Timed-Arc Petri Nets with (restricted) Urgency

S. Akshay[1], B. Genest[2], and L. Hélouët[3]

[1] Department of Computer Science and Engineering, IIT Bombay, India.
akshayss@cse.iitb.ac.in
[2] CNRS, IRISA, Rennes, France
blaise.genest@irisa.fr
[3] INRIA, Campus de Beaulieu, Rennes, France
loic.helouet@inria.fr

Abstract. In this paper we study verification for classes of Petri Nets with time. We present the first, up to our knowledge, decidability result on reachability and boundedness for Petri Net variants that combine unbounded places, time, and urgency (the ability to enforce actions to happen within some delay). For this, we introduce the class of Timed-Arc Petri Nets with Urgency, which extends Timed-Arc Petri Nets [28] to allow urgency constraints, a feature from Timed-transition Petri Nets (TPNs) [24]. In order to avoid (straightforward) undecidability, we consider restricted urgency: urgency can be used only on transitions consuming tokens from bounded places.

For Timed-Arc Petri Nets with restricted Urgency, we extend decidability results from Timed-Arc Petri Nets: control-state reachability and boundedness are decidable. Our main result concerns (marking) reachability, which is undecidable for both TPNs (because of unrestricted urgency) [21] and Timed-Arc Petri Nets (because of infinite number of "clocks") [27]. We obtain decidability of reachability for (unbounded) TPNs *with restricted urgency* under a new, yet natural, *timed-arc semantics* presenting them as Timed-Arc Petri Nets with restricted urgency. Decidability of reachability under the original semantics of TPNs is also obtained for a restricted subclass of unbounded nets.

1 Introduction

Petri nets are a simple yet powerful formalism modeling distributed systems. Several extensions have been proposed to enrich them with timing constraints, and allow specification of real-time behaviors. We first discuss expressivity and decidability of two main variants: *Time(d-Transition) Petri Nets (TPNs)* [24] and *Timed-Arc Petri Nets* [28].

TPNs can constrain each transition with a timing interval. To be fireable, a transition needs to have been enabled for an amount of time in the given interval [24]. Also, when a transition has been enabled for the maximal amount of time according to its associated interval, it must fire. This is called *urgency*. Formally, a (continuous, positive valued) *clock* is associated to each transition. Hence the number of such clocks is bounded by the number of transitions. Although the number of clocks is bounded, most problems (reachability, control-state reachability, boundedness) are undecidable for TPNs [21], as two counter machines can easily be encoded. To obtain decidability, one either restricts to bounded TPNs [8], where the number of tokens in any place is bounded, or gives up urgency [26]. In the latter case, the untimed language of a TPN without urgency, also known as its weak-time semantics, is the language of the associated Petri Net without timing constraints, weakening the interest of TPNs.

Timed-Arc Petri Nets, also called Timed Petri Nets, associate a (continuous, positive valued) age to each token [28, 2]. The number of continuous values is thus a priori unbounded. Each arc from a place to a transition can be constrained by a timing interval, meaning that only tokens with age in the interval can be consumed by this transition. Timed-Arc Petri Nets cannot encode urgency [19, 2]. Although the number of token ages is unbounded, the theory of well structured transition systems [18] can be applied because of monotonicity (a token is allowed to stay forever at a place). Thus, control-state reachability (whether a place can be filled with at least one token) and boundedness (whether the number of tokens are always bounded) are decidable for Timed-Arc Petri Nets [2]. However, the (marking) reachability problem (whether a particular marking is ever reachable) is undecidable [27].

In terms of expressivity, Timed-Arc Petri Nets cannot express that a token is produced exactly every unit of time: due to lack of urgency, it is not possible to force a transition to fire. On the other hand, TPNs cannot express that an unbounded number of tokens with slightly different ages are consumed at least two units of time after they have been created .i.e., latency [7]. In this paper, we propose a formalism which can easily specify these two characteristics, see Fig. 1. Namely, we introduce *Timed-Arc Petri nets with Urgency*, extending Timed-Arc Petri Nets with explicit urgency requirements, à la Merlin [24]. This is done by introducing urgency constraints on transitions, forcing transitions to fire if they remains enabled for long enough. Detailed comparisons with related models, e.g., [20], can be found in the *related work* section below.

Unsurprisingly, most problems are undecidable as soon as urgency is used on unbounded places (Proposition 1, and [20]). So far, decidability results have been obtained by either imposing a bound on the number of tokens (e.g.,[16, 8]) or removing urgency completely (e.g.,[23, 26]). Here, we consider classes of Timed-Arc Petri Nets and of TPNs *with restricted Urgency* to obtain decidability. With restricted urgency, transitions consuming tokens exclusively from bounded places can use urgency; other transitions consuming tokens from at least one unbounded place do not have urgency constraints. Using restricted urgency does *not* make the untimed language of a TPN with restricted Urgency the same as the language of the associated untimed Petri Net. Thus, it differs from weak-time semantics [26], where all urgency constraints are ignored.

We present up to our knowledge the first decidability results for a Petri net variant combining time, urgency and unbounded places. First, for the general class of Timed-Arc Petri Nets with restricted Urgency, we obtain decidability of *control-state reachability* (Theorem 1), i.e., whether a given place can ever be filled, and of boundedness. This extends decidability results [2] on Timed-Arc Petri Nets (without urgency). We also prove (Proposition 2) that Timed-Arc Petri Nets with restricted Urgency and Timed-Arc Petri Nets are timed language equivalent, but not timed bisimilar. Our main result concerns the *decidability of (marking) reachability*. Reachability is undecidable for Timed-Arc Petri Nets (without any urgency), due to the presence of unboundedly many "clocks" (timed tokens) [27], and also for TPNs (because of unrestricted urgency) [21]. This leads us to consider TPNs *with restricted urgency*, which inherently use a bounded number of “clocks”. We define an alternative *timed-arc semantics* for TPNs, presenting them as a subclass of Timed-Arc Petri Nets with Urgency. The timed-arc semantics is quite close to the original semantics of TPNs, in the sense that both semantics are timed bisimilar for a subclass of TPNs (Proposition 3). We then obtain our *main* result: reachability is decidable for TPNs with restricted Urgency *under our new timed-arc semantics* (Theorem 3). This allows us to decide reachability for channel systems with specified latency and throughput as illustrated in Figures 1, 2. In comparison, TPNs *with their original semantics* cannot encode channels with specified latency (and throughput), and further, the proof for deciding reachability for TPNs with restricted urgency fails. However, we obtain decidability of reachability for the subclass of TPNs *with restricted constraints, under their original semantics* (Theorem 2). This class forbids specifying upper and lower bounds on transitions leaving unbounded places.

Related work. In [20], Timed-Arc Petri Nets were extended with urgent transitions and place invariants. This model is close to our model of Timed-Arc Petri Nets with Urgency, with the following differences. While place invariants [20] can create deadlocks (i.e., time cannot elapse if increasing the age of tokens violates an invariant), in our model a timed or discrete move is always allowed. Further, urgent transitions of [20] must fire as soon as they are enabled, which corresponds to the special case of having urgency 0 in our model. Urgent transitions have also been modeled as Black transitions in generalized stochastic nets [5], but these cannot model latency constraints.

Bounded TPNs are language equivalent to timed automata [4] but not (weakly) timed bisimilar [7]. In order to make them timed bisimilar with timed automata, [9] introduced priorities. Timed-Arc Petri Nets with (restricted) urgency are not included in this class: for instance, they can impose a latency of 2 time units to an unbounded number of tokens, which cannot be done with TPNs with priorities that use only a bounded number of clocks (one per transition). Further, reachability is undecidable for (unbounded) priorities TPNs as this class subsumes the already undecidable class of TPNs, while we prove decidability of reachability for classes of unbounded TPNs.

Many more variants of Petri nets with time (and even urgency) have been proposed, but we do not mention all of them here. Our focus is to address decidability issues, and as far as we know, in all earlier results and in particular in [20, 23, 26, 5, 9], decidability is ensured only when urgency is completely disallowed, or places are all bounded. The major difference of the approach proposed in this paper is that we obtain decidability for systems with (restricted) urgency *and* unbounded places. Further, decidability of the reachability problem was known only for bounded timed systems (timed automata, bounded TPNs, bounded Timed-Arc Petri Nets), or for unbounded systems where time is not relevant (TPNs without urgency [26]), while we show in this paper that reachability can be decided for systems with both unbounded places and restricted urgency!

Finally, our framework also allows to obtain decidability for timed systems of finite state machines communicating through bag channels [13, 14]. However, unlike [13, 14], where urgency is completely disallowed, our model captures finite state processes with unrestricted urgency, communicating through unbounded bag channels that have restricted urgency. A minimal latency (i.e., delay) and a maximal throughput (i.e., rate) of a channel can be specified as shown in the channel model of Figure 1, as well as time-outs on messages.

2 Timed-Arc Petri nets with Urgency

We will denote by $\mathbb{Q}_{\geq 0}$ the set of positive rational numbers, and by $\mathcal{I}(\mathbb{Q}_{\geq 0})$ the set of intervals over $\mathbb{Q}_{\geq 0} \cup \infty$. These intervals can be of the form (a, b), $(a, b]$, $[a, b)$, or $[a, b]$. We will denote by $\mathbb{M}_{\mathbb{R}}$ the set of *multisets* of positive real numbers. For two multisets A and B, we denote by $A \sqcup B$ the disjoint union of A and B, i.e., the multiset that gathers elements of multisets A and B without deleting identical elements. Similarly, we define $A \setminus B$ as the operation that removes from A exactly one occurrence of each element of B (if it exists).

We introduce our main model, Timed-Arc Petri Nets with urgency constraints. The model is based on a semantics using *timed markings* $m : P \to \mathbb{M}_{\mathbb{R}}$ which associate to each place a multiset describing the ages of all the tokens in this place.

Definition 1 *A Timed-Arc Petri Net with Urgency, denoted* Timed-Arc PNU*, is a tuple* $\mathcal{N} = (P, T, {}^\bullet(), ()^\bullet, m_0, \gamma, U)$ *where*

- *P is a set of places, T is a set of transitions, m_0 is the initial timed marking,*
- *${}^\bullet() : T \to P$ and $()^\bullet : T \to P$ are respectively, the backward and forward flow relations indicating tokens consumed/produced by each transition.*
- *$\gamma : P \times T \to \mathcal{I}(\mathbb{Q}_{\geq 0})$ is a set of* token-age constraints on arcs *and*
- *$U : T \to \mathbb{Q}_{\geq 0} \cup \infty$ is a set of* urgency constraints on transitions*.*

For a given arc constraint $\gamma(p, t) = [\alpha(p, t), \beta(p, t)]$ we will call $\alpha(p, t)$ the lower bound and $\beta(p, t)$ the upper bound of $\gamma(p, t)$. Such constraints mean that the transition t is enabled when for each place p of its preset ${}^\bullet t$, there is a token in p of age in $\gamma(p, t)$, i.e., between $\alpha(p, t)$ and $\beta(p, t)$. The urgency constraint $U(t)$ means that a transition must fire if t has been enabled (by its preset of tokens) for $U(t)$ units of time. A Timed-Arc Petri Net [2] can be seen as a Timed-Arc PNU with $U(t) = \infty$ for all $t \in T$. Note that we do not label transitions, hence each transition can be seen as labeled by its unique name.

As an example, consider the Timed-Arc PNU $\mathcal{N}_1$ of Figure 1. Places are represented by circles, transitions by narrow rectangles, and flow relations by arcs between places

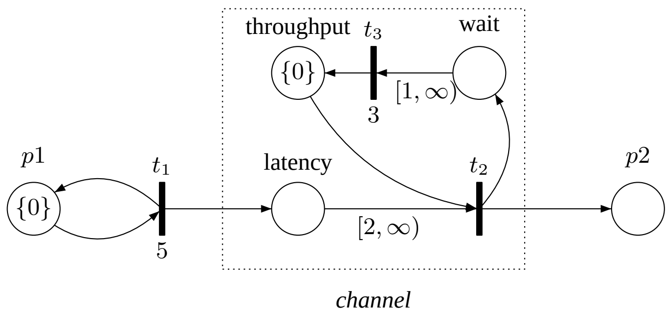

Fig. 1. Timed-Arc Petri Net with Urgency $\mathcal{N}_1$.

and transitions. Urgency of a transtion is represented below the transition (in the example, transition t_3 has urgency 3). Arc constraints γ are represented as intervals below arcs. When unspecified, an arc constraint is set to $[0, \infty)$ and an urgency constraint to ∞ (e.g. $U(t_2) = \infty$). Intuitively, Figure 1 depicts a process $p1$ that sends an unbounded number of messages to a process $p2$ through a channel. A message is sent at least every five time units (t.u.) because of the urgency constraint on t_1. Latency (or delay) for each message is at least 2 t.u. before being received, and the maximal throughput (or rate) of the channel is between 1 message every t.u. and 1 message every 4 t.u. Changing constraint $[2, \infty)$ into $[2, 100]$ models message loss, i.e., messages not received after 100 t.u. are considered lost.

Formal Semantics of Timed-Arc PNU: We now define the semantics of a Timed-Arc PNU $\mathcal{N} = (P, T, {}^\bullet(), ()^\bullet, m_0, \gamma, U)$ in terms of timed markings and discrete and timed moves. For a given place p and timed marking m, we will let age_p denote real values from $m(p)$ depicting the age of one token in place p. Note that as $m(p)$ is a multiset, two tokens in a place p may have identical ages.

We say that a transition t is *enabled* from a timed marking m if, for each $p \in {}^\bullet t$, there exists $age_p \in m(p)$ such that $age_p \in \gamma(p, t)$. A transition t is said to be *urgent* from a timed marking m if $\forall p \in {}^\bullet t, \exists age_p \in m(p)$ such that $\alpha(p, t) + U(t) \leq age_p \leq \beta(p, t)$, i.e., if the preset of t has tokens at least $U(t)$ time units older than required by $\gamma(p, t)$. Let t be an urgent transition from m. This implies that t is enabled. Further, as formally defined below, presence of urgent transitions disallows time from elapsing. Thus, there will exist a place $p \in {}^\bullet t$ such that the oldest token $age_p \in m(p)$ with $age_p \leq \beta(p, t)$ will satisfy $age_p = \alpha(p, t) + U(t)$. An urgent transition t will force occurrence of a discrete move, but not necessarily of this transition t as several transitions can be enabled (or even urgent) at the same time. Formally, the semantics of Timed-Arc PNU is decomposed into timed moves and discrete moves.

Timed moves symbolize elapsing of δ time units from a timed marking in the following way: for a given timed marking m, we denote by $m + \delta$ the timed marking obtained by adding δ to the age of every token: if $m(p) = \{age_1, \ldots, age_k\}$, then $(m + \delta)(p) = \{age_1 + \delta, \ldots, age_k + \delta\}$. A *timed move* of $\delta > 0$ time units is allowed from m if for every $0 \leq \delta' < \delta$, the timed marking $m + \delta'$ has no urgent transition, and we denote $m \xrightarrow{\delta} m + \delta$ such timed moves.

Discrete moves represent firings of transitions from a marking m. One can fire transition t from marking m and reach marking m', denoted $m \xrightarrow{t} m'$ iff t is enabled and for each place p, we have $m'(p) = (m(p) \setminus S_p) \sqcup S'_p$, where

- $S_p = \{age_p\}$ where $age_p \in m(p) \cap \gamma(p, t)$ if $p \in {}^\bullet t$, and $S_p = \emptyset$ otherwise.
- $S'_p = \{0\}$ if $p \in t^\bullet$, and $S'_p = \emptyset$ otherwise.

In other words, when a discrete transition t fires, time does not elapse, a token with age age_p that satisfies the arc-constraint $\gamma(p, t)$ is consumed in each preset place p of t (tokens that are consumed by firing a transition t are tokens which enabled t), and a new token with age 0 is produced in each post place. Note that several ages from the same place can enable the transition, and any of them can be consumed. Hence discrete moves are not a priori deterministic. Further, note that, as in TPNs, timed moves are not allowed when a transition is urgent. This ensures that a discrete move must happen when a transition t is urgent. However another transition t' (which need not be urgent) may be fired, after which, it is possible that no transitions (including t) are urgent anymore (because the corresponding tokens have been consumed), and then time can elapse. Else, urgency remains and a discrete move still needs to occur.

With the above semantics, a Timed-Arc PNU $\mathcal{N}$ defines a timed transition system $[\![\mathcal{N}]\!]$ whose states are timed markings and transitions are discrete and timed moves. We will denote by $\mathsf{Reach}(\mathcal{N})$ the set of reachable timed markings of $\mathcal{N}$ (starting from m_0). An *(untimed) marking* is a function from P to $\mathbb{N}$. For a timed marking m, we will denote by $m^\sharp : P \to \mathbb{N}$ the untimed marking that associates to every place $p \in P$ the number of tokens in $m(p)$. We will say that a place $p \in P$ of a Timed-Arc PNU is *bounded* if there exists an integer K such that for every timed marking $m \in \mathsf{Reach}(\mathcal{N})$, $m^\sharp(p) \leq K$. We will say that $\mathcal{N}$ is *bounded* iff all its places are bounded.

Continuing with our channel example of Fig. 1, the initial timed marking m_0 has $m_0(p1) = \{0\}$ and $m_0(throughput) = \{0\}$, i.e., one token with age 0 in place $p1$ and one token with age 0 in place $throughput$, and other places are empty. Transitions t_1 needs to fire between date 0 and date 5 because $U(t_1) = 5$. Assume that there is a timed move of duration 1.2 and then t_1 is fired. These two consecutive moves result in the timed marking m_1, with $m_1(p1) = \{0\}$, $m_1(throughput) = \{1.2\}$ and $m_1(latency) = \{0\}$, and where all other places are empty. From m_1, an elapse (timed move) of 2 t.u results in timed marking m_2, with $m_2(p1) = \{2\}$, $m_2(throughput) = \{3.2\}$ and $m_2(latency) = \{2\}$. There is one token in place $latency$, of age 2. Now, transition t_2 can fire from m_2 resulting in timed marking m_3 with $m_3(p1) = \{2\}$, $m_3(latency) = m_3(throughput) = \emptyset$, $m_3(wait) = \{0\}$ and $m_3(p2) = \{0\}$. More generally, if a token is produced by t_1 at date d, it cannot be used to fire t_2 before date $d+2$ (latency 2 to be received in $p2$). Also, if t_2 is fired at date d, then it cannot fire again before date $d+1$ at the earliest because of transition t_3 (max throughput 1 message per t.u.). Thus, the earliest arrival dates of tokens at $p2$ are $2, 3, 4, 5, \ldots$

3 Undecidable and Decidable Problems for Timed-Arc PNU

In this paper we will tackle the decidability of the following problems:

- Reachability: given a Timed-Arc PNU $\mathcal{N}$, given an (untimed) marking m, does there exists a timed marking $m' \in \mathsf{Reach}(\mathcal{N})$ with $m'^\sharp = m$?
- Control State reachability (also called place-reachability) : given a Timed-Arc PNU $\mathcal{N}$ and a place p, does there exist $m \in \mathsf{Reach}(\mathcal{N})$ with $m^\sharp(p) \geq 1$?
- Boundedness : given a Timed-Arc PNU $\mathcal{N}$, does there exist K such that for all $m \in \mathsf{Reach}(\mathcal{N})$, we have $m^\sharp(p) \leq K$ for all places p?

Reachability is undecidable for Timed-Arc PNU as it is undecidable for Timed-Arc Petri nets [27]. Because of urgency, the other problems defined above are undecidable for Timed-Arc PNU (see [20] for the proof for Timed-Arc Petri Nets with age invariants), following the proofs of undecidability for TPNs [21]:

Proposition 1. *Control State reachability, Reachability and Boundedness are undecidable for Timed-Arc PNU.*

To obtain decidability, two main approaches have been explored. The first involves dropping all urgency requirements. For Timed-Arc PNU, doing so we get back Timed-Arc Petri Nets and their decidability results. For TPNs, this corresponds to the weak semantics [26], under which the reachable (untimed) markings are the markings reachable by the associated (untimed) Petri net. The second approach considers only bounded nets, for which a DBM-based forward reachability algorithm can be used [16] (see also bounded TPNs [8]). Our goal in this paper is to define restrictions for Timed-Arc PNU that ensure decidability for models combining urgency and unbounded nets. Indeed, this would allow us to verify networks of timed systems communicating via unbounded channels with specified latency and throughput (as shown in Fig. 1).

3.1 Restricted Urgency

We start by defining the restriction of a Timed-Arc PNU to a subset of places. Let $\mathcal{N} = (P, T, {}^{\bullet}(), ()^{\bullet}, m_0, \gamma, U)$ be a Timed-Arc PNU, and let $P_b \subseteq P$ be a subset of places. The restriction of $\mathcal{N}$ to P_b is the Timed-Arc PNU $\mathcal{N}_{P_b} = (P_b, T, {}^{\star}()_{P_b}, ()^{\star}{}_{P_b}, m'_{0P_b}, \gamma_{P_b}, U)$, where ${}^{\star}()_{P_b}, ()^{\star}{}_{P_b}, m'_{0P_b}, \gamma_{P_b}$ are respectively restriction of ${}^{\bullet}(), ()^{\bullet}, m_0, \gamma$ to $P_b \times T, T \times P_b$ and P_b. When projecting away places $p \in P \setminus P_b$ such that every transition with p in its preset has no urgency, we have:

Lemma 1 *Let P_b be such that for all $p \in P \setminus P_b$, for all t with $p \in {}^{\bullet}(t)$, $U(t) = \infty$. Then every run of $\mathcal{N}$ is a run of $\mathcal{N}_{P_b}$.*

The proof simply projects the sequence of markings over P_b. Note that the converse is not true in general: a run of $\mathcal{N}_{P_b}$ needs not be a run of $\mathcal{N}$. We can now define our decidable subclass of Timed-Arc PNU. It is mainly based on the notion of *restricted urgency*, which intuitively means that urgency can be enforced only on the bounded part of the system.

Definition 2 *A* Timed-Arc Petri Net with restricted Urgency *(denoted* Timed-Arc PNrU*) is a triple $(\mathcal{N}, P_u, P_b)$, where $\mathcal{N} = (P, T, {}^{\bullet}(), ()^{\bullet}, m_0, \gamma, U)$ is a Timed-Arc PNU, and $P_u \sqcup P_b = P$ is a partition of places of $\mathcal{N}$ such that:*

- *For each transition $t \in T$ with ${}^{\bullet}t \cap P_u \neq \emptyset$, we have $U(t) = \infty$, and*
- *The restriction $\mathcal{N}_{P_b}$ of $\mathcal{N}$ to places of P_b is bounded.*

Lemma 2 *Let $(\mathcal{N}, P_u, P_b)$ be a Timed-Arc PNrU. Then every place $p \in P_b$ is bounded in the original Timed-Arc PNU $\mathcal{N}$ as well.*

Note that the constraint on place contents in Timed-Arc PNrU applies only to P_b. Though we will often refer to places in P_u as "unbounded places", it should be clear that this only means the contents of these places can be unbounded, not that they must be. Intuitively, in a Timed-Arc PNrU $(\mathcal{N}, P_u, P_b)$, urgency cannot be used for transitions consuming tokens from unbounded places. For instance, $\mathcal{N}_1$ in Figure 1 is a Timed-Arc PNrU (only places $latency$ and $p2$ are unbounded).

Urgency allows us to perform zero test-like operations as shown in the undecidability proof of Proposition 1 (see appendix for details). Forbidding zero testing unbounded places seems reasonable to obtain decidability. Recall that proofs showing decidability for TPNs either bound the net [8], or forbid urgency (for instance [26]), which implies that timing constraints do not have any impact on the untimed language. The approach proposed in this paper is less restrictive, as the net is unbounded and urgency is allowed in parts of the net. Timed-Arc PNrUs allow to model complex networks of timed systems, where each component is a finite state timed system that can use urgency without restriction. Communications between components are done through *unbounded* bag channels [13, 14] that can specify throughput and latency (see Fig. 1).

We can extend the proofs of [1] from Timed-Arc Petri Nets to Timed-Arc PNrUs. The proof uses well-quasi order [2] and the theory of well structured transition systems [18]. First, we define a region abstraction for markings of Timed-Arc PNrU. This abstraction is a combination of regions of a finite timed automaton representing the behavior of the net on its bounded part, and regions representing symbolically the markings of the unbounded places of the net. This set of regions is equipped with a comparison relation $\preceq$ that requires equality on the region bounded part, and comparable contents on the unbounded part. This relation is compatible with markings comparison and is a well-quasi order. We can then define a successor relation among regions that is an abstract representation of moves of a Timed-Arc PNrU. Regions equipped with their ordering and the successor relation form a well-structured transition system. Details can be found in the appendix. This shows:

Theorem 1. *Control-State reachability and Boundedness are decidable for Timed-Arc PNrU. However, reachability is undecidable for Timed-Arc PNrU.*

The undecidability comes directly from the undecidability of reachability for Timed-Arc Petri nets [1]. Timed-Arc PNrU strictly extend Timed-Arc Petri nets in terms of (timed) bisimulation, since Timed-Arc Petri Nets do not have any urgency, while Timed-Arc PNrU do. However, they are (timed language) equivalent.

Proposition 2. *Timed-Arc PNrUs are timed language equivalent to Timed-Arc Petri Nets.*

This proof is obtained by a construction that separates the bounded and unbounded part, processes the bounded part to remove urgency, and then reinserts the unbounded part carefully to keep the same set of timed behaviors. This gives an alternative proof (see appendix) of decidability of control-state reachability for Timed-Arc PNrU.

4 Decidability of the Reachability Problem

In this section we tackle the decidability of the reachability problem. On one hand, reachability is undecidable for Timed-Arc Petri Nets [27], and thus for Timed-Arc PN(r)Us, because an unbounded number of clocks can be encoded, one for each token. On the other hand, (unbounded) Timed-transition Petri Nets (TPNs) [24] only use a bounded number of clocks (one per transition). Nevertheless, (unrestricted) urgency makes reachability undecidable for TPNs [21]. To obtain decidability of reachability, we thus consider classes of TPNs with restricted urgency.

4.1 Timed-Transition Petri Nets (TPNs)

Timed-transition Petri Nets (TPNs for short), also called Time Petri Nets, introduced in [24], associate time intervals to transitions of a Petri net. Formally, a TPN $\mathcal{N}$ is a tuple $(P, T, {}^\bullet(), ()^\bullet, m_0, I)$ where P is a finite set of *places*, T is a finite set of *transitions*, ${}^\bullet(), ()^\bullet : P \to T$ are the *backward* and *forward* flow relations respectively, $m_0 \in \mathbb{N}^P$ is the *initial* (untimed) marking, and $I : T \mapsto \mathcal{I}(\mathbb{Q}_{\geq 0})$ maps each transition to a *firing interval*. We denote by $A(t)$ (resp. $B(t)$) the lower bound (resp. the upper bound) of interval $I(t)$. A *configuration* of a TPN is a pair (m, ν), where m is an untimed *marking* (recall that in untimed markings, $m(p)$ is the number of tokens in p), and $\nu : T \to \mathbb{R}_{\geq 0}$ associates a real value to each transition. A transition t is *enabled* in a marking m if $m \geq {}^\bullet t$. We denote by *En*(m) the set of enabled transitions in m. The valuation ν associates to each enabled transition $t \in$ *En*(m) the amount of time that has elapsed since this transition was last newly enabled. An enabled transition t is *urgent* if $\nu(t) \geq B(t)$, with $B(t)$ the upper bound of $I(t)$.

We first recall the classical (intermediate marking) semantics [8, 6] for TPNs defined using timed and discrete moves between configurations. A *timed move* consists of letting time elapse in a configuration. For (m, ν), $\nu+\delta$ is defined by $\nu+\delta(t) = \nu(t)+\delta$, for all $t \in$ *En*(m). A timed move from (m, ν) to $(m, \nu+\delta)$, denoted $(m, \nu) \xrightarrow{\delta} (m, \nu+\delta)$, is allowed if for every $0 \leq \delta' < \delta$, the configuration $(m, \nu + \delta')$ has no urgent transition. A *discrete move* consists of firing an enabled transition t that has been enabled for a duration that fulfills the time constraint attached to t. We have $(m, \nu) \xrightarrow{t} (m', \nu')$ if $t \in$ *En*(m), $\nu(t) \in I(t)$ and $m' = m - {}^\bullet t + t^\bullet$, for ν' defined below. We call intermediate marking the marking $m - {}^\bullet t$ which is obtained after t consumes tokens from its preset but did not create new ones yet. We will say that a transition $t' \in$ *En*(m') is *newly enabled* by firing of t if either $t' = t$, or $t' \notin$ *En*$(m - {}^\bullet t)$, i.e. is not enabled in the intermediate marking $m - {}^\bullet t$. Now, we define $\nu'(tt) = 0$ if tt is newly enabled, and $\nu'(tt) = \nu(tt)$ for all $tt \in$ *En*(m) but not newly enabled. That is, for a transition t both consuming and producing a token in p having a single token, a transition t' with $p \in {}^\bullet t'$ is disabled then newly enabled when t is fired.

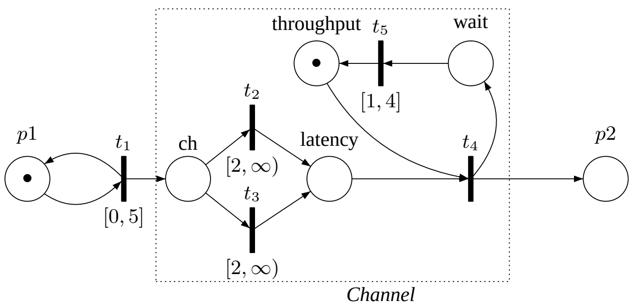

Fig. 2. A TPN $\mathcal{N}_2$.

This classical semantics of TPN is somewhat similar to that of Timed-Arc PNU, but is based on configurations instead of timed markings. The only continuous values kept in the configuration of a TPN are in ν. Hence, only $|T|$ "clock" values are kept, and configurations cannot keep track of the exact time elapsed since their creation for arbitrary number of tokens. Thus, for instance, no TPN can encode that an unbounded number of tokens is processed at least 2 units of time after each of them is created (i.e., the *latency* is at least 2), which can be easily done using Timed-Arc PNUs (and Timed-Arc PNs [28]).

Further, it is not simple to model a channel with specified latency and throughput with TPNs. Let us consider the Timed-Arc PNU $\mathcal{N}_1$ of Figure 1, which has bounded throughput (at most 1 message per t.u.). The earliest arrival dates for tokens in $p2$ are $2, 3, 4, 5, 6, 7$... These earliest arrival dates of tokens at $p2$ can be simulated by a TPN, for instance the TPN $\mathcal{N}_2$ in Figure 2 firing alternatively transitions t_2 and t_3. The two clocks attached to transitions t_2, t_3 with firing intervals $[2, \infty)$, and the lower bound of 1 attached to transition t_5 are sufficient to obtain these arrival dates. More generally, to obtain arrival dates $x, x + \delta, \ldots, x + n\delta, \ldots$, one would use $\lceil \frac{x}{\delta} \rceil$ (clocks attached to) transitions conveying tokens from place ch to place $latency$. However, this approach does not faithfully model latency of the unbounded channel. In $\mathcal{N}_2$, if two tokens are created by $p1$ at dates 0 and 1.9, then at date 2, both transitions t_2, t_3 can fire, as both have been enabled for 2 t.u. This results in sending one token that remained for 2 t.u. and one token that remained for only 0.1 t.u in place ch to place $latency$. Then these two tokens can be sent from place $latency$ to $p2$ at dates 2 and 3. Thus, there is a run of $\mathcal{N}_2$ in which a token created at date 1.9 arrives in $p2$ at date 3. This execution of $\mathcal{N}_2$ violates the latency requirement of 2 t.u. that was captured in $\mathcal{N}_1$. In general, TPNs cannot remember ages of tokens and transitions that fire need not use the tokens that enabled them.

4.2 TPNs with restricted constraints

Reachability is undecidable for general TPNs because of unrestricted urgency [21]. We now introduce two natural restrictions to allow decidability.

Definition 3 *Let $\mathcal{N} = (P, T, {}^\bullet(), ()^\bullet, m_0, I)$ be a TPN and $P = P_u \sqcup P_b$ be a partition of its places such that the restriction of $\mathcal{N}$ to places of P_b is bounded. Then,*

- *$\mathcal{N}$ is called a* TPN with restricted urgency *if for each transition $t \in T$ with ${}^\bullet t \cap P_u \neq \emptyset$, we have $I(t) = [c, \infty)$ with $c \in \mathbb{Q}_{\geq 0}$.*
- *$\mathcal{N}$ is called a* TPN with restricted constraints *if for each transition $t \in T$ with ${}^\bullet t \cap P_u \neq \emptyset$, we have $I(t) = [0, \infty)$.*

The class of TPNs with restricted constraints is strictly contained in the class of TPNs with restricted urgency. As an example, the TPN $\mathcal{N}_2$ on Fig. 2 is a TPN with restricted urgency but not a TPN with restricted constraints, since there is an arc from the unbounded place Ch to transition t_2 with constraints $[2, \infty)$, that is a constraint with non-trivial lower bound and no upper (urgency) bound.

It is not decidable whether a TPN (resp. a Timed-Arc PNU) is with restricted urgency (or constraints), because it is not decidable whether a TPN (resp. a Timed-Arc

PNU) is bounded. It is easy to come up with decidable subclasses by imposing structural boundedness on the underlying Petri net, i.e., by assuming that the Petri Net (without timing constraints) associated with $\mathcal{N}$ restricted to P_b is bounded. These subclasses are decidable as boundedness is decidable for (untimed) Petri nets. We prove our theorems for the more general classes, but our results obviously apply to the classes with structural boundedness too.

Theorem 2. *Reachability, boundedness and control-state reachability are decidable for TPNs with restricted constraints.*

Theorem 2 is obtained by a simple adaptation of Prop. 5 from Section 5.

We now introduce a new *timed-arc semantics* presenting TPNs as Timed-arc PNU. This semantics is bisimilar to the original semantics for simple systems. Compared with the original semantics, we can model channels with specified throughput and latency (as shown below). We can also prove that reachability is decidable for the more general class of TPNs with restricted urgency. The reason is that unlike in the *original semantics*, we can add components to a TPN without altering the *timed-arc semantics*. This compositionality property of timed-arcs (and non compositionality for timed-transitions) was noticed before in the context of 1-safe TPNs [12].

4.3 A new Timed-Arc Semantics for TPNs

The core idea is that the timed-arc semantics takes into account the age of tokens in input places. Formally, we define Timed($\mathcal{N}$), the Timed-Arc PNU associated with the TPN $\mathcal{N} = (P, T, {}^\bullet(), ()^\bullet, m_0, I)$. Intuitively, Timed($\mathcal{N}$) preserves all places and transitions of $\mathcal{N}$, adds one place p_t per transition t, adds p_t to the pre and post flow of t, and adapts the timing constraints. Fig.3 displays a TPN $\mathcal{N}_3$ on the left and *Timed*($\mathcal{N}_3$) on the right. We define Timed($\mathcal{N}$) $= (P', T, {}^\star(), ()^\star, m_0', \gamma, U)$ where:

- $P' = P \cup P_T$ with $P_T = \{p_t \mid t \in T\}$.
- ${}^\star(), ()^\star$ extend respectively ${}^\bullet(), ()^\bullet$ in the following way: $p \in {}^\star t$ iff $p = p_t$ or $p \in {}^\bullet t$ and $p \in t^\star$ iff $p = p_t$ or $p \in t^\bullet$.
- For all t, for $I(t) = [A(t), B(t)]$, we let $U(t) = B(t) - A(t)$ and for all $p \in {}^\star t$, we set $\gamma(p, t) = [A(t), +\infty)$ (for $I(t) = (A(t), B(t)]$ we let $\gamma(p, t) = (A(t), +\infty)$),
- We let $m_0'(p) = 0^{m_0(p)}$ for all $p \in P$ and $m_0'(p_t) = \{0\}$ for all transitions t.

TPN $\mathcal{N}_2$ under the timed-arc semantics, i.e., Timed($\mathcal{N}_2$), represents the channel with latency 2 and maximal throughput of 1 message per time unit, which is also modeled by the Timed-arc PNU of Figure 1. Indeed, a token can be sent from place ch to

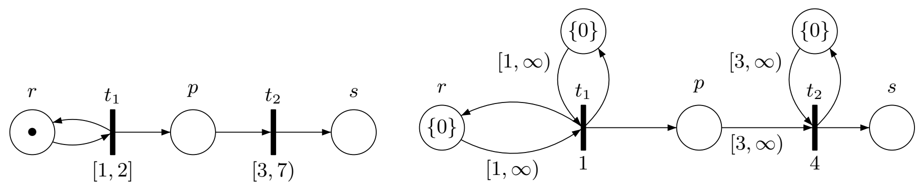

Fig. 3. A TPN $\mathcal{N}_3$ (left) which is timed bisimilar to *Timed*($\mathcal{N}_3$) (right).

place $latency$ by either transition only when it is at least 2 time units old, preserving the latency requirement. The new timed-arc semantics is quite close to the original/classical semantics though. Indeed, we show below that Timed($\mathcal{N}$) and $\mathcal{N}$ are timed bisimilar for a subclass of (possibly unbounded) TPNs, including $\mathcal{N}_3$ from Fig. 3 but not $\mathcal{N}_2$.

Proposition 3. *Let $\mathcal{N}$ be a TPN such that for all reachable configurations, for all places p, either p has at most 1 token, or only one transition t is enabled with $p \in {}^\bullet t$. Then Timed($\mathcal{N}$) and $\mathcal{N}$ are timed bisimilar.*

Notice that if $\mathcal{N}$ is a TPN with restricted Urgency, then Timed($\mathcal{N}$) is a Timed-Arc PNrU, ensuring that boundedness and control-state reachability are decidable. We can now state our main result, namely Theorem 3: reachability is decidable for TPNs with restricted urgency *under timed-arc semantics* (e.g. *Timed*($\mathcal{N}_2$) from Figure 2 is in that class). TPNs with restricted urgency *under timed-arc semantics* can model networks of (finite-state) *timed* systems with unrestricted urgency, communicating through bag channels [13, 14], specifying maximal throughput and minimal latency, assuming that the throughput is not infinite. Indeed, it suffices to modify the TPN in Figure 2 with $\lceil \frac{x}{\delta} \rceil$ transitions from ch to $latency$ in order to model a channel with latency at least x and throughput at most δ messages per unit of time.

Theorem 3. *Let $\mathcal{N}$ be a TPN with restricted Urgency. Then the reachability problem for Timed($\mathcal{N}$) is decidable.*

We show that although the *timed-arc semantics* of TPNs "formally" uses an unbounded number of clocks, a complex reduction allows to consider only a bounded number of clocks. This step is crucial in the proof of Theorem 3, and we believe that this technique can be generalized and re-used for other problems in related contexts. The rest of the paper is devoted to the proof of Theorem 3.

5 Proof of Theorem 3

The proof of Theorem 3 is done in two steps and builds on several lemmas. Let $(\mathcal{N}, P, Q)$, with $\mathcal{N} = (P \cup Q, T, {}^\bullet(), ()^\bullet, m_0, I)$ be a TPN with restricted Urgency, P (resp. Q) the set of bounded (resp. unbounded) places. In this section, we show how to check if a given (untimed) marking is reachable in Timed($\mathcal{N}$). The intuitive idea is that, under restricted urgency, a transition t which has an unbounded place from Q in its preset, has no urgency/upper constraint. Hence to fire t, it suffices to check the lower bound constraint, i.e., to check that some tokens (among an unbounded number) in its pre-places are old enough. Now, the crucial point is that to check this lower-bound, we need the ages of only a bounded number of tokens, as there are a finite number of transitions, and for each transition t, its associated "clock" p_t is reset after it is fired.

Formally, the proof (of Theorem 3) is in two steps: we first convert the TPN with restricted urgency $\mathcal{N}$ to a TPN with restricted constraints $\mathcal{N}'$ such that Timed($\mathcal{N}$) and Timed($\mathcal{N}'$) have the same set of reachable markings. In the second step, we obtain a Petri Net that is bisimilar to Timed($\mathcal{N}'$), which implies the decidability of reachability.

Step 1: Construction of the TPN with restricted constraints $\mathcal{N}'$. In order to obtain a TPN with restricted constraints $\mathcal{N}'$ from $\mathcal{N}$, we will keep (an overapproximation of)

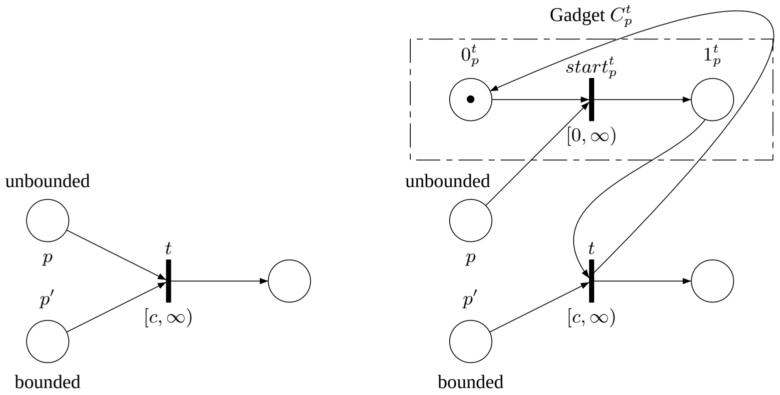

Fig. 4. Step 1 of proof: converting (part of) TPN with restricted urgency $\mathcal{N}$ to restricted constraints $\mathcal{N}'$

ages for a bounded number of tokens from each unbounded place $p \in Q$. For that, we will use $|T| \times |Q|$ gadgets $(C_p^t)_{t\in T, p\in Q}$.

Gadget C_p^t, associated with place $p \in Q$ and transition $t \in T$ (with $p \in {}^\bullet t$), is a TPN with restricted constraints. Each gadget is similar: it has 2 places, 0_p^t and 1_p^t, and in the initial marking the token is at 0_p^t. There is an associated transition $start_p^t$: we have ${}^\star start_p^t = \{p, 0_p^t\}$ and $start_p^{t\,\star} = \{1_p^t\}$, with the timing constraint $I'(start_p^t) = [0,\infty)$. That is, $\mathcal{N}'$ will non-deterministically guess the transition that will fire. The gadget for a fixed transition t and place p is shown in Figure 4. Every transition t reading from an (unbounded) place $p \in Q$ is transformed to read from (bounded) place 1_p^t of gadget C_p^t instead. That is, if a transition t reads from unbounded places $\{p_1, \cdots, p_k\} = {}^\bullet t \cap Q$, then we have ${}^\star t = {}^\bullet t \setminus \{p_1, \cdots, p_k\} \cup \{1_{p_j}^t \mid j \leq k\}$ and $t^\star = t^\bullet \cup \{0_{p_j}^t \mid j \leq k\}$. The timing constraint is left unchanged: $I'(t) = I(t)$. We obtain $\mathcal{N}' = (P', T', {}^\star(), ()^\star, m_0', I')$:

- $P' = P \cup Q \cup \{0_p^t, 1_p^t \mid p \in P, t \in T\}$,
- $T' = T \cup \{start_p^t \mid t \in T, p \in Q\}$,
- ${}^\star(), ()^\star, I'$ as defined above, and
- $m_0'(p) = m_0(p)$ for $p \in P$, and $m_0'(0_p^t) = 1$, $m_0'(1_p^t) = 0$ for all t, p.

It is clear that $\mathcal{N}'$ is a TPN with restricted constraints, with the same set $Q' = Q$ of unbounded places as for all t' with ${}^\star t' \cap Q' \neq \emptyset$, we have $t' = start_p^t$ for some $t \in T$, and thus $I(t') = [0,\infty)$.

The idea of the gadget is the following. Let $m \in \mathsf{Reach}(\text{Timed}(\mathcal{N}))$, t be a transition with $I(t) = [a,\infty)$ and $p \in Q \cap {}^\bullet t$ be an unbounded place. $m(p_t)$ is the time elapsed since the last firing of t. For firing t, we need to have both $m(p_t) \geq a$ and $age_p \geq a$, i.e., we need $min(m(p_t), age_p) \geq a$. In other words, keeping $min(m(p_t), age_p)$ instead of age_p is sufficient to know whether t is enabled. This is implemented in Timed($\mathcal{N}'$), as there can be only one token in $1_p^{t'}$, and its age $m'(1_p^{t'})$ is never older than $m'(p_t)$, as $start_p^t$ can happen only after t fired (0_p^t filled when t fired).

We now show that Timed($\mathcal{N}'$) preserves the set of reachable untimed markings of Timed($\mathcal{N}$). We start by defining a map f from untimed markings of Timed($\mathcal{N}'$) to untimed markings of Timed($\mathcal{N}$). Recall that for a timed marking m, $m^\sharp$ refers to the untimed marking obtained by counting the number of tokens in each place. Let $m' \in \mathsf{Reach}(\text{Timed}(\mathcal{N}'))$. For each place $p \in P \cup Q$ of Timed($\mathcal{N}$), we define:

$$f({m'}^\sharp)(p) = \begin{cases} {m'}^\sharp(p) + \sum_{t \in T} {m'}^\sharp(1_p^t) & \text{if } p \in Q \\ {m'}^\sharp(p) & \text{otherwise} \end{cases}$$

Now we can show that Timed($\mathcal{N}'$) can only produce (untimed) markings of Timed($\mathcal{N}$):

Lemma 3 *Let m' be a timed marking in Reach(Timed($\mathcal{N}'$)). Then there exists a timed marking $m \in \mathsf{Reach}$(Timed($\mathcal{N}$)) with $f({m'}^\sharp) = m^\sharp$.*

Next, we show that every untimed marking of Timed($\mathcal{N}$) can be simulated in Timed($\mathcal{N}'$):

Lemma 4 *Let m be a timed marking in Reach(Timed($\mathcal{N}$)). Then one can reach in Timed($\mathcal{N}'$) any timed marking m' with:*

- *$f({m'}^\sharp) = m^\sharp$ and*
- *for all $p \in P \cup P_T$, we have $m'(p) = m(p)$ and*
- *for all $t \in T$, $p \in Q$, we have either $m'(0_p^t) = \emptyset$ or $m'(0_p^t) = m'(p_t)$, and*
- *for all $q \in Q$, letting $T'_q = \{t \in T \mid m'(1_q^t) \neq \emptyset\}$, we have $m(q) = m'(q) \sqcup \{age_t \mid t \in T'_q\}$ with $m'(1_q^t) = min(m(p_t), age_t)$ for all $t \in T'_q$.*

The formal proofs of both the above lemmas can be found in the appendix. Now observe that for all places p, we have $\sum_{t \in T} {m'}^\sharp(1_p^t) \leq |T|$. Thus fixing an untimed marking c, there exist only a finite number of untimed markings ${m'}^\sharp$ such that $f({m'}^\sharp) = c$. Combining Lemmas 3, 4, we obtain:

Proposition 4. *Let c be an untimed marking of Timed($\mathcal{N}$). Let c' be any untimed marking of Timed($\mathcal{N}'$) with $f(c') = c$. Then c is reachable in Timed($\mathcal{N}$) iff c' is reachable in Timed($\mathcal{N}'$).*

This completes the first step of the proof of Theorem 3.

Step 2: From the TPN with restricted constraints $\mathcal{N}'$ to a Petri Net $\mathcal{N}''$. Now we show that for a TPN with restricted constraints $\mathcal{N}'$, it is decidable whether a marking c' is reachable in Timed($\mathcal{N}'$), by reducing $\mathcal{N}'$ to an equivalent (untimed) Petri net. As marking reachability is decidable for Petri nets, this completes the proof of Theorem 3.

Proposition 5. *For any TPN with restricted constraints $\mathcal{N}'$, one can construct a Petri Net $\mathcal{N}''$ such that $\mathcal{N}''$ and Timed($\mathcal{N}'$) are (untimed) bisimilar.*

Proof. Given a TPN with restricted constraints $\mathcal{N}'$, we first construct a 1-bounded (untimed) Petri Net $\mathcal{N}_1$ which is bisimilar to Timed($\mathcal{N}'_B$), where $\mathcal{N}_B$ is the bounded part of $\mathcal{N}$. That is,

Lemma 5 *If $\mathcal{N}_B$ is a K-bounded TPN, for some positive integer K, we can construct a* 1*-bounded Petri Net $\mathcal{N}_1$ such that $\mathcal{N}_1$ and Timed($\mathcal{N}_B$) are (untimed) bisimilar.*

The proof of this lemma is easily obtained by building a timed automaton bisimilar to Timed($\mathcal{N}_B$) and interpreting its regions as a 1-safe Petri Net [20], adapting a result for the original semantics of TPNs [15]. Once we have this, we add the unbounded places of $\mathcal{N}'$ to the Petri net $\mathcal{N}_1 = (P_1, T_1, {}^\bullet(), ()^\bullet, m_1)$. Formally, we construct the Petri net $\mathcal{N}_2 = (P_2, T_1, {}^\star(), ()^\star, m_2)$ with:

- The set P_2 of places of $\mathcal{N}_2$ is $P_2 = P_1 \cup P_u$, for P_u the unbounded places of Timed($\mathcal{N}'$).
- Initial marking m_2 is the union of m_1 and of the restriction of the initial marking of Timed($\mathcal{N}'$) to its set P_u of unbounded places.
- The set of transitions of $\mathcal{N}_2$ is the set T_1 of transitions of $\mathcal{N}_1$. Concerning the flow relations, let $t_1 \in T_1$ corresponding to a transition $t \in T$ in the original net Timed($\mathcal{N}'$). We have $p \in {}^\star t_1$ if:
  - $p \in P_1$ and $p \in {}^\bullet t_1$ (arc from p to t_1 in $\mathcal{N}_1$), or
  - $p \in P_u$ and there is an arc from p to t in Timed($\mathcal{N}'$).

  We have $p \in {t_1}^\star$ if $p \in P_1$ and $p \in {t_1}^\bullet$, or if $p \in P_u$ and there is an arc from t to p in Timed($\mathcal{N}'$).

With this, we have the following lemma, whose technical proof is again relegated to the appendix.

Lemma 6 *Timed($\mathcal{N}'$) and $\mathcal{N}_2$ are (untimed) bisimilar.*

Finally, by setting $\mathcal{N}'' = \mathcal{N}_2$, we obtain the proof of Proposition 5.

From the constructions described in Steps 1 and 2 above and using Propositions 4 and 5, we conclude the proof of Theorem 3. An easy adaptation of Proposition 5 shows that reachability, boundedness and control-state reachability are decidable for TPNs with restricted constraints under the original TPN semantics, i.e., Theorem 2.

6 Conclusion

In this paper, we extended Timed-Arc Petri Nets to express urgency constraints, thus capturing the essential feature of TPNs. While general Timed-Arc Petri Nets with Urgency are undecidable (as are TPNs), we obtain decidability when urgency is used only in the bounded part of the system. We then consider an alternative timed-arc semantics for general TPNs in terms of Timed-Arc Petri Nets with Urgency. This new timed-arc semantics allows TPNs to model some latency requirements. Decidability of reachability can be proved for a class of TPNs larger with the timed-arc semantics than with the usual one. We plan to study robustness properties, i.e, whether the system can withstand infinitesimal timing errors, as has been extensively studied for timed automata [25, 17, 11], etc. We would like to extend the study started for TPNs (e.g. [3]) to Timed-Arc Petri Nets with restricted Urgency.

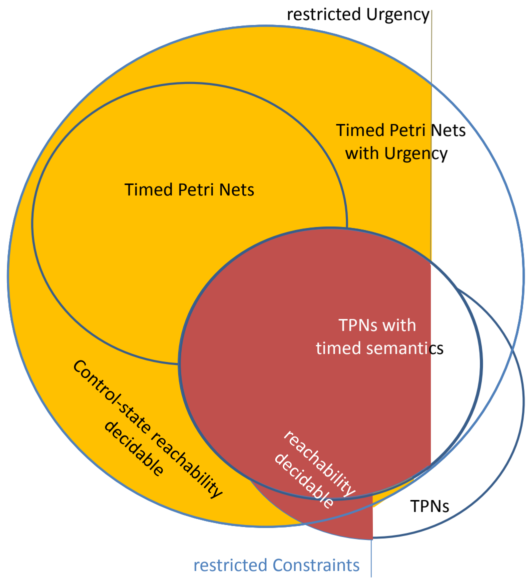

Fig. 5. Inclusion of classes of Time/Timed-Arc Petri Nets with urgency w.r.t. timed bisimilarity.

References

1. P. A. Abdulla, P. Mahata, and R. Mayr. Dense-timed Petri nets: Checking Zenoness, token liveness and boundedness. *Journal of Logical Methods in Computer Science*, 3(1), 2007.
2. P. A. Abdulla and A. Nylén. Timed Petri nets and BQOs. In *ICATPN*, LNCS 2075, p.53-70, 2001.
3. S. Akshay, L. Hélouët, C. Jard, and P.-A. Reynier. Robustness of time Petri nets under guard enlargement. In *RP, LNCS 7550*, pages 92–106, 2012.
4. R. Alur and D. Dill. A theory of timed automata. *In TCS*, 126(2):183–235, 1994.
5. F. Bause and P. S. Kritzinger. *Stochastic Petri nets - an introduction to the theory (2. ed.)*. Vieweg, 2002.
6. B. Bérard, F. Cassez, S. Haddad, D. Lime, and O. H. Roux. Comparison of different semantics for time petri nets. In *ATVA'05*, volume 3707 of *LNCS*, pages 293–307, 2005.
7. B. Bérard, F. Cassez, S. Haddad, D. Lime, and O. H. Roux. The expressive power of time petri nets. *TCS*, 474:1–20, 2013.
8. B. Berthomieu and M. Diaz. Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans. in Software Engineering*, 17(3):259–273, 1991.
9. B. Berthomieu, F. Peres, and F. Vernadat. Bridging the gap between timed automata and bounded time petri nets. In *FORMATS'06*, volume 4202 of *LNCS*, pages 82–97, 2006.
10. P. Bouyer, S. Haddad, and P.-A. Reynier. Timed Petri nets and timed automata: On the discriminating power of Zeno sequences. *Inf. Comput.*, 206(1):73–107, 2008.
11. P. Bouyer, N. Markey, and O. Sankur. Robustness in timed automata. In *Proc. of RP*, volume 8169 of *LNCS*, pages 1–18, 2013.
12. Th. Chatain and C. Jard. Complete Finite Prefixes of Symbolic Unfoldings of Safe Time Petri Nets. In *ICATPN*, volume 4024 of *LNCS*, pages 125–145, 2006.
13. L. Clemente, F. Herbreteau, A. Stainer, and G. Sutre. Reachability of communicating timed processes. In *FoSSaCS*, volume 7794 of *LNCS*, pages 81–96, 2013.

14. L. Clemente, F. Herbreteau, and G. Sutre. Decidable topologies for communicating automata with fifo and bag channels. In *CONCUR, LNCS 8704, p. 281-296*, 2014.
15. D. D'Aprile, S. Donatelli, A. Sangnier, and J. Sproston. From time Petri nets to timed automata: An untimed approach. In *TACAS, LNCS 4424*, pages 216–230, 2007.
16. A. David, L. Jacobsen, M. Jacobsen, and J. Srba. A forward reachability algorithm for bounded timed-arc Petri nets. In *SSV'12*, volume 102 of *EPTCS*, pages 125–140, 2012.
17. M. De Wulf, L. Doyen, N. Markey, and J.-F. Raskin. Robust safety of timed automata. *Formal Methods in System Design*, 33(1-3):45–84, 2008.
18. A. Finkel and Ph. Schnoebelen. Well-structured transition systems everywhere! *In TCS*, 256(1-2):63–92, 2001.
19. S. Haddad. Time and timed Petri nets. In *Disc PhD School 2011*, pages available at http://www.lsv.ens–cachan.fr/ haddad/disc11–part1.pdf, 2011.
20. L. Jacobsen, M. Jacobsen, M. H. Møller, and J. Srba. Verification of timed-arc Petri nets. In *SOFSEM'11*. LNCS 6543, p.46-72, 2011.
21. N. D. Jones, L. H. Landweber, and Y. E. Lien. Complexity of some problems in Petri nets. *Theor. Comput. Sci.*, 4(3):277–299, 1977.
22. D. Lime and O. H. Roux. Model checking of time Petri nets using the state class timed automaton. *Discrete Event Dynamic Systems*, 16(2):179–205, 2006.
23. J. A. Mateo, J. Srba, and M. G. Sørensen. Soundness of timed-arc workflow nets. In *PETRI NETS'14*. LNCS 8489, p.51-70, 2014.
24. P. M. Merlin. *A Study of the Recoverability of Computing Systems*. PhD thesis, University of California, Irvine, CA, USA, 1974.
25. A. Puri. Dynamical properties of timed automata. *In DEDS*, 10(1-2):87–113, 2000.
26. P.-A. Reynier and A. Sangnier. Weak time Petri nets strike back! In *Proc. of CONCUR 2009*, volume 5710 of *LNCS*, pages 557–571, 2009.
27. V. V. Ruiz, F. C. Gomez, and D. de Frutos-Escrig. On non-decidability of reachability for timed-arc Petri nets. In *PNPM*, pages 188–. IEEE Computer Society, 1999.
28. B. Walter. Timed Petri-Nets for Modelling and Analysing Protocols with Real-Time Characteristics. In *Proc. of PSTV*, pages 149–159, 1983.

Appendix

This appendix presents the missing details and proofs in the paper. It is organized section-wise. To clarify relations among variants of Petri nets and their properties, Figure 5 summarizes the relative expressiveness of the various classes, as well as the decidability results.

Proofs for Section 3

Proposition 1. *Control State reachability, (marking) reachability and boundedness are undecidable for Timed-Arc PNU.*

Proof. The proof follows on the same lines as the undecidability proof in [21] and is done by a standard encoding of reachability for a 2 counter machine with zero test (Minsky Machine). A Minsky machine has a set of counters $\{C_1, \dots C_k\}$, and a sequence of instructions $inst_1.inst_2 \dots inst_n$. Each Instruction $inst_i$ is of the form $inst_i : inc(j, k)$, meaning that the machine increments counter C_j and moves to instruction $inst_k$, or $inst_i : Jzdec(j, k, k')$ meaning that the machines tests whether counter C_j is zero or whether it is strictly greater. If it is 0, the machine moves to counter k'. Else it decrements C_j and moves to instruction k. A well known result is that it is undecidable whether a Minsky machine with two counters can reach an instruction x.

We now show how to encode any two-counters Minksy machine M with a Timed-Arc PNU $\mathcal{N}_M = (P_M, T_M, {}^\bullet()_M, ()^\bullet{}_M, m_0, \gamma_M, U_M)$. The set of places P_M contains two places p_1, p_2 (one per counter), and one place q_i per instruction $inst_i$ of the machine. The set of transitions T_M contains one transition t_i for each increment instruction $inst_i : inc(j, k)$, with ${}^\bullet t_{iM} = \{q_i\}$ and $t_i{}^\bullet{}_M = \{p_j, q_k\}$. Further, $U_M(t_i) = 0$ and $\gamma(q_i, t_i) = [0, \infty)$. T_M also contains a pair of transitions t_i, t_i^z for each decrement instruction $inst_i : Jzdec(j, k, k')$. Intuitively, t_i will simulate the move of the machine when counter C_j is not empty, and t_i^z the move of the machine when counter C_j is empty. We have ${}^\bullet t_{iM} = \{q_i, p_j\}$ and $t_i{}^\bullet{}_M = \{q_k\}$. Further, $U_M(t_i) = 0$ and $\gamma_M(q_i, t_i) = \gamma_M(p_j, t_i) = [0, \infty)$, i.e., transition t_i fires as soon as both places p_i and q_j contain at least one token. For transition t_i^z, we have ${}^\bullet t_i^z{}_M = \{q_i\}$ and $t_i^{z\bullet}{}_M = \{q_{k'}\}$. Further, $U(t_i^z) = 0$ and $\gamma_M(q_i, t_i^z) = [1, \infty)$, i.e., t_i^z fires exactly 1 time unit after place q_i is filled.

Let us show that urgency prevents firing of transition t_i^z from a marking m with $m^\sharp(q_i) = 1$ and $m^\sharp(p_j) > 0$ (i.e., illegally decrementing a counter of value zero). Suppose the machine moves from a marking m^- to m, with place q_i marked. This means that place q_i contains a token with age 0. At a given instant, only one place $q_j, j \in \{1, \dots, n\}$ can be marked in a reachable marking of $\mathcal{N}_M$. Suppose transition t_i^z can fire from a marking m' with $m'^\sharp(q_i) = 1$ and $m'^\sharp(p_j) > 0$, obtained from m by letting at least one time unit elapse. This means that $m'(p_j) \geq 1$, and hence, at least one time unit has elapsed from marking m. However, transition t_i is urgent from any marking with at least one token in q_i and one token in p_j, and in particular from m. Thus time cannot elapse from m, a contradiction. So, our encoding of M is faithful and a run $t_{i_1} \cdots t_{i_n}$ can happen in $\mathcal{N}_M$ iff run $inst_{i_1} \cdots inst_{i_n}$ can happen in M.

Hence a two-counter machine M reaches instruction x iff place q_x is reachable in $\mathcal{N}_M$. This reachability question can be equivalently encoded as a coverability problem where the marking to cover is $m(p_x) = \{0\}$ and $m(p) = \emptyset$ for all other places. Further, checking boundedness of a counter in a Minsky machine is also known to be undecidable. Hence, boundedness (and place boundedness) are also undecidable in general for Timed-Arc PNU.

Proofs for Section 3.1

Theorem 1. *Control-state reachability and boundedness are decidable for Timed-Arc PNrU. However, (marking) reachability is undecidable for Timed-Arc PNrU.*

Undecidability of reachability comes from the undecidability for Timed-Arc Petri nets [1]. It remains to prove the decidability for control-state reachability and boundedness of Timed-Arc PNrU. To do so, we extend the techniques of [1, 10] using well-quasi-orders to directly show decidability of control-state reachability and boundedness.

Decidability via Well-quasi-orders In order to prove Theorem 1, we extend the techniques of [1, 10] using well-quasi-orders to show decidability of control-state reachability and boundedness.

Proof. To simplify notations, we will consider that all constraints and urgencies are defined as integers, but the proof can be easily recast in a rational context, by defining intervals which size is $1/m$, where m is the smallest common multiple of rationals denominators.

We first define a region abstraction for markings. We then define an ordering on markings, and prove that this ordering is compatible with the transition relation. We then show that runs of Timed-Arc PNrU can be abstracted as runs on a finite region automaton.

We let max be the larger value appearing as an upper bound on an arc constraint. A region defines the integral parts of clock values up to max (the exact age of a token is irrelevant if it is greater than max), and also the ordering of the fractional parts. For Timed-Arc Petri Nets, we need to use a variant which also defines the place in which each token (clock) resides.

Definition 4 *A* region *is a triple* (b_0, w, b_{max}) *where*

- $b_0 \in M((P \times \{0, ..., max\}))$. b_0 *is a multiset of pairs. A pair of the form* (p, n) *represents a token with age exactly* n *in place* p.
- w *is a word from* $(M((P \times \{0, ..., max\})) - \{\emptyset\})^*$, *where each element is a non-empty multiset. Each letter* (p, n) *in* w *represents a token in place* p *with age in* $[n, n + 1]$. *Pairs in the same multiset represent tokens which ages have the same fractional part. The order of the multisets in* w *corresponds to the order of the fractional parts (i.e., smaller fractional parts come first in the word* w*).*
- $b_{max} \in M(P)$ *represent tokens which age have exceeded* max, *and can not be distinguished by transitions.*

The ordering among the multisets inside w reflects the ordering among the fractional parts of the clock values (in our cas token ages) that are increasing from left to right. These fractional parts are not made precise and could take any real value. However, if two tokens have ages a_1, a_2 with the same integral part but $a_1 < a_2$, then the token with the larger fractional part should be the first one to cause urgency or have useless age. Each region hence characterizes an infinite set of markings. A marking M satisfies a region $R = (b, w, b')$ (written $M \models R$) iff one can find an injective map from M to elements in R, that respects places localization, and age constraints imposed by w and the integral parts of tokens ages. We denote by$[\![R]\!]$ the set of markings that satisfy R. For a marking M, one can compute the unique region R_M such that $M \in [\![R_M]\!]$

The region construction allows for the definition of an equivalence among markings. We define $M_1 \equiv M_2$ iff for each R we have $M_1 \in [\![R]\!] \Leftrightarrow M_2 \in [\![R]\!]$.

Let us now consider ordering on markings. We set $M_1 \sqsubseteq M_2$ iff for every p, $M_1(p) \sqsubseteq M_2(p)$, that is M_1 can be obtained from M_2 by removing tokens. Let M be a marking of a Timed-Arc PNrU. We define M^U the restriction of M to unbounded places, and M^B the restriction of M to bounded places. Obviously, the restriction ${R_M}^U$ of R_M to unbounded places is equal to R_{M^U}, the restriction ${R_M}^B$ of R_M to unbounded places is equal to R_{M^B}.

We next define $M_1 \preceq M_2$ if $R_{M_1^B} = R_{M_2^B}$ and there exists M_2' such that $M_2' \sqsubseteq M_2^U$ and $M_1^U \equiv M_2'$.

Claim : if $M_1 \preceq M_2$ and $M_1 \longrightarrow M_1'$, then there exists M_2' such that $M_2 \longrightarrow M_2'$. Intuitively, this property holds because moves in the bounded part of a net, can be seen as moves of a timed automaton, for which a finite region abstraction exists and is bisimilar to the original model. Then, unbounded places contents can not forbid time to elapse, so if the bounded part of the net allows a timed move, the unbounded part can not forbid it. Similarly, discrete moves are allowed if there are enough tokens in the preset places of the fired transition, that is if bounded and unbounded parts contains enough tokens with adequate dates. Hence, having a larger marking can not prevent discrete firing.

Next we define an ordering on regions abstractions. For two regions obtained from a Timed-Arc PNrU, we set $R_M \preceq R'_M$ iff ${R_M}^B = {R'_M}^B$, and for the unbounded parts ${R_M}^U = (b_0, b_1 \dots b_m, b_{m+1})$ and ${R'_M}^U = (c_0, c_1 \dots {}_l, c_{l+1})$, there exists a monotone injective mapping $g : \{0, \dots, m{+}1\} \to \{0, \dots, l{+}1\}$ with $g(0) = 0, g(m{+}1) = l{+}1$, and for every $i \in 0..m+1, b_i \leq c_{g(i)}$.

Clearly, the order on regions agree with the order on markings: if $R \preceq R'$, then for every pair of markings $M \in [\![R]\!]$ $M' \in [\![R']\!]$, we have $M \preceq M'$

Our definition of regions only differ from the definitions of [1] on their bounded part. We can hence reuse the proofs of [1] to show that $[\![R]\!]^{\uparrow} = [\![R^{\uparrow}]\!]$, where $M^{\uparrow}$ denotes the set of markings with equivalent bounded region and unbounded part greater than M^U, and $R^{\uparrow}$ denotes the set of regions greater than R w.r.t $\preceq$.

So far, we have that $\preceq$ is a WQO on regions. It is an effectively decidable relation. Furthermore, one can effectively compute a finite representation for sets of predecessors for an upward closed set of regions (i.e. a representation of a set of markings with upward closed unbounded part).

For a region $R = (b_0, b_1 \ldots . b_m, b_{m+1})$ representing a set of markings, one can compute a finite set of regions $R_1, \ldots R_k$ such that markings of each R_i are transformed into markings of R via discrete moves of timed moves. Discrete moves due to firing of a transition t simply remove from b_0 one tokens of the form $(p, 0)$ for each place in the postset of t, and add tokens with adequate age and ordering (i.e ages that are at least $\alpha(p,t)$ but that do not exceed the urgency constraint of t) to b_0 and $b_1 \ldots . b_m$. As only an integer part with value in $0..max - 1$ has to be chosen together with a respective ordering on a finite number of fractional parts (i.e choose a place of insertion in w for at most one token per place), there is a finite number of such choices. Note that during a discrete move, tokens which age is greater than max remain in b_{m+1} as they can not be consumed by a transition. Timed moves consist in a shift of markings by δ time units. Calling $R_{-\delta}$ the region obtained from R by subtracting δ to all clocks, we have that for any urgent transition t enabled in $R_{-\delta}$, δ is such that it can not exceed $\alpha(p,t)+U(t)$ for at least one place p in the preset of t. As we only memorize integral parts in $[0..max-1]$ of clocks and the respective ordering on fractional parts, the set of regions that precede R by a timed move is finite and can be effectively computed as a reorganization of w accoridng to a circular permutation, and an insertion of a representation of some tokens in b_{max} in the other parts (b_0 and w) of the region (we do not give details of the computation here).

Construction of predecessors is hence effective. Furthermore, comparing regions is effective too, as it suffices to consider a finite set of mappings from R to R'. So one can maintain a finite set of minimal representatives for predecessors of upward closed sets of regions S.

This allows for the use of standard backward algorithms to compute $Pre^*(S)$ and check coverability of a particular marking, and hence to verify the control state reachability problem (see for instance [18]). We start from a region representation with exactly one token in place p to be covered. We have to test all possibilities regarding the interval in which this token's age lays (yielding as many tests as there are intervals). For each such region $R_i \in R_0, R_1, \ldots, R_{max}$, we perform a backward search to compute (a finite representation of) $Pre^*(R_i)$, and check whether M_0 belongs to the upward closure of $Pre^*(R_i)$.

Similarly, one can compute a finite region representation for the set of successors $Post(S)$ for a region R. This gives a way to build a coverability tree labeled by regions. We start from an open node labeled by region R_{M_0}. At each step of the algorithm, we compute $post(R)$ for a node n with region R attached, and that does not yet has successors. If $Post(R)$ contains a region R' that is strictly greater than a region attached to some predecessor of n, then we can deduce that the net is unbounded. If $Post(R)$ is empty, we need not continue the construction at this node, and close it. Otherwise, we add one open node per successor in $Post(R)$. The algorithm terminates when a witness of unboundedness has been found, or when the built tree constains no open node. Termination is guaranteed, because of well-quasi-ordering, as there is no infinite sequence of regions containing ony incomparable or decreasing regions. □

Expressive equivalence of Timed-Arc PNrU and Timed-Arc Petri nets In this part of the appendix, we show that Timed-Arc PNrU are in fact expressively equivalent to Timed-Arc Petri nets.

Proposition 2. *Timed-Arc PNrUs are timed language equivalent to Timed-Arc Petri Nets.*

Note that, this immediately gives an alternative proof of the decidability of control-state reachability for Timed-Arc PNrU.

The proof is done in two steps. First, given a Timed-Arc PNrU $\mathcal{N}$, we construct a K-bounded (labeled) Timed-Arc Petri Net $\mathcal{N}_1$ which is *timed (language) equivalent* to the bounded part $\mathcal{N}_B$ of $\mathcal{N}$ (it is not possible to get bisimilarity because Timed-Arc Petri Nets do not use urgency). This is done by converting the bounded part of the Timed-Arc PNrU into a timed automaton (using Prop. 6 below), and then using the following result from [10] (Theorem 7 and Corollary 1) to convert timed automata to Timed-Arc Petri Nets with equivalent (finite) language.

Lemma 7 *[10] Let $\mathcal{A}$ be a timed automaton. Then we can construct a K'-bounded Timed-Arc Petri Net $\mathcal{N}'$ (for some non-negative integer K') such that $\mathcal{N}'$ and $\mathcal{A}$ are timed equivalent.*

In the second step, we show that the Timed-Arc PNrU $\mathcal{N}$ is timed (language) equivalent to the Timed-Arc Petri net $\mathcal{N}_2$ formed by adding unbounded places of $\mathcal{N}$ to the Timed-Arc Petri Net $\mathcal{N}_1$ (see Lemma 8 below).

The proof of the bounded part (first step) goes via timed automata. We start by showing that the semantics of *bounded* Timed-Arc PNU can be encoded in a bisimilar way by timed automata. Let X be a finite set of real-valued variables called clocks. We write $\mathcal{C}(X)$ for the set of *constraints* over X, $\mathcal{C}_{ub}(X)$ of *upper bound* constraints over X as usual [4].

Definition 5 (Timed Automata (TA) [4]) *A* timed automaton $\mathcal{A}$ *over Σ_ε is a tuple (Q, q_0, X, δ, I) where Q is a finite set of* locations*, $q_0 \in Q$ is the* initial location*, X is a finite set of* clocks*, $I \in \mathcal{C}_{ub}(X)^Q$ assigns an* invariant *to each location and $\delta \subseteq Q \times \mathcal{C}(X) \times \Sigma_\varepsilon \times 2^X \times Q$ is a finite set of* edges.

A *valuation* v is a mapping in $\mathbb{R}_{\geq 0}^X$. For $R \subseteq X$, the valuation $v[R]$ is the valuation v' such that $v'(x) = v(x)$ when $x \notin R$ and $v'(x) = 0$ otherwise. Finally, constraints of $\mathcal{C}(X)$ are interpreted over valuations: we write $v \models \gamma$ when constraint γ is satisfied by v. The semantics of a TA $\mathcal{A} = (Q, q_0, X, \delta, I)$ is the transition system $[\![\mathcal{A}]\!] = (S, s_0, \rightarrow)$ where $S = \{(q, v) \in Q \times (\mathbb{R}_{\geq 0})^X \mid v \models Inv(q)\}$, $s_0 = (\ell_0, \mathbf{0})$ and $\rightarrow$ is defined by **delay moves:** $(\ell, v) \xrightarrow{d} (\ell, v + d)$ if $d \in \mathbb{R}_{\geq 0}$ and $v + d \models Inv(\ell)$; and **discrete moves:** $(\ell, v) \xrightarrow{a} (\ell', v')$ if there exists some $e = (\ell, \gamma, a, R, \ell') \in E$ s.t. $v \models \gamma$ and $v' = v[R]$. The (untimed) language of $\mathcal{A}$ is that of $[\![\mathcal{A}]\!]$ and is denoted by $\mathcal{L}(\mathcal{A})$.

The encoding of bounded Timed-Arc PNU into timed automata is not a surprise, as such bisimilar encodings have already been proposed for bounded TPNs. [15] propose to compute a marking timed automaton from TPN: the construction starts from the finite set of markings, and creates transitions of the form (m, g_t, t, m') for any pair of markings such that $m' = m - {}^\bullet t + t^\bullet$ with guard $g_t = \nu(t) \in [A(t), B(t)]$. We can then obtain the construction of finite timed automata from bounded Timed-Arc PNrUs.

Proposition 6. *Let $\mathcal{N}$ be a K-bounded Timed-Arc PNrU for some integer K. Then one can compute a finite timed automaton $\mathcal{A}^B_{\mathcal{N}}$ that is timed bisimilar to $\mathcal{N}$.*

Proof. (of proposition 6) We set $\mathcal{A}^B_{\mathcal{N}} = (Q, q_0, X, \delta, I)$, where Q is a set of states, δ a transition relation, $I : Q \to CST$ is an invariant relation attaching a constraint on clock values to each state, and X is a set of clocks.

The set of clocks X is computed as follows: $X = \{x_{p,i} \mid p \in P \wedge i \in 1 \dots K\}$. Intuitively, $x_{p,i}$ will measure the age of the i^{th} token in place p. We denote by CST the set of all linear constraints that can be defined over X. Note that as we associate one clock per token in each place, and a place can be marked with less than K tokens, all clocks are not meaningful. We will compute inductively a set $Q \subseteq 2^X \times CST$. Each state $q = (A, cst)$ of the automaton will remember a set A of active clocks, and a constraint cst on the values of clocks. This construction is almost the state class construction of [22], with the slight difference that an active clock measures the age of a tokens instead of measuring the time since a transition was last newly enabled. To each state $q = (A, cst)$ and valuation ν, we can associate a timed marking $m_{q,\nu}$ such that $m_{q,\nu}(p) = \biguplus_{x_{p,i} \in A} \{\nu(x_{p,i})\}$. Similarly, to each clock selection A, we can associate a marking $m^\sharp(A)(p) = |\{x_{p,k} \in A\}|$. We will say that A enables t if for every place p in ${}^\bullet t$, $m^\sharp(A)(p) \geq 1$.

Let us now define the transition relation. We do not need to define timed moves, that will simply consist in letting time elapse from a state of the automaton. However, transitions of the automaton have to encode discrete moves of the net. As already pointed out, from a timed marking m, a discrete firing of a transition t can result in a finite number of new markings $m'_1, \dots, m'_k$, as the tokens to be consumed from each place can be chosen non-deterministically. Note however that as places are bounded by K, the number of markings that can be reached is bounded by $K^{|P|}$.

Let $q = (A, cst)$, and let t be a transition such that $\forall p \in {}^\bullet t$ there exists at least one clock $x_{p,i}$ in A. We define $EXS(A, t)$ as the set of sets of clocks that contain exactly one clock of A per place in ${}^\bullet t$. To fire, a transition needs to select a set of tokens that will be consumed, that is choose a element Y of $EXS(A, t)$. To fire t with set of clocks Y, every clock $x_{p,i}$ in Y has to satisfy $x_{p,i} \in \gamma(p, t)$. For every $Y \in EXS(A, t)$, we define $g_{t,Y} = \bigwedge_{x_{p,i} \in Y} x_{p,i} \in \gamma(p, t)$ as the guard attached to transition t when choosing set Y of token clocks. For a fixed origin state $q = (A, cst)$, a transition t and a chosen set of clocks Y, we will denote by $q' = Next(q, t, Y)$ the state reached after firing t from q, provided clocks in Y satisfy $g_{t,Y}$. Hence, such a transition will exist only if $cst \wedge g_{t,Y}$ is satisfiable. State q' is of the form $q' = (A', cst')$, where $A' = A \setminus Y \cup R(A, t, Y)$, where $R(A, t, Y) = \{x_{p_1,k_1}, \dots x_{p_q,k_q}\}$ contains exactly one clock per place in $t^\bullet$, and such that each k_j is the minimal index of clock attached to place p_j that does not appear in $A \setminus Y$ (this avoids introducing non-determinism when reusing clocks). This index exists, as $\mathcal{N}$ is bounded. The constraint attached to the new state is computed as follows:

We add to cst the guard $g_{t,Y}$, and the conjunction of constraints $x'_{p,j} \leq x_{p',i}$ for each pair of clocks in $R, A \setminus Y$. Intuitively, this additional constraint means that the age of newly created tokens must be smaller or equal to the age of remaining tokens. We then eliminate variables $x_{p,i} \in Y$, using a variable elimination technique such as Fourier-Motzkin elimination. We then rename variables $x'_{p,j}$ to $x_{p,j}$ to obtains cst'.

We can now define δ as follows $\delta = \{(q = (A, cst), g_{t,Y}, t, R(A, t, Y), q') \mid g_{t,Y} \wedge cst$ is satisfiable $\wedge\ t$ enabled by $A \wedge q' = Next(q, t, Y)\}$. One can remark that a transition t can be enabled from several states, as two distinct clock selections may represent the same markings. The initial state of the automaton is $q_0 = (A_0, cst_0)$ where A_0 is the set $A_0 = \bigcup\{x_{p,1}, x_{p,k} \mid m_0^\sharp(p) = k\}$, and cst_0 is the set of constraints $cst_0 = \{x = y \mid x, y \in A_0\}$. One can note that by repeteadly combining conjunctions of inequations of the form $x \leq c, c \leq x$ where c is a constant and x a clock and then eliminating variables, we can only reach a bounded number of constraints from cst_0. Hence, Q is finite.

The last ingredient of our timed automaton $\mathcal{A}_\mathcal{N}$ is the invariant attached to each state. Invariants must guarantee that clock values must not increase beyond urgency of a transition, i.e one can not reach a clock valuation ν such that for each place of the preset of some enabled transition t there exists a clock $x_{p,i}$ such that $\alpha(t) + U(t) < \nu(x_{p,i}) \leq \beta(t)$. □

We can now come back to the proof of proposition 2: As we can build a timed automaton $\mathcal{A}^B_\mathcal{N}$ from any bounded $\mathcal{N}$, using lemma 7, one can build Timed-Arc Petri Net $\mathcal{N}_1$ which is *timed equivalent* to $\mathcal{N}$. One can apply this technique to build a Timed-Arc Petri Net $\mathcal{N}_1$ which is *timed equivalent* to the bounded part $\mathcal{N}_B$ of $\mathcal{N}$ (it is not possible to get bisimilarity because Timed-Arc Petri nets do not use urgency).

We can now end the proof of proposition 2 by showing that the original Timed-Arc PNrU $\mathcal{N}$ is timed equivalent to the Timed-Arc Petri net $\mathcal{N}_2$ formed by adding unbounded places of $\mathcal{N}$ to the Timed-Arc Petri Net $\mathcal{N}_1$.

Lemma 8 *$\mathcal{N}$ and $\mathcal{N}_2$ are timed equivalent.*

Proof. (of Lemma 8) First, let $(t^1, d^1) \cdots (t^n, d^n)$ be a finite word in the language of $\mathcal{N}$, where t^i are transitions of $\mathcal{N}$ and $d^1 < \cdots < d^n$ are dates at which t^n occurs (that is, $d^i - d^{i-1}$ units of time occured between t^{i-1} and t^i). Let m be some timed marking reached in $\mathcal{N}$ after $(t^1, d^1) \cdots (t^n, d^n)$. First, as $(t^1, d^1) \cdots (t^n, d^n)$ is a timed word of $\mathcal{N}$, it is also a timed word of $\mathcal{N}_B$, the bounded restriction of $\mathcal{N}$, and thus of $\mathcal{N}_1$. Let m_1 be a timed marking reached in $\mathcal{N}_1$ after $(t^1, d^1) \cdots (t^n, d^n)$. We now show by induction that the timed word $(t^1, d^1) \cdots (t^n, d^n)$ is a timed word of $\mathcal{N}_2$, reaching a timed marking $m_2 = m_1 \cup m_u$ with m_u the restriction of m to P_u. First, it is trivial for the empty timed word.

Induction step: Let $\delta = d^n - d^{n-1}$ and $t = t^n$. Let m'_1, m' be two markings such that $m'_1 \xrightarrow{\delta} m'_1 + \delta \xrightarrow{t} m_1$ in $\mathcal{N}_1$ and $m' \xrightarrow{\delta} m' + \delta \xrightarrow{t} m$ in $\mathcal{N}$. By the induction hypothesis, the timed marking $m'_2 = m'_1 \cup m'_u$ is reached in $\mathcal{N}_2$ after $(t^1, d^1) \cdots (t^{n-1}, d^{n-1})$. We want to show that from m'_2, δ timed units can elapse and then t is allowed, leading to m_2. First, δ time units can elapse from m'_2 in $\mathcal{N}_2$ as there is no urgency in $\mathcal{N}_2$ (it is a Timed-Arc Petri Net). Further, tokens consumed in $m'_1 + \delta \xrightarrow{t} m_1$ can be consumed in

$m'_2 + \delta$ as well, and so do tokens consumed from P_u in the transition $m' + \delta \xrightarrow{t} m$. Thus we indeed have $m'_2 \xrightarrow{\delta} m'_2 + \delta \xrightarrow{t} m_2$ and the proof is completed by induction.

Conversely, take $(t^1, d^1) \cdots (t^n, d^n)$ be a finite word in the language of $\mathcal{N}_2$. Denote by $m_2 = m_1 \cup m_u$ a timed marking reached by this timed word, with m_1 a timed marking of $\mathcal{N}_1$ and m_u the timed marking over P_u the infinite places of $\mathcal{N}$. We have that $(t^1, d^1) \cdots (t^n, d^n)$ is a timed word of $\mathcal{N}_1$ and thus of $\mathcal{N}_B$ the bounded part of $\mathcal{N}$.Let m_b be a timed marking of $\mathcal{N}_B$ reached by this timed word. Then we prove by induction that $(t^1, d^1) \cdots (t^n, d^n)$ is a timed word of $\mathcal{N}$ reaching $m = m_b \cup m_u$. The initial step is trivial for the empty timed word.

Induction step: Let $\delta = d^n - d^{n-1}$ and $t = t^n$. Let m'_2 such that $m'_2 \xrightarrow{\delta} m'_2 + \delta \xrightarrow{t} m_2$ in $\mathcal{N}_2$, with $m'_2 = m'_1 \cup m'_u$ decomposed between P_1 and P_u, and let m'_b such that $m'_b \xrightarrow{\delta} m'_b + \delta \xrightarrow{t} m_b$ in $\mathcal{N}_B$ the bounded part of $\mathcal{N}$. By the induction hypothesis, the timed marking $m' = m'_b \cup m'_u$ is reached in $\mathcal{N}_2$ after $(t^1, d^1) \cdots (t^{n-1}, d^{n-1})$. We want to show that from m', δ timed units can elapse and then t is allowed, leading to m. First, δ time units can elapse from m'_b in $\mathcal{N}_B$, and thus it can elapse from m' in $\mathcal{N}$ because $\mathcal{N}$ is a Timed-Arc PNrU (there is no urgency constraint on transitions consuming tokens from P_u). Further, tokens consumed from $m'_b + \delta \xrightarrow{t} m_b$ can be consumed in m' as well, and so do tokens consumed from P_u in the transition $m'_2 + \delta \xrightarrow{t} m_2$. Thus we indeed have $m' \xrightarrow{\delta} m' + \delta \xrightarrow{t} m$ and the proof is completed by induction. □

Thus, from Lemma 8 we obtain that for each Timed-Arc PNrU $\mathcal{N}$, one can construct a Timed-Arc Petri Net $\mathcal{N}_2$ such that $\mathcal{N}$ and $\mathcal{N}_2$ are timed equivalent. This concludes the proof of Proposition 2.

The Timed-Arc Petri net $\mathcal{N}_2$ built in Proposition 2 can be used to check control state reachability for a place p of $\mathcal{N}$. We know that this property is decidable for Timed-Arc Petri nets, and boils down to checking that some place of $\mathcal{N}_2$ representing a marking where p contains a token can contain a token. This property is decidable for Timed-Arcs Petri nets [2]. Note that there might be several such places due to the translation to and from timed automata.

Proofs for Section 4.3

Proposition 3. *Let $\mathcal{N}$ be a TPN such that for all reachable configurations, for all places p, either p has at most 1 token, or only one transition t is enabled with $p \in {}^\bullet t$. Then $\mathcal{N}$ and Timed($\mathcal{N}$) are timed bisimilar.*

Proof. Consider the relation $\mathcal{R}$ between reachable configurations of $\mathcal{N}$ and reachable timed markings of Timed($\mathcal{N}$) defined as follows: $((m, \nu), m')) \in \mathcal{R}$ if, for all $p \in P$ we have $m'^\sharp(p) = m(p)$ for all $p \in P$, and the following property (P1) holds:

(P1) for all $t \in En(m)$, $\nu(t) = min_{p \in {}^\star t}(\max'_p)$, where $\max'_p$ is the oldest token in $m'(p)$ for all p.

Notice that $p_t \in {}^\star t$. Also, if $t \in En(m)$, then the token $\max'_p$ can be consumed as by definition $\beta(p, t) = +\infty$ for all p, t.

We now show that $\mathcal{R}$ is a timed bisimilation, i.e., we prove that for each pair $\big((m,\nu),m'\big)$ in $\mathcal{R}$ a timed or discrete move a is allowed from (m,ν) if and only if a is also allowed from m', and the resulting configurations and timed markings are in $\mathcal{R}$.

Clearly, $((m_0,\nu_0),m'_0) \in \mathcal{R}$. Now, let $((m,\nu),m') \in \mathcal{R}$. Thus $m'^{\sharp}(p) = m(p)$ for all $p \in P$ and (P1) holds. First, assume that the elapse of δ units of time is possible from (m,ν). This means that no urgent transition is met after $\delta' < \delta$ units of time. In particular, for all $t \in En(m)$, $\nu(t) + \delta' \leq B(t)$. As $\nu(t) = min_{p\in {}^{\star}t}(\max'_p)$, there exists $p \in {}^{\star}t$ with $\max'_p + \delta' \leq B(t) = A(t) + (B(t) - A(t)) = \alpha(p,t) + U(t)$, thus t is not urgent in m' either for any δ'.

Both moves $(m,\nu) \xrightarrow{\delta} (m,\nu+\delta)$ and $m' \xrightarrow{\delta} (m'+\delta)$ are allowed. We have $((m,\nu+\delta),m'+\delta) \in \mathcal{R}$ as $\nu(t) = min_{p\in {}^{\star}t}(\max'_p)$, with $\max'_p$ the oldest token in $m'(p)$. We also have $(\nu+\delta)(t) = \nu(t)+\delta = min_{p\in {}^{\star}t}(\max'_p)+\delta = min_{p\in {}^{\star}t}(\max''_p)$, with $\max'_p$ the oldest token in $m' + \delta(p)$. Symmetrically, if a δ time units move is possible from m',then it is possible from (m,ν) as well and lead to this same bisimilar configuration.

Now we address discrete moves: let t be a transition fireable from configuration (m,ν), and leading to configuration (m_2,ν_2). Thus $t \in En(m)$ and $\nu(t) \in I(t)$. For all $p \in {}^{\star}t$, $\max'_p \geq \nu(t) \geq A(t) = \alpha(t)$, hence t is enabled in m'. When firing t from a timed marking of a Timed-Arc PNU, any of the tokens at each pre-place may be consumed. Hence, we can obtain several timed markings m'_2. However, for all such m'_2 obtained, we get easily $m'_2{}^{\sharp}(p) = m_2(p)$ for all $p \in P$.

To complete the proof that $((m_2,\nu_2),m'_2) \in \mathcal{R}$, we need to show that $P1$ holds. Let $\max''_p$ denote the maximal value of a token in timed marking m'_2. For $tt \in En(m_2)$, we have several cases:

- if ${}^{\bullet}tt \cap {}^{\bullet}t \neq \emptyset$, then there exists a place p in ${}^{\bullet}t \cap {}^{\bullet}tt$. If $tt = t$, we have by definition $\nu_2(t) = 0 = age_t$ and $m_2(p_t) = \{age_t\}$. If $tt \neq t$ and both tt and t are enabled in m, then by the assumption on the TPN, $m(p) = 1$. Thus tt is newly enabled in (m_2,ν_2), as the intermediate marking reached during firing of t from m satisfies $m(p) - {}^{\bullet}t(p) = 0$. Thus $\nu_2(tt) = 0$. On the other hand, we also have that $p \in t^{\bullet}$ (else tt would not be in $En(m_2)$) and $m'_2(p) = \{0\}$. Hence $max''_p = 0$ and $min_{q\in m'_2}(max''_q) = 0$, and thus $P1$ holds.
- Else, if $tt \in En(m)$, we have $\max'_p = \max''_p$ for any place p in ${}^{\star}tt$, as no token from ${}^{\star}tt$ has been consumed. Hence, $\nu_2(tt) = \nu(tt) = min_{p\in {}^{\star}tt}(\max'_p) = min_{p\in {}^{\star}tt}(\max''_p)$, and $P1$ holds.
- Else, tt is newly enabled, with some $p \in {}^{\bullet}tt \cap t^{\bullet}$ and $m'_2(p) = \{0\}$. Thus $\nu_2(tt) = 0 = min_{p\in m'_2}(max''_p)$ as $max''_p = m'_2(p)$ which implies that $P1$ holds.

Thus we have proved that $((m_2,\nu_2),m'_2) \in \mathcal{R}$ and this completes one direction of the bisimulation.

For the reverse direction, let transition t be fireable from timed marking m' and leading to timed marking m'_2. As $m'^{\sharp}(p) = m(p)$ for any place $p \in P$, we have $t \in En(m)$. As t is fireable from m', there exists $age'_p \in m'(p)$ with $\alpha(p,t) \leq age'_p \leq \max'_p$. Hence $\alpha(p,t) \leq min_{p\in {}^{\star}t} \max'_p = \nu(t)$. As no urgency constraint is ever violated (the timed marking and configurations are reachable), we also have

$\nu(t) = min_{p \in {}^\star t}(max_p) \leq \alpha(t) + U(t) = A(t) + B(t) - A(t) = B(t)$. Hence t is fireable from (m, ν), obtaining marking m_2 with $m_2(p) = {m'_2}^\sharp(p)$ for all $p \in P$. The proof that (P1) holds for $(m_2, \nu_2), m'_2$ is exactly the same as above. As a conclusion, we have that $\mathcal{R}$ is a bisimilation. $\square$

Proofs and details for Section 5

We start with Lemmas 3 and 4, that are used to prove Proposition 4. We recall that $\mathcal{N} = (P \cup Q, T, {}^\bullet(), ()^\bullet, m_0, \lambda, I)$ is a Timed-Arc PNrU, where P is the set of bounded places and Q the set of unbounded places, and $\mathcal{N}'$ is a TPN with restricted constraints obtained from $\mathcal{N}$.

Let us start with a definition which will simplify notations. Let m' be a timed marking in $\mathsf{Reach}(\text{Timed}(\mathcal{N}'))$ and m a timed marking in $\mathsf{Reach}(\text{Timed}(\mathcal{N}))$. Then, we write $m \equiv m'$, if $f(m'^\sharp) = m^\sharp$.

Lemma 3 *Let m' be a timed marking in $\mathsf{Reach}(\text{Timed}(\mathcal{N}'))$. Then there exists a timed marking in $\mathsf{Reach}(\text{Timed}(\mathcal{N}))$ with $m \equiv m'$.*

Proof. We will prove by induction on the size of a path that if one can reach m' in Timed($\mathcal{N}'$), then one can reach m in Timed($\mathcal{N}'$) with

- $m \equiv m'$,
- for all $p \in P \cup \{p_t \mid t \in T\}$, $m(p) \geq m'(p)$,
- letting T_q be the set of $t \in T$ such that $m(1_q^t) \neq \emptyset$ for all $q \in Q$, then for all $q \in Q$, $m(q) = m'(q) \sqcup \{age_1, \cdots age_k\}$ and there exists a bijection $g : T_q \mapsto [1, k]$ with $m'(1_q^t) \leq age_{g(t)}$ for all $t \in T_q$.

For $m' = m'_0$, this property holds trivially. We can now proceed by induction on the length of runs needed to reach m'. Let m' be a reachable marking of Timed($\mathcal{N}$). A path reaching m' ends with a move $m'' \xrightarrow{e} m'$, where e can be a timed move, a firing of a transition $start_p^t$, or a firing of a transition t of the original net. Hence m'' is reached in less steps than m'. We can hence apply the induction hypothesis, i.e., one can reach m in Timed($\mathcal{N}'$) with:

- $m \equiv m''$,
- for all $p \in P \cup \{p_t \mid t \in T\}$, $m(p) \geq m''(p)$,
- letting T_q be the set of $t \in T$ such that $m(1_q^t) \neq \emptyset$ for all $q \in Q$, then for all $q \in Q$, $m(q) = m''(q) \sqcup \{age_1, \cdots age_k\}$ and there exists a bijection $g : T_q \mapsto [1, k]$ with $m''(1_q^t) \leq age_{g(t)}$ for all $t \in T_q$.

Case $e = start_p^t$: Then it is easy to see that $m'(q) = m(q)$ for all q but for $m(p) = m'(p) \sqcup \{age\}, m(1_q^t) = \emptyset$ and $m'(1_q^t) = 0$. It is easy to check that m satisfies the hypothesis wrt m'. As m is known to be reachable in *Timed*($\mathcal{N}$), we conclude.

Case $e = \delta$ (time elapses by δ units): m and m'' agrees on the bounded places, hence as the urgency is not violated in m'' by elapsing δ unit of time, it is not violated in m by elapsing δ unit of time. Thus $m + \delta$ is reachable in *Timed*($\mathcal{N}$). Last, it is easy to see that $m + \delta$ satisfies the hypothesis wrt $m' = m'' + \delta$.

Case $e = t$ for some $t \in T$: If ${}^\bullet t$ has only bounded places, then m'' and m agree on these places, and one can apply t from m to obtain m^+ which satisfies the hypothesis wrt m'. Else, consider an unbounded place $q \in {}^\bullet t \cap Q$. Thus $I(t) = [a, +\infty)$. We have $m'(1_q) = \{nage_q\}$ with $nage_q \geq a$. Thus there exists $age_q \in m(q)$ with $age'_q \geq age_q$. In particular, t is enabled wrt q. This is true for all q, so t is enabled from m. Denote m^+ the reached configuration of Timed($\mathcal{N}$) after deleting the chosen tokens age_q with $age_q \geq nage_q$.

Now, firing t in m creates the same tokens in the same place with the same age 0 as firing t from m'. It deletes token $m'(1_q)$ and one token in q of age bigger or equal. Hence we still have for all $q \in Q$, $m^+(q) = m'(q) \sqcup \{age_1, \cdots age_k\}$ and there exists a bijection $g : T_q \mapsto [1, k]$ with $m'(1^t_q) \leq age_{g(t)}$ for all $t \in T_q$. That is, m^+ satisfies the hypothesis wrt m'. □

Lemma 4 *Let m be a timed marking in Reach(Timed($\mathcal{N}$)). Then one can reach in Timed($\mathcal{N}'$) any timed marking m' with:*

- *$f(m'^\sharp) = m^\sharp$ and*
- *for all $p \in P \cup P_T$, we have $m'(p) = m(p)$ and*
- *for all $t \in T$, $p \in Q$, we have either $m'(0^t_p) = \emptyset$ or $m'(0^t_p) = m'(p_t)$, and*
- *for all $q \in Q$, letting $T'_q = \{t \in T \mid m'(1^t_q) \neq \emptyset\}$, we have $m(q) = m'(q) \sqcup \{age_t \mid t \in T'_q\}$ with $m'(1^t_q) = min(m(p_t), age_t)$ for all $t \in T'_q$.*

Proof. We proceed by induction on the length of run reaching m in $\mathcal{N}$. For a run of length 0, this is trivial. Assume that m is reached after a move $m^- \xrightarrow{e} m$. Let m' be any marking satisfying the condition above wrt m. We will show that we can reach m' in Timed($\mathcal{N}'$).

Assume that e lets $\delta > 0$ units of time elapse. For all $p \in P \cup Q$, for all $age_p \in m(p)$, $age_p \geq \delta$. We first show that for all $p' \in P'$, for all $age'_p \in m'(p')$, $age'_p \geq \delta$. For $q = 1^t_p$ for some t, p, we have $m'(q) = \{min(m(p_t), age_p)\}$ with $age_p \in m(p)$. Now, we have $m(p_t) \geq \delta$ and $age_p \geq \delta$. For other $p' \in P'$, $age' \in m'(p)$, there exists $p \in P \cup Q$ with $age' \in m(p)$. We thus obtain for all $p' \in P'$, for all $age' \in m'(p')$, $age' \geq \delta$. Hence we can define $m'' = m' - \delta$ as a marking.

The number of tokens in m, m^- is the same, and the number of tokens in m', m'' is the same (only timing changed). Thus, as $m \equiv m'$, we have $m^- \equiv m''$. Now, $m^- \equiv m''$ satisfies the conditions above, we can thus apply the induction hypothesis telling us that m'' is reachable in Timed($\mathcal{N}'$). Now, waiting δ units of time from m'' is allowed in Timed($\mathcal{N}'$) as it does not violate any urgency. Otherwise, i.e., if the urgency of some transition t was violated, it would imply that $I(t) = [a, b]$ and thus ${}^\bullet t \subseteq P$ is made of only bounded places. As m^- and m'' coincide on bounded places, this would also violate urgency on m^-, which is by definition not the case, a contradiction. Hence δ units of time can elapse from m'', and the marking reached is m'. That is, m' is reachable in Timed($\mathcal{N}'$).

Now, consider a discrete move e firing a transition t. Let us define as $(age_p)_{p \in {}^\bullet t}$ the tokens which are consumed by t from m^- to m, that is $age_p = m^-(p) \setminus m(p)$. We have $m(p_t) = 0$ (after t is fired). Thus for all $q \in {}^\bullet t$, $m'(1^t_q) = \emptyset$ or $m'(1^t_q) = 0$ for m' satisfying the condition above wrt m.

We then define m'^- such that $m^- \equiv m'^-$ in the following manner:

- For all $p \in P \cup P_T$, we have $m'^-(p) = m^-(p)$.
- For all $q \in Q$ with $q \in {}^\bullet t$, $m'^-(1_q^t) = \min(m^-(t), age_q)$ for age_q the age of the token consumed by t from $m^-(q)$ to produce $m(q)$ defined above.
- For all $q \in Q \cap t^\bullet$, we have two cases. In the first case, the token 0 created by t needs to be consumed immediately after t by a $start_q^{t_q}$ for some transition t_q we define below. This is the case if the number of transitions tt with $m'(1_q^{tt}) = \{0\}$ is exactly the number of tokens 0 in $m(q)$. In this case, we choose t_q to be any transition tt with $m'(1_q^{tt}) = \{0\}$ and let $S_q = \emptyset$ (the token is created then consumed, hence it is not in $m'(q) \setminus m(q)$).
  Otherwise, we will consider that all $m'(1_q^{tt})$ have been already filled in $m'^-(1_q^{tt})$: we let $t_q = \bot$ be undefined and we define $S_q = \{0\}$ (a token 0 is added in $m'(q)$ by t). Last, for all $q \in Q \setminus t^\bullet$, we define $S_q = \emptyset$ (there is no added token for these q).
- For all $q \in Q$ and transition tt with $q \in {}^\bullet tt$,
  - if $tt \neq t, t_q$, we let $m'^-(1_q^{tt}) = m'(1_q^{tt})$
  - if $tt = t_q \neq t$ (in particular t_q is defined), then $m'^-(1_q^{tt}) = \emptyset$ (if $t_q = t$ it is already defined above).
- for all $q \in Q$, if $q \notin {}^\bullet t$ or if $m'(1_q^t) = \emptyset$, then we let $m'^-(q) = m'(q) \setminus S_q$. Else, $m'(1_q^t) = \{0\}$ has been created by firing transition $start_q^t$ after t, consuming a token from $m'^-(q)$, which is no more in m'. In this case, we define $nage_q$ the age of the token consumed as $nage_q = age_t$ (otherwise, $nage_q = \bot$ is undefined). By definition, we have $m(q) = m'(q) \sqcup \{age_{t'} | t' \in T'_q\}$. We then let $m'^-(q) = m'(q) \sqcup \{nage_q\} \setminus S_q$.

Let us now show that m^-, m'^- satisfy the induction hypothesis. The three first items are trivial. Let us show the last item. Let $q \in Q$ and let $New_q = 0$ if $q \in t^\bullet$ and $New_q = \emptyset$ otherwise. Let $Old_q = m'(q) \setminus m(q)$. We have $m^-(q) \sqcup New_q = m(q) \sqcup Old_q$. We also have $m(q) = m'(q) \sqcup \{age_{tt} \mid tt \in T'_q\}$ by hypothesis on m'.

Hence $m^-(q) = m'(q) \sqcup \{age_{tt} \mid tt \in T'_q\} \sqcup Old_q \setminus New_q$, with the T'_q associated with m'. Replacing $m'(q)$ by the definition above, we obtain $m^-(q) = m'^-(q) \sqcup S_q \setminus \{nage_q\} \sqcup \{age_{tt} \mid tt \in T'_q\} \sqcup Old_q \setminus New_q$. Now, Old_q is associated with $m^-(1_q^t)$ (both sets may be empty).

It remains to explain $\{age_{tt} \mid tt \in T'_q\} \sqcup S_q \setminus \{nage_q\} \setminus New_q$. We have $S_q \subseteq New_q$, with equality unless t_q is defined, in which case $New_q \setminus S_q = \{0\}$. This is to take into account the fact that either $t_q = t$ is already taken care of above, or $t_q \neq t$ and $m'^-(1_q^{t_q}) = \emptyset$. Last, $\{nage_q\} \neq \emptyset$ if $m'(1_q^t) = \{0\}$, which was already taken care of above for m'^-. The remaining in $\{age_{tt} \mid tt \in T'_q\}$ correponds to $m'(1_q^t) = m'^-(1_q^t)$. Hence the induction hypothesis is proved.

Hence we can apply the induction hypothesis, and obtain that m'^- is reachable in Timed($\mathcal{N}'$). Now performing t from m'^- then $start_q^{t_q}$ for all q such that $t_q \neq \bot$ and then $start_q^t$ for all q with $nage_q \neq \bot$, we obtain m'. That is, m' is reachable in Timed($\mathcal{N}'$), and we proved the next step of the induction. □

Finally, we give the proof for lemma 6, that is needed to prove Proposition 5. We recall that $\mathcal{N}''$ is a Petri net obtained from a TPN with restricted constraints $\mathcal{N}'$ by first computing the behavior of the bounded part of $\mathcal{N}'$, and then appending the unbounded part of the net.

Lemma 6 *Timed($\mathcal{N}'$) and $\mathcal{N}''$ are (untimed) bisimilar.*

Proof. A timed marking m of Timed($\mathcal{N}$) can be decomposed as $m = m_b \cup m_u$, where m_b is the restriction of m to bounded places, and m_u the restriction to unbounded places. Similarly, a marking of $\mathcal{N}''$ can be decomposed into $m_2 = m_1 \cup m_u$ again by restriction to bounded/unbounded places. From Lemma 5 and from the construction of $\mathcal{N}'$, we know that Timed($\mathcal{N}_B$) is bisimilar to $\mathcal{N}'$ (the details are on the same lines as Proposition 6 explained earlier in this Appendix). We denote by $R_{B,1}$ the unique largest bisimulation from timed markings of Timed($\mathcal{N}_B$) to markings of $\mathcal{N}'$.

We denote by R a relation from timed markings of Timed($\mathcal{N}$) to markings of $\mathcal{N}''$ as follows. Let $m = m_b \cup m_u$ be a marking of Timed($\mathcal{N}$) and $m_2 = m_1 \cup m'_u$ be a marking of $\mathcal{N}''$. Then, $(m, m_2) \in R$ iff $(m_b, m_1) \in R_{B,1}$, and $m'_u = m^\sharp_u$. Obviously, we have $(m_0, m_2) \in R$. We can now prove that R is a bisimulation.

Let $(m, m_2) \in R$. Assume that $m \xrightarrow{\delta} m+\delta \xrightarrow{t} m'$ in $\mathcal{N}$. Thus $m_b \xrightarrow{\delta} m_b+\delta \xrightarrow{t} m'_b$ in $\mathcal{N}_b$ with m'_b the bounded part of m'. Furthermore, $m^\sharp_u \geq {}^\bullet t \cap P_u$. Thus we have $m_1 \xrightarrow{t} m'_1$ in $\mathcal{N}_1$, and furthermore, $(m'_1, m'_b) \in R_{B,1}$. By definition of $\mathcal{N}''$, firing t results in a flow of tokens among places of P_u that is identical (regardless of ages) in $\mathcal{N}$ and in $\mathcal{N}''$, so we indeed have $m_1 \cup m^\sharp_u \xrightarrow{t} m'_1 \cup m'^\sharp_u$. Furthermore $m'^\sharp_u = m'_u$, so $(m', m'_1 \cup m'^\sharp_u) \in R$.

Conversely, assume that $m_2 \xrightarrow{t} m'_2$. We denote $m_2 = m_1 \cup m_3$ and $m'_2 = m'_1 \cup m'_3$ where m_3, m'_3 denote respectively the projections of m_2 and m'_2 on P_u. In particular, as t can fire, we have $m_1 \xrightarrow{t} m'_1$. So, there exists a reachable marking m'_b of Timed($\mathcal{N}_B$) such that $(m'_b, m'_1) \in R_{B,1}$ and there exists δ such that $m_b \xrightarrow{\delta} m_b + \delta \xrightarrow{t} m'_b$. In particular, δ does not violate any urgency constraints in the bounded part of the net.

Now, $\mathcal{N}'$ is a TPN with restricted constraints. First, this means that all urgency constraints are in the bounded part of $\mathcal{N}$. Hence, $m \xrightarrow{\delta} m + \delta$ does not violate any urgency constraints. Now, we want to show that $m + \delta \xrightarrow{t} m'$ is possible in $\mathcal{N}'$, for some m'_u with $m' = m'_b \cup m'_u$ and $(m'_u)^\sharp = m'_3$. First, because $(m, m_2) \in R$, with $m_2 = m_1 \cup m_3$, we have $m^\sharp(p) = m_3(p) \geq 1$ for all $p \in P_u \cap {}^\bullet t$. Also, we have trivially that $m^\sharp(p) = m^\sharp_b(p) \geq 1$ for all $p \in P_B \cap {}^\bullet t$ as t is enabled from m_1, and $(m_b, m_1) \in R_{B,1}$. Thus t is enabled. Now, $m + \delta$ respects all the timings constraints of t: as $\mathcal{N}'$ is a TPN with restricted constraints, all constraints apply to the bounded part. Transition t is enabled from m_1, thus t can fire from $m + \delta$. For the unbounded part, firing of t can consume any token in places of $P_u \cap {}^\bullet t$ and we easily get $(m'_u)^\sharp = m'_3$. For the bounded part, we chose to consume the tokens consumed during the transition $m_b + \delta \xrightarrow{m}{}'_b$. We thus obtain $m' = m'_b + m'_u$, and $(m', m'_2) \in R$. Hence R is a bisimulation relation. $\square$