Towards Decision of Testing Equivalence for Petri Nets with Dense Time

IRINA VIRBITSKAITE, *Institute of Informatics Systems, Siberian Division of the Russian Academy of Sciences, Novosibirsk, Russia.*
E-mail: virb@iis.nsk.su [1]

Abstract

The intention of the paper is to extend the testing methodology to a model of Petri nets with a dense time domain. Alternative characterizations of timed testing relations are provided. We also show the decidability of timed testing for a subclass of the model under consideration. The key to our decision procedures is that timed testing equivalences can be reduced to appropriate symbolic bisimulations.

Keywords: time Petri nets, timed testing, symbolic bisimulation, decidability

1 Introduction

An important ingredient of every theory of concurrency is a notion of equivalence between processes. Typically, equivalences are used in the setting of specification and verification both to compare two distinct systems and to reduce the structure of a system.
The best known behavioural equivalence is bisimulation [8]. Two processes are assumed to be bisimilar, if the observer can not see any difference in their behaviours. The standard definitions of bisimulation equivalences and preorders are given in terms of recursively defined relations. The formulation is mathematically elegant and yields efficient algorithms for determining if finite-state processes are related. However, it is often the case that there is no smooth connection between the preorders and equivalences.
Another of the major equivalences of concurrency theory is that of testing, presented in [6]. Two processes are considered testing equivalent if there is no test that can distinguish them. This notion is intuitively appealing and has led to a well-developed mathematical theory of processes that ties together the equivalences and preorders. However, no characterization of these equivalences has led to an algorithmic solution for finite-state processes. Therefore testing decision procedures are based on reduction of testing to bisimulation [3].
Those equivalences were considered for formal system models without time delays. Recently, the demand for correctness analysis of real time systems, i.e. systems whose descriptions involve a quantitative notion of time, increases rapidly. Timed extensions of interleaving models have been investigated thoroughly in the last 5-10 years (see,

[1] This work is supported in part by the Russian Fund of Basic Research (Grant N 00-01-00898), and the Russian Ministry of High Education (Grant for Basic Research in Mathematics).

for example, [1]).
As a representive of a true concurrent approach, time Petri nets are considered in [7] to model real time systems over a dense time domain. A time Petri net is a Petri net whose transitions are labelled by two timing constraints that indicate their earliest and latest firing times. A state of a time Petri net is a set of its places containing tokens, equipped with a finite set of clocks corresponding to the transitions and recording the time elapsed since the transitions were enabled. A time Petri net progresses through a sequence of states in one of two ways: by firing transitions and letting a certain amount of time pass. The model appeared to us as a most simple and natural approach and, at the same time, it provably offers enhanced expressive capabilities w.r.t. several other timed extensions of Petri nets.
In this paper, we first seek to develop a framework for testing preorders and equivalences in the setting of time Petri nets to take into account processes' timing behaviour in addition to their degrees of relative nondeterminism and concurrency, and then try to apply the framework to characterize and, also, to decide timed testing equivalences reducing them to some variants of symbolic bisimulation.
There have been several motivations for this work. One has been given by the paper [10] which has provided alternative characterizations for testing preorders in the context of Petri net models with discrete time. A next origin of this study has been the paper [2], where both strong and weak timed bisimulations are shown to be decidable for dense time processes, by reducing the equivalences to the corresponding symbolic bisimulations. Finally, another origin has been the papers [4] and [9], which have treated timed testing for iterleaving models with discrete time and dense time, respectively. The latter paper has also provided a testing decision procedure that uses an untimed bisimulation between deterministic graphs built from mutually refined timer region graphs that are a finite abstraction of an operational semantics of the timed transition model considered. When we surveyed these results, we discovered that it was not the case for time Petri nets. Working further on the question we looked for a subclass of nets in which we could reduce timed testing relations to the corresponding variants of symbolic bisimulations.
The rest of the paper is organized as follows. The basic notions concerning time Petri nets are introduced in the next section. The definitions and alternative characterizations of timed testing and symbolic bisimulation are given in sections 3 and 4, respectively. In the following section, we introduce a subclass of deterministic time Petri nets, in which timed testing relations coincide with appropriate symbolic bisimulations. Decision procedures for the equivalences are further provided. Due to lack of the space, some of the proofs are omitted they can be found in a forthcoming paper.

2 Time Petri Nets

In this section, we define some terminology concerning time Petri nets [7]. Before introducing the basic concepts, we need to consider some auxiliary notations. Let Act be a finite set of visible actions and τ an internal action. Then $Act_\tau = Act \cup \{\tau\}$. Further, let $\mathbf{N}$ be the set of natural numbers, $\mathbf{R}^+$ the set of positive real numbers, and $\mathbf{R}_0^+$ the set of nonnegative real numbers. Define the set $Interv(\mathbf{R}_0^+) = \{(d_1, d_2), (d_1, d_2], [d_1, d_2), [d_1, d_2] \subset \mathbf{R}_0^+ \mid d_1, d_2 \in \mathbf{N}\}$.
We are now ready to introduce the concept of time Petri nets.

DEFINITION 2.1

A *time Petri net, labelled over* Act_τ, is a tuple $TN = (P, T, F, l, D, m_{TN})$, where P is a finite set of places; T is a finite set of transitions ($P \cap T = \emptyset$); $F \subseteq (P \times T) \cup (T \times P)$ is the flow relation; $l : T \to Act_\tau$ is a labelling function; $D : T \to Interv(\mathbf{R}_0^+)$ is a timing function such that $D(t)$ is a closed interval from $Interv(\mathbf{R}_0^+)$ for all $t \in T$ with $l(t) \in Act$ [2]; $m_{TN} \subseteq P$ is the initial marking.

Fig. 1 shows an example of a time Petri net, where action labels and time intervals are drawn near to transitions. If no confusion arises, we will often omit place identities and use action labels rather transition identities to denote transitions.

For $t \in T$, ${}^\bullet t = \{p \in P \mid (p, t) \in F\}$ and $t^\bullet = \{p \in P \mid (t, p) \in F\}$ denote the *preset* and *postset* of t, respectively. To simplify the presentation, we assume that ${}^\bullet t \cap t^\bullet = \emptyset$ for every transition t. Let $\mathcal{TN}_\tau$ denote the set of all time Petri nets labelled over Act_τ.

A *marking* m of TN is any subset of P. A transition t is *enabled* in a marking m if ${}^\bullet t \subseteq m$ (all its input places have tokens in m), otherwise it is *disabled*. Let $enable(m)$ be the set of transitions, enabled in m. Let $\Gamma(TN) = [T \to \mathbf{R}_0^+]$ be the set of *time assignments* for transitions from T. Given $\natural \in \Gamma(TN)$ and $d \in \mathbf{R}_0^+$, we let $\natural + d$ denote the time assignment of the value $\natural(t) + d$ to each t from T.

TN_1 :

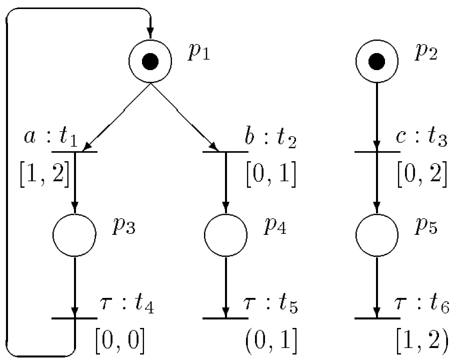

FIG. 1.

A *state* q of TN is a pair $<m, \natural>$, where m is a marking and $\natural \in \Gamma(TN)$. The *initial state* of TN is a pair $q_{TN} = <m_{TN}, \natural_{TN}>$, where $\natural_{TN}(t) = 0$ for all $t \in T$.

The states of TN change, if a time passes or if a transition fires. In a state $q = <m, \natural>$, a transition $t \in T$ *can fire* (denoted $q \xrightarrow{t}$), if $t \in enable(m)$ and $\natural(t) \in D(t)$. We shall write $q \xrightarrow{a}$, if $q \xrightarrow{t}$ and $l(t) = a$. The firing of t in q *leads to* the state $q' = <m', \natural'>$ (denoted $q \xrightarrow{t} q'$), if $m' = (m \setminus {}^\bullet t) \cup t^\bullet$, and forall $t' \in T$, $\natural'(t') = 0$, if $t' \in enable(m') \setminus enable(m)$, else $\natural'(t') = \natural(t')$. We shall write $q \xrightarrow{a} q'$, if $q \xrightarrow{t} q'$ and $l(t) = a$. In a state $q = <m, \natural>$, a time $d \in \mathbf{R}^+$ *can pass* if $\forall t \in enable(m)\ \exists d' \geq d \diamond \natural(t) + d' \in D(t)$. The passage of d in q *leads to* the state $q' = <m', \natural'>$ (denoted $q \xrightarrow{d} q'$), if $m' = m$, and $\natural'(t) = \natural(t) + d$ for all $t \in T$. A state q is *reachable* if either $q = q_{TN}$ or there exists a reachable state q' such that $q' \xrightarrow{x} q$ for some $x \in T \cup \mathbf{R}^+$. Let $RS(TN)$ denote the set of all reachable states of TN.

[2] Notice, τ-actions with open intervals are essential for alternative characterizing timed testing (see below).

A *run* r is a sequence of $q_{TN}, q_1, \ldots, q_{n-1}, q_n \in RS(TN)$ and $x_1, \ldots, x_n \in T \cup \mathbf{R}^+$ of the form: $r = q_{TN} \stackrel{x_1}{\to} q_1 \ \ldots \ q_{n-1} \stackrel{x_n}{\to} q_n$. We define the timing duration of a run r as follows: $time(r) = \sum_{1 \leq i \leq n,\ x_i \in \mathbf{R}^+} \ x_i$. Let $R(TN)$ denote the set of all runs in TN. For $t \in T$, we let $time(t) = \{time(r) \mid r = q_{TN} \stackrel{x_1}{\to} q_1 \ldots q_{n-1} \stackrel{x_n}{\to} q_n \in R(TN),\ q_n \stackrel{t}{\to}\}$ (the set of times at which a transition t can fire). As an illustration, we construct the following run r of the time Petri net TN_1 (see Fig. 1): $r = \langle\{p_1, p_2\}, \natural \equiv 0\rangle \stackrel{\delta_1 = 0.7}{\to} \langle\{p_1, p_2\}, \natural \equiv 0.7\rangle \stackrel{\delta_2 = 0.3}{\to} \langle\{p_1, p_2\}, \natural \equiv 1\rangle \stackrel{\delta_3 = 0}{\to} \langle\{p_1, p_5\}, \natural \equiv 1\rangle \stackrel{\delta_4 = 0}{\to} \langle\{p_4, p_5\}, \natural \equiv 1\rangle \stackrel{\delta_5 = 1}{\to} \langle\{p_4, p_5\}, \natural \equiv 2\rangle \stackrel{\delta_6 = 0}{\to} \langle\{p_5\}, \natural \equiv 2\rangle \stackrel{\delta_7 = 0}{\to} \langle\emptyset, \natural \equiv 2\rangle$. Moreover, we have $time(r) = 2$ and $time(t_6) = [1, 4)$.

We say that TN is *one-safe*, if for every $<m, \natural> \in RS(TN)$ and for every $t \in enable(m)$ it holds: $t^\bullet \cap m = \emptyset$. To guarantee that in any run of TN time is increasing beyond any bound, we need the following *progress condition*: for every set $\{t_1, t_2, \ldots, t_n\} \subseteq T$ (s.t. $\forall\, 1 \leq i < n \diamond t_i^\bullet \cap {}^\bullet t_{i+1} \neq \emptyset$ and $t_n^\bullet \cap {}^\bullet t_1 \neq \emptyset$) there exists $1 \leq j \leq n$ s.t. $0 \notin D(t_j)$. In the sequel, TN and TN' will always denote one-safe time Petri nets satisfying the progress condition.

The *weak leading relation* $\Rightarrow$ *on states of* TN is a relation defined by: $\stackrel{\epsilon}{\Rightarrow} \iff \stackrel{\tau}{\to}^*$ and $\stackrel{x}{\Rightarrow} \iff \stackrel{\epsilon}{\Rightarrow}\stackrel{x}{\to}\stackrel{\epsilon}{\Rightarrow}$, where $\stackrel{\tau}{\to}^*$ is the reflexive and transitive closure of $\stackrel{\tau}{\to}$, and $x \in Act \cup \mathbf{R}^+$. We consider the relation $\stackrel{d}{\Rightarrow}$ as possessing the time continuity property: $q \stackrel{d_1 + d_2}{\Rightarrow} \iff q \stackrel{d_1}{\Rightarrow}\stackrel{d_2}{\Rightarrow}$ for some $d_1, d_2 \in \mathbf{R}^+$.

From now on, we shall use the following notions and notations. Let $Act(\mathbf{R}_0^+) = \{a(d) \mid a \in Act,\ d \in \mathbf{R}_0^+\}$ be the set of *timed actions* of Act over $\mathbf{R}_0^+$. Then $(Act(\mathbf{R}_0^+))^*$ is the set of finite *timed words* over $Act(\mathbf{R}_0^+)$. The function $\Delta : (Act(\mathbf{R}_0^+))^* \to \mathbf{R}_0^+$, measuring the *duration* of a timed word, is defined by: $\Delta(\epsilon) = 0,\ \Delta(w.a(d)) = \Delta(w) + d$. The domain for real-time languages is denoted by $Dom(Act,\ \mathbf{R}_0^+) = \{\langle w, d\rangle \mid w \in (Act(\mathbf{R}_0^+))^*,\ d \in \mathbf{R}_0^+,\ d \geq \Delta(w)\}$. The weak leading relation $\Rightarrow$ is extended to timed words from $(Act(\mathbf{R}_0^+))^*$ and $Dom(Act,\ \mathbf{R}_0^+)$ as follows. For given $d \in \mathbf{R}_0^+$, $d' \in \mathbf{R}^+$, $a \in Act$, and $w \in (Act(\mathbf{R}_0^+))^*$, we define: $q \stackrel{a}{\Rightarrow} q'$ iff $q \stackrel{a(0)}{\Rightarrow} q'$; $q \stackrel{d'}{\Rightarrow}\stackrel{a}{\Rightarrow} q'$ iff $q \stackrel{a(d')}{\Rightarrow} q'$; $q \stackrel{w}{\Rightarrow}\stackrel{a(d)}{\Rightarrow} q'$ iff $q \stackrel{w.a(d)}{\Longrightarrow} q'$; $q \stackrel{w}{\Rightarrow} q'$ iff $q \stackrel{\langle w,\ \Delta(w)\rangle}{\Longrightarrow} q'$; $q \stackrel{\langle w, d\rangle}{\Longrightarrow}\stackrel{d'}{\Rightarrow} q'$ iff $q \stackrel{\langle w,\ d + d'\rangle}{\Longrightarrow} q'$. The set $L(TN) = \{\langle w, d\rangle \in Dom(Act, \mathbf{R}_0^+) \mid q_{TN} \stackrel{\langle w, d\rangle}{\Longrightarrow}\}$ is the *language* of TN.

3 Timed Testing

In this section, we develop *timed testing* relations and provide their alternative characterizations in the setting of time Petri nets.

In general, testing preorders and equivalences relate processes on the basis of their behaviour in response to tests. One natural approach is taken by DeNicola and Hennessy in [6]: a test may be any process in which the set of actions contains a distinguished action ω, the *success* action; then, a process' behaviour in response to a test is given in terms of the *computations* that result when the test is 'applied' to the process. These intuitions may be formalized in the setting of time Petri nets as follows.

Let $\omega \notin Act_\tau$ and $Act_{\tau,\omega} = Act_\tau \cup \{\omega\}$. We shall use $\mathcal{TN}_{\tau,\omega}$ to denote the set of all time Petri nets labelled over $Act_{\tau,\omega}$ (the set of *tests*). For a test $TTN \in \mathcal{TN}_{\tau,\omega}$, let $\widetilde{q}_{TTN}$ be its *initial* state.

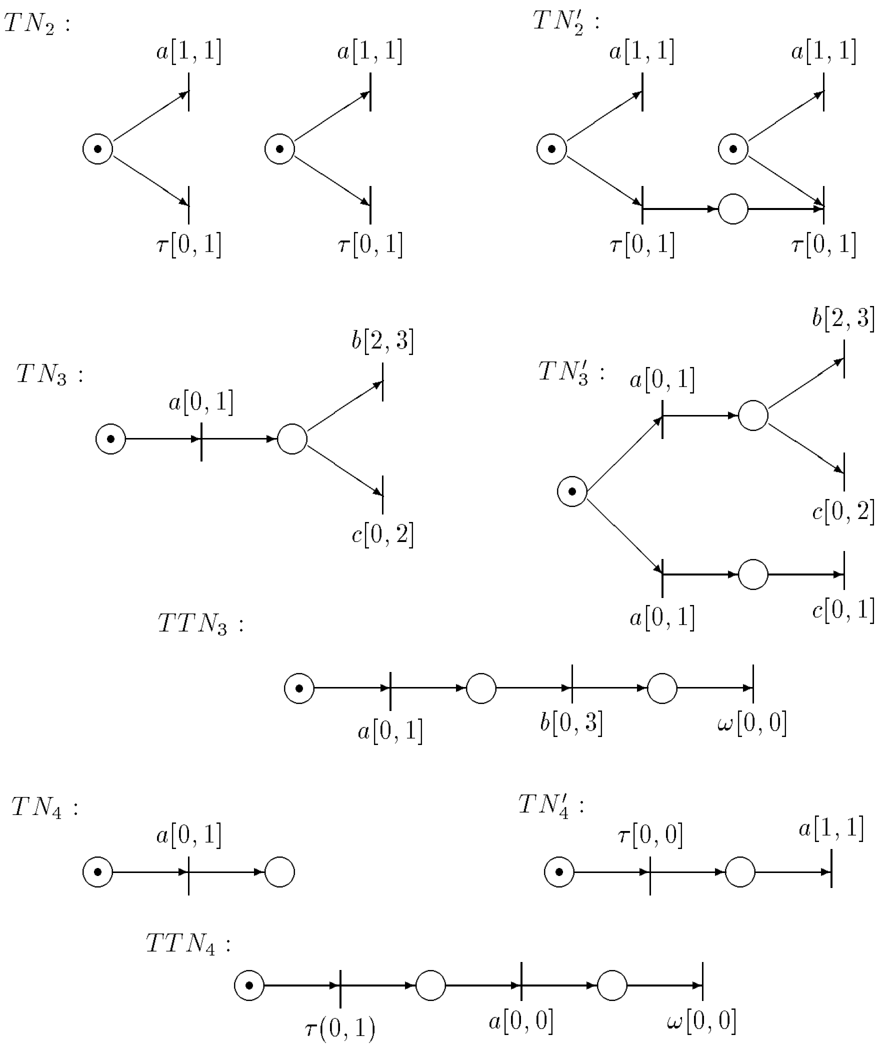

FIG. 2.

Let $TN \in \mathcal{TN}_\tau$, $TTN \in \mathcal{TN}_{\tau,\omega}$ and $q \in RS(TN)$, $\widetilde{q} \in RS(TTN)$. Then an expression $q||\widetilde{q}$ is called a *testing state*. A testing state is *successful*, if $\widetilde{q} \stackrel{\omega}{\rightarrow}$. The *leading relation on testing states* is defined by: $q||\widetilde{q} \stackrel{x}{\rightarrow} q'||\widetilde{q}'$, if $q \stackrel{x}{\rightarrow} q'$ and $\widetilde{q} \stackrel{x}{\rightarrow} \widetilde{q}'$ ($x \in Act \cup \mathbf{R}^+$); $q||\widetilde{q} \stackrel{\tau}{\rightarrow} q'||\widetilde{q}$, if $q \stackrel{\tau}{\rightarrow} q'$; $q||\widetilde{q} \stackrel{\tau}{\rightarrow} q||\widetilde{q}'$, if $\widetilde{q} \stackrel{\tau}{\rightarrow} \widetilde{q}'$. The *weak leading relation on testing states* is defined in a similar way as on states of a time Petri net. A maximal sequence $\natural = q_0||\widetilde{q}_0 \stackrel{y_1}{\rightarrow} q_1||\widetilde{q}_1 \stackrel{y_2}{\rightarrow} \cdots \stackrel{y_n}{\rightarrow} q_n||\widetilde{q}_n$ ($y_i \in Act_\tau \cup \mathbf{R}^+,\ 1 \leq i \leq n,\ n \geq 0$) is a *computation* of $q_0||\widetilde{q}_0$, if $\exists c \in \mathbf{R}^+\ \forall y_i \in \mathbf{R}^+ \diamond y_i \geq c$. We say that $y_1 y_2 \ldots y_n$ is the corresponding sequence of a computation $\natural$. A computation $\natural$ is called *successful*, if $q_i||\widetilde{q}_i$ is successful for some $0 \leq i \leq n$. We write q *may* $\widetilde{q}$, if there is a successful computation of $q||\widetilde{q}$; TN *may* TTN, if q_{TN} *may* $\widetilde{q}_{TTN}$. We write q *must* $\widetilde{q}$, if every

computation of $q\|\widetilde{q}$ is successful; TN *must* TTN, if q_{TN} *must* $\widetilde{q}_{TTN}$.
We are now ready to define the concepts of timed testing preorders and equivalences for time Petri nets TN and TN': $TN \leq_\alpha TN' \iff \forall\, TTN \in \mathcal{TN}_{\tau,\omega} \circ TN\ \alpha\ TTN \Rightarrow TN'\ \alpha\ TTN$, where $\alpha \in \{may,\ must\}$; $TN \leq_{test} TN' \iff TN \leq_{may} TN'$ and $TN \leq_{must} TN'$; $TN \simeq_\alpha TN' \iff TN \leq_\alpha TN'$ and $TN' \leq_\alpha TN$, where $\alpha \in \{may,\ must,\ test\}$.
To illustrate the concepts introduced, consider the time Petri nets shown in Fig. 2. The *test*-equivalent nets TN_2 and TN_2' are first depicted. The next nets TN_3 and TN_3' are *may*-equivalent, but not *must*-equivalent. Consider the test TTN_3 (see Fig. 2). We have TN_3 *must* TTN_3, but $\neg(TN_3'$ *must* $TTN_3)$, because there exists an unsuccessful computation of $M_{TN_3'}\|T_{TTN_3}$ with the corresponding sequence $a.1$. The non-*may*-equivalent nets TN_4 and TN_4' are finally shown. Consider the test TTN_4 (see Fig. 2). We have TN_4 *may* TTN_4, but $\neg(TN_4'$ *may* $TTN_4)$, because every computation of $M_{TN_4'}\|T_{TTN_4}$ is unsuccessful.
In the following, we shall need some additional notions and notations. Let $q \in RS(TN)$ and $\langle w,\ d\rangle \in Dom(Act,\ \mathbf{R}_0^+)$. Then $S(q) = \{x \in Act \cup \mathbf{R}^+ \mid q \xrightarrow{x}\}$, and $Acc(TN,\ \langle w,\ d\rangle) = \{S(q') \mid q_{TN} \overset{\langle w,\ d\rangle}{\Longrightarrow} q',\ q' \overset{\tau}{\not\to}\}$ (timed acceptance set). Let $N, N' \subset 2^{Act \cup \mathbf{R}^+}$. Then $N \subset\subset N' \iff \forall S \in N\ \exists S' \in N' \circ [(S' \mid_{Act} \subseteq S \mid_{Act}) \wedge (S \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S' \mid_{\mathbf{R}^+} = \emptyset)]$; $N \equiv N' \iff N \subset\subset N' \wedge N' \subset\subset N$.
Timed testing can be characterized by means of timed languages and timed acceptance sets.

Lemma 3.1
Let $\langle w, d\rangle = \langle a_1(d_1)\ldots a_n(d_n), \sum_{i=1}^n d_i + d_{n+1}\rangle \in L(TN)(n \geq 0)$. Let $A_{\langle w,d\rangle} = \{\langle a_1(d_1')\ldots a_n(d_n'), \sum_{i=1}^n d_i' + d_{n+1}'\rangle \in Dom(Act \cup \mathbf{R}_0^+) \mid \forall 1 \leq k \leq l \leq n+1 \circ \sum_{i=k}^l d_i' \in \wr \sum_{i=k}^l d_i \wr\}$. Then **(i)** $A_{\langle w,d\rangle} \subseteq L(TN)$, **(ii)** $\forall \langle w', d'\rangle \in A_{\langle w,d\rangle} \circ Acc(TN, \langle w, d\rangle) \equiv Acc(TN, \langle w', d'\rangle)$. Here, for $d \in \mathbf{R}_0^+$, we let $\wr\ d\ \wr = [\ d, d\ ]$, if $d \in \mathbf{N}$, else $\wr\ d\ \wr = (\lfloor d\rfloor, \lceil d\rceil)$.

Proposition 3.2
(i) $TN \leq_{may} TN' \iff L(TN) \subseteq L(TN')$; **(ii)** $TN \leq_{must} TN' \iff \forall\langle w,\ d\rangle \in Dom(Act, \mathbf{R}_0^+) \circ Acc(TN', \langle w, d\rangle) \subset\subset Acc(TN, \langle w, d\rangle)$.

4 Symbolic Bisimulation

In this section, we define and characterize some notions of symbolic bisimulation.
Central to defining the notions is a synchronization of different time Petri nets, gained by using the product of their states. Let $q \in RS(TN)$ and $q' \in RS(TN')$. Then a pair $q|q'$ is called an *extended state* of TN and TN'. We use $ES(TN, TN')$ to denote the set of all extended states of TN and TN'. Let $q|q',\ q_1|q_1' \in ES(TN, TN')$. Then the *leading relation on extended states* is defined as follows: $q|q' \xrightarrow{x} q_1|q_1'$, if $q \xrightarrow{x} q_1$ and $q' \xrightarrow{x} q_1'$ $(x \in Act \cup \mathbf{R}^+)$; $q|q' \xrightarrow{\tau} q_1|q_1'$, if either $q \xrightarrow{\tau} q_1$ and $q' = q_1'$ or $q' \xrightarrow{\tau} q_1'$ and $q = q_1$. The *weak leading relation on extended states* is defined and further extended to timed words from $Dom(Act, \mathbf{R}_0^+)$ in a similar way as on states of a time Petri net. Since time Petri nets are defined over a dense time domain, the number of their extended states is infinite. In order to get a discrete representation of the state-space of time Petri nets, we use the concept of regions (equivalence classes of states) [1]. Let $q = (m, \natural)|q' = (m', \natural')$, $q_1 = (m_1, \natural_1)|q_1' = (m_1', \natural_1') \in ES(TN, TN')$.

Then $q|q' \simeq q_1|q'_1$, if (1) $m = m_1$, $m' = m'_1$, (2) (i) for each $1 \leq i \leq n$ it holds: $\lfloor \natural|\natural'(t_i) \rfloor = \lfloor \natural_1|\natural'_1(t_i) \rfloor$ or $\natural|\natural'(t_i) + d \notin D(t_i)$ and $\natural_1|\natural'_1(t_i) + d_1 \notin D(t_i)$ for all $d, d_1 \in \mathbf{R}_0^+$; (ii) for each $1 \leq i, j \leq n$ s.t. $\natural|\natural'(t_i) + d \in D(t_i)$ for some $d \in \mathbf{R}_0^+$ and $\natural_1|\natural'_1(t_j) + d_1 \in D(t_j)$ for some $d_1 \in \mathbf{R}_0^+$ it holds: $\{\natural\ |\natural'(t_i)\} \leq \{\natural\ |\natural'(t_j)\} \iff \{\natural_1\ |\natural'_1(t_j)\} \leq \{\natural_1\ |\natural'_1(t_j)\}$, and $\{\natural\ |\natural'(t_i)\} = 0 \iff \{\natural_1\ |\natural'_1(t_i)\} = 0$. Here, $n = |\ T_{TN}\ | + |\ T_{TN'}\ |$, $\natural|\natural'$ is the concatenation of vectors $\natural$ and $\natural'$, and for $d \in \mathbf{R}_0^+$, $\{d\}$ and $\lfloor d \rfloor$ denote its fractional and smallest integer parts, respectively. A set $R = [q|q'] = \{q_1|q'_1 \in ES(TN, TN') \mid q|q' \simeq q_1|q'_1\}$ is called a *region* of TN and TN'.

Let R, R_1 be regions of TN and TN' such that $R \neq R_1$. Then the *leading relation on regions* is defined as follows: $R \xrightarrow{a} R_1$, if there exists $q|q' \in R$ and $q_1|q'_1 \in R_1$ such that $q|q' \xrightarrow{a} q_1|q'_1$ ($a \in Act_\tau$); $R \xrightarrow{\chi} R_1$, if there exists $d \in \mathbf{R}^+$, $q|q' \in R$ and $q_1|q'_1 \in R_1$ such that $q|q' \xrightarrow{d} q_1|q'_1$, and $q|q' \xrightarrow{d'} \widetilde{q}_1|\widetilde{q}'_1 \in R \cup R_1$ for all $0 < d' \leq d$. We shall call a partition of $ES(TN, TN')$ into regions *stable* iff it holds: if $R \xrightarrow{a} R_1$, then $\forall q|q' \in R \diamond q|q' \xrightarrow{a} q_1|q'_1$ for some $q_1|q'_1 \in R_1$ ($a \in Act_\tau$); and if $R \xrightarrow{\chi} R_1$, then $\forall q|q' \in R\ \exists d \in \mathbf{R}^+ \diamond q|q' \xrightarrow{d} q_1|q'_1$ for some $q_1|q'_1 \in R_1$, and $q|q' \xrightarrow{d'} \widetilde{q}_1|\widetilde{q}'_1 \in R \cup R$ for all $0 < d' \leq d$. The *region graph* of TN and TN' is the labelled directed graph $RG(TN, TN') = (V_{RG}, E_{RG}, l_{RG})$. The set of vertices V_{RG} is a stable partition of $ES(TN, TN')$ into regions, the set of edges E_{RG} is the leading relation on regions of V_{RG}, the labelling function $l_{RG} : E_{RG} \longrightarrow Act_\tau \cup \{\chi\}$ is defined as: $l((R, R_1)) = z \iff R \xrightarrow{z} R_1$. For $R \in V_{RG}$ and $z \in Act \cup \{\chi\}$, we define $Der(R, z) = \{R_1 \in V_{RG} \mid R \xrightarrow{z} R_1\}$. We also need a notion of the *weak leading relation on regions* of V_{RG}, given by: $R \overset{\epsilon}{\Rightarrow} R_1 \iff R \xrightarrow{\tau}{}^{*} R_1$, and $R \overset{z}{\Rightarrow} R_1 \iff R \overset{\epsilon}{\Rightarrow}\xrightarrow{z}\overset{\epsilon}{\Rightarrow} R_1$ ($z \in Act \cup \{\chi\}$). We next define the concepts of a class (the τ-closure of regions) and the class graph [3] in the setting of time Petri nets. Let $RG(TN, TN') = (V_{RG}, E_{RG}, l_{RG})$ and $Q \subseteq V_{RG}$. A set $Q^\tau = \{R' \in V_{RG} \mid \exists R \in Q\ .\ R \overset{\epsilon}{\Rightarrow} R'\}$ is called a *class* of TN and TN'. Define $Q_0 = \{[q_{TN}|q_{TN'}]\}^\tau$, and $Der(Q,\ z) = \bigcup_{R \in Q} Der(R,\ z)$. For classes Q, Q_1 and $z \in Act \cup \{\chi\}$, the *leading relation on classes* is given by: $Q \xrightarrow{z} Q_1$, if $Q_1 = (Der(Q,\ z))^\tau$. Define $S(Q) = \{z \in Act \cup \{\chi\} \mid Q \xrightarrow{z}\}$. A class Q is called *reachable*, if either $Q = Q_0$ or there exists a reachable class Q_1 and $z \in Act \cup \{\chi\}$ such that $Q_1 \xrightarrow{z} Q$. The *class graph* of TN and TN' is the labelled directed graph $CG(TN, TN') = (V_{CG}, E_{CG}, l_{CG})$. The set of vertices V_{CG} is the set of reachable classes of TN and TN', the set of edges E_{CG} is the projection of the leading relation on classes to the set $V_{CG} \times V_{CG}$, the labelling function $l_{CG} : E_{CG} \longrightarrow (Act \cup \{\chi\})$ is defined as: $l((Q, Q_1)) = z \iff Q \xrightarrow{z} Q_1$.

For the correctness of our decision algorithms (see below), we need to introduce the following notion.

Definition 4.1

Let $\langle w, d \rangle \in L(TN) \cap L(TN')$ and $CG(TN, TN') = (V_{CG},\ E_{CG},\ l_{CG})$. Let $p = Q_0\ \ldots\ Q$ be a path in $CG(TN, TN')$. Then $q|q' \in ES(TN, TN')$ is *reachable by* $\langle w, d \rangle$ *consistent with* p iff $[q|q'] \in Q$ and either:

- $p = Q_0$ and $\langle w, d \rangle = \langle \epsilon, 0 \rangle$, or
- $p = p' \xrightarrow{z} Q$, there exists $q_1|q'_1 \in ES(TN, TN')$ is reachable by $\langle w', d' \rangle$ consistent with p', and either:

– $z = a \in Act$, $q_1|q_1' \stackrel{a\ d''}{\Rightarrow\!\!\rightarrow} q|q'$, and $\langle w, d\rangle = \langle w' a(d' - \Delta(w'), d' + d''\rangle$, for some $d'' \in \mathbf{R}_0^+$, or

– $z = \chi$, $q_1|q_1' \stackrel{d''}{\Rightarrow} q|q'$, and $\langle w, d\rangle = \langle w', d' + d''\rangle$, for some $d'' \in \mathbf{R}^+$.

Introduce some auxiliary notations. Let $q_1 = (m_1, \natural_1)|q_2 = (m_2, \natural_2) \in ES(TN, TN')$, and $i, j = 1, 2$ $(i \neq j)$. Then $q_i \xrightarrow{a}_{q_j}$ iff $(a \in Act\ \wedge\ q_i \xrightarrow{a})$, and $q_i \xrightarrow{\chi}_{q_j}$ iff $(\exists d \in \mathbf{R}^+ \circ q_i \xrightarrow{d}\ \wedge (m_1, \natural_1 + d)|(m_2, \natural_2 + d) \notin [q_1|q_2]$.
We are now ready to define some notions of symbolic bisimulation.

DEFINITION 4.2
Let $CG(TN, TN') = (V_{CG}, E_{CG}, l_{CG})$, $\Pi \subseteq V_{CG}$ and $z \in (Act \cup \{\chi\})$ be given. Then

- $\beta \subseteq V_{CG}$ is called a *symbolic* Π-*bisimulation* between TN and TN', if $\beta \subseteq \Pi$, $Q_0 \in \beta$, and for all $Q \in \beta$ and $[q|q'] \in Q$ it holds:
  – if $q \xrightarrow{z}_{q'}$, then $\exists Q_1 \in \beta$ s.t. $[q|q''] \xrightarrow{z} [q_1|q_1']$ for some $[q|q''] \in Q$ and $[q_1|q_1'] \in Q_1$,
  – if $q' \xrightarrow{z}_{q}$, then $\exists Q_1 \in \beta$ s.t. $[q''|q'] \xrightarrow{z} [q_1|q_1']$ for some $[q''|q'] \in Q$ and $[q_1|q_1'] \in Q_1$,
- TN and TN' are *symbolic* Π-*bisimilar* (denoted $TN \approx_\Pi TN'$), if there exists a symbolic Π-bisimulation between TN and TN'.

A symbolic V_{CG}-bisimulation is a symbolic bisimulation (denoted $\approx$). We further define the relation $\Pi \subseteq V_{CG}$ as follows: a vertex Q of $CG(TN, TN')$ belongs to Π, iff for all $[q|q'] \in Q$ $([q|q'] \stackrel{\overline{z}}{\not\rightarrow})$ it holds:
– $\exists [q|q_1'] \in Q$ $([q|q_1'] \stackrel{\overline{z}}{\not\rightarrow}) \circ (S(q_1') \mid_{Act} \subseteq S(q) \mid_{Act})$, $(S(q) \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S(q_1') \mid_{\mathbf{R}^+} = \emptyset)$,
– $\exists [q_1|q'] \in Q$ $([q_1|q'] \stackrel{\overline{z}}{\not\rightarrow}) \circ (S(q_1) \mid_{Act} \subseteq S(q') \mid_{Act})$, $(S(q') \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S(q_1) \mid_{\mathbf{R}^+} = \emptyset)$.

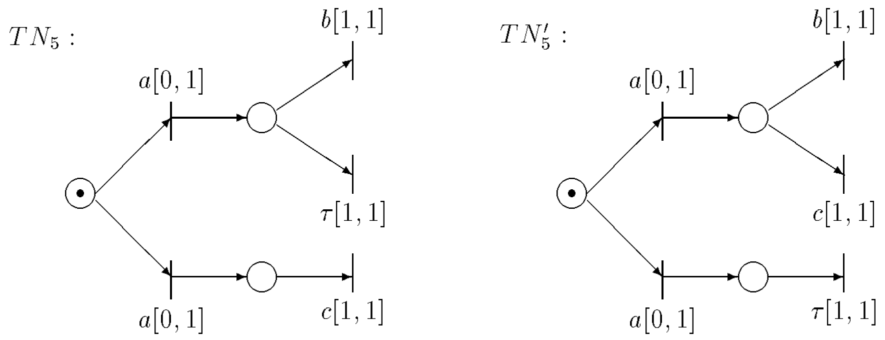

FIG. 3.

Illustrate the notions introduced by the following examples. First consider the non-bisimilar time Petri nets TN_2 and TN_2', shown in Fig. 2. There is no symbolic bisimulation between TN_2 and TN_2', because in $G(TN_2, TN_2')$ there exists a vertex Q with a region $[q|q']$ ($q|q'$ is reachable by $\langle a(1), 1\rangle$ consistent with the path from Q_0 to Q) such that $q \xrightarrow{a}_{q'}$, but in $G(TN_2, TN_2')$ there is no vertex Q_1 such that $[q|q''] \xrightarrow{a} [q_1|q_1']$ for some $[q|q''] \in Q$ and $[q_1|q_1'] \in Q_1$. Next consider the time Petri nets TN_3 and TN_3', shown in Fig. 2. We have $TN_3 \approx TN_3'$, but $TN_3 \not\approx_\Pi TN_3'$, because in any symbolic bisimulation between TN_3 and TN_3' there exists a vertex Q, which contains only two regions $[q|q'], [q|q'']$ ($q|q', q|q''$ are reachable by $\langle a(1), 1\rangle$ consistent with the path from Q_0 to Q) such that $S(q) = \{\{c\} \cup (0,1]\}$ and $S(q') = \{\{c\}\}$, i.e. $\neg(S(q') \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S(q) \mid_{\mathbf{R}^+} = \emptyset)$. Finally, consider the symbolic Π-bisimilar nets TN_5 and TN_5' shown in Fig. 3.

We further provide alternative characterizations of the concepts of symbolic bisimulation.

PROPOSITION 4.3

Let $CG(TN, TN') = (V_{CG}, E_{CG}, l_{CG})$ be given. Then **(i)** $TN \approx TN' \iff V_{CG}$ is a symbolic bisimulation, **(ii)** $TN \approx_\Pi TN' \iff V_{CG}$ is a symbolic Π-bisimulation.

5 Deciding the Equivalences

In this section, we consider the problem of deciding timed testing equivalences. The point of our decision algorithms is that timed testing relations are to be reduced to appropriate symbolic bisimulations. However, it has turned out that not every *test*-equivalent time Petri nets from the class $\mathcal{TN}_\tau$ are symbolic Π-bisimilar and vice versa. As was shown above, we have $TN_2 \simeq_{test} TN_2'$, but $TN_2 \not\approx TN_2'$ (see Fig. 2). On the other hand, consider the time Petri nets $TN_6 = [(a[0,1] \parallel b[0,1]) \parallel c[1,2]] \;\#\; [a[0,1]\;, (c[1,2] \parallel b[0,1])] \;\#\; [c[1,2] \parallel (a[0,1]\;, b[0,0])] \;\#\; [c[1,1] \parallel (a[0,1]\;, b[0,1])] \;\#\; [(b[0,1]\;, a[0,1]) \parallel c[1,2]] \;\#\; [(a[0,1] \;\#\; (a[0,1]\;, \tau[0,0])) \parallel (b[1,1] \parallel (\tau[1,2], \tau[0,0]) \parallel c[1,2])]$ and $TN_6' = [a[0,1]\;, (c[1,2] \parallel b[0,1])] \;\#\; [c[1,2] \parallel (a[0,1]\;, b[0,0])] \;\#\; [c[1,1] \parallel (a[0,1]\;, b[0,1])] \;\#\; [(b[0,1]\;, a[0,1]) \parallel c[1,2]] \;\#\; [(a[0,1] \;\#\; (a[0,1]\;, \tau[0,0])) \parallel (b[1,1] \parallel (\tau[1,2]\;, \tau[0,0]) \parallel c[1,2])]$. We have $TN_6 \approx_\Pi TN_6'$ and $TN_6 \not\simeq_{may} TN_6'$, because the timed word $\langle a(0.5)b(0.2)c(0.5), 1.2\rangle$ belongs to $L(TN_6)$, but does not to $L(TN_6')$. Here, to make our examples easier to understand, we gave for them algebraic expressions (see [5]) over actions with the time intervals of the corresponding transitions.

Therefore we now introduce a subclass of time Petri nets in which timed *may*- and *must*-equivalences are proved to coincide with symbolic and symbolic Π-bisimulation equivalences, respectively. A time Petri net TN is *deterministic*, if $\forall t, t' \in T \diamond \exists r = q_{TN} \xrightarrow{x_1} q_1 \ldots q_{n-1} \xrightarrow{x_n} q_n \in R(TN)$ s.t. $q_n \xrightarrow{t}$ and $q_n \xrightarrow{t'} \Rightarrow (l(t) \neq l(t'))$, TN is *τ-discrete*, if $\forall t \in T \diamond l(t) = \tau \Rightarrow D(t) \subset \mathbf{N}$. Let $\mathcal{DTN}_\tau$ denote the class of time Petri nets having the determinacy and τ-discreteness properties. Notice, none of the example nets mentioned in the section possesses the properties.

The following theorem establishes the coincidence between timed testing relations and appropriate symbolic bisimulations.

THEOREM 5.1

Let $TN, TN' \in \mathcal{DTN}_\tau$. Then **(i)** $TN \simeq_{may} TN' \iff TN \approx TN'$, **(ii)** $TN \simeq_{must} TN' \iff TN \approx_\Pi TN'$ [3].

We are now ready to decide the problem of checking the equivalences.

THEOREM 5.2

Let $TN, TN' \in \mathcal{DTN}_\tau$. The checking problems for $TN \approx TN'$, $TN \approx_\Pi TN'$, $TN \simeq_{test} TN'$ are decidable.

PROOF. The algorithm for checking symbolic bisimulation between TN and TN' from $\mathcal{DTN}_\tau$ consists of the following steps. First, we construct the region graph of TN and

[3] It has turned out that the reduction of timed testing to symbolic bisimulation is crucial to decide testing even in the case of time Petri nets from $\mathcal{DTN}_\tau$, because the mutual synchronization of nets is still necessary.

TN', $RG(TN, TN')$, using the algorithm from [1]. In addition, each vertex $R = [q|q']$ of $RG(TN, TN')$ is equipped with a pair $\langle S_R, S'_R\rangle$, where $S_R = \{z \in (Act \cup \chi) \mid q \xrightarrow{z}_{q'}\}$ and $S'_R = \{z \in (Act \cup \chi) \mid q' \xrightarrow{z}_{q}\}$. Second, we construct the class graph of TN and TN', $CG(TN, TN')$, using the algorithm from [3]. In addition, each vertex Q of $CG(TN, TN')$ is equipped with $S_Q = \bigcup_{R\in Q}(S_R \cup S'_R)$. Let V_{CG} denote the set of all vertices of $CG(TN, TN')$. Using It is easy to show that for deciding whether V_{CG} is a symbolic bisimulation, it is sufficient to check whether $S_Q \subseteq S(Q)$ for each $Q \in V_{CG}$. The algorithm for checking symbolic Π-bisimulation between TN and TN' from $\mathcal{DTN}_\tau$ is based on that presented above. In addition, each vertex R of $RG(TN, TN')$ is equipped with acc_R ($acc_R \iff (R \overset{\mathcal{I}}{\not\Rightarrow} S_R = S'_R)$, and each vertex Q of $CG(TN, TN')$ is equipped with acc_Q ($acc_Q \iff \bigwedge_{R\in Q} acc_R$). It is easy to show that for deciding $V_{CG} \subseteq \Pi$, it is sufficient to check the truth of acc_Q for each $Q \in V_{CG}$, if V_{CG} is a symbolic bisimulation.
The decidability of checking $TN \simeq_{test} TN'$ follows from the existence of the above algorithms. ∎

Make some remarks on the complexity of the algorithm for checking a symbolic Π-bisimulation between time Petri nets TN and TN' from the class $\mathcal{DTN}_\tau$ (see the proofs of Theorem 5.2): according to [1], the procedure of constructing the region graph is exponential in the size of TN and TN'; according to [3], the procedure of constructing the class graph is polynomial in the size of the region graph; checking, whether V_{CG} is a symbolic Π-bisimulation, takes time complexity linear in the size of the class graph.

References

[1] R. Alur, C. Courcoubetis, D. Dill. Model checking in dense real time. *Information and Computation* **104** (1993) 2–34.

[2] K. Čerāns. Decidability of bisimulation equivalences for parallel timer processes. *Lecture Notes in Computer Science* **663** (1993) 302–315.

[3] R. Cleaveland, M. Hennessy. Testing equivalence as a bisimulation equivalence. *Lecture Notes in Computer Science* **407** (1989) 11–23.

[4] R. Cleaveland, A.E. Zwarico. A theory of testing for real-time. Proc. *LICS'91* 110–119.

[5] R. van Glabbeek, F. Vaandrager. Petri Net Models for Algebraic Theories of Concurrency. *Lecture Notes in Computer Science* **259** (1987) 224–242.

[6] R. De Nicola, M. Hennessy. Testing equivalence for processes. *Theoretical Computer Science* **34** (1984) 83–133.

[7] Merlin, P., Faber, D.J. Recoverability of communication protocols. *IEEE Trans. of Communication* COM-**24**(9) (1976).

[8] D. Park. Concurrency and automata on infinite sequences. *Lecture Notes in Computer Science* **154** (1981) 561–572.

[9] B. Steffen, C. Weise. Deciding testing equivalence for real-time processes with dense time. *Lecture Notes in Computer Science* **711** (1993) 703–713.

[10] Vogler W. Faster Asynchronous Systems. *Lecture Notes in Computer Science* **962** (1995) 299–312.