Joint NCC & IIS Bull., Comp. Science, 8 (1998), 57–79

Equivalence notions and refinement for timed Petri nets*

Irina B. Virbitskaite, Igor V. Tarasyuk

The paper is contributed to develop a family of equivalence notions for real-time systems represented by labelled Merlin's time Petri nets with zero length time intervals (i.e., with fixed time delays). We call them "timed Petri nets". In particular, we introduce timed (time-sensitive), untimed (time-abstracting) and region (based on the notion of region [1]) equivalences in both the trace and bisimulation semantics. The interrelations of all the equivalence notions are examined for a general class of timed nets as well as for a subclass of untimed nets (timed nets with time delays equal to zero's). Further we define a timed variant of state-mashine refinement [4] and investigate how the proposed equivalence notions behave with respect to this class of refinements.

Key words & phrases: timed and untimed Petri nets, timed, untimed and region equivalences, trace and bisimulation semantics.

1. Introduction

An important ingredient of every theory of concurrence is a notion of equivalence between systems. 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. Over the past several years, a variety of equivalences — most notably, perhaps, trace and bisimulation ones — have been promoted, and the relationship between them has been quite well-understood (see, for example, [8]).

Those equivalences were considered for formal system models without time delays. Recently, a growing interest can be observed in modeling real-time systems which imply a need of a representation of the lapse of time. Several formal methods for specifying and reasoning about such systems have been proposed in recent years (see [3] as a survey). Whereas, the incorporation of real time into equivalence notions is less advanced. There are a few papers (see, for example, [2, 5, 11]), where decidability questions of time-sensitive equivalences are investigated. In these studies, real-time

*Partially supported by the INTAS-RFBR under Grant 95-01-0378). The first author is also supported by the Russian State Committee of High Education for Basic Research in Informatics (Graph Theoretical Methods in Programming). The second author is also supported by the Foundation for Promotion of Young Scientists of Siberian Division of the Russian Academy of Sciences.

systems are represented by parallel timer processes or timed automata, containing fictitious time measuring elements called clocks. However, concurrence cannot be modelled directly by such timed states graphs. On the other hand, to model real-time systems over dense time domain, Petri nets with time delays were considered in [9, 13]. A timed net proceeds in one of two ways: by firing transitions or letting a certain amount of real time pass.

For the design of concurrent systems, it can be useful to consider a hierarchy of their descriptions, which allows refinement of unstructured entities on a more abstract design level by complex structures on a lower level. The notion of refinement has found considerable interest in the literature, but, to the best of our knowledge, with reference to untimed systems (see, for example, [4, 6]). Contributions usually consist of congruence results, i.e., an equivalence is defined and for abstract system descriptions it is shown: if two such descriptions are equivalent and both are refined in the same way, then the resulting detailed system descriptions are equivalent again.

Our main aim here is to develop a family of equivalence notions and establish their interrelations for real-time systems represented by Petri nets with fixed time delays (timed nets). Essentially, our model is labelled Merlin's time nets with time intervals of zero length. A second point of the paper is to define a timed variant of action refinement and to investigate how the proposed equivalence notions behave with respect to this class of refinements.

There have been various motivations for this work. One has been the papers [2, 5, 10, 12], in which the definitions of timed (time-sensitive) and untimed (time-abstracting) equivalence notions were given for parallel timer processes and timed automata. Timed equivalences can measure the exact real-numbered duration of every delay, whereas untimed ones abstract away from the lapse of time. The paper [1] has proposed the notion of region (an equivalence class of states) to be able to construct a finite representation of the space of states of timed automata. However, hitherto the literature of timed Petri nets has lacked such the equivalences. Therefore we attempted to introduce timed, untimed and region equivalence notions in both the trace and bisimulation semantics and to establish their interrelations, resulting in a lattice of implications. Furthermore, the coincidence between timed and region variants of the equivalences was proved, implying simplification of timed equivalence checking. A next origin of this paper was the notion of action refinement that was not received much attention in the field of the design of real-time systems. In this regard, the paper [7] is a welcome exception, in that a design method for real time systems developed through a sequence of reginement steps, applied to timed Petri nets was put forward. Following [4], where a notion of state-machine refinement (SM-refinement) was introduced for untimed Petri nets, we considered a timed variant of the refinement under which the transitions with the fixed label and time delay were replaced by timed state machine nets. Finally, another origin of the study was the question whether or not the introduced equivalences are preserved by the transition refinement. Summarizing the extensive discussion in [6], we looked for conditions under which timed bisimulation is preserved with respect to the refinement.

The remainder of the paper is organized as follows. In the next section, we give a short presentation of a notion of timed Petri net. In Section 3, timed, untimed and region equivalence notions in both the trace and bisimulation semantics are defined. Section 4 establishes the coincidence between timed and region variants of the equivalences. Comparing the equivalences, we further construct the lattice of their implications. In Section 5, it is shown that timed and untimed variants of the equivalences are collapsed for untimed nets in which all time delays associated with transitions are equal to zero's. In Section 6, we first define a notion of timed SM-refinement and then derive some conditions for timed bisimulation to be preserved by the refinement. Section 7, finally, contains a few concluding remarks.

2. Basic definitions

In the following, we define some basic notions concerning timed nets that are a slight simplification of Merlin's model of Petri nets with time [9, 13]. Let L be a set of *action names* ranged over by a with and without subscripts. We denote the set of natural numbers by $\mathbf{N}$, and the set of *nonnegative real numbers* by $\mathbf{R}^+$. Let $L \cup \mathbf{R}^+$ be ranged over by x with and without subscripts.

Definition 1. A *timed net* is a 6-tuple $N = \langle P_N, T_N, F_N, l_N, M_N, \Upsilon_N \rangle$, where:

- $\langle P_N, T_N, F_N, l_N, M_N \rangle$ is a safe Petri net (with labelling over L);
- $\Upsilon_N : T_N \to \mathbf{N}$ is a *time delay* function.

Example 1. An example of a timed net can be seen in Figure 1, where the time delays are depicted by the numbers in the double brackets near by the corresponding transitions.

As usual, we introduce the following notations. For $y \in T_N \cup P_N$, ${}^\bullet y = \{z \mid (z, y) \in F_N\}$ and $y^\bullet = \{z \mid (y, z) \in F_N\}$ denote the *preset* and *postset* of y, respectively. We will need to refer to the set of places without ingoing arcs or without outgoing arcs. Let ${}^\circ N = \{p \in P_N \mid {}^\bullet p = \emptyset\}, N^\circ = \{p \in P_N \mid p^\bullet = \emptyset\}$. We use $\mathcal{N}$ to denote a class of timed nets, ranged over by N and N'.

A mapping $\beta : N \to N'$ is an *isomorphism* between N and N', denoted by $\beta : N \simeq N'$, if:

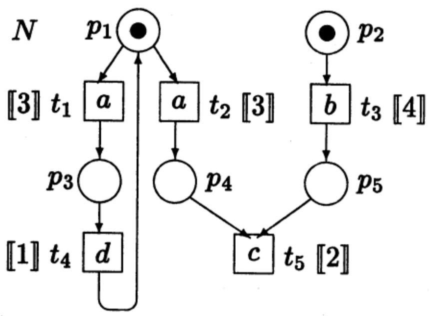

Figure 1. An example of a timed net

1. β is a bijection s.t. $\beta(P_N) = P_{N'}$ and $\beta(T_N) = T_{N'}$;
2. $\forall p \in P_N\ \forall t \in T_N\ (p,t) \in F_N \Leftrightarrow (\beta(p), \beta(t)) \in F_{N'}$ and $(t,p) \in F_N \Leftrightarrow (\beta(t), \beta(p)) \in F_{N'}$;
3. $\forall t \in T_N\ l_N(t) = l_{N'}(\beta(t))$;
4. $\forall p \in P_N\ p \in M_N \Leftrightarrow \beta(p) \in M_{N'}$;
5. $\forall t \in T_N\ \Upsilon_N(t) = \Upsilon_{N'}(\beta(t))$.

N and N' are *isomorphic*, denoted by $N \simeq N'$, if $\exists \beta : N \simeq N'$.

A *marking* M of $\mathcal{N}$ is any subset of P_N. Let $Mark(N)$ denote the set of markings of N. A transition t is *enabled* in $M \in Mark(N)$, 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 $\Gamma = [T_N \to \mathbf{R}^+]$ be the set of *time assignments* for transitions from T_N. Assume $\Upsilon \in \Gamma$ and $\delta \in \mathbf{R}^+$. Then $\Upsilon + \delta$ denotes the time assignment of the value $\Upsilon(t) + \delta$ for each t from T_N.

A *state* of N is a pair $Q = (M, \Upsilon)$, where $M \in Mark(N)$ is a marking of N and $\Upsilon \in \Gamma$. The *initial* state of N is $Q_N = (M_N, \Upsilon_0)$, where $\Upsilon_0(t) = 0$ for all $t \in T_N$.

The states of N change if time passes or if a transition fires.

In a state $Q = (M, \Upsilon)$ of N, time $\delta \in \mathbf{R}^+$ *can pass* if for all $t \in Enable(M)$, $\Upsilon(t) + \delta \leq \Upsilon_N(t)$. In this case, the state $\widetilde{Q} = (\widetilde{M}, \widetilde{\Upsilon})$ of N is *obtained by passing* δ from Q (written $Q \xrightarrow{\delta} \widetilde{Q}$), iff:

1. $\widetilde{M} = M$;
2. $\widetilde{\Upsilon} = \Upsilon + \delta$.

We consider the relation $\xrightarrow{\delta}$ as having time continuity property: $Q \xrightarrow{\delta} \widetilde{Q} \iff$ there exist δ_1, δ_2 and Q_1 s.t. $Q \xrightarrow{\delta_1} Q_1 \xrightarrow{\delta_2} \widetilde{Q}$ and $\delta_1 + \delta_2 = \delta$.

In a state $Q = (M, \Upsilon)$ of N, a transition $t \in T_N$ is *firable* if $t \in Enable(M)$, and $\Upsilon(t) = \Upsilon_N(t)$. In this case, the state $\widetilde{Q} = (\widetilde{M}, \widetilde{\Upsilon})$ of N is *obtained by firing* t from Q (written $Q \stackrel{t}{\to} \widetilde{Q}$), iff:

1. $\widetilde{M} = M \setminus {}^\bullet t \cup t^\bullet$;
2. $\forall t' \in T_N \; \widetilde{\Upsilon}(t') = \begin{cases} 0, & t' \in Enable(\widetilde{M}) \setminus Enable(M); \\ \Upsilon(t'), & \text{otherwise.} \end{cases}$

We write $Q \stackrel{a}{\to} \widetilde{Q}$ if there exists $t \in T_N$ s.t. $Q \stackrel{t}{\to} \widetilde{Q}$ and $l_N(t) = a$.

A state Q of N is *reachable* if $Q = Q_N$, or there exists a reachable state Q' of N s.t. $Q' \stackrel{x}{\to} Q$. Let $States(N)$ denote the *set of all reachable* states of N.

Example 2. Let us consider some reachable states of the timed net shown in Fig. 1. These are: $(\{p_2, p_3\}, (3.5, 3.5, 3.5, 3.5, 3.5))$, $(\{p_4, p_5\}, (4, 4, 4, 4, 0))$ and $(\{p_2, p_4\}, (3.8, 3.8, 3.8, 3.8, 3.8))$, etc.

3. Equivalences

3.1. Timed equivalences

We start with considering timed equivalences in the trace and bisimulation worlds which can measure the exact real-numbered duration of every delay.

Definition 2. A *timed trace* of N is a sequence $x_1 \cdots x_n$ s.t. $Q_N \stackrel{x_1}{\to} Q_1 \stackrel{x_2}{\to} \ldots \stackrel{x_n}{\to} Q_n$. We denote a set of *all timed traces* of N by $TimedTraces(N)$. N and N' are *timed trace equivalent*, denoted by $N \equiv_t N'$, iff $TimedTraces(N) = TimedTraces(N')$.

This means that two timed nets are timed trace equivalent, iff their timed traces are compared.

Definition 3. A relation $\mathcal{B} \subseteq States(N) \times States(N')$ is a *timed bisimulation* between N and N', denoted by $\mathcal{B} : N \underline{\leftrightarrow}_t N'$, iff:

1. $(Q_N, Q_{N'}) \in \mathcal{B}$;
2. $(Q, Q') \in \mathcal{B}$, $Q \stackrel{x}{\to} \widetilde{Q} \;\Rightarrow\; \exists \widetilde{Q}'$ s.t. $Q' \stackrel{x}{\to} \widetilde{Q}'$, $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{B}$;
3. As item 2, but the roles of N and N' are reversed.

N and N' are *timed bisimilar*, denoted by $N \underline{\leftrightarrow}_t N'$, iff $\exists \mathcal{B} : N \underline{\leftrightarrow}_t N'$.

This means that two timed nets are timed bisimilar if there exists a timed bisimulation between them, i.e., a relation between their timed bisimilar states, among which the initial ones, such that the states obtained by firing transitions with the same label or by passing the same amount of time are again timed bisimilar.

3.2. Untimed equivalences

We next define untimed versions of trace and bisimulation equivalences which abstract away from the duration of time delays. Before doing so, we need to define the following notions and notations.
Let $Q, \widetilde{Q} \in States(N)$. Then $Q \overset{t}{\mapsto} \widetilde{Q}$ (time abstracting firing), iff $Q \overset{\delta_1}{\to} Q_1 \overset{t}{\to} Q_2 \overset{\delta_2}{\to} \widetilde{Q}$ for some $Q_1, Q_2 \in States(N)$ and $\delta_1, \delta_2 \in \mathbf{R}^+$. We write $Q \overset{a}{\mapsto} \widetilde{Q}$ if $Q \overset{t}{\mapsto} \widetilde{Q}$ and $l_N(t) = a$.

Definition 4. An *untimed trace* of N is a sequence $a_1 \cdots a_n$ s.t. $Q_N \overset{a_1}{\mapsto} Q_1 \overset{a_2}{\mapsto} \ldots \overset{a_n}{\mapsto} Q_n$. We denote a set of *all untimed traces* of a timed net N by $UntimedTraces(N)$. N and N' are *untimed trace equivalent*, denoted by $N \equiv_u N'$, iff
$UntimedTraces(N) = UntimedTraces(N')$.

This means that two timed nets are untimed trace equivalent, iff their untimed traces are compared.

Definition 5. A relation $\mathcal{B} \subseteq States(N) \times States(N')$ is a *untimed bisimulation* between N and N', denoted by $\mathcal{B} : N \underline{\leftrightarrow}_u N'$, iff:

1. $(Q_N, Q_{N'}) \in \mathcal{B}$;
2. $(Q, Q') \in \mathcal{B},\ Q \overset{a}{\mapsto} \widetilde{Q} \ \Rightarrow\ \exists \widetilde{Q}'$ s.t. $Q' \overset{a}{\mapsto} \widetilde{Q}',\ (\widetilde{Q}, \widetilde{Q}') \in \mathcal{B}$;
3. As item 2, but the roles of N and N' are reversed.

N and N' are *untimed bisimilar*, denoted by $N \underline{\leftrightarrow}_u N'$, iff $\exists \mathcal{B} : N \underline{\leftrightarrow}_u N'$.

This means that two timed nets are untimed bisimilar if there exists a untimed bisimulation between them, i.e., a relation between their untimed bisimilar states, among which the initial ones, such that the states obtained by time abstracting firing transitions with the same label are again untimed bisimilar.

3.3. Region equivalences

In this section, we introduce trace and bisimulation equivalences defined in terms of regions [1]. The main idea behind a region is to partition states of a system into equivalence classes to be able to construct a finite representation of its behavior. We shall say that two states of the timed net are in the same region, i.e., they are in some sense equivalent, iff the corresponding time assignments agree on the integral parts and also on the ordering of the fractional parts. Since a value of time assignment for a transition can be arbitrary large, it is never compared with a constant greater than the time delay of the transition. The ordering of fractional parts is needed to decide which time assignment will change its integral part first. This leads to the following definion of the region of a state of the timed net.

For any $\delta \in \mathbf{R}^+$, let $\lceil\delta\rceil$ denote its integral part, and $\{\delta\}$ denote its fractional part.

Let $\Upsilon, \Upsilon' \in \Gamma$. Then Υ and Υ' are *region equivalent*, denoted by $\Upsilon \cong \Upsilon'$, iff:

1. $\forall t \in T_N$, either $\lceil\Upsilon(t)\rceil = \lceil\Upsilon'(t)\rceil$ or both $\Upsilon(t)$ and $\Upsilon'(t)$ greater than $\Upsilon_N(t)$;
2. $\forall t, t' \in T_N$ s.t. $\Upsilon(t) \leq \Upsilon_N(t)$ and $\Upsilon(t') \leq \Upsilon_N(t')$: $\{\Upsilon(t)\} \leq \{\Upsilon(t')\} \iff \{\Upsilon'(t)\} \leq \{\Upsilon'(t')\}$.
3. $\forall t \in T_N$ s.t. $\Upsilon(t) \leq \Upsilon_N(t)$: $\{\Upsilon(t)\} = 0 \iff \{\Upsilon'(t)\} = 0$;

Let $Q = (M, \Upsilon)$, $Q' = (M', \Upsilon') \in States(N)$. Then Q and Q' are *region equivalent*, denoted by $Q \cong Q'$ if $M = M'$ and $\Upsilon \cong \Upsilon'$. We define a *region* of Q as follows: $[Q] = \{Q' \in States(N) \mid Q \cong Q'\}$, i.e., $[Q]$ is an equivalence class of Q w.r.t. $\cong$. We use $RegStates(N)$ to denote the set of all regions of N. Let $R_N = [Q_N]$ be the initial region of N.

Example 3. Let us consider some timed net with four transitions and its states: $Q_1 = (M, \Upsilon_1)$, $Q_2 = (M, \Upsilon_2)$ and $Q_3 = (M, \Upsilon_3)$, with $\Upsilon_1 = (0.1, 5, 1.33, 7.42)$, $\Upsilon_2 = (0.25, 5, 1.5, 7.999)$ and $\Upsilon_3 = (0.75, 5, 1.5, 7.999)$. We then have $Q_1 \cong Q_2$, but $Q_1 \not\cong Q_3$, since the ordering requirement is not valid, because $\{\Upsilon_1(t_1)\} < \{\Upsilon_1(t_3)\}$, whereas $\{\Upsilon_3(t_1)\} > \{\Upsilon_3(t_3)\}$.

Lemma 1. *Let* $(M, \Upsilon) \in States(N)$. *Then* $\{\Upsilon(t)\} = \{\Upsilon(t')\}$ *for all* $t, t' \in T_N$.

Proof. Assume $Q = (M, \Upsilon) \in States(N)$. By the definition of reachability, we have two cases.

1. $Q = Q_N$. Then $\forall t \in T_N$ $\{\Upsilon(t)\} = 0$, by the definition of Q_N.
2. $\exists \widehat{Q} = (\widehat{M}, \widehat{\Upsilon}) \in States(N)$ s.t. $\widehat{Q} \xrightarrow{x} Q$. Then $\{\widehat{\Upsilon}(t)\} = \{\widehat{\Upsilon}(t')\}$ for all $t, t' \in T_N$. Two cases are admissible.

   (a) $x = a \in L$. This means $\widehat{Q} \xrightarrow{t} Q$ for some $t \in T_N$ with $l_N(t) = a$. Then $\{\widehat{\Upsilon}(t)\} = \{\Upsilon(t')\} = 0$ for all $t, t' \in T_N$, by the definition of $\xrightarrow{t}$.

(b) $x = \delta \in \mathbf{R}^+$. This means $\Upsilon \doteq \widehat{\Upsilon} + \delta$, due to the definition of $\xrightarrow{\delta}$. Then we have: $\{\Upsilon(t)\} = \{\widehat{\Upsilon}(t) + \delta\} = \{\{\widehat{\Upsilon}(t)\} + \{\delta\}\} = \{\{\widehat{\Upsilon}(t')\} + \{\delta\}\} = \{\widehat{\Upsilon}(t') + \delta\} = \{\Upsilon(t')\}$, for all $t, t' \in T_N$.

□

Using the lemma above, we can introduce some additional notations that will be helpful throughout the rest of this section. For $\Upsilon \in \Gamma$ and $t \in T_N$, we use $\{\Upsilon\}$ to denote $\{\Upsilon(t)\}$.

Let $(M, \Upsilon) \in States(N)$. Then we define

$$\zeta(Q) = \begin{cases} 1/2, & \text{if } \{\Upsilon\} = 0; \\ 1 - \{\Upsilon\}, & \text{otherwise.} \end{cases}$$

In order to formulate trace and bisimulation equivalences in terms of regions, we need to define some relations on regions.

Assume $R, \widetilde{R} \in RegStates(N)$. Then

$R \overset{t}{\dashrightarrow} \widetilde{R}$ iff there exist $(M, \Upsilon) \in R$, $(\widetilde{M}, \widetilde{\Upsilon}) \in \widetilde{R}$ and $\delta \geq 0$ s.t. $(M, \Upsilon) \xrightarrow{t} (\widetilde{M}, \widetilde{\Upsilon})$ and $\{(M, \Upsilon + \delta') \mid 0 \leq \delta' \leq \delta\} \subseteq R$.

We write $R \overset{a}{\dashrightarrow} \widetilde{R}$ iff there exists $t \in T_N$ s.t. $R \overset{t}{\dashrightarrow} \widetilde{R}$ and $l_N(t) = a$.

$R \overset{\surd}{\dashrightarrow} \widetilde{R}$ iff $R \neq \widetilde{R}$ and there exist $(M, \Upsilon) \in R$, $(\widetilde{M}, \widetilde{\Upsilon}) \in \widetilde{R}$ and $\delta > 0$ s.t. $\{(M, \Upsilon + \delta') \mid 0 \leq \delta' \leq \delta\} \subseteq R \cup \widetilde{R}$.

Lemma 2.

(i) $R \overset{t}{\dashrightarrow} \widetilde{R} \;\Rightarrow\; \forall Q \in R \;\exists \widetilde{Q} \in \widetilde{R}\; Q \xrightarrow{t} \widetilde{Q}$;

(ii) $R \overset{\surd}{\dashrightarrow} \widetilde{R} \;\Rightarrow\; \forall Q \in R \;\exists \widetilde{Q} \in \widetilde{R}\; Q \xrightarrow{\zeta(Q)} \widetilde{Q}$.

Proof.

(i) By the definition of $\overset{t}{\dashrightarrow}$, there exist $(M_1, \Upsilon_1) \in R$ and $(\widetilde{M_1}, \widetilde{\Upsilon}_1) \in \widetilde{R}$ s.t. $(M_1, \Upsilon_1) \xrightarrow{t} (\widetilde{M_1}, \widetilde{\Upsilon}_1)$. Take arbitrary $(M_2, \Upsilon_2) \in R$. Since $\Upsilon_1 \cong \Upsilon_2$ and t is firable in (M_1, Υ_1), then $\Upsilon_1(t) = \Upsilon_2(t) = \Upsilon_N(t)$. Hence, t is firable in (M_2, Υ_2), because $M_1 = M_2$. Assume $(M_2, \Upsilon_2) \xrightarrow{t} (\widetilde{M_2}, \widetilde{\Upsilon}_2)$. Then $\widetilde{M_1} = \widetilde{M_2}$ and $\widetilde{\Upsilon}_1 \cong \widetilde{\Upsilon}_2$, due to the definition of $\xrightarrow{t}$. Thus $(\widetilde{M_2}, \widetilde{\Upsilon}_2) \in \widetilde{R}$.

(ii) By the definition of $\overset{\surd}{\dashrightarrow}$ and ζ, there exist $Q_1 = (M_1, \Upsilon_1) \in R$ and $\widetilde{Q}_1 = (\widetilde{M_1}, \widetilde{\Upsilon}_1) \in \widetilde{R}$ s.t. $Q_1 \xrightarrow{\zeta(Q_1)} \widetilde{Q}_1$. Then $\widetilde{\Upsilon}_1 = \Upsilon_1 + \zeta(Q_1)$, by the definition of $\xrightarrow{\zeta(Q_1)}$. Take arbitrary $Q_2 = (M_2, \Upsilon_2) \in R$. Since $\Upsilon_1 \cong \Upsilon_2$ and time can pass in Q_1, then time can pass in Q_2. Hence

$Q_2 \stackrel{\zeta(Q_2)}{\rightarrow} \tilde{Q}_2 = (\widetilde{M}_2, \tilde{\Upsilon}_2)$. Then $\tilde{\Upsilon}_2 = \Upsilon_2 + \zeta(Q_2)$, by the definition of $\stackrel{\zeta(Q_2)}{\rightarrow}$. Two cases are admissible.

1. $\{\Upsilon_1\} = 0$. Then $\tilde{\Upsilon}_1 = \Upsilon_1 + \zeta(Q_1) = \lceil \Upsilon_1 \rceil + 1/2 = \lceil \Upsilon_2 \rceil + 1/2 = \Upsilon_2 + \zeta(Q_2) = \tilde{\Upsilon}_2$.
2. $\{\Upsilon_1\} \neq 0$. Then $\tilde{\Upsilon}_1 = \Upsilon_1 + \zeta(Q_1) = (\lceil \Upsilon_1 \rceil + \{\Upsilon_1\}) + (1 - \{\Upsilon_1\}) = \lceil \Upsilon_1 \rceil + 1 = \lceil \Upsilon_2 \rceil + 1 = (\lceil \Upsilon_2 \rceil + \{\Upsilon_2\}) + (1 - \{\Upsilon_2\}) = \Upsilon_2 + \zeta(Q_2) = \tilde{\Upsilon}_2$.

Clearly, $\widetilde{M_1} = \widetilde{M_2}$. Thus $\tilde{Q}_2 \in \tilde{R}$.

□

Let $L \cup \{\surd\}$ ranged over y with and without subscripts. We are now ready to define region equivalence notions in the setting of the trace and bisimulation semantics.

Definition 6. A *region trace* of N is a sequence $y_1 \cdots y_n$ s.t. $R_N \stackrel{y_1}{-\rightarrow} R_1 \stackrel{y_2}{-\rightarrow} \ldots \stackrel{y_n}{-\rightarrow} R_n$. Let $RegTraces(N)$ denote the set of *all region traces* of N.
N and N' are *region trace equivalent*, denoted by $N \equiv_r N'$, iff $RegTraces(N) = RegTraces(N')$.

This means that two timed nets are region trace equivalent, iff their region traces are compared.

Definition 7. A relation $\mathcal{B} \subseteq RegStates(N) \times RegStates(N')$ is a *region bisimulation* between N and N', denoted by $\mathcal{B} : N \underline{\leftrightarrow}_r N'$, iff:

1. $(R_N, R_{N'}) \in \mathcal{B}$;
2. $(R, R') \in \mathcal{B}$, $R \stackrel{y}{-\rightarrow} \tilde{R} \Rightarrow \exists \tilde{R}'$ s.t. $R' \stackrel{y}{-\rightarrow} \tilde{R}'$ and $[\tilde{Q}, \tilde{Q}'] \in \mathcal{B}$;
3. As item 2, but the roles of N and N' are reversed.

N and N' are *region bisimilar*, denoted by $N \underline{\leftrightarrow}_r N'$, iff $\exists \mathcal{B} : N \underline{\leftrightarrow}_r N'$.

This means that two timed nets are region bisimilar if there exists a region bisimulation between them, i.e., a relation between their bisimilar regions of states, among which the initial ones, such that the regions of states obtained by firing transitions with the same label or by passing time are again bisimilar.

4. Comparison of equivalences

In this section first we show the coincidence of the region equivalence notions with timed ones, implying simplification of timed equivalence checking.

Theorem 1. *Let $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}\}$. For time nets N and N' $N \leftrightarrow_t N' \Leftrightarrow N \leftrightarrow_r N'$.*

Proof. We shall consider the case $\leftrightarrow = \underline{\leftrightarrow}$, because the case $\leftrightarrow = \equiv$ is a simpler one.

'$\Leftarrow$' Assume $\mathcal{B} : N \underline{\leftrightarrow}_r N'$. Let $Q = (M, \Upsilon)$, $Q' = (M', \Upsilon')$, and $\widetilde{Q} = (\widetilde{M}, \widetilde{\Upsilon})$. We define a relation $\mathcal{C}$ as follows. $\mathcal{C} = \{(Q, Q') \mid (R, R') \in \mathcal{B},\ Q \in R,\ Q' \in R',\ \{\Upsilon\} = \{\Upsilon'\}\}$. Let us show $\mathcal{C} : N \underline{\leftrightarrow}_t N'$.

1. $(Q_N, Q_{N'}) \in \mathcal{C}$, by the definitions of $\mathcal{B}$ and $\mathcal{C}$.
2. Suppose $(Q, Q') \in \mathcal{C}$ and $Q \xrightarrow{x} \widetilde{Q}$. We consider two cases.
   - $x = a \in L$. By the definition of $\overset{a}{- \rightarrow}$, we have $R \overset{a}{- \rightarrow} \widetilde{R}$. Since $(R, R') \in \mathcal{B}$, due to the definition of $\mathcal{C}$, then $R' \overset{a}{- \rightarrow} \widetilde{R}'$ and $(\widetilde{R}, \widetilde{R}') \in \mathcal{B}$, due to the definition of $\mathcal{B}$. Hence $Q' \xrightarrow{a} \widetilde{Q}'$ for some $\widetilde{Q}' \in \widetilde{R}'$, according to Lemma 2(i) and the definition of $\xrightarrow{a}$. Then we have $\{\widetilde{\Upsilon}\} = \{\widetilde{\Upsilon}'\} = 0$, because $\{\Upsilon\} = \{\Upsilon'\} = 0$. Thus $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{C}$.
   - $x = \delta \in R^+$. By time continuity of $\xrightarrow{\delta}$ and the definition of ζ, we can find a sequence $Q = Q_1 \xrightarrow{\zeta(Q_1)} Q_2 \cdots Q_{k-1} \xrightarrow{\zeta(Q_{k-1})} Q_k \xrightarrow{\delta'} Q_{k+1} = \widetilde{Q}$ s.t. $\sum_{i=1}^{k-1} \zeta(Q_i) + \delta' = \delta$ and $0 \leq \delta' < \zeta(Q_k)$. Let us show by inducution on k that there exists a sequence $Q' = Q'_1 \xrightarrow{\zeta(Q_1)} Q'_2 \cdots Q'_{k-1} \xrightarrow{\zeta(Q_{k-1})} Q'_k \xrightarrow{\delta'} Q'_{k+1} = \widetilde{Q}'$ s.t. $(Q_i, Q'_i) \in \mathcal{C}$ for all $1 \leq i \leq k+1$.

$k = 1$ Two cases are admissible.

(a) $\{\Upsilon\} = 0$. Then $R \overset{\surd}{- \rightarrow} \widetilde{R}$. Since $(R, R') \in \mathcal{B}$, by the definition of $\mathcal{C}$, then $R' \overset{\surd}{- \rightarrow} [\widetilde{Q}']$ and $(\widetilde{R}, \widetilde{R}') \in \mathcal{B}$, by the definition of $\mathcal{B}$. This means $Q' \xrightarrow{\zeta(Q')} \widetilde{Q}'$ for some $\widetilde{Q}' \in \widetilde{R}'$, due to Lemma 2(ii). Since $\{\Upsilon\} = \{\Upsilon'\}$, due to the definition of $\mathcal{C}$, then $\zeta(Q) = \zeta(Q') = \delta''$, due to the definition of ζ. Hence $\{\widetilde{\Upsilon}\} = \{\widetilde{\Upsilon}'\}$, according to the definition of $\xrightarrow{\delta''}$. Thus $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{C}$.

(b) $\{\Upsilon\} \neq 0$. Then $\{\Upsilon\} = \{\Upsilon'\} \neq 0$, by the definition of $\mathcal{C}$. This means $0 \leq \delta' < \zeta(Q')$. Hence $Q' \xrightarrow{\delta'} \widetilde{Q}'$, $\widetilde{Q} \in R$ and

$\widetilde{Q}' \in \widetilde{R}'$, according to the definitions of $\mathcal{C}$ and ζ. By the definition of $\stackrel{\delta'}{\rightarrow}$, we have $\{\widetilde{\Upsilon}\} = \{\widetilde{\Upsilon}'\}$, because $\{\Upsilon\} = \{\Upsilon'\}$. Thus $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{C}$.

$k > 1$ It follows from the induction hypothesis and reasonings analogous to those in item (a).

3. Similar to item 2, but the roles of N and N' are reversed.

Thus $\mathcal{C} : N \underline{\leftrightarrow}_t N'$.

'$\Rightarrow$' Assume $\mathcal{B} : N \underline{\leftrightarrow}_t N'$. We define $\mathcal{C} = \{(R, R') \mid R = [Q],\ R' = [Q'],\ (Q, Q') \in \mathcal{B}\}$. Let us show $\mathcal{C} : N \underline{\leftrightarrow}_r N'$.

1. $(R_N, R_{N'}) \in \mathcal{C}$, by the definitions of $\mathcal{B}$ and $\mathcal{C}$.
2. Suppose $(R, R') \in \mathcal{C}$ and $R \stackrel{y}{-\!\!\rightarrow} \widetilde{R}$. Take $Q \in R$ and $Q' \in R'$ s.t. $\zeta(Q) = \zeta(Q')$. We consider two cases.
   - $y = a \in L$. According to Lemma 2(i), we have $Q \stackrel{a}{\rightarrow} \widetilde{Q}$ for some $\widetilde{Q} \in \widetilde{R}$. Since $(Q, Q') \in \mathcal{B}$, due to the definition of $\mathcal{C}$, then $Q' \stackrel{a}{\rightarrow} \widetilde{Q}'$ and $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{B}$, due to the definition of $\mathcal{B}$. Hence $R' \stackrel{a}{-\!\!\rightarrow} \widetilde{R}'$, by the definition of $\stackrel{a}{-\!\!\rightarrow}$ and $(\widetilde{R}, \widetilde{R}') \in \mathcal{C}$, by the definition of $\mathcal{C}$.
   - $y = \surd$. From Lemma 2(ii) and the definition of ζ, it follows $Q \stackrel{\zeta(Q)}{\rightarrow} \widetilde{Q}$ for some $\widetilde{Q} \in \widetilde{R}$. Since $(Q, Q') \in \mathcal{B}$, due to the definition of $\mathcal{C}$, then $Q' \stackrel{\zeta(Q)}{\rightarrow} \widetilde{Q}'$ and $(\widetilde{Q}, \widetilde{Q}') \in \mathcal{B}$, due to the definition of $\mathcal{B}$. Hence $R' \stackrel{\surd}{\rightarrow} \widetilde{R}'$, by the definitions of $\stackrel{\surd}{-\!\!\rightarrow}$ and ζ, and $(\widetilde{R}, \widetilde{R}') \in \mathcal{C}$, by the definition of $\mathcal{C}$.
3. Similar to item 2, but the roles of N and N' are reversed.

Thus $\mathcal{C} : N \underline{\leftrightarrow}_r N'$. □

The following theorem establishes the interrelations between the equivalence relations defined prior to that. It states that all interrelations of the equivalences may be depicted by arrows of directed graph in Figure 2, and no additional non-trivial arrow (which cannot be obtained on the basis of existing implications) may be added. Hence, one equivalence implies an other one iff there exists a directed path from one equivalence to second one in this graph.

Theorem 2. *Let* $\leftrightarrow, \twoheadleftarrow\!\!\twoheadrightarrow \in \{\equiv_t, \underline{\leftrightarrow}_t, \equiv_u, \underline{\leftrightarrow}_u, \simeq\}$. *For time nets* N *and* N' $N \leftrightarrow N' \Rightarrow N \twoheadleftarrow\!\!\twoheadrightarrow N'$ *iff in the graph in Figure 2 there exists a directed path from* $\leftrightarrow$ *to* $\twoheadleftarrow\!\!\twoheadrightarrow$.

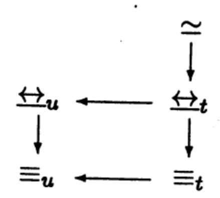

Figure 2. Interrelations of equivalences

Proof. ($\Leftarrow$) Let us prove that all implications are valid.

- The implications $\leftrightarrow_t \rightarrow \leftrightarrow_u$, $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}\}$, are valid, since the time abstracting equivalences are weaker than time-sensitive ones.
- The implications $\underline{\leftrightarrow}_\star \rightarrow \equiv_\star$, $\star \in \{t, u\}$, are valid, since bisimulation equivalences imply trace ones.
- The implication $\simeq \rightarrow \underline{\leftrightarrow}_t$ is obvious.

($\Rightarrow$) Let us prove that there exist no additional non-trivial implications.

- In Figure 3(a), $N \underline{\leftrightarrow}_u N'$ but $N \not\equiv_t N'$, since only in N' one time unit can pass before an occurrence of an action a.
- In Figure 3(b), $N \equiv_t N'$ but $N \underline{\not\leftrightarrow}_u N'$, since only in N' an action a can happen after one time unit so that an action b cannot happen afterward.
- In Figure 3(c), $N \underline{\leftrightarrow}_t N'$ but $N \not\simeq N'$, since the upper transitions of N and N' are labelled by different actions (a and b).

□

5. Untimed nets

In this section first we introduce a subclass of timed nets in which all time delays associated with transitions are equivalent to zero's.

Definition 8. A *untimed net* is a timed net $N = \langle P_N, T_N, F_N, l_N, M_N, \Upsilon_N \rangle$ s.t. $\forall t \in T_N \ \Upsilon_N(t) = 0$.

For untimed nets the coincidence of the timed and untimed equivalences is established, reported in the following theorem.

Theorem 3. *Let* $\leftrightarrow \in \{\equiv, \underline{\leftrightarrow}\}$. *For untimed nets N and N'* $N \leftrightarrow_u N' \Leftrightarrow N \leftrightarrow_t N'$.

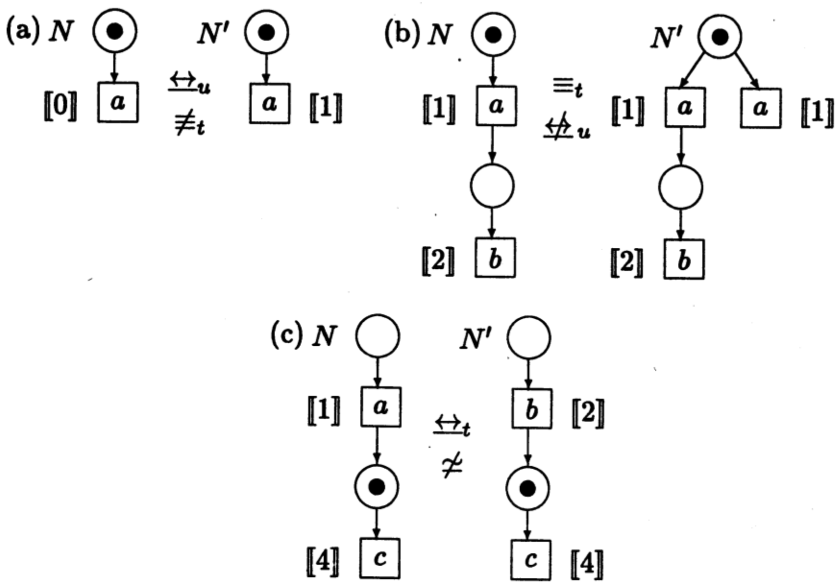

Figure 3. Examples of the equivalences

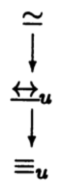

Figure 4. Interrelations of equivalences on untimed nets

Proof. Obvious, since time cannot pass in untimed nets. □

Theorem 4. *Let* $\leftrightarrow, \leftrightarrow\!\!\!\!\leftrightarrow \in \{\equiv_u, \underline{\leftrightarrow}_u, \simeq\}$. *For untimed nets* N *and* N' $N \leftrightarrow N' \Rightarrow N \leftrightarrow\!\!\!\!\leftrightarrow N'$ *iff in the graph in Figure 4 there exists a directed path from* $\leftrightarrow$ *to* $\leftrightarrow\!\!\!\!\leftrightarrow$.

Proof. ($\Leftarrow$) By Theorem 1.

($\Rightarrow$)

- Consider the untimed nets $\overline{N}$ and $\overline{N}'$ being obtained from the timed nets N and N' in Figure 3(b) by letting all the time delays equal to zero. We have $\overline{N} \equiv_u \overline{N'}$ but $\overline{N} \underline{\not\leftrightarrow}_u \overline{N'}$.
- Consider the untimed nets $\overline{N}$ and $\overline{N}'$ being obtained from the timed nets N and N' in Figure 3(c) by letting all the time delays equal to zero. We have $\overline{N} \underline{\leftrightarrow}_u \overline{N'}$ but $\overline{N} \not\simeq \overline{N'}$.

□

6. The timed SM-refinement

One of the most important features of an equivalence notion is its stability over an action refinement. Since we introduce a number of equivalences, it is interesting to see whether or not they are preserved under this operation. In our context, this means that if two timed nets are equivalent in some sence and we transform them accordingly, whether or not the transformed timed nets will be again equivalent in the same sense. Incorporating a time notion into SM-refinement [4], we consider the refinement of the net transitions with the fixed label and time delay by timed state-machine nets.

Definition 9. *Timed SM-net* is a timed net $D = \langle P_D, T_D, F_D, l_D, M_D, \Upsilon_D \rangle$ s.t.:

1. $\forall t \in T_D \ |{}^\bullet t| = |t^\bullet| = 1$,
   i.e., each transition has exactly one input and one output place;
2. $\exists p_{in},\ p_{out} \in P_D$ s.t. $p_{in} \neq p_{out}$ and ${}^\circ D = \{p_{in}\},\ D^\circ = \{p_{out}\}$,
   i.e., D has a unique input and a unique output place;
3. $M_D = \{p_{in}\}$,
   i.e., at the beginning there is a unique token in p_{in}.
4. for any two timed traces $\delta_1 t_1 \cdots \delta_n t_n$ and $\delta'_1 t'_1 \cdots \delta'_m t'_m$ of D s.t. $Q_D \stackrel{\delta_1}{\rightarrow} \widehat{Q}_1 \stackrel{t_1}{\rightarrow} \ldots \stackrel{\delta_n}{\rightarrow} \widehat{Q}_n \stackrel{t_n}{\rightarrow} Q_n = (\{p_{out}\}, \Upsilon_n)$ and $Q_D \stackrel{\delta'_1}{\rightarrow} \widehat{Q}'_1 \stackrel{t'_1}{\rightarrow} \ldots \stackrel{\delta'_n}{\rightarrow} \widehat{Q}'_n \stackrel{t'_n}{\rightarrow} Q'_n = (\{p_{out}\}, \Upsilon'_n)$, we have $\sum_{i=1}^{n} \Upsilon_D(t_i) = \sum_{j=1}^{m} \Upsilon_D(t'_j)$. Let us denote this constant sum by $\Upsilon(D)$.

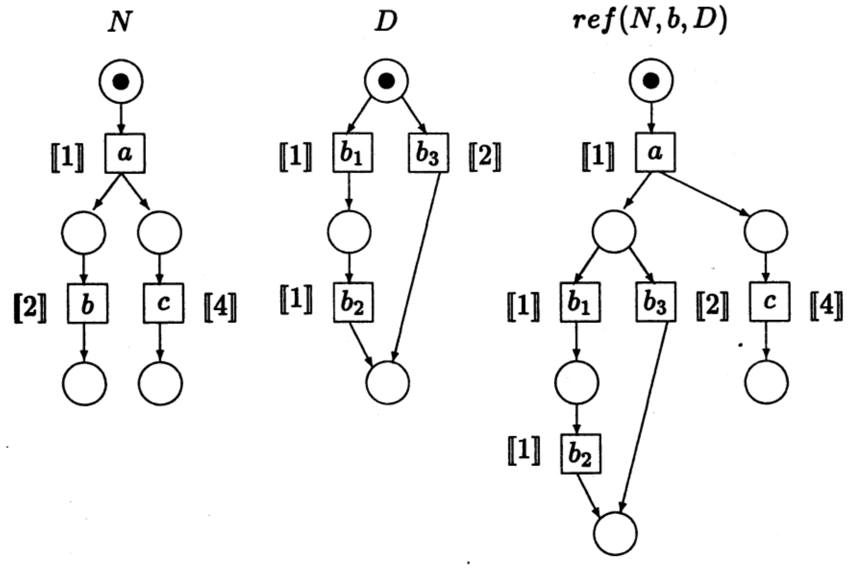

Figure 5. An example of an application of the timed SM-refinement

Definition 10. Let $D = \langle P_D, T_D, F_D, l_D, M_D, \Upsilon_D\rangle$ be a timed SM-net, $a \in l_N(T_N)$ and $T = \{u \in T_N \mid (l_N(u) = a) \wedge (\Upsilon_N(u) = \Upsilon(D))\}$. *The timed SM-refinement*, denoted by $ref(N, a, D)$, is (up to isomorphism) the timed net $\overline{N} = \langle P_{\overline{N}}, T_{\overline{N}}, F_{\overline{N}}, l_{\overline{N}}, M_{\overline{N}}, \Upsilon_{\overline{N}}\rangle$, where:

- $P_{\overline{N}} = P_N \cup \{\langle p, u\rangle \mid p \in P_D \setminus \{p_{in}, p_{out}\},\ u \in T\}$;
- $T_{\overline{N}} = (T_N \setminus T) \cup \{\langle t, u\rangle \mid t \in T_D,\ u \in T\}$;
- $F_{\overline{N}}(\bar{x}, \bar{y}) = \begin{cases} F_N(\bar{x}, \bar{y}), & \bar{x}, \bar{y} \in P_N \cup (T_N \setminus T); \\ F_D(x, y), & \bar{x} = \langle x, u\rangle,\ \bar{y} = \langle y, u\rangle,\ u \in T; \\ F_N(\bar{x}, u), & \bar{y} = \langle y, u\rangle,\ \bar{x} \in {}^\bullet u,\ u \in T,\ y \in p_{in}^\bullet; \\ F_N(u, \bar{y}), & \bar{x} = \langle x, u\rangle,\ \bar{y} \in {}^\bullet u,\ u \in T,\ x \in {}^\bullet p_{out}; \\ 0, & \text{otherwise}; \end{cases}$
- $l_{\overline{N}}(\bar{t}) = \begin{cases} l_N(\bar{t}), & \bar{t} \in T_N \setminus T; \\ l_D(t), & \bar{t} = \langle t, u\rangle,\ t \in T_D,\ u \in T; \end{cases}$
- $M_{\overline{N}} = M_N$;
- $\Upsilon_{\overline{N}}(\bar{t}) = \begin{cases} \Upsilon_N(\bar{t}), & \bar{t} \in T_N \setminus T; \\ \Upsilon_D(t), & \bar{t} = \langle t, u\rangle,\ t \in T_D,\ u \in T. \end{cases}$

Example 4. Fig. 5 demonstrates an application of the timed SM-refinement.

Now we consider the question of preservation of the equivalences introduced above by the timed SM-refinements.

Example 5.

- In Figure 6 $N \underline{\leftrightarrow}_t N'$, but $ref(N, a, D) \not\equiv_u ref(N', a, D)$, since only in $ref(N', a, D)$ there is not the untimed trace $a_1 b a_2$. Hence, the equivalences between $\equiv_u$ and $\underline{\leftrightarrow}_t$ are not preserved by the timed SM-refinements. The main reason of this is that in N' a transition with a label a is in conflict with an other transition.

- In Figure 7 $N \underline{\leftrightarrow}_t N'$, but $ref(N, a, D) \not\equiv_u ref(N', a, D)$, since only in $ref(N', a, D)$ there is not untimed trace $a_1 a_1$. Hence, the equivalences between $\equiv_u$ and $\underline{\leftrightarrow}_t$ are not preserved by the timed SM-refinements. The main reason of this is that in N two transitions with label a can fire concurrently.

A careful examination of the above examples shows that the problem here arises from the fact that we applied the timed SM-refinement to transitions labelled the actions deciding choices or involved in autoconcurrency. demonstrate that the timed SM-refinement should not replace transitions which are in conflict with other ones or may be fired together with other transitions having the same label, since otherwise most of the equivalences are not preserved by refinements. Similar problems were discussed [6] in the framework of event structures. To overcome these difficulties, a new notion of the safe refinement was proposed in that paper. Now we apply this idea to timed Petri nets.

An action $a \in L$ is said to be *conflicting* if there is $Q \in States(N)$, such that two transitions $t, t' \in T_N$ (necessary distinct) are firable in Q, $l_N(t) = a$ and ${}^\bullet t \cap {}^\bullet t' \neq \emptyset$.

An action $a \in L$ is said to be *autoconcurrent* if there is $Q \in States(N)$, such that two transitions $t, t' \in T_N$ (necessary distinct) are firable in Q, $l_N(t) = l_N(t') = a$ and ${}^\bullet t \cap {}^\bullet t' = \emptyset$.

We define the *safe* timed SM-refinement operation, $sref$, which differs from the timed SM-refinement one by the additional requirement to refine only transitions which are not labelled by conflicting or autoconcurrent actions.

The following two theorems demonstrate how the considered equivalences behave by the safe timed SM-refinements.

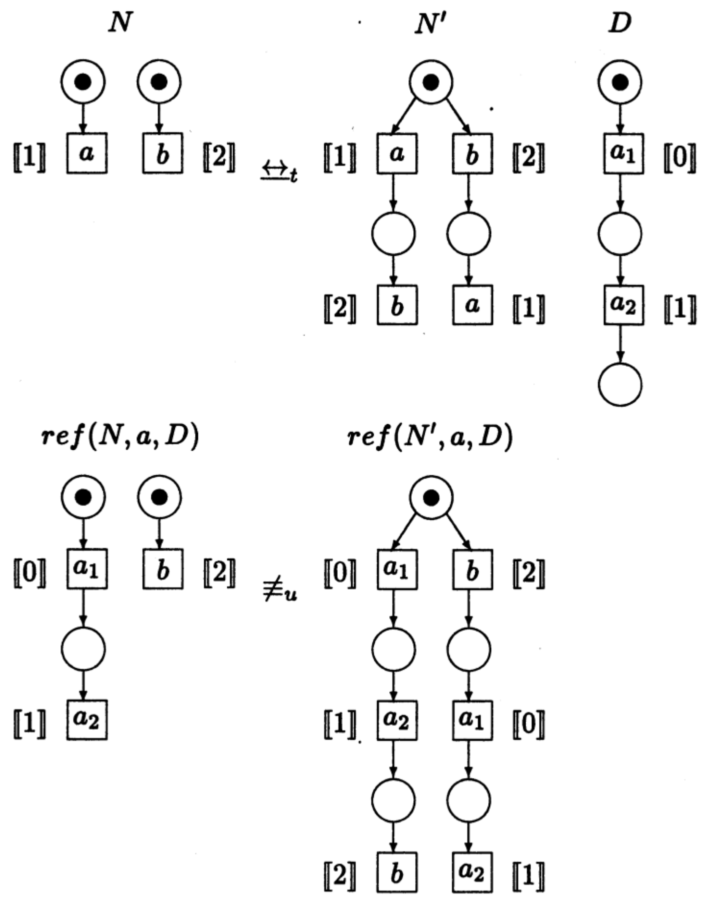

Figure 6. The equivalences between $\equiv_u$ and $\underline{\leftrightarrow}_t$ are not preserved by the timed **SM**-refinements (conflict)

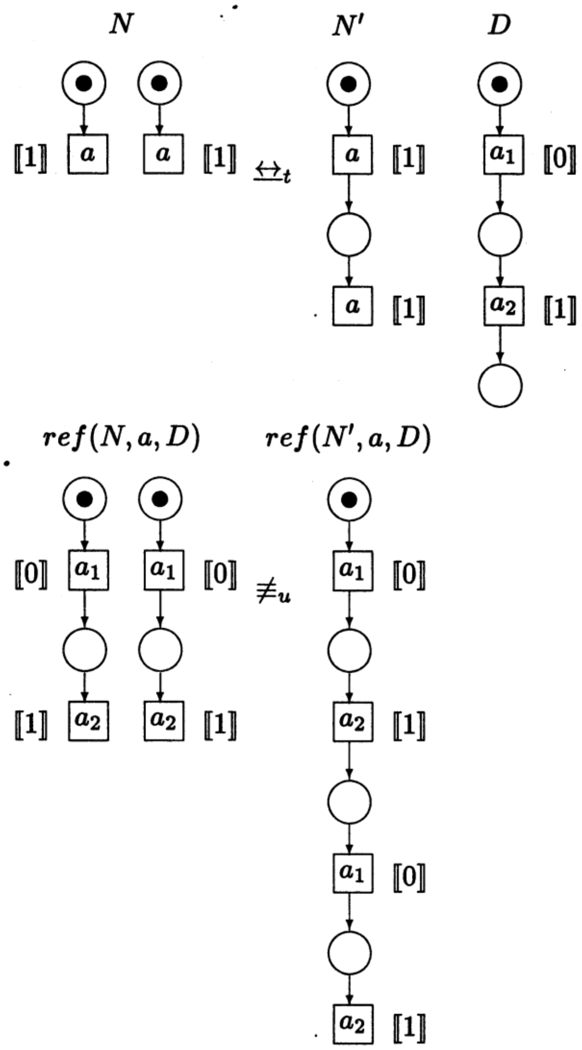

Figure 7. The equivalences between $\equiv_u$ and $\underline{\leftrightarrow}_t$ are not preserved by the timed SM-refinements (autoconcurrency)

Theorem 5. *The equivalences $\equiv_t, \equiv_u, \underline{\leftrightarrow}_u$ are not preserved by the safe timed SM-refinements.*

Proof.

- In Figure 8 $N \equiv_t N'$, but $sref(N, b, D) \not\equiv_u sref(N', b, D)$, since only in $sref(N', b, D)$ there is not untimed the ab_1c. Hence, the equivalences $\equiv_u$ and $\equiv_t$ are not preserved by the safe timed SM-refinements.
- In Figure 9 $N \underline{\leftrightarrow}_u N'$, but $sref(N, a, D) \not\equiv_u sref(N', a, D)$, since only in $sref(N', a, D)$ action b can happen. Hence, the equivalences $\equiv_u$ and $\underline{\leftrightarrow}_u$ are not preserved by the safe timed SM-refinements.

□

Theorem 6. *Let N and N' be timed nets s.t. $a \in l_N(T_N) \cap l_{N'}(T_{N'})$ and D be a timed SM-net. Then $N \underline{\leftrightarrow}_t N' \Rightarrow sref(N, a, D) \underline{\leftrightarrow}_t sref(N', a, D)$.*

Proof. Let us present the main stages of the proof.

1. Let us note that any SM-net may be combined from elementary SM-nets (consisting of one transition with the only input and the only output place) with use of alternative (choice) and sequential composition operations.

   Hence, a refinement by general time SM-net may be replaced by sequence of *simple* time SM-refinements: *renaming*, *simple choice* and *simple splitting*, which substitute transitions by time SM-nets D_1, D_2 и D_3, depicted in Figure 10 respectively. Then the requirement 4 from general time SM-net is turned into the following conditions.

   - For $D_1 : \Upsilon_{D_1}(t) = \Upsilon(D_1)$.
   - For $D_2 : \Upsilon_{D_2}(t_1) = \Upsilon_{D_2}(t_2) = \Upsilon(D_2)$.
   - For $D_3 : \Upsilon_{D_3}(t_1) + \Upsilon_{D_3}(t_2) = \Upsilon(D_3)$.

   Hence, we can consider only simple time SM-refinements in the proof.
2. After some substantiations, one can see that the only type of simple time SM-refinement which is sufficient for the proof is simple splitting. Hence, we can consider only such a refinement in the proof.
3. The rest of the proof is based on the following fact. If one net does not model a behavior of another net after application of simple splitting, then one of these refined nets has conflicting or autoconcurrent actions. We have contradiction with construction of time SM-refinement.

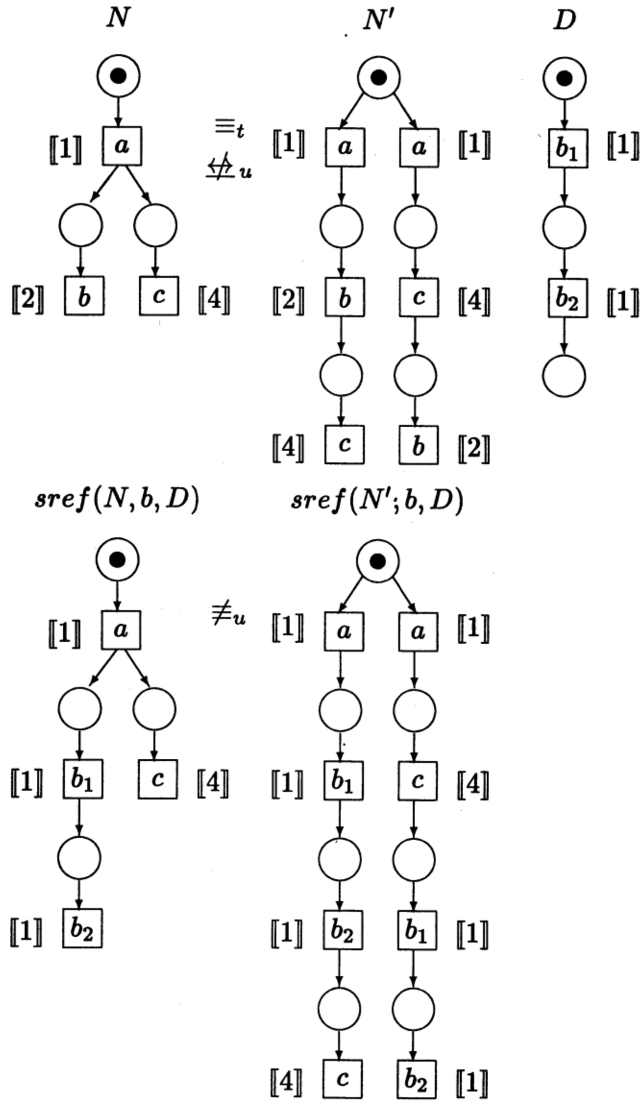

Figure 8. The equivalences $\equiv_u$ and $\equiv_t$ are not preserved by the safe timed SM-refinements

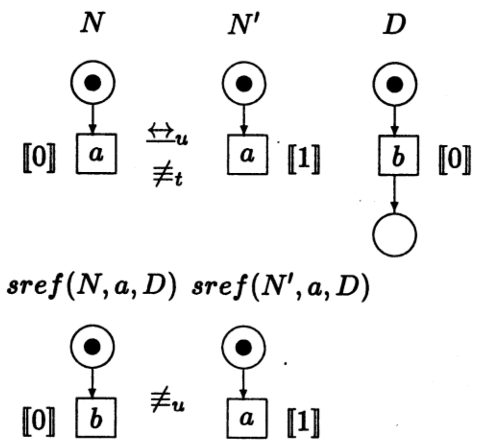

Figure 9. The equivalences $\equiv_u$ and $\underline{\leftrightarrow}_u$ are not preserved by the safe timed SM-refinements

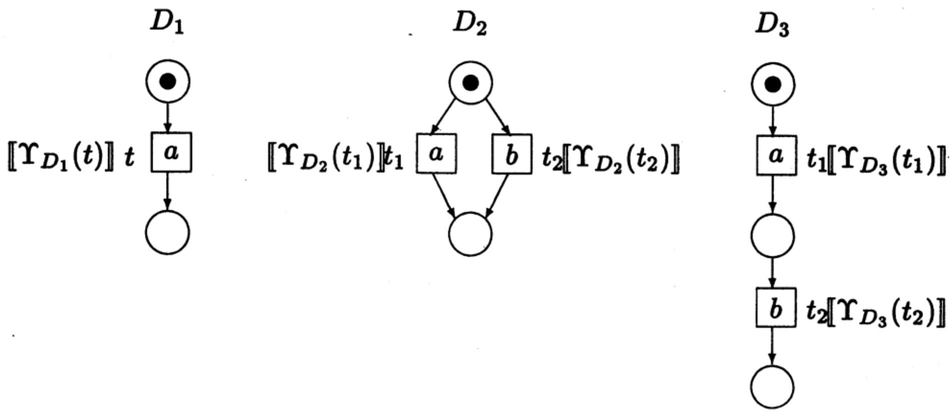

Figure 10. Time SM-nets of renaming, simple choice and simple splitting

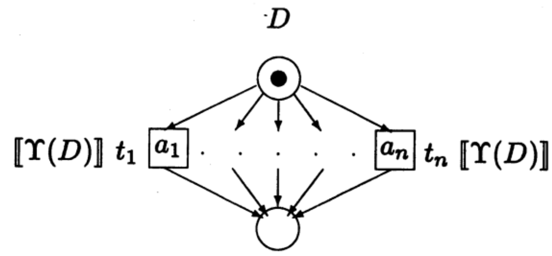

Figure 11. An example of a timed n-choice net

□

It is worth nothing that the refinement theorem above is also valid for an extension of the safe timed SM-refinement by letting replacement of transitions with conflicting and autoconcurrent labels by a special kind of timed SM-nets — *timed n-choice nets* (see Figure 11).

7. Conclusion

In this paper we introduced and investigated rather a complete set of equivalence notions for timed nets with silent actions. All the equivalences were compared on a general class of timed nets and their subclass of untimed nets, resulting in lattices of implications. The coincidence of timed and region variants of equivalences was established, implying simplification of timed equivalence checking. The equivalences were treated for preservation by a new operation of the timed SM-refinement, and the only candidate, timed bisimulation equivalence ($\underline{\leftrightarrow}_t$), that may be useful for the multilevel design, was found. All these results provide us a basis for behavioral reasoning about concurrent systems with time delays represented by timed Petri nets.

A further development may consists in an attempt to introduce time-sensitive equivalence notions in the linear-branching time spectrum, e.g., failure and testing semantics. It is also worth extending the results obtained to timed nets with interval time delays which are a more expressive formalism that timed nets with constant time delays.

References

[1] R. Alur, C. Courcoubetis, D. Dill, *Model-checking for real-time systems*, Proceedings of LICS'90, 1990, 414–425.

[2] R. Alur, C. Courcoubetis, T.A. Henzinger, *The observational power of clocks*, LNCS 836, 1994, 162–177.

[3] R. Alur R, T.A. Henziger, *Logics and models of real time: a survey*, LNCS 600, 1992, 74–106.

[4] E. Best, R. Devillers, A. Kiehn, L. Pomello, *Concurrent bisimulations in Petri nets*, Acta Informatica, **28**, 1991, 231–264.

[5] K. Čerāns, *Decidability of bisimulation equivalences for parallel timer processes*, LNCS 663, 1993, 302–315.

[6] I. Czaja, R.J. van Glabbeek, U. Goltz, *Interleaving semantics and action refinement with atomic choice*, LNCS 609, 1992, 89–107.

[7] M. Felder, A. Morzenti, *A temporal logic approach to implementation and refinement in timed Petri nets*, LNCS 575, 1994, 365–381.

[8] R.J. van Glabbeek, *The linear time – branching time spectrum II: the semantics of sequential systems with silent moves. Extended abstract*, LNCS 715, 1993, 66–81.

[9] P. Merlin, D.J. Farber, *Recoverability of communication protocols*, IEEE Transactions on Communication Protocols COM-24(9), 1976.

[10] S. Tripakis, S. Yovine, *Analysis of timed systems based on time-abstracting bisimulations*, LNCS 967, 1996, 232–243.

[11] C. Weise, D. Lenzkes, *Efficient scaling-invariant checking of timed bisimulation*, LNCS 1200, 1997, 177–188.

[12] W. Yi, *Real time behaviour of asynchronous agents*, LNCS 458, 1990, 502–520.

[13] T. Yoneda, A. Shibayama, B.-H. Schlingloff, E.M. Clarke, *Efficient verification of parallel real-time systems*, LNCS 697, 1993, 321–332.