable. On the other hand, coverability and boundedness are proved now to be decidable. This fact is a consequence of the close interrelationship between TAPN's and transfer nets, for which similar results have been recently proved. Finally, we see that if dead tokens are defined as those that cannot be used for firing any transition in the future, we can detect these kind of tokens in an effective way.

1 Introduction

Petri nets are widely used for the modeling and analysis of concurrent systems, because of their graphical nature and the solid mathematical foundations supporting them. Several timed extensions of the basic model have been proposed to expand their application areas to those systems which exhibit a time-dependent behaviour that should be considered both in the modeling and the analysis process, such as distributed systems, communication systems and real-time systems.

A survey of the different approaches to introduce time into Petri nets is presented in [7]. We can identify a first group of models, which assign time delays to transitions, either using a fixed and deterministic value [19–21] or choosing it from a probability distribution [3]. Other models use time intervals to establish the enabling times of transitions [16]. Finally, we have also some models that introduce time on tokens [1, 6]. In such a case, tokens become classified into two different classes: available and unavailable ones. Available tokens are those that can be immediately used for firing a transition, while unavailable tokens cannot. We have to wait for a certain period of time for these tokens to become available, although it is also possible for a token to remain unavailable forever (such tokens are said to be *dead*). More recently, Cerone and Maggiolo-Schettini

[8] have defined a very general model (statically timed Petri nets), where timing constraints are intervals statically associated with places, transitions and arcs. Thus, models with timing constraints attached only to places, transitions or arcs can be obtained as particular subclasses of this general framework.

In this paper we analyze timed-arc Petri nets [6, 23, 14], a timed extension of Petri nets in which tokens have associated a natural[1] value indicating the elapsed time from its creation (its *age*), and arcs from places to transitions are also labelled by time intervals, which establish restrictions on the age of the tokens that can be used to fire the adjacent transitions.

In [6], Bolognesi *et. al* describe timed-arcs Petri nets, comparing them with Merlin's model in the framework of design and analysis of concurrent systems. The interpretation and use of timed-arcs Petri nets can be obtained from a collection of processes interacting with each other according to a rendez-vous mechanism. Each process may execute either local actions or synchronization ones. Local actions are those that the process may execute without cooperation from another process, and thus in the Petri net model of the whole system they would appear as transitions with a single precondition place, while synchronization actions would have several precondition places, which correspond to the states at which each one of the involved processes is ready to execute the action. Then, each time interval establishes some timing restrictions related to a particular process (for instance the time that a local processing may require). In consequence, the firing of a synchronization action can be done in a time window, which depends on the age of the tokens on its precondition places.

One of the applications of timed-arc Petri nets comes from the fact that it is quite easy to get a timed-arc Petri net modeling a system that has been described by means of a *Timed-LOTOS* specification [18]. Therefore, in particular, the interest of the model can be justified as a graphical tool for the design and analysis of concurrent systems.

In [5], it is proved that timed-arc Petri nets are not Turing complete, since in particular they cannot correctly simulate a 2-counter machine. Thus, we could expect that the differences with untimed Petri nets in terms of expressiveness would not be rather significant. Nevertheless, we have shown in [22] that the difference is big enough to make the reachability problem undecidable for timed-arc Petri nets.

In this paper we extend the study of decidability of properties of TAPN's proving, in particular, that coverability and boundedness are both decidable. This is a consequence of the close connection between TAPN's and transfer nets, which can simulate each other, as we show in this paper. As a consequence, the

[1] Although it is usual to define TAPN's taking real numbers to measure the passage of time, in this paper we will only allow natural numbers, since, for the time being, we have not been able to generalize our decidability results to the continuous time models. Instead, we considered general TAPN's when proving undecidability of reachability in [22], but it is important to note that our counterexample proving undecidability makes no use of continuous time. Reachability is also undecidable when discrete time is considered.

(un)decidability of any property preserved by these simulations can be translated from any of these classes to the other.

Transfer nets and reset nets have been thoroughly studied in [10, 9, 11] as particular cases of the so called Generalized Self-Modifying nets (shortly G-nets). In particular, it is proved there that coverability is decidable for both classes of nets, by applying a new general[2] *backward* technique, presented in [2, 13]. On the contrary, boundedness is decidable for transfer but not for reset nets. A corollary of this decidability result is that place boundedness is undecidable in both cases.

After studying the decidability of several classic properties of TAPN's, we will concentrate on the detection of *dead* tokens. Due to the time restrictions in the nets, and also to their structure, some tokens may become *dead*, since they are too old to fire any transition in the future. Thus they stay attached to their places forever, growing and growing, and never being available. We will prove that this kind of tokens can be effectively detected, because of the fact that the firability of transitions is closely related with coverability, that it is proved to be decidable.

The paper is structured as follows. In Section 2 we present timed-arc Petri nets and their semantics; in Section 3 we recall the undecidability of reachability, and prove that, instead, it becomes decidable if we fix the duration of the considered computations. In section 4 we prove the decidability of coverability. Section 5 is the main section of the paper; there we prove the close relationship between TAPN's and transfer nets, and as a consequence we transfer to TAPN's most of the known (un)decidability results for transfer nets. In Section 6 we show that the problem of detecting dead tokens is decidable, and we also study a weaker version of that property. Finally, in Section 7 we discuss the work to be done in the future.

2 Timed-Arc Petri Nets

We deal with timed-arc Petri nets, which have their tokens annotated with an age (an integer value indicating the elapsed time from its creation), and where arcs connecting places with transitions have associated a time interval, which limits the age of the tokens consumed to fire the adjacent transition.

However, a transition is not forced to be fired when all its preconditions contain tokens with an adequate age, and the same is true even if the age of any of these tokens is about to expire. More in general, in the model there is not any kind of urgency, which can be interpreted in the sense that the model is *reactive*, as transitions will only be fired when the external context requires it. But then, it may be the case that the external context may lose the ability to fire a transition if some needed tokens become too old. Even more, it is possible that

[2] What is mainly new in these papers is their generality, which comes from the fact that the presented results can be applied to any well structured transition system. Instead, a *backward* algorithm to decide coverability for reset nets was presented in [4] as long ago as in 1976.

some tokens become *dead*, which means definitely useless because the increasing of their age will not allow in the future the firing of any of their postcondition transitions.

Definition 1. *(Timed-arc Petri nets)*
We define a **timed-arc Petri net** *(TAPN) as a tuple* [3] $N = (P, T, F, \texttt{times})$, *where* P *is a finite set of* **places**, T *is a finite set of* **transitions** $(P \cap T = \emptyset)$, F *is the* **flow relation** $(F \subseteq (P \times T) \cup (T \times P))$, *and* $\texttt{times}$ *is a function that associates to each arc* (p, t) *in* F *a pair of natural numbers, the second of which can be infinite, i.e.:* $\texttt{times} : F|_{P\times T} \longrightarrow \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$.

When $\texttt{times}(p, t) = [t_1, t_2]$ we write $\pi_i(p, t)$ to denote t_i, for $i = 1, 2$. Since $\texttt{times}$ defines the intervals of age of the tokens to be consumed by the firing of each transition (see Def.2), we will always have $\pi_1(p, t) \leq \pi_2(p, t)$. Moreover, we will write $x \in \texttt{times}(p, t)$ to denote $\pi_1(p, t) \leq x \leq \pi_2(p, t)$.

As we previously mentioned, tokens are annotated with natural numbers, so markings are defined by means of multisets on $\mathbb{N}$. More exactly, a marking M is a function $M : P \longrightarrow \mathcal{B}(\mathbb{N})$ where $\mathcal{B}(\mathbb{N})$ denotes the set of finite multisets of natural numbers. Thus, as usual, each place is annotated with a certain number of tokens, but each one of them has associated a natural number (its *age*). We will denote the set of markings of N by $\mathcal{M}(N)$, and using classical set notation, we denote the total number of tokens on a place p by $|M(p)|$. Finally, by some abuse of notation, sometimes we will denote the individual tokens in a marking M by pairs (p, i) with $p \in P$ and $i \in \mathbb{N}$ denoting its age, also writing $(p, i) \in M$.

As initial markings we only allow markings M such that for all p in P, and any $x > 0$ we have $M(p)(x) = 0$ (i.e., the *initial age* of any token is 0)[4]. Then, we define *marked timed-arc Petri nets* (MTAPN) as pairs (N, M), where N is a timed-arc Petri net, and M is an initial marking on it. As usual, from this initial marking we will obtain new markings, as the net evolves, either by firing transitions, or by the passage of time. In consequence, given a non-zero marking, even if we do not fire any transition at all, starting from this marking we get an infinite reachability set of markings, due to the token aging.

A timed-arc Petri net with an arbitrary marking can be graphically represented by extending the usual representation of P/T nets with the corresponding time information. In particular, we will use the age of each token to represent it. Therefore, MTAPN's have initially a finite collection of zero values labelling each place. In Fig.1 we show a MTAPN modeling a producer/consumer system.

Let us now see how we can fire transitions, and how we model the passage of time.

[3] To simplify some definitions we consider only arcs with weight 1, but the extension to general arcs with greater weights is straightforward.

[4] In fact, it would not be a problem to allow initial markings containing older tokens, as long as we only have a finite number of tokens. Then, the main reason for imposing this restriction is to capture the intuitive idea that initial tokens have not had yet time to become old.

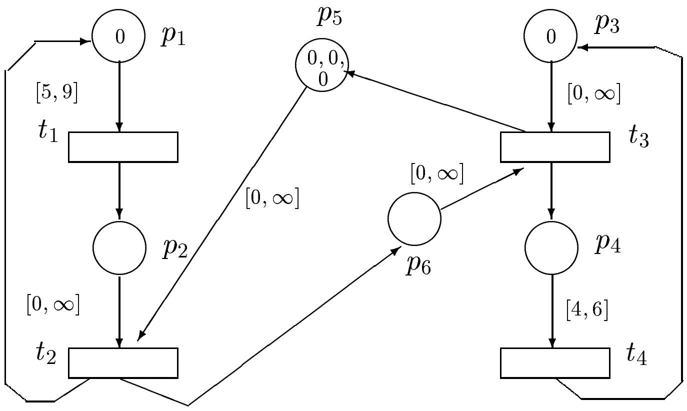

Fig. 1. Timed-arc Petri net modeling the PC-problem

Definition 2. *(Firing rule)*
Let $N = (P, T, F, \mathtt{times})$ *be a TAPN,* M *a marking on it, and* $t \in T$.

(1) We say that t *is* **enabled** *at marking* M *if and only if:*

$$\forall p \in {}^\bullet t \;\exists x_p \in \mathbb{N} \textit{ such that } M(p)(x_p) > 0 \wedge x_p \in \mathtt{times}(p, t)$$

i.e., on each precondition of t *we have some token whose age belongs to* $\mathtt{times}(p, t)$.

(2) If t *is enabled at* M*, it can be fired, and by its firing we reach any marking* M' *which can be obtained as follows:*

$$M'(p) = M(p) - C^-(p, t) + C^+(t, p),\ \forall p \in P$$

where both the subtraction and the addition operators work on multisets, and

- $C^-(p, t) \in \begin{cases} \{\{x_p\} \mid x_p \in M(p) \text{ and } x_p \in \mathtt{times}(p, t)\} & \text{if } p \in {}^\bullet t \\ \{\varnothing\} & \textit{otherwise} \end{cases}$
- $C^+(t, p) = \begin{cases} \varnothing & \textit{if } p \notin t^\bullet \\ \{0\} & \textit{otherwise} \end{cases}$

Thus, from each precondition place of t *we remove a token fulfilling (1), and we add a new token (with age 0) on each postcondition place of* t.

As usual, we denote these evolutions by $M[t\rangle M'$*, but it is noteworthy that these evolutions are in general non-deterministic, because when we fire a transition* t*, some of its precondition places could hold several tokens with different ages, that could be used to fire it. Besides, we see that the firing of transitions does not consume any time. Therefore, to model the passage of time we need the function* `age`*, defined below. By applying it, we age all the tokens of the net by the same time:*

(3) The function $\mathtt{age} : \mathcal{M}(N) \times \mathbb{N} \longrightarrow \mathcal{M}(N)$ *is defined by:*

$$\mathtt{age}(M, x)(p)(y) = \begin{cases} M(p)(y - x) & \textit{if } y \geq x \\ 0 & \textit{otherwise} \end{cases}$$

The marking obtained from M after x units of time without firing any transitions will be that given by $\texttt{age}(M, x)$.

Although we have defined the evolution by firing single transitions, this can be easily extended to the firing of *steps* or *bags* of transitions, since those transitions that could be fired together in a single step could also be fired in sequence in any order, because no *aging* is produced by the firing of transitions. In this way, we obtain step transitions that we denote by $M[R\rangle M'$. By alternating step transitions and the passage of time we can define a timed step semantics, where timed step sequences are those sequences $\sigma = M_0[R_1\rangle_{x_1} M_1 \ldots M_{n-1}[R_n\rangle_{x_n} M_n$, where $M_i's$ are markings, $R_i's$ multisets of transitions and $x_i's \in \mathbb{N}$, in such a way that $M_i[R_{i+1}\rangle M'_{i+1}$ and $M_{i+1} = \texttt{age}(M'_{i+1}, x_{i+1})$. Note that we admit $x_i = 0$ in order to allow null intervals of time between two causally related steps. However, note that this does not imply that both steps could be fired together in a single step. Therefore, we maintain the distinction between parallelism and sequentiality. Finally, we denote sequences σ by $M_0[\,\overline{R}\,\rangle_x M_n$, where $\overline{R} = R_1 R_2 \ldots R_n$ and $x = \sum\limits_{i=1}^{n} x_i$, whenever we do not need to keep track of the values of each x_i.

Most of the interesting properties of TAPN's, as the terminology to define them, are direct translations of properties of untimed Petri nets, although in some cases the introduction of time makes necessary to change the way in which these properties are defined. Besides, we also have some new properties for which time plays a central role.

Given a MTAPN (N, M_0), we define $[M_0\rangle$ as the set of reachable markings on N starting from M_0. Given $m \in \mathbb{N}$ we define by $[M_0\rangle_m$ the set of reachable markings by a computation of total duration m:

$$[M_0\rangle_m = \{\ M \mid \exists\, \overline{R},\ M_0[\,\overline{R}\,\rangle_m M\ \}$$

When $M \in [M_0\rangle_m$ we say that M is reachable in time m. The MTAPN (N, M_0) *terminates* if there is no infinite computation. A place $p \in P$ is *bounded* if there exists $n \in \mathbb{N}$ such that for all $M \in [M_0\rangle$ we have $|M(p)| \leq n$; we say that it is *time-locally bounded* if for all $t \in \mathbb{N}$ there exists $n \in \mathbb{N}$ such that for all $M \in [M_0\rangle$ we have $|M(p)(t)| \leq n$; finally, it is *uniformly time-locally bounded* if there exists $n \in \mathbb{N}$ such that for all $t \in \mathbb{N}$ and all $M \in [M_0\rangle$ we have $|M(p)(t)| \leq n$. We say that (N, M_0) is *bounded* (resp. *time-locally bounded, uniformly time-locally bounded*) if every $p \in P$ in it is bounded (resp. time-locally bounded, uniformly time-locally bounded)[5].

As a matter of fact, we have only defined uniformly time-locally boundedness for completeness, since for TAPN's it is immediate to prove the following property:

Proposition 3. *Given a MTAPN (N, M_0) and $p \in P$, we have that p is time-locally bounded in N iff it is uniformly time-locally bounded in it.*

[5] Note that we cannot define boundedness equivalently by saying that the reachability set is finite, since by aging a marking we can always obtain older and older different markings.

Proof. Since only tokens of age 0 are created, for all $M_1 \in [M_0\rangle_N$ and all $m \in \mathbb{N}$ there exists $M_2 \in [M_0\rangle_N$ such that for all $p \in P$ we have $M_1(p)(m) \leq M_2(p)(0)$. As a consequence, if p is time-locally bounded in N, the corresponding bound for age 0 is a uniform bound for any age. □

3 Decidability of Timed Reachability

In [22] we have proved the following result:

Theorem 4. *The reachability problem for MTAPN's is undecidable.*

However, timed reachability is decidable, since given (finite) time m one can simulate the behaviour of a MTAPN up to that instant by means of a plain Petri net N^m, which is defined as follows:

Definition 5. *Given a TAPN $N = (P, T, F, \mathtt{times})$, an initial marking M_0 for it* [6] *and $m \in \mathbb{N}$, we define the associated P/T Petri net $N^m = (P^m, T^m, F^m)$ as follows:*

- $P^m = (P \times 0..m) \cup \{ c_i \mid i \in 0..m \}$,
- *The set of transitions is the set*
  $T^m = \{ (t, \mathtt{ages}, l) \mid \mathtt{ages} : \{ p \mid (p,t) \in F \} \to (\mathtt{times}(p,t) \cap 0..m),\ l \in 0..m \}$
  $\cup \{ \mathtt{tick}_i \mid i \in 1..m \}$
- *The flow relation, $F^m \subseteq (P^m \times T^m) \cup (T^m \times P^m)$, is defined as the set*

$$\begin{aligned} F^m = \{ & (c_{i-1}, \mathtt{tick}_i), (\mathtt{tick}_i, c_i) \mid i \in 1..m \} \cup \\ \{ & (c_i, (t, \mathtt{ages}, i)) \mid i \in 0..m \} \cup \\ \{ & ((p,j), (t, \mathtt{ages}, i)) \mid (p,t) \in F,\ j = i - \mathtt{ages}(p) \} \cup \\ \{ & ((t, \mathtt{ages}, i), (p,i)) \mid (t,p) \in F\ ,\ a \in 0..i \} \end{aligned}$$

Finally, as initial marking we would take M_0^m defined by

$$\begin{cases} M_0^m(c_0) = 1, & \\ M_0^m(c_i) = 0 & \textit{for all } i > 0, \\ M_0^m((p,0)) = M_0(p) & \textit{for all } p \in P, \\ M_0^m((p,i)) = 0 & \textit{for all } p \in P \textit{ and } i > 0. \end{cases}$$

It is not difficult to see how N^m works, simulating the behaviour of N up to the instant m. The idea is that the age of the tokens is controlled in a static way: tokens do not move when they become older; instead the *clock places* c_i do the work, since depending on the value of the global clock represented by them, the firable transitions would take the consumed tokens from different places. In this way, we capture the fact that the current age of a token on a place $(p,j) \in P^m$ is equal to $i - j$, where c_i is the currently marked clock place. All this is formalized as follows:

[6] It would not be difficult to extend this construction in order to allow any finite marking as the initial marking M_0. In such a case, we take $P^m = P \times -\mathtt{old}(M_0)..m$, where $\mathtt{old}$ is defined as in Def.6. Each token (p,i) in M_0 would be represented by a token on the corresponding place $(p, -i) \in P^m$.

Definition 6. *Let $N = (P, T, F, \texttt{times})$ be a TAPN with an initial marking M_0. Given M an arbitrary marking of N and $m \in \mathbb{N}$:*

(1) We define $\texttt{old}(M) = max\{\, i \mid \exists i \in M(p)\,\}$.

(2) If $\texttt{old}(M) \leq m$ we say that the marking M^m of N^m defined by

$$\begin{cases} M^m(c_m) = 1, & \\ M^m(c_i) = 0 & \text{for all } i < m, \\ M^m((p,j)) = M(p)(m-j) & \text{for all } j \in 0..m \end{cases}$$

is the representation of M in N^m.

Theorem 7. *For each marking M of a TAPN N such that $\texttt{old}(M) \leq m$ we have that $M \in [M_0\rangle_m$ iff $M^m \in [M_0^m\rangle_{N^m}$.*

Corollary 8. *Timed reachability is decidable for TAPN's.*

4 Decidability of Coverability

In this section we will prove that coverability is decidable for timed-arc Petri nets. Nevertheless, the proof of this result cannot be obtained by generalizing the coverability tree [12] in a proper way. Instead, we have to use a *backward* technique recently presented in [2, 13].

Definition 9. *([12, 2] Well-structured transition systems)*
A **well-structured transition system** *(WSTS) is a structure $S = \langle Q, \rightarrow, \sqsubseteq \rangle$ such that:*

(1) $Q = \{M, \ldots\}$ is a set of **states**,

(2) $\rightarrow \subseteq Q \times Q$ is a set of **transitions**,

(3) $\sqsubseteq \subseteq Q \times Q$ is a **well-quasi-ordering** *(wqo) on the set of states, that is, a reflexive, transitive and well founded relation such that*

$$M \rightarrow M' \text{ and } M_1 \sqsupseteq M \text{ imply } M_1 \rightarrow M_1' \text{ for some } M_1' \sqsupseteq M'.$$

Definition 10. *(Upward closed sets)*
Let S be a WSTS and Q its set of states.

(1) We say that $K \subseteq Q$ is **upward closed** *iff whenever $M \in K$ we also have $M' \in K$ for all $M \sqsubseteq M'$.*

(2) If $K \subseteq Q$ is upward closed and $s \in \mathbb{N}$, we define $\texttt{cover}_s(K)$ as the set of all the states from which one can reach a state in K in exactly s steps.

(3) If K is upward closed and $B \subseteq K$, we say that B is a **basis** *of K iff $K = \{\, M' \mid \exists M \in B,\ M \sqsubseteq M' \,\}$.*

¿From now on, for $K \subseteq Q, \uparrow K$ denotes the set $\{M' \in Q | \exists M \in K, M \sqsubseteq M'\}$.

Proposition 11. *If S is a WSTS, any upward closed set in it admits a finite basis.*

Definition 12. *(Effective WSTS)*
Let S be a WSTS. We say that S is **effective** *iff the following conditions hold:*

(1) $\sqsubseteq$ is decidable.
(2) For each finite set $B \subseteq Q$ we can effectively compute a finite basis of the set $\uparrow(\mathtt{cover}_1(\uparrow B))$.

Theorem 13. *[2, 13] For effective WSTS's coverability is decidable.*

Next we recall the main ideas of the proof of this result in the case of Petri nets:

- If M can be covered from M_0 there exists $M' \in [M_0\rangle$ with $M \sqsubseteq M'$. Then we consider the last step of these computations, taking $M'' \in [M_0\rangle$ such that $M''[t\rangle M'$. We take as $\mathtt{cover}_1(M)$ the set of these markings M''.
- Iterating this construction we obtain $\mathtt{cover}_k(M)$ for all $k \in \mathbb{N}$. It can be proved that each one of these sets is upward closed.
  Then we define $\mathtt{cover}^*_k(M)$ as $\bigcup\limits_{l \leq k} \mathtt{cover}_l(M)$. Obviously, $k \leq k'$ implies $\mathtt{cover}^*_k(M) \subseteq \mathtt{cover}^*_{k'}(M)$. But by applying the fact that $\sqsubseteq$ is a well-quasi-ordering, there must exist some $k \in \mathbb{N}$ such that $\mathtt{cover}^*_{k+1}(M) = \mathtt{cover}^*_k(M)$. If we take as $\mathtt{cover}^*(M)$ such a set, it is clear that M can be covered from M_0 iff $M \in \mathtt{cover}^*(M_0)$.
- Since the sets $\mathtt{cover}^*_k(M_0)$ are infinite we could not cope directly with them in an effective way. But applying Prop.11 we can effectively characterize these sets by their finite basis.

In order to apply this technique to prove the decidability of the coverability problem for MTAPN's, we have to find an adequate ordering relation between their markings. The natural candidate would be the multiset inclusion, as it is the immediate generalization of the adequate ordering for ordinary nets. Unfortunately, that would not work since $\mathbb{N}$ is an infinite set, and then if for any place $p \in \mathbb{N}$ we consider the singleton markings $M_i = \{(p,i)\}$ we have $M_i \not\subseteq M_j$, whenever $i \neq j$. Nevertheless, it is possible to find an adequate ordering taking into account the fact that when a token becomes old enough its age becomes unimportant since it can be only used to fire those transitions for which $M_2(p,t) = \infty$. This is formalized as follows:

Definition 14. *Let $N = (P, T, F, \mathtt{times})$ be a TAPN.*

(1) For each $p \in P$ we define
$$Max(p) = Max\{\{\,\pi_1(p,t) \mid t \in p^\bullet\} \cup \{\pi_2(p,t) \mid t \in p^\bullet\,,\ \pi_2(p,t) < \infty\,\}\}$$
and $S(p) = Max(p) + 1$.
(2) We define the **stable time** *of N, $St(N)$, as $Max\{\,S(p) \mid p \in P\,\}$.*

Thus, we have that once the age of a token on p exceeds $Max(p)$ the only postcondition transitions $t \in p^\bullet$ that could be fired by using that token are those for which $\pi_2(p,t) = \infty$. Obviously, in order to fire such a transition t the age of the involved token on p is unimportant once it exceeds $S(p)$. The same is true in a uniform way for all the places of the net if we take $St(N)$ instead of the values $S(p)$.

Now we can define an adequate ordering relation $\sqsubseteq'$ between markings of a time-arc Petri net as follows:

Definition 15. *Let $N = (P, T, F, \texttt{times})$ be a TAPN and M,M' two of its markings. $M \sqsubseteq' M'$ iff the following conditions hold:*

(1) $M|_{St(N)-1} \subseteq M'|_{St(N)-1}$ where for all $p \in P$, $i \in \mathbb{N}$, for any marking M and any $k \in \mathbb{N}$ we have

$$M|_k(p)(i) = \begin{cases} M(p)(i) & \textit{if } i \leq k \\ 0 & \textit{otherwise} \end{cases}$$

(2) For all $p \in P$ $\quad |\{i \in M(p) \mid i \geq St(N)\}| \leq |\{i \in M'(p) \mid i \geq St(N)\}|$.

Proposition 16.

(1) $\sqsubseteq'$ is a well-quasi-ordering.

(2) If we consider the equivalence relation $\sim'$ induced by $\sqsubseteq'$, whenever $M_1 \sim' M_2$, for all $t \in T$ we have that $M_1[t\rangle \Leftrightarrow M_2[t\rangle$, and if $M_1[t\rangle M_1'$ there exists M_2' such that $M_2[t\rangle M_2'$ and $M_1' \sim' M_2'$.

As an immediate consequence we have the following

Theorem 17.

(1) The transition system generated by the firing rule of any TAPN, N, becomes a well-structured transition system when we take as ordering relation between its markings the corresponding relation $\sqsubseteq'$.

(2) Coverability is decidable for timed-arc Petri nets.

Proof.

(1) Immediate, by application of the firing rule for TAPN's.

(2) Let M be the marking that we want to know if it is coverable from M_0. If $M = M|_{St(N)-1}$ then we can apply the procedure in the proof of Th.13 in order to decide if it is coverable. Otherwise, this procedure cannot be directly used, because of the fact that $\sqsubseteq'$ does not preserve the information about the exact age of the tokens once they become older than $St(N)$. Nevertheless, since markings are finite, we can consider $max(M) = max\{i \mid \exists(p,i) \in M\} < \infty$, taking as ordering relation $\sqsubseteq''$ that is defined exactly as $\sqsubseteq'$ but replacing $St(N)$ by $max(M) + 1 > St(N)$. Then we have that $\sqsubseteq''$ has the same good properties as $\sqsubseteq'$, and in addition it fully preserves the information about the age of the tokens when they are younger than $max(M)$. Therefore, using this ordering relation one can decide if the given marking M is coverable from M_0.

$\square$

We could continue the study of some other properties of timed-arc Petri nets in a direct way, but in order to avoid the repetition of similar reasonings to those made in [9, 10] when studying reset and transfer nets, we will prove in the next section that transfer nets and timed-arc Petri nets can simulate each other. This makes possible to translate to TAPN's most of the known results about decidability and undecidability for transfer nets.

5 Relating Timed-Arc Petri Nets and Transfer Nets

We start this section by presenting reset and transfer nets. Both are particular cases of Generalized Self-Modifying nets which have been thoroughly studied in [10].

Definition 18. *(Reset and transfer nets)*

(1) A **reset net** *is a 3-tuple $N = \langle P, T, F\rangle$ where*

- $P = \{p_1, \ldots, p_{|P|}\}$ *is a finite set of* **places***,*
- T *is a finite set of* **transitions** *(with $P \cap T = \varnothing$),*
- $F : (P \times T) \cup (T \times P) \to \mathbb{N}^\infty$ *is a* **flow function** *where $\mathbb{N}^\infty = \mathbb{N} \cup \{\infty\}$ and for all $(t, p) \in (T \times P)$, $F(t, p) \in \mathbb{N}$.*

Whenever $F(p, t) = \infty$ we say that (p, t) is a **reset arc** *and that t* **resets** *p.*

(2) A **marked reset net** *is a pair (N, m_0) where N is a reset net and $m_0 \in \mathbb{N}^{|P|}$ is the* **initial marking***.*

(3) A **transfer net** *is a 4-tuple $N = \langle P, T, F, TA\rangle$ where $\langle P, T, F\rangle$ defines an ordinary P/T net and TA is a function $TA : T \to \mathcal{P}(P \times P)$, that defines the* **transfer arcs***, , verifying*

$$(p, p') \in TA(t) \Rightarrow \Big((\neg\exists p'' \neq p' \ (p, p'') \in TA(t)) \ \wedge \ F(p, t) = F(t, p') = 0\Big).$$

(4) A **marked transfer net** *is a pair (N, m_0) where N is a transfer net and $m_0 \in \mathbb{N}^{|P|}$ is the* **initial marking***.*

Firing rules for reset and transfer nets are defined as follows:

Definition 19. *(Firing rules)*
A transition t is **firable** *in a reset or transfer net whenever it is firable in the corresponding plain P/T net obtained by removing reset and transfer arcs. The firing of t from M produces a new marking M' which is defined exactly as for ordinary nets, but whenever we have a reset or a transfer arc we have instead:*

(1) If $F(p, t) = \infty$ then $M'(p) = F(t, p)$, which means that we take all the tokens from p, and we only return someone if p is a postcondition of t.

(2) If $(p, p') \in TA(t)$, we have the following (non disjoint) cases [7]:

(a) There is no place p'' such that $(p', p'') \in TA(t)$: $M'(p') = M(p') + M(p)$.

[7] We have to distinguish whether we have one or more transfer arcs associated to the same transition.

(b) *There is a place p'' such that $(p',p'') \in TA(t)$: $M'(p') = M(p)$.*
(c) *There is no place p'' such that $(p'',p) \in TA(t)$: $M'(p) = F(t,p)$.*
(d) *If there is a place p'' such that $(p'',p) \in TA(t)$, one can proceed as in case (b), since we have $(p'',p),(p,p') \in TA(t)$.*

Next we recall some of the results about decidability of properties of reset and transfer nets which have been presented in [10, 15].

Theorem 20.

(1) *Coverability and termination are decidable for reset and transfer nets.*
(2) *Boundedness is decidable for transfer nets, but undecidable for reset nets.*
(3) *Structural boundedness is decidable for transfer nets.*
(4) *Place boundedness is undecidable for both reset and transfer nets.*

In the following we will see that timed-arc Petri nets and transfer nets are closely related. First we will see how to simulate timed-arc Petri nets by transfer nets. In order to do it, we need some previous definitions:

Definition 21. *Let $N = (P, T, F, \mathtt{times})$ be a TAPN and $B \in \mathbb{N}$ such that $B \geq St(N)$. We define the transfer net $N^B = (P^B, T^B, F^B, TA^B)$ which simulates N by preserving the age information up to the instant B, as follows:*

- $P^B = P \times 0..B$,
- $T^B = \{(t, \mathtt{ages}) \mid t \in T, \mathtt{ages} : \{p \mid (p,t) \in F\} \to (\mathtt{times}(p,t) \cap 0..B)\} \cup \{\mathtt{tick}\}$,
- $F^B = \{((p,a),(t,\mathtt{ages})) \mid \mathtt{ages}(p) = a\} \cup \{((t,\mathtt{ages}),(p,0)) \mid (t,p) \in F\}$,
- *For all* $(t, \mathtt{ages}) \in T^B \quad TA^B((t,\mathtt{ages})) = \varnothing$,
- $TA^B(\mathtt{tick}) = \{((p,a),(p,a+1)) \mid p \in P, a < B\}$.

In this definition, each place (p,a) with $a < B$ represents the tokens in the original place $p \in P$ with age a, while (p,B) represents those tokens whose age is greater or equal than B. Each transition $(t, \mathtt{ages})$ represents the firing of t by consuming tokens in its precondition places whose ages are defined by the values of $\mathtt{ages}$. So, each possible value of the function $\mathtt{ages}$ corresponds to a different selection of the ages of the consumed tokens (thus to any of the different ways to fire t in N, as defined in Def.2(2)). To be exact, we are identifying all the ages older than B, what has no negative influence, taking into account that $B \geq St(N)$. Finally, the passage of time is modelled by the transition $\mathtt{tick}$ that ages all the tokens in the net that are not older than B.

As a consequence, the markings of both nets N and N^B are related as follows:

Definition 22.

(1) *Given a marking M of N we define the associated marking M^B of N^B by*
$$\begin{cases} M^B(p,a) = M(p)(a) & \text{if } a < B \\ M^B(p,B) = \sum\limits_{a \geq B} M(p)(a) & \end{cases}$$

(2) Given a marking M' of N^B we define the set of markings represented by M', $tr(M')$, as $tr(M') = \{ M \in \mathcal{M}(N) \mid M^B = M' \}$.

The following theorem formalizes the fact that N^B is a full simulation of N up to age B:

Theorem 23. *Let N be a TAPN, $B \geq St(N)$ and N^B the associated transfer net, we have:*

(1) For any marking M' of N^B, if $M_1, M_2 \in tr(M')$ then $M_1|_{B-1} = M_2|_{B-1}$.
(2) Whenever we have $M_1[t\rangle_N M_2$, there exists some $(t, \mathtt{ages}) \in T^B$ such that $M_1^B[(t, \mathtt{ages})\rangle_{N^B} M_2^B$.
(3) If $M_1'[(t, \mathtt{ages})\rangle_{N^B} M_2'$, then for all $M_1 \in tr(M_1')$ we have $M_1[t\rangle_N M_2$ for some $M_2 \in tr(M_2')$.

Therefore, TAPN's can be smoothly simulated by transfer nets; as a consequence, any decidable property for these latter preserved by the simulation is also decidable for the former. In particular, we could have used this simulation to prove decidability of coverability, but we prefered to do it in a direct way in order to emphasize the role of the bound $St(N)$, which allow us to reason about the ages of the tokens in a finitary way. We also have the following:

Corollary 24.

(1) Boundedness of MTAPN's is a decidable property.
(2) Structural boundedness of MTAPN's is also decidable.

By applying this simulation we can also conclude the decidability of termination of TAPN's. Nevertheless, this cannot be done in a straightforward way, since the computations of the nets N^B never terminate (because `tick` transition has no precondition and therefore it can be executed forever). Instead, we have to study if any infinite computation of this net does not terminate with an infinite suffix containing only `tick` transitions. But, due to the structure of these nets, whenever we iterate the execution of `tick` for at least $St(N)$ times we reach a stable marking where all the tokens are $St(N)$ old. Besides, it is easy to see that before this happens we can only reach a marking that is bigger than or equal to some previous one in the computation if we have executed some non-`tick` transition in between. Therefore, by applying the finite reachability technique in [12], and not taking into account the execution of `tick` when it does not change the current marking, we can prove the following:

Corollary 25. *Termination of TAPN's is decidable.*

In order to prove that the undecidability results for transfer nets are also preserved on TAPN's we need the converse simulation. This is a much more elaborated construction, that is somehow based on the simulation of counter machines that was used in [22] to prove undecidability of reachability for MTAPN's, and in [10, 11], to prove undecidability of boundedness for reset nets. The main ideas of this simulation are to introduce transitions to represent transfer arcs, and to force the transitions to consume tokens one instant old. Together with the *good* computations also some bad but *non-dangerous* computations will be generated. Besides, older tokens become dead.

Definition 26. *Let $N = \langle P, T, F, TA\rangle$ be a transfer net. We define the associated TAPN, $N^I = \langle P^I, T^I, F^I, \mathtt{times}^I\rangle$, as follows:*

- *$P^I = P \cup \{n\} \cup \{r_t \mid t \in T\}$, where n and each r_t will be control states to capture the different (normal and reset) working states of the net.*
- *$T^I = T \cup \{v_{t,p} \mid p \in P, t \in T\} \cup \{rn_t \mid t \in T\}$, where the transitions $v_{t,p}$ will be used to reset the age of the tokens after the firing of t and transfer them to the corresponding place when there exists some transfer arc $(p, p') \in TA(t)$. Finally, the transitions rn_t are used to recover the normal state of the net.*
- *The flow function, $F^I \subseteq (P^I \times T^I) \cup (T^I \times P^I)$ is defined as the set*

$$\begin{aligned} F^I = F \; & \cup \{(p, v_{t,p}), (r_t, v_{t,p}), (v_{t,p}, r_t) \mid p \in P, t \in T\} \\ & \cup \{(v_{t,p}, p') \mid p \in P, t \in T, (p, p') \in TA(t)\} \\ & \cup \{(v_{t,p}, p) \mid p \in P, t \in T, \neg\exists (p, p') \in TA(t)\} \\ & \cup \{(r_t, rn_t), (rn_t, n) \mid t \in T\} \\ & \cup \{(n, t), (t, r_t) \mid t \in T\} \end{aligned}$$

- *$\mathtt{times}^I : F^I|_{P^I \times T^I} \to \mathbb{N} \times \mathbb{N}$ is defined by*

$$\mathtt{times}^I(p^I, t^I) = \begin{cases} [0,0] \text{ if } p^I \in \{r_t \mid t \in T\} \\ [1,1] \text{ otherwise.} \end{cases}$$

It is easy to see that N^I can simulate the behaviour on N in the following way: Each marking M of N will be represented by the marking M^I given by

- For each $p \in P$ with $M(p) = k$ we take $M^I(p) = 0^k$, where 0^k denotes the multiset containing k zero values.
- $M^I(n) = 0^1$ and $M^I(r_t) = \varnothing$, for all $t \in T$.

Then, whenever we have $M[t\rangle_N M'$, we can represent in N^I the firing of t by means of the following steps:

(a) $M^I[\varnothing\rangle_1 \mathtt{ages}(M^I, 1)$, that is, we age one unit all the tokens in M^I.
(b) $\mathtt{ages}(M^I, 1)[t\rangle_0 M''^I$, that is, we fire (the copy of) t. As a consequence, the tokens in ${}^\bullet t$ are removed from these places, having age 1, while the tokens in $t^\bullet$ are added to these places, having age 0.
(c) In order to obtain the marking representing M' in N^I, M'^I, we have to transfer the tokens in each $p \in P$, for which there exists some $(p, p') \in TA(t)$, to the corresponding p'. This is made by firing $M''^I(p)(1)$ copies of $v_{t,p}$. Besides, for each $p \in P$ for which there is no transfer arc $(p, p') \in TA(t)$ we have to rejuvenate the tokens in $M''^I(p)$ by firing $M''^I(p)(1)$ copies of the corresponding transition $v_{t,p}$.
(d) Finally, we recover the "normal" state of N^I by firing rn_t.

It is easy to see that no other transition can be fired until we terminate a step of the simulation by firing the last transition rn_t. Therefore, we only have two possibilities: either we strictly follow the procedure above, thus obtaining M'^I, or we fire rn_t in advance to get some marking M'''^I. In this last case, if we denote by M^{*I} the submarking of M'''^I constituted by the tokens having age 0, we have $M^{*I} \subseteq M'^I$. However, the older tokens in M'''^I, having age 1, become dead, since the only way in which N^I can evolve is by aging M^{*I}, and in such a case these tokens become too old to be used in the future, since no transition in N^I can consume a token older than 1.

So we obtain the following:

Theorem 27. *N^I weak-simulates N, which means that by means of the computations of transitions in N^I we obtain as reachable markings, when restricting ourselves to tokens having age 0, the representations M'^I of the reachable markings in (N, M_0) and their approximations $M^{*I} \subseteq M'^I$.*

As a consequence, most of the undecidability results for transfer nets can be translated to TAPN's. In particular, we have the following:

Proposition 28. *Timed-locally boundedness is undecidable for TAPN's.*

Proof. Since the projections of the markings of the simulating net N^I on the set of tokens having age 0 are just the representations of the reachable markings in N and their approximations, for each $p \in P$ we have that p is bounded in N iff there exists some $k \in \mathbb{N}$ such that for all $M' \in [M^I\rangle_{N^I}$ we have $M'(p)(0) \leq k$. Therefore, since place boundedness is undecidable for transfer nets, we conclude that timed-locally boundedness is also undecidable for TAPN's. $\square$

In the following section we will present some more consequences of the mutual reductions between transfer nets and TAPN's. Before, we will note that the same construction can be used to simulate reset nets, by only including recovering transitions $v_{t,p}$ for those places $p \in P$ with $F(p,t) \in \mathbb{N}$. Of course, in this case, even when we exactly replicate the behaviour of the original net, something is lost; namely, the global boundedness character of the net. This is because we are not able to accurately capture the resetting character of reset arcs: instead of removing the corresponding tokens we just avoid its rejuvenation; so they become dead and thus have no other influence in the forthcoming behaviour of the net, but they remain in the reachable markings, and thus they probably alter the boundedness character of the net. As a consequence of this fact, we do not know yet if place boundedness is or not decidable for TAPN's.

6 Eliminating Dead Tokens

Let us now turn our attention to *dead tokens*. It is obvious that they can be eliminated without affecting the future behaviour of the net, although the reachability set will be affected when doing it. Moreover, the reachable markings with dead tokens are exactly those which are obtained by inserting these dead tokens into the markings that are reachable after their removal. Then, we are interested in eliminating dead tokens, because by doing so we decrease the number of reachable markings, thus slightly reducing the state explosion. At first, we have to exactly define what a dead token is.

Definition 29. *(Dead tokens)*
Given a TAPN $N = (P, T, F, \mathtt{times})$ and a marking M of it, we say that a token in M is **dead** *if there is no reachable marking $M' \in [M\rangle$ and no transition $t \in T$ such that t is enabled at M' and some of its firings can consume that token.*

We will see that, as a consequence of the decidability of coverability for TAPN's, dead tokens can be effectively detected. However, this is not obtained as an immediate corollary. Instead, we have to slightly change the coding of a TAPN by a transfer net, by adding a clock:

Definition 30.

(1) We define the **clocked transfer net** *N^{CB} which simulates a TAPN N by representing the global clock of N and preserving the age information up to B, by $N^{CB} = \langle P^B \cup \{\mathtt{clock}\}, T^B, F^{CB}, TA^B\rangle$ where $F^{CB} = F^B \cup \{(\mathtt{tick}, \mathtt{clock})\}$.*
(2) We define the ordering $\sqsubseteq_B$ between markings of the net N^{CB} by $M_1 \sqsubseteq_B M_2$ iff $M_1(\mathtt{clock}) = M_2(\mathtt{clock})$ and $M_1|_{P^B} \subseteq M_2|_{P^B}$.

We have defined in this way the covering ordering $\sqsubseteq_B$ since we need to control the passage of time and this is clearly lost if `tick` tokens are treated in an accumulative way. But it is clear that $\sqsubseteq_B$ is not a well-quasi-order since the number of `tick`-tokens is not bounded a priori. Fortunately, it is only necessary to measure time up to a known bound. This can be done by slightly changing the definition of N^{CB}:

Definition 31. *Given $B' \in \mathbb{N}$ we define $N^{B,B'}$ as N^{CB}, but taking*

- $P^{B,B'} = P^B \cup \{c_0, \ldots, c_{B'}\}$,
- $T^{B,B'} = T^B \cup \{\mathtt{tick}_1, \ldots, \mathtt{tick}_{B'}, \mathtt{tick}_\infty\}$,
- $F^{B,B'} = F^B \cup \{(c_{i-1}, \mathtt{tick}_i), (\mathtt{tick}_i, c_i) \mid i \in 1..B'\} \cup \{(c_{B'}, \mathtt{tick}_\infty), (\mathtt{tick}_\infty, c_{B'})\}$.

Then we define $\sqsubseteq_{B,B'}$, taking $M_1 \sqsubseteq_{B,B'} M_2$ iff

$$M_1|_{P^B} \subseteq M_2|_{P^B} \ \wedge \ M_1|_{P^{B,B'}-P^B} = M_2|_{P^{B,B'}-P^B}$$

Now, we define *clock safe markings* as those markings M having $\sum_{i=0}^{B'} M(c_i) = 1$, so it is clear that if we take as initial marking any M_0 with $M_0(c_0) = 1$ and $M_0(c_i) = 0,\ \forall i \neq 0$, all the reachable markings are clock safe. And if we restrict $\sqsubseteq_{B,B'}$ to these clock safe markings, it is easy to see that the corresponding transition system is well structured.

Let us now consider a token (p, i) in a given marking of a TAPN N that we take as its initial marking M_0. If it is not dead then we have some reachable marking $M_0[\,\overline{R}\,\rangle_m M$, where m denotes the total ellapsed time, and some firable t, $M[t\rangle$, with $(p,t) \in F$ and $i + m \in \mathtt{times}(p,t)$. As a consequence, we have the following:

Proposition 32. *For each token (p,i) on the initial marking M_0 of a MTAPN N and each $t \in T$ with $(p,t) \in F$, there exists some reachable marking $M_0[\overline{R}\rangle_m M$ with $M[t\rangle$ and $i + m \in \mathtt{times}(p,t)$ iff there exist some $m \in 0..St(N)$ with $i + m \in \mathtt{times}(p,t)$ and some $M' \in \mathcal{M}(N^{St(N),St(N)})$ with $M'(c_m) = 1$ and $M'[t\rangle$, such that M' is coverable from $M_0^{B,B'}$ with respect to $\sqsubseteq_{St(N),St(N)}$, where $M_0^{B,B'}$ is defined as M_0^B, adding $M_0^{B,B'}(c_0) = 1$.*

Corollary 33. *We can decide when a token of a marking of a TAPN is dead.*

Proof. We only have to apply Th.13 for each one of the minimal markings of the set $\{M \in \mathcal{M}(N^{St(N),St(N)}) \mid M[t\rangle \text{ and } M(c_m) = 1 \text{ with } i+m \in \mathtt{times}(p,t)\}$ $\square$

We can also consider a weaker version of dead tokens. The reason why we introduce this new notion is because by means of dead tokens we wanted to capture the fact that they are useless, since they cannot be used to fire any transition in the future. However, when we have several identical tokens in the same place it is possible that some (but not all) of them can be used to fire such a transition. If this were the case, under our Def.29 any such token would be said to be *non-dead*. We need a weaker notion if we want to say that some of these identical tokens is indeed (weakly) dead.

Definition 34. *(Weakly dead tokens) Given a TAPN $(P, T, F, \mathtt{times})$ and a marking M of it, a token in M at the place p is* **not**[8] **weakly dead** *if there is some reachable marking $M' \in [M\rangle$ such that M' contains at p a single token as old* [9] *as the given one, and there is some enabled transitions at M' whose firing can consume that token.*

The detection of weakly dead tokens in TAPN's turns out to be a rather pathological task, as the following result shows:

Theorem 35.

(1) It is not possible to effectively detect weakly dead tokens with age 0.
(2) We can effectively detect any weakly dead token having a positive age.

Proof.

(1) We use the fact that reachability in TAPN's can be reduced to zero-reachability of a single given place. This is proved in a similar way as for classic P/T nets.

[8] We define this concept in a negative way in order to have a more clear definition
[9] Taking into account the fact that the given token has possibly grown during this computation.

(2) (sketch) We consider the transfer net N^B simulating N, defined as in Def.21, but introducing into it clock places, as in Def.31, by means of which we can control when a token from the same place and having the same age that the given one, is consumed by the firing of a transition. Then we introduce a counter place where we put a token each time that happens. Then, it is clear that the given token is weakly dead iff we can put in this counter place (at least) as many tokens as we had in the given marking at the same place having the same age as the original one. Since coverability is decidable for TAPN's we conclude that, in this case, weakly dead tokens are effectively detectable.

□

The reason why the situation is different depending on the age of the given token is that whenever we create a new token its age is 0. Therefore, when the age of the given token is positive, we know in advance how many tokens like it have to be consumed in order that the given one will be necessary to fire some new transition. Instead, when the age is 0, we could create some new copies of the given token before the marking is aged, thus making impossible to control when the last clone of the given token is consumed.

7 Conclusions, Discussion and Future Work

We have studied timed-arc Petri nets, for which we have proved that coverability, termination and boundedness are decidable, but timed-locally boundedness is not. We have also seen that *dead tokens* can be effectively detected. All of this is due to the close relationship between TAPN's and transfer nets, because enabling of transitions is closely related to coverability.

Besides place boundedness, there are some other timed properties of TAPN in which we are interested but unfortunately we do not know yet if they are decidable or not. For instance, we could say that N is *non-Zeno* if it has no infinite computation with finite duration, i.e. such that, the number of aging steps in it is finite.

If we take into account the relationship between TAPN's and transfer nets we see that in order to decide this kind of properties we should be able to decide on transfer nets properties like:

Is there any infinite computation along which only finitely many instances of transitions having transfer arcs are executed?

This kind of properties have been proved to be decidable for plain nets. But the technique to do it is to characterize infinite computations by means of looping ones. This cannot be done for transfer nets since the coverability tree does not give enough information to characterize the behaviour of such a net.

So, in order to extend our knowledge about decidability of properties of TAPN's, or equivalently on transfer nets, we have to study which are the logics for which the model checking problem is decidable for this kind of nets. In order to obtain these logics we should take into account that the corresponding decidability proofs should be based on the use of the *backward* technique in [2, 12], instead of using the coverability tree.

Concerning the possibility of extending our results to continuous time, we conjecture that it can be done in the case where the limits of the intervals defining the age of the tokens to be consumed by the firing of a transition are integer (or rational) values, but allowing arbitrary passage of time along computations to be in $\mathbb{R}$. The idea to prove such a result would be to take into account that the age of a token is only defined by the instant at which it is produced. Then, we should prove that any timed step sequence could be represented by some *equivalent* one where aging steps are defined by integer values.

If this conjecture is proved to be correct, we should look for a continuity argument by means of which these results for the rational case could be extended to arbitrary TAPN's whose firing intervals are defined by real values.

We already succeeded on the extension of decidability of coverability for timed Petri nets to the rational and real cases [21], although the ideas there applied cannot be directly translated to the case of timed-arc Petri nets. It could be the case that such a generalization would be not possible in this framework. In fact, in [21] we can also find some result for the discrete time case whose proof cannot be translated to dense time domains. But perhaps, by an alternative proof one could get such a generalization.

References

1. W.M.P. van der Aalst. *Interval Timed Coloured Petri Nets and their Analysis.* LNCS vol. 691, pp. 451-472. 1993.
2. P.A. Abdulla, K. Čerāns, B. Jonsson and T. Yih-Kuen. *General Decidability Theorems for Infinite-State Systems.* Proc. 11th IEEE Symp. Logic in Computer Science (LICS'96), New Brunswick, NJ, USA, July 1996, pages 313-321. 1996.
3. M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte and A. Cumani. *On Petri Nets with Stochastic Timing.* Proc. of the International Workshop on Timed Petri Nets, IEEE Computer Society Press, pp. 80-87. 1985.
4. A. Arnold and M. Latteux. *Vector Addition Systems and Semi-Dyck Language.* Research Report 78, Laboratoire de Calcul de l'Université des Sciences et Techniques de LILLE, December 1976.
5. T. Bolognesi and P. Cremonese. *The Weakness of Some Timed Models for Concurrent Systems.* Technical Report CNUCE C89-29. CNUCE-C.N.R. October 1989.
6. T. Bolognesi, F. Lucidi and S. Trigila. *From Timed Petri Nets to Timed LOTOS.* Proc. of the 10th International IFIP WG6.1 Symposium on Protocol Specification, Testing and Verification, North-Holland. 1990.
7. F.D.J. Bowden. *Modelling Time in Petri Nets.* Proc. Second Australia-Japan Workshop on Stochastic Models. 1996.
8. A. Cerone and A. Maggiolo-Schettini. *Time-based Expressivity of Time Petri Nets for System Specification.* Theoretical Computer Science (216)1-2, pp. 1-53. 1999.

9. C. Dufourd, A. Finkel and Ph. Schnoebelen. *Reset Nets between Decidability and Undecidability*. Proc. 25th. Int. Coll. Automata, Languages, and Programming (ICALP'98), Aalborg, Denmark, July 1998, LNCS vol. 1443, pp:103-115. Springer-Verlag, 1998.
10. C. Dufourd. *Réseaux de Petri avec Reset/Transfert: Décidabilité et Indécidabilité*. Thése de Docteur en Sciences de l'École Normale Supérieure de Cachan. October 1998.
11. C. Dufourd, P. Jančar and Ph. Schnoebelen. *Boundedness of Reset P/T Nets*. Proc. 26th. Int. Coll. Automata, Languages, and Programming (ICALP'99), Prague, Czech Rep., July 1999, LNCS vol. 1644, pp. 301-310. Springer-Verlag, 1999.
12. A. Finkel. *Reduction and Covering of Infinite Reachability Trees*. Information and Computation, 89(2):144-179, 1990.
13. A. Finkel and Ph. Schnoebelen. *Fundamental Structures in Well-Structured Infinite Transition Systems*. Proc. 3rd Latin American Theoretical Informatics Symposium (LATIN'98), Campinas, Brazil, Apr. 1998, LNCS vol. 1380, pp:102-118. Springer-Verlag, 1998.
14. H.-M. Hanisch. *Analysis of Place/Transition Nets with Timed-Arcs and its Application to Batch Process Control*. Application and Theory of Petri Nets, LNCS vol. 691, pp:282-299. Springer-Verlag, 1993.
15. R. Mayr. *Lossy Counter Machines*. Technical report TUM-I9827, TU-München, 1998.
16. P. Merlin. *A Study of the Recoverability of Communication Protocols*. PhD. Thesis, Univ. of California. 1974.
17. J. L. Peterson. *Petri net Theory, and the Modeling of Systems*. Prentice-Hall. 1981.
18. J. Quemada, A. Azcorra, and D. de Frutos. *A Timed Calculus for LOTOS*. Proc. FORTE'89, pp. 245-264. 1989.
19. C. Ramchandani. *Performance Evaluation of Asynchronous Concurrent Systems by Timed Petri Nets*. PhD. Thesis, Massachusetts Institute of Technology, Cambridge. 1973.
20. J. Sifakis. *Use of Petri Nets for Performance Evaluation*. Proc. of the Third International Symposium IFIP W.G.7.3., Measuring, Modelling and Evaluating Computer Systems. Elsevier Science Publishers, pp. 75-93. 1977.
21. V. Valero, D. de Frutos Escrig, and F. Cuartero. *Decidability of the Strict Reachability Problem for TPN's with Rational and Real Durations*. Proc. 5th. International Workshop on Petri Nets and Performance Models, pp. 56-65. 1993.
22. V. Valero, D. de Frutos Escrig, and F. Cuartero. *On Non-Decidability of Reachability for Timed-Arc Petri Nets*. Proc. 8th. International Workshop on Petri Nets and Performance Models, pp. 188-196. 1999.
23. B. Walter. *Timed Petri-Nets for Modelling and Analysing Protocols with Real-Time Characteristics*. Proc. 3rd IFIP Workshop on Protocol Specification, Testing and Verification, North-Holland. 1983.