Minimal Cost Reachability/Coverability in Priced Timed Petri Nets

Minimal Cost Reachability/Coverability in Priced Timed Petri Nets

Parosh Aziz Abdulla[1] and Richard Mayr[2]

[1] Uppsala University, Sweden
[2] University of Edinburgh, UK

Abstract. We extend discrete-timed Petri nets with a cost model that assigns token storage costs to places and firing costs to transitions, and study the minimal cost reachability/coverability problem. We show that the minimal costs are computable if all storage/transition costs are non-negative, while even the question of zero-cost coverability is undecidable in the case of general integer costs.

1 Introduction

Petri nets are one of the most widely used models for the study and analysis of concurrent systems. Furthermore, several different models have been introduced in [1, 7, 14, 13, 4, 9, 6] which extend the classical model by introducing timed behaviors.

We consider *Timed Petri Nets (TPNs)* in which each token has an 'age' represented by a natural number [1, 7]. A marking of the net is a mapping which assigns a multiset of natural numbers to each place. The multiset represents the numbers and ages of the tokens in the corresponding place. Each arc of the net is equipped with an interval defined by two natural numbers (or ω). A transition may fire iff its input places have tokens with ages satisfying the intervals of the corresponding arcs. Tokens generated by transitions will have ages in the intervals of the output arcs. In fact, this model is a generalization of the one in [7], since the latter only allows generating tokens of age 0.

In parallel, there have been several works on extending the model of timed automata [2] with *prices* (*weights*) (see e.g., [3, 11, 5]). Weighted timed automata are suitable models for embedded systems, where we have to take into consideration the fact that the behavior of the system may be constrained by the consumption of different types of resources. Concretely, weighted timed automata extend classical timed automata with a cost function C that maps every location and every transition to a nonnegative integer (or rational) number. For a transition, C gives the cost of performing the transition. For a location, C gives the cost per time unit for staying in the location. In this manner, we can define, for each run of the system, the accumulated cost of staying in locations and performing transitions along the run.

We study a natural extension of TPNs, namely *Priced TPNs (PTPNs)*. We allow the cost function to map transitions and places of the Petri net into vectors of integers (of some given length k). Again, for a transition, C gives the cost of performing the transition; while for a place, C gives the cost per time unit per token in the place. We consider the *cost-optimal problem* for PTPNs where, given an *initial marking* M_0 and a set F of *final markings*, the task is to compute the minimal accumulated cost of a run that reaches F from M_0. We consider two variants of the problem: the *reachability* problem in which F is a single marking; and the *coverability* problem in which F is an upward closed set of markings. Since the set of costs within which we can reach a

set F from a set M_0 is upward closed (regardless of the form of F), the cost-optimality problem can be reduced, using the construction of Valk and Jantzen [17], to the *cost threshold* problem. In the latter, we are given a cost vector v, and we want to check whether it is possible to reach F from M_0 with a cost that does not exceed v.

We also consider two models related to PTPNs. The first, called *Priced Petri Nets (PPNs)*, is a priced extension of classical (untimed) Petri nets, and is a special case of PTPNs. The other model is an (interesting and nontrivial) extension of classical Petri nets, in which a fixed place p and a fixed transition t are connected by a so called *inhibitor arc*. In this case, the transition t can fire only if p is empty. It has been shown that the reachability problem for Petri nets with one inhibitor arc is decidable [15].

For the above mentioned models, we show a number of (un)decidability results. First, we recall that the reachability problem is undecidable for TPNs [16], which immediately implies that the cost threshold reachability problem is undecidable for PTPNs. With this undecidability result in mind, the two natural (and simpler) problems to consider are the cost threshold coverability problem for PTPNs, and the cost threshold reachability problem for PPNs. We prove that cost threshold coverability problem is decidable for PTPNs with non-negative costs (where all components of the cost vectors are non-negative). We show that this gives in a straightforward manner also a proof of the decidability of the coverability problem for Petri nets with one inhibitor arc. Furthermore, we show that the cost threshold reachability problem for PPNs and the reachability problem for Petri nets with one inhibitor arc are reducible to each other. These results show a close (and surprising) relationship between our models and that of Petri nets with one inhibitor arc. Finally, we show that if we allow negative costs then even the cost threshold coverability problem for PPNs becomes undecidable.

2   Preliminaries

2.1   Priced Timed Petri Nets

The timed Petri net model (TPN) defined by Escrig et al. [7, 16] is an extension of Petri nets where tokens have integer ages measuring the time since their creation, and transition arcs are labeled with time-intervals (whose bounds are natural numbers or ω) which restrict the ages of tokens which can be consumed and produced. We extend this model to priced timed Petri nets (PTPN) by assigning multidimensional costs to both transitions (action costs) and places (storage costs). Each firing of a discrete transition costs the assigned cost vector. The cost of a timed transition depends on the marking. Storing k_1 tokens for k_2 time units on a place with cost vector v costs $k_1 * k_2 * v$.

Let $\mathbb{N}$ denote the non-negative integers and $\mathbb{N}^k$ and $\mathbb{N}_\omega^k$ the set of vectors of dimension k over $\mathbb{N}$ and $\mathbb{N} \cup \{\omega\}$, respectively ($\omega$ represents the first limit ordinal). We use a set *Intrv* of intervals $\mathbb{N} \times \mathbb{N}_\omega$. We view a multiset M over A as a mapping $M : A \mapsto \mathbb{N}$.

Given a set A with an ordering $\preceq$ and a subset $B \subseteq A$, B is said to be *upward closed* in A if $a_1 \in B, a_2 \in A$ and $a_1 \preceq a_2$ implies $a_2 \in B$. Given a set $B \subseteq A$, we define the *upward closure* $B \uparrow$ to be the set $\{a \in A \mid \exists a' \in B : a' \preceq a\}$.

Definition 1. *A* Priced Timed Petri Net (PTPN) *is a tuple* $N = (P, T, In, Out, C)$ *where* P *is a finite set of places,* T *is a finite set of transitions,* In, Out *are partial functions from* $T \times P$ *to* $Intrv$ *and* $C : P \cup T \to \mathbb{Z}^k$ *is the cost function assigning*

(multidimensional, and possibly also negative) firing costs to transitions and storage costs to places.

If $In(t, p)$ (respectively $Out(t, p)$) is defined, we say that p is an *input (respectively output) place* of t. Let max denote the maximal finite number $\in \mathbb{N}$ appearing on the time intervals of the given PTPN.

A *marking M* of N is a finite multiset over $P \times \mathbb{N}$. It defines the numbers and ages of tokens in each place in the net. We identify a token in a marking M by the pair (p, x) representing its place and age in M. Then, $M((p, x))$ defines the number of tokens with age x in place p. Abusing notation, we define, for each place p, a multiset $M(p)$ over $\mathbb{N}$ where $M(p)(x) = M((p, x))$. We sometimes denote multisets as lists. For a marking M of the form $[(p_1, x_1), \ldots, (p_n, x_n)]$ we use M^{+1} to denote the marking $[(p_1, x_1 + 1), \ldots, (p_n, x_n + 1)]$. For PTPN, let $\leq$ denote the partial order on markings given by multiset-inclusion.

Transitions. We define two transition relations on the set of markings: timed and discrete. The *timed transition relation* increases the age of each token by one. Formally, $M_1 \rightarrow_{time} M_2$ iff $M_2 = M_1^{+1}$.

We define the *discrete transition relation* $\rightarrow_D$ as $\bigcup_{t \in T} \longrightarrow_t$, where $\longrightarrow_t$ represents the effect of *firing* the discrete transition t. More precisely, $M_1 \longrightarrow_t M_2$ if the set of input arcs $\{(p, \mathcal{I}) | In(t, p) = \mathcal{I}\}$ is of the form $\{(p_1, \mathcal{I}_1), \ldots, (p_k, \mathcal{I}_k)\}$, the set of output arcs $\{(p, \mathcal{I}) | Out(t, p) = \mathcal{I}\}$ is of the form $\{(q_1, \mathcal{J}_1), \ldots, (q_\ell, \mathcal{J}_\ell)\}$, and there are multisets $b_1 = [(p_1, x_1), \ldots, (p_k, x_k)]$ and $b_2 = [(q_1, y_1), \ldots, (q_\ell, y_\ell)]$ over $P \times \mathbb{N}$ such that the following holds:
- $b_1 \leq M_1$ and $M_2 = (M_1 - b_1) + b_2$
- $x_i \in \mathcal{I}_i$, for $i : 1 \leq i \leq k$ and $y_i \in \mathcal{J}_i$, for $i : 1 \leq i \leq \ell$

We say that t is *enabled* in M if such a b_1 exists. A transition t may be fired only if for each incoming arc, there is a token with the right age in the corresponding input place. These tokens will be removed when the transition is fired. The newly produced tokens have ages which are chosen nondeterministically from the relevant intervals on the output arcs of the transition.

We write $\longrightarrow = \rightarrow_{time} \cup \rightarrow_D$ to denote all transitions, $\overset{*}{\longrightarrow}$ to denote the reflexive-transitive closure of $\longrightarrow$ and $\rightarrow_D^+$ to denote the transitive closure of $\rightarrow_D$. It is easy to extend $\overset{*}{\longrightarrow}$ to sets of markings. We define $Reach(M) := \{M' | M \overset{*}{\longrightarrow} M'\}$ as the set of markings reachable from M.

The cost of computations. A computation $\sigma := M_1 \longrightarrow M_2 \longrightarrow \ldots \longrightarrow M_n$ is a sequence of transitions, and also denoted by $M_1 \overset{\sigma}{\longrightarrow} M_n$. The cost of a discrete transition t is defined as $Cost(M \longrightarrow_t M') := C(t)$ and the cost of a timed transition is defined as $Cost(M \rightarrow_{time} M^{+1}) := \sum_{p \in P} |M(p)| * C(p)$. The cost of a computation σ is the sum of all transition costs, i.e., $Cost(\sigma) := \sum_{i=1}^{n-1} Cost(M_i \longrightarrow M_{i+1})$.

If the prices are ignored in PTPN then the model becomes equivalent to the timed Petri nets of [7, 16], except that we also allow the creation of tokens with nonzero ages.

2.2 Priced Petri Nets.

Priced Petri Nets (PPN) are a simple extension of standard Petri nets (i.e., without token clocks and time constraint arcs) by adding prices and transition delays. Later we will

show that PPN are a weaker model than PTPN (Lemma 3), but most undecidability results hold even for the weaker PPN model (Theorem 14).

Definition 2. *A* Priced Petri Net (PPN) *is a tuple* $N = (P, T, T_0, T_1, In, Out, C)$ *where* P *is a finite set of places,* $T = T_0 \uplus T_1$ *is a disjoint union of the sets of instantaneous transitions and timed transitions,* $In, Out : T \rightarrow \{0, 1\}^P$, *and* $C : P \cup T \rightarrow \mathbb{Z}^k$ *is the cost function assigning (multidimensional) firing costs to transitions and storage costs to places.*

The markings $M : P \rightarrow \mathbb{N}$ and the firing rules are exactly as in standard Petri nets. Transition t is enabled at marking M iff $M \geq In(t)$ (componentwise), and firing t yields the marking $M - In(t) + Out(t)$.

The cost of an instantaneous transition $t \in T_0$ is defined as $Cost(M_1 \longrightarrow_t M_2) := C(t)$ and the cost of a timed transition $t \in T_1$ is defined by $Cost(M_1 \longrightarrow_t M_2) := C(t) + \sum_{p \in P} M(p) * C(p)$. As before, the cost of a computation is the sum of all transition costs in it.

2.3 The Priced Reachability/Coverability Problem

We study the problem of computing the minimal cost for reaching a marking in a given target set from the initial marking.

COST-THRESHOLD

Instance: A PTPN (or PPN) N with an initial marking M_0, a set of final markings F and a vector $v \in \mathbb{N}_\omega^k$.

Question: Does there exist a marking $M_f \in F$ and a computation $M_0 \xrightarrow{\sigma} M_f$ s.t. $Cost(\sigma) \leq v$?

We call this problem cost-threshold-coverability if F is upward-closed, and cost-threshold-reachability if F is a single marking, i.e., $F = \{M_f\}$ for a fixed marking M_f.

Lemma 3. *The cost threshold reachability/coverability problem for PPN is polynomial time reducible to the cost threshold reachability/coverability problem for PTPN.*

If all costs are non-negative (i.e., in $\mathbb{N}^k$ rather than $\mathbb{Z}^k$) then the standard componentwise ordering on costs is a well-order and thus every upward-closed set of costs has finitely many minimal elements. Furthermore, if we have a positive instance of cost-threshold with some allowed cost v then any modified instance with some allowed cost $v' \geq v$ will also be positive. Thus the set of possible costs in the cost-threshold problem is upward-closed. In this case the Valk-Jantzen Theorem [17] implies that the set of minimal possible costs can be computed iff the Cost-Threshold problem is decidable.

Theorem 4. *(Valk & Jantzen [17])   Given an upward-closed set $V \subseteq \mathbb{N}^k$, the finite set V_{min} of minimal elements of V is computable iff for any vector $\boldsymbol{v} \in \mathbb{N}_\omega^k$ the predicate $\boldsymbol{v}\!\downarrow \cap\, V \neq \emptyset$ is decidable.*

COMPUTING MINIMAL POSSIBLE COSTS

Instance: A PTPN (or PPN) N with $C : P \cup T \rightarrow \mathbb{N}^k$, initial marking M_0, and a set of final markings F.

Question: Compute the minimal possible costs of reaching F, i.e., the finitely many minimal elements of $\{v \in \mathbb{N}^k \,|\, \exists M_f \in F, \sigma.\, M_0 \xrightarrow{\sigma} M_f \wedge Cost(\sigma) \leq v\}$

2.4 Petri Nets with Control-states and Petri Nets with One Inhibitor Arc

Several other versions of Petri nets will be used in our constructions. We define Petri nets with an extra finite control. This does not increase their expressive power, since they can easily be encoded into standard Petri nets with some extra places. However, for some constructions, we need to make the distinction between infinite memory and finite memory in Petri nets explicit.

A *Petri net with control-states (cPN)* is a tuple $N = (P, Q, T, In, Out)$ where P is a finite set of places, Q a finite set of control-states, T a finite set of transitions and $In, Out : T \rightarrow Q \times \mathbb{N}^P$.

A marking is a tuple (q, M) where $q \in Q$ and $M : P \rightarrow \mathbb{N}$. A transition t with $In(t) = (q_1, M_1)$ and $Out(t) = (q_2, M_2)$ is enabled at marking (q, M) iff $q_1 = q$ and $M_1 \leq M$. If t fires then the resulting marking is $(q_2, M - M_1 + M_2)$.

The reachability problem for Petri nets is decidable [12] and a useful generalization to sets of markings was shown by Jančar [10].

Theorem 5. *([10]) Given a cPN, we can use a simple constraint logic to describe properties of markings (q, M). Any formula Φ in this logic is a boolean combination of predicates of the following form: $q = q_i$ (the control-state is q_i), $M(p_i) \geq k$, or $M(p_i) \leq k$, where $k \in \mathbb{N}$. In particular, the logic can describe all upward-closed sets of markings. Given an initial marking and a constraint logic formula Φ, it is decidable if there exists a reachable marking that satisfies Φ.*

A *Petri Net with One Inhibitor Arc* [15] is defined as an extension of cPN. We fix a place p and a transition t, which are connected by a so-called inhibitor arc (p, t), and modify the firing rule for t such that t can only fire if place p is empty. Decidability of the reachability problem for Petri nets with one inhibitor arc has been shown in [15]. This result can be extended to the case where one place inhibits several transitions.

Lemma 6. *The reachability problem for Petri nets with many inhibitor arcs $(p, t_1), \ldots, (p, t_k)$ which all connect to the same place p can be reduced to the reachability problem for Petri nets with just one inhibitor arc.*

3 Decidability for Non-negative Costs

Theorem 7. *The cost-threshold coverability problem is decidable for PTPN with non-negative costs.*

Consider an instance of the problem. Let $N = (P, T, In, Out, C)$ be a PTPN with $C(P \cup T) \subseteq \mathbb{N}^k$, M_0 the initial marking, F an upward-closed target set (represented by the finite set F_{min} of its minimal elements) and $v = (v_1, \ldots, v_k) \in \mathbb{N}_\omega^k$.

First, for every i, if $v_i = \omega$ then we replace v_i by 0 and set the i-th component of the cost function C of N to 0, too. This does not change the result of the cost-threshold problem. So we can assume without restriction that $v = (v_1, \ldots, v_k) \in \mathbb{N}^k$ and let $b := \max_{1 \leq i \leq k} v_i \in \mathbb{N}$. Let $max \in \mathbb{N}$ be the maximal finite constant appearing on time intervals of the PTPN N.

Our proof has two parts. In part 1 we construct a cPN N' that simulates the behavior of

the PTPN N w.r.t. discrete transitions. In part 2 we define an operation g on markings of N' which encodes the effect of timed transitions in N.

Construction (part 1). We now construct the cPN $N' = (P', Q, T', In', Out')$ that encodes discrete transitions of N.

The set of places $P = \{p_1, \ldots, p_n\}$ of N can be divided into two disjoint subsets $P = P_1 \uplus P_0$ where $\forall p \in P_0. \, C(p) = 0$ and $\forall p \in P_1. \, C(p) > 0$. We call the places in P_0 *free-places* and the places in P_1 *cost-places*. Let $m := |P_1|$. Without restriction let $p_1, \ldots, p_m$ be cost-places and $p_{m+1}, \ldots, p_n$ be free-places.

The places of N' are defined as $P' := \{p(j, x) \mid p_j \in P \land 0 \le x \le max + 1\}$. The index x is used to encode the age of the token. So if in N there is a token of age x on place p_j then in N' there is a token on place $p(j, x)$. The number of tokens on place $p(j, max + 1)$ in N' encodes the number of tokens with ages $> max$ on p_j in N. This is because token ages $> max$ cannot be distinguished from each other in N. In the following we always consider equivalence classes of markings of N by identifying all token ages $> max$.

The set of control-states Q of N' is defined as a tuple $Q = Q' \times R$, where $Q' = \{0, \ldots, b\}^{\{1, \ldots, m\} \times \{0, \ldots, max+1\}}$. The intuition is that for every cost-place $p(j, x)$ of N' the number of tokens on $p(j, x)$ is partly stored in the control-state $q(j, x)$ of Q' up-to a maximum of b, while only the rest is stored on the place $p(j, x)$ directly. So if place $p(j, x)$ contains $y \ge b$ tokens then this is encoded as just $y - b$ tokens on place $p(j, x)$ and a control-state q with $q(j, x) = b$. If place $p(j, x)$ contains $y \le b$ tokens then this is encoded as 0 tokens on place $p(j, x)$ and a control-state q with $q(j, x) = y$. Furthermore, $R = \{0, \ldots, b\}^k$ and is used to store the remaining maximal allowed cost.

A marking M' of N' is given as $M' = ((q, r), M'')$ where $(q, r) \in Q$ and $M'' : \{1, \ldots, n\} \times \{0, \ldots, max + 1\} \to \mathbb{N}$. For every $r \in R$ we define a mapping f_r of markings of N to markings of N'. Given a marking M of N, let $M' = ((q, r), M'') := f_r(M)$ be defined as follows.

- $q(j, x) = \min\{M((p_j, x)), b\}$ for $1 \le j \le m, 0 \le x \le max$
- $M''(j, x) = \max\{0, M((p_j, x)) - b\}$ for $1 \le j \le m, 0 \le x \le max$
- $q(j, max + 1) = \min\{\sum_{x > max} M((p_j, x)), b\}$ for $1 \le j \le m$
- $M''(j, max + 1) = \max\{0, \sum_{x > max} M((p_j, x)) - b\}$ for $1 \le j \le m$
- $M''(j, x) = M((p_j, x))$ for $j > m, 0 \le x \le max$
- $M''(j, max + 1) = \sum_{x > max} M((p_j, x))$ for $j > m$

This ensures that token numbers up-to b on cost-places are encoded in the control-state and only the rest is kept on the cost-places themselves. Free-places are unaffected. The parameter $r \in R$ assigns the allowed remaining cost, which is independent of M, but also stored in the finite control of M'. The initial marking M_0' of N' is defined as $M_0' = f_v(M_0)$. The upward-closed set of final markings F of N is represented by the finite set F_{min} of its minimal elements. We also represent the upward-closed set of final markings F' of N' by the finite set F'_{min} of its minimal elements. Let $F'_{min} := \bigcup_{0 \le r \le v} f_r(F_{min})$, i.e., we take the union over all possibilities of remaining allowed (unused) cost r.

The Petri net N' only encodes the effect of discrete transitions of the PTPN N' (the effect of timed transitions will be handled separately). The set T' of transitions of N' is defined as follows. Let $t \in T$. We say that a pair of functions $I, O : \{1, \ldots, n\} \times \{0, \ldots, max + 1\} \to \{0, 1\}$ are compatible with t iff $\forall p_j \in P = \{p_1, \ldots, p_n\}$

- If $In(t, p_j)$ is defined then $\exists_{=1} x \in In(t, p_j) \cap \{0, \ldots, max + 1\}$ s.t. $I(j, x) = 1$ and $I(j, x') = 0$ for every $x' \neq x$. Otherwise $I(j, x) = 0$ for all x.
- If $Out(t, p_j)$ is defined then $\exists_{=1} x \in Out(t, p_j) \cap \{0, \ldots, max + 1\}$ s.t. $O(j, x) = 1$ and $I(j, x') = 0$ for every $x' \neq x$. Otherwise $O(j, x) = 0$ for all x.

The set of all these compatible pairs of functions represents all possible ways of choosing the age of tokens consumed/produced by t out of the specified time intervals. All ages $> max$ are lumped together under $max + 1$, since they are indistinguishable in N.

Then for every combination of matrices $v_1, v_3 \in \{0, 1\}^{\{1, \ldots, m\} \times \{0, \ldots, max+1\}}$ and $v_2, v_4 \in \{0, 1\}^{\{1, \ldots, n\} \times \{0, \ldots, max+1\}}$ and every pair of functions $I, O : \{1, \ldots, n\} \times \{0, \ldots, max+1\} \rightarrow \{0, 1\}$ which are compatible with t and every control-state $(q, r) \in Q$ we have a transition $t' \in T'$ with $In(t') = ((q, r), I')$ and $Out(t') = ((q', r'), O')$ iff the following conditions are satisfied.

- $(v_1, \mathbf{0}) + v_2 = I$ and $(v_3, \mathbf{0}) + v_4 = O$
- $q \geq v_1$
- $q' = q - v_1 + v_3 \in Q'$ (in particular, every component of q' is $\leq b$).
- $r' = r - C(t) \geq \mathbf{0}$ (the cost of t is deducted from the remaining allowed cost).
- $I'(p(j, x)) = v_2(j, x)$ for $1 \leq j \leq n, 0 \leq x \leq max + 1$
- $O'(p(j, x)) = v_4(j, x)$ for $1 \leq j \leq n, 0 \leq x \leq max + 1$

The choice of the v_1, v_2, v_3, v_4 encodes all possible ways of splitting the effect of the transition on places between the part which is encoded in the finite control and the part remaining on the places. Consume v_1 from the finite control and v_2 from the real places. Produce v_3 in the finite control and v_4 on the real places. Note that v_1, v_3 have a smaller dimension than v_2, v_4, because they operate only on cost-places. So v_1, v_3 are extended from dimension $\{1, \ldots, m\} \times \{0, \ldots, max + 1\}$ to $\{1, \ldots, n\} \times \{0, \ldots, max + 1\}$ by filling the extra entries (the free places) with zero to yield $(v_1, \mathbf{0})$ and $(v_3, \mathbf{0})$ which can be combined with v_2, v_4. The transitions cannot guarantee that cost-places are always properly stored up-to b in the finite control. E.g., if $b = 7$ there could be reachable markings where a cost-place holds 5 tokens, but only 3 are encoded in the finite control while 2 remain on the place. However, proper encodings as described above are always possible. Our constructions will ensure that such non-standard encodings as in this example do not change the final result. Intuitively, the reason is the following. Non-standard encodings differ from proper encodings by having more tokens on the real cost-places and fewer encoded in the finite control. However, at key points (i.e., where timed transitions happen) our constructions enforce that the real cost-places are empty, thus filtering out the computations with non-standard encodings. Furthermore, by forcing the cost-places to be empty, we ensure that all contents of cost-places are encoded in the finite control and that they are below the bound b. This makes it possible to deduct the correct place-costs during timed transitions (see Construction (part 2) below).

Lemma 8. *Let M_1, M_2 be markings of N. Then there is a computation σ using only discrete transitions s.t. $M_1 \overset{\sigma}{\longrightarrow} M_2$ with $Cost(\sigma) \leq v$ if and only if in N' there are computations σ' where $f_r(M_1) \overset{\sigma'}{\longrightarrow} f_{r'}(M_2)$ for every r, r' with $v \geq r$ and $r' = r - Cost(\sigma) \geq \mathbf{0}$.*

Proof. Directly from the construction of N' and induction over the lengths of σ, σ'. □

Construction (part 2). The cPN N' only encodes the behavior of N during discrete transitions. It does not encode the effect of timed transitions, nor the place-costs of delays. A crucial observation is that, in computations of N, since the maximal allowed cost v is bounded by b (componentwise), the maximal number of tokens on any cost-place must be $\leq b$ before (and thus also after) every timed transition, or else the cost would exceed the limit v. Since in N' token numbers on cost-places are encoded in the finite control up-to b, we can demand without restriction that in N' all cost-places are empty before and after every simulated timed transition. These simulated timed transitions are not encoded into N' directly, but handled in the following construction.

We define a (non-injective) function g on markings of N' which encodes the effect of a timed transition. For technical reasons we restrict the domain of g to markings of N' which are empty on all cost-places. Let $((q, r), M)$ be a marking of N' where M is empty on all cost-places. The marking $((q', r'), M') := g(((q, r), M))$ is defined by

- $q'(j, 0) = 0$ for $1 \leq j \leq m$. (No token has age 0 after time has passed.)
- $q'(j, x + 1) = q(j, x)$ for $1 \leq j \leq m$, $0 \leq x < max$. (All tokens age by 1.)
- $q'(j, max + 1) = q(j, max) + q(j, max + 1)$ for $1 \leq j \leq m$.
- $M'(j, 0) = 0$ for $1 \leq j \leq n$. (No token has age 0 after time has passed.)
- $M'(j, x + 1) = M(j, x)$ for $1 \leq j \leq n$, $0 \leq x < max$. (All tokens age by 1.)
- $M'(j, max + 1) = M(j, max) + M(j, max + 1)$ for $1 \leq j \leq n$.
- $r' = r - \sum_{j=1}^{m} \sum_{x=0}^{max+1} q(j, x) * C(p_j)$ (Deduct the correct place costs).
- $r \leq v$. (Costs above the limit v are not allowed.)
- $M(j, x) = 0$ for $1 \leq j \leq m$. (All cost-places are empty in M and thus also in M'.)

The last two conditions ensure that g is only defined for $r \leq v$ and markings where all cost-places are empty. Also g is not injective, since ages $> max$ are encoded as $max+1$.

Lemma 9. *Let M_1, M_2 be markings of N. Then $M_1 \rightarrow_{time} M_2$ with $Cost(M_1 \rightarrow_{time} M_2) \leq v$ if and only if in N' we have $f_{r'}(M_2) = g(f_r(M_1))$ for every r, r' with $v \geq r$ and $r' = r - Cost(M_1 \rightarrow_{time} M_2) \geq \mathbf{0}$.*

Proof. Since $Cost(M_1 \rightarrow_{time} M_2) \leq v$, the content of the cost-places in M_1 and M_2 is below b. Thus the cost places are completely encoded in the finite control in $f_r(M_1)$ and $f_{r'}(M_2)$, while the cost-places themselves are empty. Therefore the remaining cost r' is computed correctly, and depends only on the finite control. The rest follows directly from the definition of N' and g. □

Lemma 10. *The following three conditions are equivalent.*

1. *The PTPN N with M_0, F and v is a positive instance of cost-threshold.*
2. *There exist markings $M_1, \ldots, M_{j-1}$ and $A_1, \ldots, A_j$ of N with $M_i \xrightarrow{\sigma_i} A_{i+1}$ and $A_i \rightarrow_{time} M_i$ and $A_j \in F$ where σ_i consists only of discrete transitions and $\sum_{i=0}^{j-1} Cost(\sigma_i) + \sum_{i=1}^{j-1} Cost(A_i \rightarrow_{time} M_i) \leq v$.*
3. *There exist markings $M'_0 = f_v(M_0)$ and $M'_1, \ldots, M'_{j-1}$ and $A'_1, \ldots, A'_j$ of N' with $M'_i \xrightarrow{\sigma'_i} A'_{i+1}$ and $M'_i = g(A'_i)$ and $A'_j \in F'$.*

Proof. Conditions 1. and 2. are equivalent by definition. For the equivalence of 2. and 3. let $r_0 = v$, $a_{i+1} = r_i - Cost(\sigma_i)$ and $r_i = a_i - Cost(A_i \rightarrow_{time} M_i)$. Then $M'_i = f_{r_i}(M_i)$ and $A'_i = f_{a_i}(A_i)$. The proof follows directly from Lemmas 8 and 9.

In the following we consider an extended notion of computations of N' which contain both normal transitions of N' and applications of function g.

Let F^i be the set of markings M' of N' where: (1) M' can reach F' via an extended computation that starts with an application of function g and contains i applications of function g (i.e., i encoded timed transitions), and (2) M' is 0 on all cost-places, i.e., $M'((j, x)) = 0$ for $1 \leq j \leq m$ and all x. The set G^i is defined analogously, except that the extended computation must start with a normal transition of N'. We have $G^0 = \{M' = (\mathbf{0}, \boldsymbol{x}) \mid M' \xrightarrow{*} M'' \in F'\}$ and $G^i = \{M' = (\mathbf{0}, \boldsymbol{x}) \mid M' \xrightarrow{*} M'' = (\mathbf{0}, \boldsymbol{x}') \in F^i\}$ for $i > 0$, and $F^{i+1} = g^{-1}(G^i)$ for $i \geq 0$.

Since F is upward-closed w.r.t. the (multiset-inclusion) order on markings of N, the set F' is upward-closed w.r.t. the order on markings of N'. Therefore, all sets F^i and G^i are upward-closed w.r.t. the free-places (i.e., their projection on the free-places), by monotonicity of Petri nets and the monotonicity of function g. Furthermore, the markings in F^i and G^i are all zero on the cost-places. So F^i and G^i can be characterized by their finitely many minimal elements. The finitely many minimal elements of G^0 (resp. G^i) can be computed from F' (resp. F^i) by generalized Petri net reachability (Theorem 5) and the Valk-Jantzen Theorem (Theorem 4) applied to the Petri net N'.

The step from G^i to F^{i+1} encodes the effect of a timed transition. Let G^i_{min} be the finite set of minimal elements of G^i. We compute the finite set F^{i+1}_{min} of minimal elements of F^{i+1} as $F^{i+1}_{min} := g^{-1}(G^i_{min})$. Even though g is not injective, $g^{-1}(G^i_{min})$ is still finite, because G^i_{min} is finite and every marking in G^i_{min} contains only finitely many tokens. Finally, let $H^l := \bigcup_{i \leq l} F^i$. Now we can prove the main theorem.

Proof. (of Theorem 7) Given the instance of cost-threshold, we construct the Petri N' and the sets of markings H^l for $l = 0, 1, 2, \ldots$. The markings in H^l are all empty on the cost-places, but the sets H^l are upward-closed w.r.t. the free-places. Thus, by Dickson's Lemma [8], the construction converges at H^y for some finite index y. By the construction of F^i we obtain that the set H^y contains all markings which are empty on the cost-places and which can reach F' via some extended computation that begins with an application of function g and costs at most v.

Thus, by Lemma 10, the instance of cost-threshold is positive iff $M_0' = f_v(M_0)$ can reach $H^y \cup F'$ by normal transitions in Petri net N'. This is decidable by Theorem 5. $\quad\square$

It was shown in [15] that reachability, and thus also coverability, is decidable for Petri nets with one inhibitor arc. However, our result on PTPN also yields a more direct proof of decidability of the coverability problem.

Corollary 11. *Given a Petri net with one inhibitor arc, an initial marking M_0 and an upward-closed set of final markings F, it is decidable if $M_0 \to^* F$.*

Proof. We reduce the coverability problem for a Petri net with one inhibitor arc N to the cost threshold problem for a PPN N' which is constructed as follows. Let (p, t) be the inhibitor arc. We remove the inhibitor arc, make t a timed transition and all other transitions instantaneous transitions. Place p has cost 1 while all other places and transitions have cost 0. In N', any computation with cost 0 has the property that transition t is only used if place p is empty. Therefore, in N' the set F is reachable with cost 0 iff the set F is reachable in N.

Furthermore, by Lemma 3, the cost threshold problem for PPN is reducible to the cost threshold problem for PTPN. Since F is upward-closed and all costs are non-negative the problem is decidable by Theorem 7. □

Now we consider the cost-threshold reachability problem. This is the case where F is not upward-closed, but a fixed marking, i.e., $F = \{M_f\}$.

Theorem 12. *The cost-threshold reachability problem is undecidable for PTPN, even if all costs are zero.*

Proof. Directly from the undecidability of the reachability problem for TPN [16]. □

However, for the simpler PPN model, the cost-threshold reachability problem is equivalent to the reachability problem for Petri nets with one inhibitor arc. The reduction from Petri nets with one inhibitor arc to the cost-threshold reachability problem for PPN is similar to the construction in Corollary 11. Now we show the other direction.

Theorem 13. *The cost-threshold reachability problem for PPN is decidable.*

Proof. Consider a PPN $N = (P, T, T_0, T_1, In, Out, C)$, cost v, initial marking M_0 and a target marking M_f. We construct a Petri net $N' = (P', Q, T', In', Out')$, with inhibitor arcs (p_0, t) (for many different transitions t, but always the same place p_0), an initial marking M'_0 and markings $M'_f, \widehat{M_f}$ s.t. $M'_0 \rightarrow^* M'_f \rightarrow \widehat{M_f}$ in N' iff N is a positive instance of cost-threshold.

Construction. Let $v = (v_1, \ldots, v_k) \in \mathbb{N}_\omega^k$. First, for every i, if $v_i = \omega$ then we replace v_i by 0 and set the i-th component of the cost function C of N to 0, too. This does not change the result of the cost-threshold problem. Thus we can assume without restriction that $v = (v_1, \ldots, v_k) \in \mathbb{N}^k$. Let $b := \max_{1 \leq i \leq k} v_i \in \mathbb{N}$.

The set of places $P = \{p_1, \ldots, p_n\}$ of N can be divided into two disjoint subsets $P = P_1 \uplus P_0$ where $\forall p \in P_0. C(p) = 0$ and $\forall p \in P_1. C(p) > 0$. We call the places in P_0 *free-places* and the places in P_1 *cost-places*. Let $m := |P_1|$. Without restriction let $p_1, \ldots, p_m$ be cost-places and $p_{m+1}, \ldots, p_n$ be free-places.

The set of control-states Q of N' is defined as a tuple $Q = Q' \times R$, where $Q' = \{0, \ldots, b\}^m$. The intuition is that for every cost-place p_j the number of tokens on p_j is partly stored in the j-th component of Q' up-to a maximum of b, while only the rest is stored on the place directly. So if place p_j contains $x \geq b$ tokens then this is encoded as just $x - b$ tokens on place p_j and a control-state where the j-th component is b. If place p_j contains $x \leq b$ tokens then this is encoded as just 0 tokens on place p_j and a control-state where the j-th component is x. Furthermore, $R = \{0, \ldots, b\}^k$ and is used to store the remaining maximal allowed cost of the computation.

Let $P' := P \cup \{p_0\}$. The extra place p_0 will be used to store the sum of all cost-places. So every marking M' of N' will satisfy the invariant $M'(p_0) = \sum_{p \in P_1} M'(p)$. In particular $M'(p_0) = 0 \Leftrightarrow \forall p \in P_1. M'(p) = 0$.

The set T' of transitions of N' is defined as follows. Let $t \in T_0$. Then for every combination of vectors $v_1, v_3 \in \mathbb{N}^m$ and $v_2, v_4 \in \mathbb{N}^n$ and every control-state $(q, r) \in Q$ we have a transition $t' \in T'$ with $In(t') = ((q, r), I)$ and $Out(t') = ((q', r'), O)$ iff the following conditions are satisfied. (The intuition for the vectors v_1, v_2, v_3, v_4 is

to model all possible ways of splitting the consumption/production of tokens by the transition between tokens encoded in the finite control and tokens present on the real places; similarly as in Construction (part 1) of the proof of Theorem 7.)

- $(v_1, \mathbf{0}) + v_2 = In(t)$, and $(v_3, \mathbf{0}) + v_4 = Out(t)$
- $q \geq v_1$
- $q' = q - v_1 + v_3 \leq (b, \ldots, b)$
- $r' = r - C(t) \geq \mathbf{0}$
- $I(p_j) = v_2(p_j)$ for $j \geq 1$ and $I(p_0) = \sum_{i=1}^m v_2(p_i)$
- $O(p_j) = v_4(p_j)$ for $j \geq 1$ and $O(p_0) = \sum_{i=1}^m v_4(p_i)$

Let $t \in T_1$. Then for every combination of vectors $v_1, v_3 \in \mathbb{N}^m$ and $v_2, v_4 \in \mathbb{N}^n$ and every control-state $(q, r) \in Q$ we have a transition $t' \in T'$ with $In(t') = ((q, r), I)$ and $Out(t') = ((q', r'), O)$ and inhibitor arc (p_0, t') iff the following conditions hold.

- $(v_1, \mathbf{0}) + v_2 = In(t)$, and $(v_3, \mathbf{0}) + v_4 = Out(t)$
- $q \geq v_1$
- $q' = q - v_1 + v_3 \leq (b, \ldots, b)$
- $r' = r - C(t) - \sum_{i=1}^m q_i * C(p_i) \geq \mathbf{0}$
- $I(p_j) = v_2(p_j)$ for $j \geq 1$ and $I(p_0) = \sum_{i=1}^m v_2(p_i) = 0$
- $O(p_j) = v_4(p_j)$ for $j \geq 1$ and $O(p_0) = \sum_{i=1}^m v_4(p_i)$

Finally, for every $(q, r) \in Q'$, we add another transition t' to T with $In(t') = ((q, r), \mathbf{0})$ and $Out(t') = ((q, \mathbf{0}), \mathbf{0})$. This makes it possible to set the remaining allowed cost to zero at any time.

For every $r \in R$ we define a mapping f_r of markings of N to markings of N'. Given a marking M of N, let $M' := f_r(M)$ be defined as follows. $M' = ((q, r), M'')$ where $q_i = \min\{M(p_i), b\}$ for $1 \leq i \leq m$ and $M''(p_i) = \max\{0, M(p_i) - b\}$ for $1 \leq i \leq m$ and $M''(p_i) = M(p_i)$ for $i > m$ and $M''(p_0) = \sum_{i=1}^m M''(p_i)$. This ensures that token numbers up-to b on cost-places are encoded in the control-state and only the rest is kept on the cost-places themselves. Free-places are unaffected. The parameter $r \in R$ assigns the allowed remaining cost, which is independent of M, but also stored in the finite control of M'. The initial marking M'_0 of N' is defined as $M'_0 = f_v(M_0)$ and the final marking is defined as $\widehat{M_f} = f_{\mathbf{0}}(M_f)$.

Proof of correctness. Assume that there is a computation σ of N of the form $M_0 \to M_1 \to M_2 \to \cdots \to M_f$ such that $Cost(\sigma) \leq v$. Then there is a corresponding computation σ' in N' of the form $M'_0 \to M'_1 \to M'_2 \to \cdots \to M'_f \to \widehat{M_f}$ such that $M'_i = f_{r_i}(M_i)$, where $r_i = v - Cost(M_0 \to \cdots \to M_i)$ and $\widehat{M_f} = f_{\mathbf{0}}(M_f)$. The step $M'_f \to \widehat{M_f}$ uses the special transition that sets the remaining allowed cost to zero.

The crucial observation is that whenever a timed transition $M_i \xrightarrow{t} M_{i+1}$ is used in σ then the number of tokens on every cost-place p_j in M_i is $\leq b$, because $Cost(\sigma) \leq v$. Therefore, in M'_i every cost-place p_j is empty, since all the $\leq b$ tokens are encoded into the finite control. Thus $M'_i(p_0) = 0$ and the inhibitor arc (p_0, t) does not prevent the transition from $M'_i \xrightarrow{t} M'_{i+1}$. Furthermore, the remaining allowed cost r_i is always non-negative, because, by our assumption, $v \geq Cost(\sigma)$. Thus the cost restrictions do

not inhibit transitions in σ' either. Finally, we apply the special transition which sets the remaining allowed cost to zero and thus we reach $\widehat{M_f}$ as required.

In order to show the other direction, we need a reverse mapping g from markings M' of N' to markings M of N. For $M' = ((q, r, M'')$ we define $M = g(M')$ as follows. For cost-places p_j (with $1 \leq j \leq m$) we have $M(p_j) = M''(p_j) + q_j$. For free-places p_j (with $j > m$) we have $M(p_j) = M''(p_j)$. Assume now that we have a computation σ' of N' of the form $M'_0 \to M'_1 \to M'_2 \to \cdots \to M'_f \to \widehat{M_f}$. Without restriction we can assume that the special transition which sets the remaining allowed cost to zero is used exactly only in the last step $M'_f \to \widehat{M_f}$, because the set of possible computations is monotone increasing in the allowed remaining cost. In the special case where the remaining allowed cost is already 0 in M'_f we have $M'_f = \widehat{M_f}$. There then exists a computation σ of N of the form $M_0 \to M_1 \to M_2 \to \cdots \to M_f$ such that $M_i = g(M'_i)$ and $Cost(M_0 \to \cdots \to M_i) = v - r_i$, where $M'_i = ((q, r_i), M''_i)$ (for some $q \in Q'$). The crucial argument is about the timed-transition steps $M'_i \xrightarrow{t} M'_{i+1}$. The inhibitor arc (p_0, t) in N' ensures that $M'_i(p_0) = 0$. Thus all cost-places p_j are empty in M'_i and only the part up-to b which is encoded in the finite control part q remains. Therefore, we deduct the correct cost $C(t) + \sum_{i=1}^{m} q_i * C(p_i)$ from r_i to obtain r_{i+1} and so we maintain the above invariant by $Cost(M_0 \to \cdots \to M_{i+1}) = v - r_{i+1}$. So we obtain $Cost(M_0 \to \cdots \to M_f) = v - r_f \geq \mathbf{0}$ and thus $Cost(\sigma) \leq v$.

Finally, since all the inhibitor arcs in N' connect to the same place p_0, the reachability problem for N' can be reduced to the reachability problem for some Petri net with just one inhibitor arc by Lemma 6, and this is decidable by [15]. $\qquad \square$

4   Undecidability for Negative Costs

The cost threshold coverability problem for PTPN is undecidable if negative transition costs are allowed. This holds even for the simpler PPN and one-dimensional costs.

Theorem 14. *The cost threshold problem for PPN $N = (P, T, T_0, T_1, In, Out, C)$ is undecidable even if $C(P) \subseteq \mathbb{N}$ and $C(T) \subseteq \mathbb{Z}_{\leq 0}$ and F is upward-closed.*

Proof. We prove undecidability of the problem through a reduction from the control-state reachability problem for 2-counter machines. We recall that a 2-counter machine M, operating on two counters c_0 and c_1, is a triple (Q, δ, q_{init}), where Q is a finite set of *control states*, δ is a finite set of *transitions*, and $q_{init} \in Q$ is the *initial control-state*. A transition $r \in \delta$ is a triple (q_1, op, q_2), where op is of one of the three forms (for $i = 0, 1$): (i) $c_i ++$ which increments the value of c_i by one; (ii) $c_i --$ which decrements the value of c_i by one; or (iii) $c_i = 0$? which checks whether the value of c_i is equal to zero. A *configuration* γ of M is a triple (q, x, y), where $q \in Q$ and $x, y \in \mathbb{N}$. The configuration gives the control-state together with the values of the counters c_0 and c_1. The initial configuration c_{init} is $(q_{init}, 0, 0)$. The operational semantics of M is defined in the standard manner. In the control-state reachability problem, we are given a counter machine M and a (final) control-state q_F, and check whether we can reach a configuration of the form (q_F, x, y) (for arbitrary x and y) from γ_{init}.

Given an instance of the control-state reachability problem for 2-counter machines, we derive an instance of the cost threshold coverability problem for a PPN with only
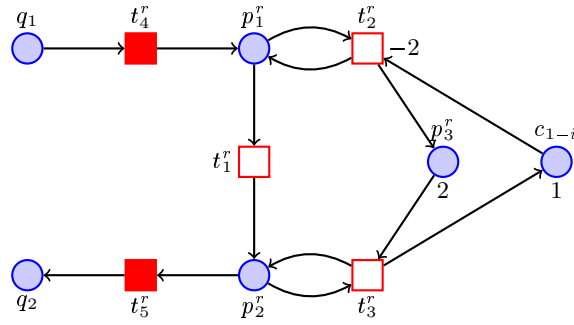
Fig. 1. Simulating zero testing in a PPN. Timed transitions are filled. The cost of a place or transition is shown only if it is different from 0.

non-negative costs on places and only non-positive costs on transitions; and where the threshold vector is given by (0) (i.e., the vector is one-dimensional, and its only component has value 0). We define the PPN $N = (P, T, T_0, T_1, In, Out, C)$ as follows. For each control-state $q \in Q$ we have a place $q \in P$. A token in the place q indicates that M is in the corresponding control-state. For each counter c_i we have a place $c_i \in P$ with $C(c_i) = 1$. The number of tokens in the place c_i gives the value of the corresponding counter. We define the set F as $(q_F, 0, 0) \uparrow$. Increment and decrement transitions are simulated in a straightforward manner. For a transition $r = (q_1, c_i++, q_2) \in \delta$ there is a transition $r \in T_0$ such that $In(r) = \{q_1\}$, $Out(r) = \{q_2, c_i\}$, and $C(t) = 0$. For a transition $r = (q_1, c_i--, q_2) \in \delta$ there is a transition $r \in T_0$ such that $In(r) = \{q_1, c_i\}$, $Out(r) = \{q_2\}$, and $C(t) = 0$. The details of simulating a zero testing transition $r = (q_1, c_i = 0?, q_2)$ are shown in Figure 1. The main idea is to put a positive cost on the counter places c_i and c_{1-i}. If the system 'cheats' and takes the transition from a configuration where the counter value c_i is positive (the corresponding place is not empty), then the transition will impose a cost which cannot be compensated in the remainder of the computation. Since the other counter c_{1-i} also has a positive cost, we will also pay an extra (unjustified) price corresponding to the number of tokens in c_{1-i}. Therefore, we use a number of auxiliary places and transitions which make it possible to reimburse unjustified cost for tokens on counter c_{1-i}. The reimbursement is carried out (at most completely, but possibly just partially) by cycling around the tokens in c_{1-i}. Concretely, we have three transitions $t_1^r, t_2^r, t_3^r \in T_0$ and two transitions $t_4^r, t_5^r \in T_1$. Furthermore, we have three extra places $p_1^r, p_2^r, p_3^r \in P$. The costs are given by $C(t_2^r) = -2$, $C(p_3^r) = 2$, $C(c_{1-i}) = C(c_{1-i}) = 1$; while the cost of the other places and transitions are all equal to 0. Intuitively, N simulates the transition r by first firing the transition t_4^r which will add a cost which is equal to the number of tokens in c_i and c_{1-i}. Notice that the (only) token in the place q_i has now been removed. This means that there is no token in any place corresponding to a control-state in M and hence the rest of the net has now become passive. We observe also that t_4^r puts a token in p_1^r. This token will enable the next phase which will make it possible to reclaim the (unjustified) cost we have for the tokens in the place c_{1-i}. Let n be the number of tokens in place c_{1-i}. Then, firing t_4^r costs n. We can now fire the transition t_2^r m times, where $m \leq n$,

thus moving m tokens from c_{1-i} to p_3^r and gaining $2m$ (i.e., paying $-2m$). Eventually, t_1^r will fire enabling t_3^r. The latter can fire k times (where $k \le m$) moving k tokens back to c_{1-i}. The places p_3^r and c_{1-i} will now contain $m - k$ resp. $n + k - m$ tokens. Finally, the transition t_5^r will fire, costing $2(m - k) + (n + k - m) = n + m - k$, moving a token to q_2, and resuming the "normal" simulation of M. The total cost ℓ for the whole sequence of transitions is $n - 2m + n + m - k = 2n - m - k$. This means $0 \le \ell$ and that $\ell = 0$ only if $k = m = n$, i.e., all the tokens of c_{1-i} are moved to p_3^r and back to c_{1-i}. In other words, we can reimburse all the unjustified cost (but not more than that). This implies correctness of the construction as follows. Suppose that the given instance of the control-state reachability problem has a positive answer. Then there is a faithful simulation in N (which will eventually put a token in the place q_F). In particular, each time we perform a transition which tests the value of counter c_i, the corresponding place is indeed empty and hence we pay no cost for it. We can also choose to reimburse all the unjustified cost paid for counter c_{1-i}. Recall that all transitions are instantaneous except the ones which are part of simulations of zero testing (the ones of the form t_4^r and t_5^r). It follows that the computation has an accumulated cost equal to 0. On the other hand, suppose that the given instance of the control-state reachability problem has a negative answer. Then the only way for N to put a token in q_F is to 'cheat' during the simulation of a zero testing transition (as described above). However, in such a case either $m < n$ or $k < n$. In either case, the accumulated cost for simulating the transition is positive. Since simulations of increment and decrement transition have zero costs, and simulations of zero testing transitions have non-negative costs, the extra cost paid for cheating cannot be recovered later in the computation. This means that the accumulated cost for the whole computation will be strictly positive, and thus we have a negative instance of the cost threshold coverability problem. $\qquad\square$

Corollary 15. *The cost threshold problem for PTPN $N = (P, T, T_0, T_1, In, Out, C)$ is undecidable even if $C(P) \subseteq \mathbb{N}$ and $C(T) \subseteq \mathbb{Z}_{\le 0}$ and F is upward-closed.*

Proof. Directly from Theorem 14 and Lemma 3. $\qquad\square$

5 Conclusion

We have considered Priced Timed Petri nets (PTPN), which is an extension of discrete-timed Petri nets with a cost model. We have shown decidability of the priced coverability problem when prices of places and transitions are given as vectors of natural numbers. On the other hand, allowing negative costs, even the priced coverability problem becomes undecidable even for the simpler model of Priced Petri Nets (PPNs) which is an extension of the the classical model of Petri nets with prices.

The (un)decidability results of can be extended in several ways, using constructions similar to the ones provided in the paper. For instance, if we consider a model where we allow control-states and where the place-costs depend on the control-state, then the coverability problem is undecidable for PPNs even if there are no costs on the transitions. In fact, the result can be shown using a simplified version of the proof of Theorem 14. The idea is to use the control-state to avoid paying wrongly-paid costs for the counter which is not currently tested for zero. Furthermore, if all place-costs are 0, then the cost threshold reachability/coverability problem can be encoded into the standard Petri

net problems, even if transition-costs can be negative. Finally, if all places have non-positive costs then everything is still decidable, even for general integer transition costs. The reason is that, instead of negative place costs, we could in every time-passing phase 'cycle' the tokens on cost-places at most once through negative transitions. Since the costs are negative, there is an 'incentive' to do this cycling fully.

A challenging problem which we are currently considering is to extend our results to the case of dense-timed Petri nets.

References

1. P. A. Abdulla and A. Nylén. Timed Petri nets and BQOs. In *Proc. ICATPN'2001: 22nd Int. Conf. on application and theory of Petri nets*, volume 2075 of *Lecture Notes in Computer Science*, pages 53 –70, 2001.
2. R. Alur and D. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.
3. R. Alur, S. L. Torre, and G. J. Pappas. Optimal paths in weighted timed automata. In *HSCC*, pages 49–62, 2001.
4. B. Berthomieu and M. Diaz. Modeling and verification of time dependent systems using time Petri nets. *IEEE Trans. on Software Engineering*, 17(3):259–273, 1991.
5. P. Bouyer, T. Brihaye, V. Bruyère, and J.-F. Raskin. On the optimal reachability problem of weighted timed automata. *Formal Methods in System Design*, 31(2):135–175, 2007.
6. F. D. J. Bowden. Modelling time in Petri nets. In *Proc. Second Australian-Japan Workshop on Stochastic Models*, 1996.
7. D. de Frutos Escrig, V. V. Ruiz, and O. M. Alonso. Decidability of properties of timed-arc Petri nets. In *ICATPN 2000*, volume 1825 of *Lecture Notes in Computer Science*, pages 187–206, 2000.
8. L. E. Dickson. Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors. *Amer. J. Math.*, 35:413–422, 1913.
9. C. Ghezzi, D. Mandrioli, S. Morasca, and M. Pezzè. A unified high-level Petri net formalism for time-critical systems. *IEEE Trans. on Software Engineering*, 17(2):160–172, 1991.
10. P. Jančar. Decidability of a temporal logic problem for Petri nets. *Theoretical Computer Science*, 74:71–93, 1990.
11. K. G. Larsen, G. Behrmann, E. Brinksma, A. Fehnker, T. Hune, P. Pettersson, and J. Romijn. As cheap as possible: Efficient cost-optimal reachability for priced timed automata. In *Proc. 13th Int. Conf. on Computer Aided Verification*, volume 2102 of *Lecture Notes in Computer Science*, 2001.
12. E. Mayr. An algorithm for the general Petri net reachability problem. *SIAM Journal of Computing*, 13:441–460, 1984.
13. P. Merlin and D. Farber. Recoverability of communication protocols - implications of a theoretical study. *IEEE Trans. on Computers*, COM-24:1036–1043, Sept. 1976.
14. R. Razouk and C. Phelps. Performance analysis using timed Petri nets. In *Protocol Testing, Specification, and Verification*, pages 561–576, 1985.
15. K. Reinhardt. Reachability in Petri nets with inhibitor arcs. In *Proc. RP08, 2nd Workshop on Reachability Problems*, 2008.
16. V. V. Ruiz, F. C. Gomez, and D. de Frutos Escrig. On non-decidability of reachability for timed-arc Petri nets. In *Proc. 8th Int. Workshop on Petri Net and Performance Models (PNPM'99), 8-10 October 1999, Zaragoza, Spain*, pages 188–196, 1999.
17. R. Valk and M. Jantzen. The residue of vector sets with applications to decidability problems in Petri nets. *Acta Inf.*, 21, 1985.

Appendix

Lemma 3. The cost threshold reachability/coverability problem for PPN is polynomial time reducible to the cost threshold reachability/coverability problem for PTPN.

Proof. Given a PPN $N = (P, T, T_0, T_1, In, Out, C)$, an initial marking M_0 a set of final markings F and a vector $v \in \mathbb{N}_\omega^k$, we construct in polynomial time a PTPN $N' = (P', T', In', Out', C')$, an initial marking M_0' and a set of final markings F' s.t. N', F', v is a positive instance of cost-threshold iff N, F, v is.
Construction. Let $P' = P \uplus \{p_0, p_1\}$, $T' = T \uplus \{t_1\}$,

For any instantaneous transition $t \in T_0$ we have: $In'(t, p) = [0 : \omega]$ if $In(t)(p) = 1$. $Out'(t, p) = [0 : \omega]$ if $Out(t)(p) = 1$. Furthermore, $In'(t, p_0) = [0 : \omega] = Out'(t, p_0)$.

For any timed transition $t \in T_1$ we have: $In'(t_1, p_0) = [0 : \omega]$, $Out'(t_1, p_1) = [0 : 0]$, $In'(t, p) = [0 : \omega]$ if $In(t)(p) = 1$, $Out'(t, p) = [0 : \omega]$ if $Out(t)(p) = 1$, $In'(t, p_1) = [1 : 1]$ and $Out'(t, p_0) = [0 : \omega]$.

The cost function C' is defined by $C'(p) = C(p)$ for all $p \in P$, $C'(t) = C(t)$ for all $t \in T$, $C'(p_0) = C'(p_1) = \mathbf{0}$, and $C(t_1) = \mathbf{0}$.

The initial marking M_0' is a multiset over $P' \times \mathbb{N}$ which contains $(p, 0)$ exactly n times iff $M_0(p) = n$. Additionally, M_0' contains $(p_0, 0)$.

The set of final markings F' contains exactly all such multisets M' over $P' \times \mathbb{N}$ such that there is a corresponding marking $M \in F$ s.t. $\forall p \in P$. $\sum_{n \in \mathbb{N}} M'((p, n)) = M(p)$ and $\sum_{n \in \mathbb{N}} M'((p_0, n)) = 1$ and $\sum_{n \in \mathbb{N}} M'((p_1, n)) = 0$. In particular, if F is upward-closed then so is F'.
Correctness. The construction ensures that, except for place p_1, the age of tokens in N' does not matter. Instantaneous transitions of N are simulated directly by a transition in N' while timed transitions are simulated in two steps where the correct delay of exactly one time unit is enforced by the age of the token on p_1. Thus, for every run of N with a given cost, there is a run of N' with the same cost.

On the other hand, N' can have some runs with extra delays which do not correspond to any run in N. However, these extra delays do not enable any new transitions, but only leads to higher costs and/or deadlocks (with a token of age > 1 on p_1). Thus if N does not have a run to reach F within the given cost v then N' does not have a run to reach F' within cost v either. □

Lemma 6. The reachability problem for Petri nets with many inhibitor arcs $(p, t_1), \dots, (p, t_k)$ which all connect to the same place p can be reduced to the reachability problem for Petri nets with just one inhibitor arc.

Proof. Let N be a Petri net with inhibitor arcs $(p, t_1), \dots, (p, t_k)$, initial marking M_0 and final marking M_f. We construct a new net N' from N as follows. Add two new places x and y and add x to the input and output places of all transitions, except for $t_1, \dots, t_k$. Remove all inhibitor arcs $(p, t_1), \dots, (p, t_k)$. Add a new transition t with input place x, output place y and a single inhibitor arc (p, t). For every $i \in \{1, \dots, k\}$ we add y to the input places of t_i and x to the output places of t_i. The initial marking M_0' of N' is defined by $M_0'(x) = 1$, $M_0'(y) = 0$ and $M_0'(p') = M_0(p')$ for all other places p'. The final marking M_f' of N' is defined by $M_f'(x) = 1$, $M_f'(y) = 0$ and

$M'_f(p') = M_f(p')$ for all other places p'. The new net N' has just one inhibitor arc (p, t) and simulates the behavior of N with possibly some additional deadlocks (with a token on place y). We get that $M_0 \to^* M_f$ in N iff $M'_0 \to^* M'_f$ in N'. $\qquad\square$