Joint NCC & IIS Bull., Comp. Science, 18 (2002), 27–43

Timed testing for models with internal actions

E. N. Bozhenkova

In the paper, we construct a formula that characterizes a timed event structure with discrete internal actions up to the timed *must*-preorder.

1. Introduction

A notion of equivalence is an important part of every process theory. As a matter of fact, equivalences are used in 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 have been proposed, and the relationship between them has been quite well-understood (see, for example, [13]).

The major equivalences include testing ones presented in [12]. Two processes are considered to be testing equivalent, if there is no test that can distinguish them. A test itself is usually a process applied to another process by computing them together in parallel. A particular computation is considered to be successful, if the test reaches a designated successful state, and the process passes the test if every computation is successful. This notion is intuitively appealing; it 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 a decision algorithm for finite-state processes. Therefore, testing decision procedures are based on reduction of testing to bisimulation [8]. These equivalences have been considered for synchronous and asynchronous formal system models without time delays [1, 7, 8, 12, 14].

Testing equivalences have also been developed for models with time (see, for example, [4, 10, 11, 15, 18, 21, 22]) and with probability ([9, 17, 20],). Papers [10, 18], and [22] have treated timed testing for discrete and dense time transition models, respectively. The alternative characterization of timed testing given in these papers uses a notion similar to that of an acceptance set in the testing theory. The paper [22] tries to provide a testing decision procedure that uses the untimed bisimulation between deterministic graphs built from mutually refined timer region graphs that are a finite abstraction of the operational semantics of the model under consideration. Papers

[15] and [11] have investigated timed testing for the discrete and dense time process algebra. In a model of [11], time from the interval $[0, 1]$ associated with an action means the latest time of an action being performed. They prove that, in the context of that model, dense-timed testing is reduced to discrete-time one.

In paper [4], a framework for testing preorders and equivalences in the setting of timed event structures has been developed. In that model, a time interval associated with an event means the interval during which the event can occur. Occurrence of the event does not take any time. The model is a timed generalization of Winskel's prime event structures [23]. We have chosen it instead of [5] and [16] because we can give the notions of a state and a leading relation. As for the characterization and decision procedure, it turns out that the results mentioned above ([11, 22]) were not the case for some timed event structures. So, we try to give the alternative characterization of the timed testing relations. Moreover, we have found a subclass of structures in which we could reduce timed testing relations to the corresponding variants of symbolic bisimulations.

In [6], the problem of decidability of timed *must*-equivalences is reduced to the model-checking one. As a basic logic, we take the timed logic L_ν. This logic has been defined in [19] and used for construction of a characteristic formula for a timed automaton up to the timed bisimilarity and, as a consequence, for reduction of the timed bisimilarity decidability problem to the model-checking one. It is known that the latter problem is decidable ([2, 3]). We construct a characteristic formula up to the timed *must*-preorders. We do it only for timed event structures without internal actions.

This paper is devoted to construction of the characteristic formula for timed event structures with discrete internal actions. Existence of internal actions gives increment of states reachable by the same word. We try to collect them together in a class (the τ-closure of regions) and to unite classes into a class graph. And only then we construct the formulas for each class of the class graph.

The rest of the paper is organized as follows. In Section 2, we remind the basic notions concerned with timed event structures and timed testing. The timed modal logic L_ν is described in Section 3. In Section 4, we obtain a class graph from the state-space. In Section 5, we construct a formula which characterizes a timed event structure up to the timed *must*-preorders.

2. Timed event structures

In this section, we introduce a model of timed event structures, that is a real-time extension of Winskel's model of prime event structures [23] by equipping events with time intervals.

We first recall a notion of an event structure. The main idea behind event structures is to view the distributed computations as action occurrences, called events, together with a notion of causal dependence between events (which are reasonably characterized via a partial order). Moreover, to model nondeterminism, there is a notion of conflicting (mutually incompatible) events. A labelling function determines which action corresponds to an event.

Let Act be a finite set of visible actions and τ be an internal action. Then $Act_\tau = Act \cup \{\tau\}$.

Definition 1. A (*labelled*) *event structure* over Act_τ is a 4-tuple $S=(E,\leq,\#,l)$, where

- E is a countable set of events;
- $\leq \subseteq E \times E$ is a partial order (the *causality relation*) satisfying the *principle of finite causes*: $\forall e \in E \;.\; \{e' \in E \mid e' \leq e\}$ is finite;
- $\# \subseteq E \times E$ is a symmetric and irreflexive relation (the *conflict relation*) satisfying the *principle of conflict heredity*: $\forall e, e', e'' \in E \;.\; e \# e' \leq e'' \Rightarrow e \# e''$;
- $l : E \to Act_\tau$ is a labelling function.

Let $C \subseteq E$. Then C is *left-closed* iff $\forall e, e' \in E \;.\; e \in C \;\wedge\; e' \leq e \Rightarrow e' \in C$; C is *conflict-free* iff $\forall e, e' \in C \;.\; \neg(e \# e')$; C is a *configuration of* S iff C is left-closed and conflict-free. Let $Conf(S)$ denote the set of all configurations of S. For $C \in Conf(S)$, we define the set of events enabled in C as $En(C) = \{e \in E \mid C \cup \{e\} \in Conf(S)\}$.

In the following, we will consider only finite event structures, i.e., the structures whose sets of events are finite.

Before introducing the concept of a timed event structure, we need to propose some auxiliary notations. Let $\mathbf{N}_0$ be the set of natural numbers with zero, $\mathbf{R}^+$ be the set of positive real numbers, and $\mathbf{R}_0^+$ be the set of nonnegative real numbers. For any $d \in \mathbf{R}_0^+$, $\{d\}$ denotes its fractional part, $\lfloor d \rfloor$ and $\lceil d \rceil$ — its smallest and largest integer parts, respectively.
Define the set $Interv(\mathbf{R}_0^+) = \{(d_1, d_2), (d_1, d_2], [d_1, d_2), [d_1, d_2] \subset \mathbf{R}_0^+ \mid d_1, d_2 \in \mathbf{N}_0\}$.

We are now ready to introduce the concept of timed event structures.

Definition 2. A (*labelled*) *timed event structure* over Act_τ is a pair $TS = (S, D)$, where

- $S = (E, \leq, \#, l)$ is a (labelled) event structure over Act_τ;
- $D : E \to Interv(\mathbf{R}_0^+)$ is a timing function such that $D(e)$ is a closed interval from $Interv(\mathbf{R}_0^+)$ for all $e \in E$ with $l(e) \in Act$.

In a graphic representation of a timed event structure, the corresponding action labels and time intervals are drawn close to events. If no confusion arises, we will often use action labels instead of the event identifiers to denote events. The $<$-relations are depicted by arcs (omitting those derivable by transitivity), and conflicts are depicted by "#" (omitting those derivable by the conflict heredity). Following these conventions, a trivial example of a labelled timed event structure is shown in Figure 1.

$$
\begin{array}{llcl}
TS_1 \qquad & [0,1]\ a : e_1 & \longrightarrow & b : e_2\ \ [0,1] \\
 & & & \quad \# \\
 & & & \tau : e_3\ \ [0,1)
\end{array}
$$

Figure 1. A simple example

Let $\mathcal{E}_\tau$ denote the set of all labelled timed event structures over Act_τ. For convenience, we fix timed event structures $TS = (S = (E, \leq, \#, l), D)$, $TS' = (S' = (E', \leq', \#', l'), D')$ from the class $\mathcal{E}_\tau$ and work with them further.

A *state* of TS is a pair $M = (C, \delta)$, where $C \in Conf(S)$ and $\delta : E \to \mathbf{R}_0^+$. The *initial state* of TS is $M_{TS} = (C_0, \delta_0) = (\emptyset, 0)$. A state $M = (C, \delta)$ is said to be *terminated*, if $En(C) = \emptyset$. Let $ST(TS)$ denote the set of all states of TS.

A timed event structure progresses through a sequence of states in one of two ways given below.

Let $M_1 = (C_1, \delta_1), M_2 = (C_2, \delta_2) \in ST(TS)$ such that M_1 is a non-terminated state. An event $e \in En(C_1)$ *may occur* in M_1 (denoted $M_1 \stackrel{e}{\to}$) if $\delta_1(e) \in D(e)$ and $\forall e' \in En(C_1)\ \exists d \in \mathbf{R}_0^+\ .\ \delta_1(e') + d \in D(e)$. We write $M_1 \stackrel{a}{\to}$, if $M_1 \stackrel{e}{\to}$ and $l(e) = a$. The occurrence of e in M_1 *leads* to M_2 (denoted $M_1 \stackrel{e}{\to} M_2$), if $M_1 \stackrel{e}{\to}$, $C_2 = C_1 \cup \{e\}$ and

$$\delta_2(e') = \begin{cases} 0, & \text{if } e' \in En(C_2) \setminus En(C_1); \\ \delta_1(e'), & \text{otherwise.} \end{cases}$$

We write $M_1 \stackrel{a}{\to} M_2$, if $M_1 \stackrel{e}{\to} M_2$ and $l(e) = a$.

A time $d \in \mathbf{R}^+$ *may pass* in M_1 (denoted $M_1 \stackrel{d}{\to}$), if $\forall e \in En(C_1)\ \exists d' \in \mathbf{R}_0^+ (d' \geq d)\ .\ \delta_1(e) + d' \in D(e)$. The passage d in M_1 *leads* to M_2 (denoted $M_1 \stackrel{d}{\to} M_2$), if $C_2 = C_1$ and $\delta_2(e) = \delta_1(e) + d$ for all $e \in E$.

The *weak leading relation* $\Rightarrow$ *on states of* TS is the largest relation defined by: $\stackrel{\epsilon}{\Rightarrow} \Longleftrightarrow \stackrel{\tau}{\to}^*$ and $\stackrel{x}{\Rightarrow} \Longleftrightarrow \stackrel{\epsilon}{\Rightarrow}\stackrel{x}{\to}\stackrel{\epsilon}{\Rightarrow}$, 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: $M \stackrel{d_1+d_2}{\Longrightarrow} \Longleftrightarrow M \stackrel{d_1}{\Rightarrow}\stackrel{d_2}{\Rightarrow}$ for some $d_1, d_2 \in \mathbf{R}^+$.

From now on, we will use the following notions and notations. Let $Act(\mathbf{R}_0^+) = \{a(d) \mid a \in Act \wedge d \in \mathbf{R}_0^+\}$ 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 $\triangle : (Act(\mathbf{R}_0^+))^* \to \mathbf{R}_0^+$ measuring the *duration* of a timed word is defined by: $\triangle(\epsilon) = 0$, $\triangle(w.a(d)) = \triangle(w) + d$. The domain for real-time languages is denoted by $Dom(Act, \mathbf{R}_0^+) = \{\langle w, d\rangle \mid w \in (Act(\mathbf{R}_0^+))^*, d \in \mathbf{R}_0^+, d \geq \triangle(w)\}$. The weak leading relation $\Rightarrow$ is extended to timed words from $(Act(\mathbf{R}_0^+))^*$ and $Dom(Act, \mathbf{R}_0^+)$ as follows. Let $d \in \mathbf{R}_0^+$, $d' \in \mathbf{R}^+$, $a \in Act$ and $w \in (Act(\mathbf{R}_0^+))^*$. Then

$$\text{if } M \stackrel{a}{\Rightarrow} M', \text{ then } M \stackrel{a(0)}{\Rightarrow} M'; \text{ if } M \stackrel{d'}{\Rightarrow}\stackrel{a}{\Rightarrow} M', \text{ then } M \stackrel{a(d')}{\Rightarrow} M';$$
$$\text{if } M \stackrel{w}{\Rightarrow}\stackrel{a(d)}{\Rightarrow} M', \text{ then } M \stackrel{w.a(d)}{\Longrightarrow} M'; \text{ if } M \stackrel{w}{\Rightarrow} M', \text{ then } M \stackrel{\langle w,\, \triangle(w)\rangle}{\Longrightarrow} M';$$
$$\text{if } M \stackrel{\langle w,d\rangle}{\Longrightarrow}\stackrel{d'}{\Rightarrow} M', \text{ then } M \stackrel{\langle w,\, d+d'\rangle}{\Longrightarrow} M'.$$

The set $L(TS) = \{\langle w, d\rangle \in Dom(Act, \mathbf{R}_0^+) \mid M_{TS} \stackrel{\langle w,d\rangle}{\Longrightarrow}\}$ is the *language* of TS. For instance, for the timed event structure TS_1 in Figure 1 we have $L(TS_1) = \{\langle \epsilon, d_1\rangle, \langle \epsilon, 1\rangle, \langle a(d_1), d_1 + d_2\rangle, \langle a(1), 1\rangle, \langle a(d_1)b(d_2), d_1 + d_2\rangle \mid d_1 + d_2 < 1\}$.

The timed testing relations may be defined in terms of the responses of timed event structures to a collection of tests. We will, however, use an alternative characterization that relies on the following definitions. Let $M \in ST(TS)$ and $\langle w, d\rangle \in Dom(Act, \mathbf{R}_0^+)$. Then $S(M) = \{x \in Act_\tau \cup \mathbf{R}^+ \mid M \stackrel{x}{\to}\}$ and $Acc(TS, \langle w, d\rangle) = \{S(M') \mid M_{TS} \stackrel{\langle w,\, d\rangle}{\Longrightarrow} M',\ M' \stackrel{\tau}{\not\to}\}$ (timed acceptance set). Let $N, N' \subset 2^{Act \cup \mathbf{R}^+}$. Then $N \subset\subset N' \iff \forall S \in N\ \exists S' \in N' . [(S' \mid_{Act} \subseteq S \mid_{Act}) \wedge (S \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S' \mid_{\mathbf{R}^+} = \emptyset)]$; $N \equiv N' \iff N \subset\subset N' \wedge N' \subset\subset N$.

Definition 3.

- $TS \leq_{must} TS' \Longleftrightarrow \forall \langle w, d\rangle \in Dom(Act, \mathbf{R}_0^+) . Acc(TS', \langle w, d\rangle) \subset\subset Acc(TS, \langle w, d\rangle)$;
- $TS \simeq_{must} TS' \Longleftrightarrow TS \leq_{must} TS'$ and $TS' \leq_{must} TS$.

An example of timed *must*-equivalent structures is shown in Figure 2(a). The timed event structures TS_3 and TS_3' shown in Figure 2(b) are not timed *must*-equivalent. Let us consider the timed word $\langle w, d\rangle = \langle a(0.5), 1.5\rangle \in L(TS_3) \cap L(TS_3')$. We have

$Acc(TS_3, \langle w, d\rangle) = \{\{b, c\} \cup (0, 1]\}$,
$Acc(TS_3', \langle w, d\rangle) = \{\{b, c\} \cup (0, 1], \{c\}\}$, i.e.
$\neg\big(Acc(TS_3', \langle w, d\rangle) \subset\subset Acc(TS_3, \langle w, d\rangle)\big)$.

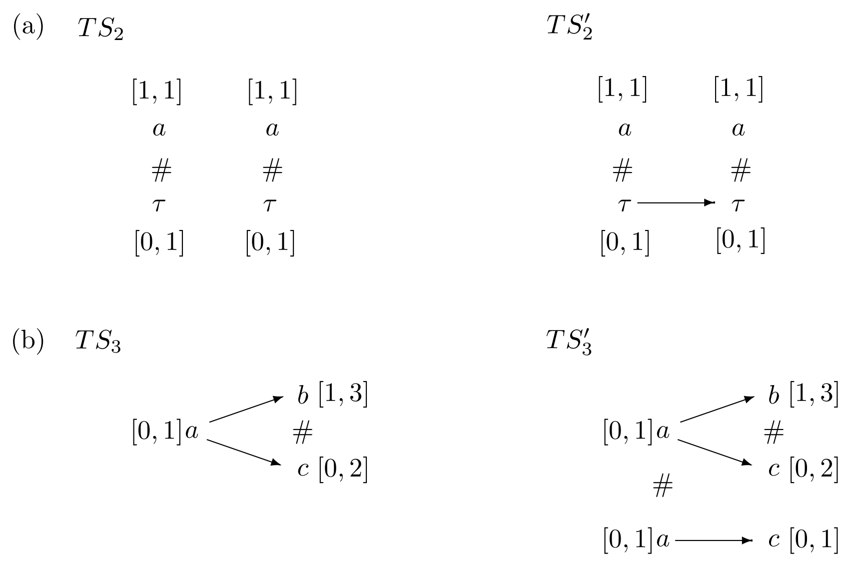

Figure 2. An example of timed *must*-equivalent (a) and non-timed *must*-equivalent (b) timed event structures

3. Timed modal logic

Here we will recall a dense-timed logic L_ν [19] and modify a satisfiability relation for timed event structures.

Definition 4. Let K be a finite set of clocks, Id be a set of identifiers and k be an integer. The set of formulas of L_ν over K, Id and k is generated by the abstract syntax with ϕ and ψ ranging over L_ν:

$$\phi := tt \mid ff \mid \phi \wedge \psi \mid \phi \vee \psi \mid \exists\phi \mid \forall\phi \mid \langle a\rangle\phi \mid [a]\phi \mid x \text{ in } \phi \mid x + n \bowtie y + m \mid x \bowtie m \mid Z,$$

where $a \in Act$, $x, y \in K$, $n, m \in \{0, 1, \ldots, k\}$, $\bowtie \in \{=, <, \leq, >, \geq\}$ and $Z \in Id$.

The meaning of identifiers from Id is specified by a declaration D that assigns a formula of L_ν to each identifier. When D is clear, we write $Z := \phi$ for $D(Z) = \phi$. The clocks from K are called formula clocks and a formula ϕ is said to be *closed* if every formula clock x occurs in ϕ in the scope of an "x in ..." operator. Given a timed event structure TS, we interpret the formulas from L_ν over *an extended* state $(C, \delta u)$, where (C, δ) is a state of TS and u is a time assignment for K. Transitions between extended states are defined by: $(C, \delta u) \stackrel{\epsilon(d)}{\to} (C, (\delta + d)(u + d))$ and $(C, \delta u) \stackrel{a}{\to} (C', \delta' u')$ iff $(C, \delta) \stackrel{a}{\to} (C', \delta')$ and $u = u'$. Formally, the satisfaction relation between extended states and formulas is defined just as in [19] and differs from [6] for $\forall\!\!\!\forall$- and $\exists\!\!\!\exists$- operators.

Definition 5. Let TS be a timed event structure and D be a declaration. The satisfaction relation $\models_D$ is the largest one that satisfies the following implications:

$$
\begin{array}{rcl}
(C, \delta u) \models_D tt & \Rightarrow & \text{true;} \\
(C, \delta u) \models_D ff & \Rightarrow & \text{false;} \\
(C, \delta u) \models_D \phi \wedge \psi & \Rightarrow & (C, \delta u) \models_D \phi \text{ and } (C, \delta u) \models_D \psi; \\
(C, \delta u) \models_D \exists\!\!\!\exists \phi & \Rightarrow & \exists d \in \mathbf{R}_0^+ \, . \, (C, \delta) \stackrel{\epsilon(d)}{\Rightarrow} (C', \delta') \\
 & & \text{and } (C', \delta' u + d) \models_D \phi; \\
(C, \delta u) \models_D \forall\!\!\!\forall \phi & \Rightarrow & \forall d \in \mathbf{R}_0^+ (C, \delta) \stackrel{\epsilon(d)}{\Rightarrow} (C', \delta') \\
 & & \text{implies } (C', \delta' u + d) \models_D \phi; \\
(C, \delta u) \models_D [a]\phi & \Rightarrow & \forall (C', \delta') \in ST(TS) \, . \, (C, \delta) \stackrel{a}{\to} (C', \delta') \\
 & & \text{implies } (C', \delta' u) \models_D \phi; \\
(C, \delta u) \models_D \langle a \rangle \phi & \Rightarrow & \exists (C', \delta') \in ST(TS) \, . \, (C, \delta) \stackrel{a}{\to} (C', \delta') \\
 & & \text{and } (C', \delta' u) \models_D \phi; \\
(C, \delta u) \models_D x + m \bowtie y + n & \Rightarrow & u(x) + m \bowtie u(y) + n; \\
(C, \delta u) \models_D x \text{ in } \phi & \Rightarrow & (C, \delta u') \models_D \phi, \text{ where } u' = [\{x\} \to 0]u; \\
(C, \delta u) \models_D Z & \Rightarrow & (C, \delta u) \models_D D(Z).
\end{array}
$$

Any relation that satisfies the above implications is called a satisfiability relation. We say that TS *satisfies* a closed formula ϕ from L_ν and write $TS \models \phi$ when $(C_0, \delta_0 u) \models_D \phi$ for any u. Note that if ϕ is closed, then $(C, \delta u) \models_D \phi$ iff $(C, \delta u') \models_D \phi$ for any $u, u' \in \mathbf{R}_0^{+K}$.

4. From state-space to class graph

Before constructing a characterictic formula, we need to transform the infinite state-space to a finite representation in such a way that states reachable by the same timed word be collected together in one class. We will do it through this section.

4.1. Region graph

As usually, in order to get a discrete representation of the state-space of a timed event structure, we use the concept of regions (equivalence classes of states) [2]. And the characteristic formula will be constructed for classes (τ-closure of regions). But we do not construct regions over the states of $ST(TS)$ by the reasons discussed in the previous works ([4, 6]).

Namely, one of the problems is existence of several regions which contain states reachable by the same timed word. So, we construct a region over common states that collect the states of $ST(TS)$ which we get by passing some timed word. But even after doing that, if there are internal actions with a dense time interval, we can get different classes which contain states reachable by the same timed word (we will consider such an example later). So, we allow internal actions to be only discrete.

The problem of synchronization of executions in two timed event structures is decided here by including counters into regions of one timed event structure in order to restrict the states of the second one for which a region formula has to be checked.

TS is *τ-discrete* if $\forall e \in E_{TS}$. $l(e) = \tau \Rightarrow D(e) = [n, n]$ ($n \in \mathbf{N}$). Let $\mathcal{E}_{d-\tau}$ denote the class of timed event structures having τ-discreteness property. Further we suppose that TS and $TS' \in \mathcal{E}_{d-\tau}$.

Definition 6. A subset $\mu \subseteq ST(TS)$ is called a *common state* of TS. The *initial* common state of TS is $\mu_0 = \{M_{TS}\}$. We will sometimes denote μ as $(C_1, \ldots, C_n, \delta_1, \ldots, \delta_n)$, where $(C_i, \delta_i) \in \mu$ $(1 \leq i \leq n)$.

Let us introduce another useful notation:

$$En(\mu) = \bigcup \{En(C) \mid \exists \delta \ . \ (C, \delta) \in \mu\}.$$

The relation $\xrightarrow{z}$ is modified on common states as follows:

- $\mu \xrightarrow{\tau} \mu'$ iff $\mu' = \{(C', \delta') \mid \exists (C, \delta) \in \mu \ . \ (C, \delta) \xrightarrow{\tau} (C', \delta')\} \cup \mu$ and $\mu \neq \mu'$;
- $\mu \xrightarrow{z} \mu'$ iff $\mu \not\xrightarrow{\tau}$ and $\mu' = \{(C', \delta') \mid \exists (C, \delta) \in \mu \ . \ (C, \delta) \xrightarrow{z} (C', \delta')\}$ $(z \in Act \cup \mathbf{R}^+)$.

Let $STC(TS)$ denote the set of all common states reachable from μ_0. The leading relation on common states of $STC(TS)$ is extended to timed words from $Dom(Act, \mathbf{R}_0^+)$ just as on the states of $ST(TS)$.

Then the notion of a region is defined analogously to Alur's one. Let $\mu = (C_1, \ldots, C_n, \delta_1, \ldots, \delta_n) \neq \mu' = (C'_1, \ldots, C'_n, \delta'_1, \ldots, \delta'_n)$. Then $\mu \simeq \mu'$ iff there exists renaming $\pi(n) : \{1, \ldots, n\} \to \{1, \ldots, n\}$, such that $(C_1, \ldots, C_n) = (C'_{\pi(n)(1)}, \ldots, C'_{\pi(n)(n)})$ and

(i) $\forall 1 \le i \le m \;.\; \lfloor \delta_1 | \dots | \delta_n(i) \rfloor = \lfloor \delta'_{\pi(n)(1)} | \dots | \delta'_{\pi(n)(n)}(i) \rfloor$;

(ii) $\forall 1 \le i, j \le m$.

— $\{\delta_1 | \dots | \delta_n(i)\} \le \{\delta_1 | \dots | \delta_n(j)\} \iff \{\delta'_{\pi(n)(1)} | \dots | \delta'_{\pi(n)(n)}(i)\} \le \{\delta'_{\pi(1)} | \dots | \delta'_{\pi(n)}(j)\}$,

— $\{\delta_1 | \dots | \delta_n(i)\} = 0 \iff \{\delta'_{\pi(n)(1)} | \dots | \delta'_{\pi(n)(n)}(i)\} = 0$,

where $\delta_1 | \dots | \delta_n$ is the concatenation of vectors δ_i $(1 \le i \le n)$ and $m = \sum_{1 \le i \le n} | C_i |$.

A set $R = [\mu] = \{\mu' \mid \mu \simeq \mu'\}$ is called a *region* of TS. We define $R_0 = [\mu_0]$.

Let R and R' be regions of TS. Then the leading relation on regions is defined as follows:

- $R \xrightarrow{a} R'$ iff $\exists \mu \in R,\ \mu' \in R' \;.\; \mu \xrightarrow{a} \mu'$ $(a \in Act_\tau)$;
- $R \xrightarrow{\chi} R'$ iff $\exists \mu \in R,\ \mu' \in R'\ \exists d \in \mathbf{R}^+ \;.\; \mu \xrightarrow{d} \mu' \wedge \forall\ 0 < d' < d\ \ \mu \xrightarrow{d'} \widetilde{\mu} \in R \cup R'$.

The leading relation on regions is extended to timed words from $Dom(Act, \mathbf{R}_0^+)$ just as on the states of $ST(TS)$.

We will call a partition of $STC(TS)$ into regions *stable* if the following holds:

- if $R \xrightarrow{a} R'$, then $\forall \mu \in R \;.\; \mu \xrightarrow{a} \mu'$ for some $\mu' \in R'$ $(a \in Act_\tau)$;
- if $R \xrightarrow{\chi} R'$, then $\forall \mu \in R\ \exists d \in \mathbf{R}^+ \;.\; \mu \xrightarrow{d} \mu'$ for some $\mu' \in R'$ and $\mu \xrightarrow{d'} \widetilde{\mu} \in R \cup R'$ for all $0 < d' \le d$.

So, we can define the notion of a region graph of TS.

Definition 7. *The region graph* of TS is a tuple $RG(TS) = (V_{RG}, E_{RG}, l_{RG})$, where the set of vertices V_{RG} is the stable partition of $STC(TS)$, the set of edges E_{RG} is the leading relation on regions of V_{RG} and the labelling function $l_{RG} : E_{RG} \longrightarrow Act_\tau \cup \{\chi\}$ is defined as $l((R, R')) = z \iff R \xrightarrow{z} R'$.

We define $Der(R,\ z) = \{R' \mid R \xrightarrow{z} R'\}$.

Lemma 1. *Let* $R \in V_{RG}$. *Then* $\forall \mu,\ \mu' \in R\ \forall (C, \delta) \in \mu\ \exists\ (C',\ \delta') \in \mu' \;.\; C = C' \ \wedge\ S((C, \delta)) \mid_{Act} = S((C', \delta')) \mid_{Act}\ \wedge\ S((C, \delta)) \mid_{\mathbf{R}^+} = \emptyset \iff S((C', \delta')) \mid_{\mathbf{R}^+} = \emptyset$.

4.2. Adding of counters

Let $RG(TS)$ be the region graph and X be a countable set of counters. Let all regions of $RG(TS)$ get a unique number, then with each region R_i we will associate its own counter x_{R_i}. For simplicity, sometimes we will denote x_{R_i} by x_i.

Moreover, with each region R we will associate a triple $(RC(R), \mu_R, \sigma_R)$, where $RC(R)$ is the set of counters, $\mu_R = (C_1, \ldots, C_{n_R}, \delta_1, \ldots, \delta_{n_R}) \in R$ is the region representative and the function $\sigma_R : RC(R) \longrightarrow 2^{n_R}$ associates the set of numbers of configurations from μ_R with each counter of $RC(R)$.

At first, we suppose $RC(R_0) = \{x_0\}$ and take μ_0 as a representative of R_0, $\sigma_{R_0}(x_0) = \{0\}$. For others $R \in RG(TS)$ we suppose $RC(R) = \emptyset$ and take an arbitrary $\mu \in R$ as its representative, $\sigma_R \equiv \emptyset$. Then the leading relation on regions is modified so that we add x_R into $RC(R)$, if after execution of some action we get $\mu \in R$ and some event becomes enabled in $C \in \mu$. Then the configuration C is associated with x_R. Additionally, we delete the counters, for which there are no configurations associated with them, from $RC(R)$. More formally:

- $(R, RC(R)) \xrightarrow{a} (R', RC(R'))$ $(a \in Act)$ iff $R \xrightarrow{a} R'$ (suppose $\mu_R \xrightarrow{a} \widetilde{\mu}$ for some $\widetilde{\mu} \in R'$) and the set $RC(R')$ is modified in two steps:

  1. $RC(R') = RC(R') \cup (R \setminus OLD(R, a))$, where
     $OLD(R, a) = \{x_i \mid \forall j \in \sigma_R(x_i) \ . \ (C_j, \delta_j) \stackrel{a}{\not\rightarrow}\}$;
  2. $RC(R') = RC(R') \cup \{x_{R'}\}$ if $\exists e \in En(\widetilde{\mu}) \setminus En(\mu_R) \wedge \forall (C, \delta) \in \mu_R \ \forall e \in C \cup En(C)$
     $\delta(e) \neq 0$;

  and $\sigma_{R'}$ is modified as follows:

  1. for all $x \in RC(R') \cap RC(R)$
     $\sigma_{R'}(x) = \sigma_{R'}(x) \cup \{j \mid \big[\exists i \in \sigma_R(x) \ \exists (\widetilde{C}_k, \widetilde{\delta}_k) \in \widetilde{\mu} \ . \ (C_i, \delta_i) \xrightarrow{a} (\widetilde{C}_k, \widetilde{\delta}_k)\big] \ \wedge \ \big[\exists \pi(n_{R'}) \ .$
     $(C'_j, \delta'_j) = (\widetilde{C}_{\pi(n_{R'})(k)}, \widetilde{\sigma}_{\pi(n_{R'})(k)}) \in \mu_{R'}\big]\}$;
  2. if $x_{R'} \in RC(R')$ then $\sigma_{R'}(x_{R'}) = \{i \mid \ (C_i, \delta_i) \in \mu_{R'} \ . \ \exists e \in En(C_i) \ \delta_i(e) = 0\}$;

- $(R, RC(R)) \xrightarrow{\tau} (R', RC(R'))$ is defined analogously to the previous item, except the first step of modifying $RC(R')$. Namely, $RC(R') = RC(R)$.

- $((R, RC(R)) \xrightarrow{\chi} (R', RC(R'))$ iff $R \xrightarrow{\chi} R'$ (suppose $\mu_R \xrightarrow{d} \widetilde{\mu}$ for some $d \in \mathbf{R}^+$ and $\widetilde{\mu} \in R'$) and

1. $RC(R') = RC(R') \cup (R \setminus OLD(R,\chi))$, where $OLD(R,\chi) = \{x_i \mid \forall j \in \sigma_R(x_i)\big(\neg\exists(\widetilde{C},\widetilde{\delta}) \in \widetilde{\mu} \ . \ (C_j,\delta_j) \xrightarrow{d} (\widetilde{C},\widetilde{\delta})\big)\}$;
2. for all $x \in RC(R') \cap RC(R)$ $\sigma_{R'}(x) = \sigma_{R'}(x) \cup \{j \mid \big[\exists i \in \sigma_R(x) \ \exists(\widetilde{C}_k,\widetilde{\delta}_k) \in \widetilde{\mu} \ . \ (C_i,\delta_i) \xrightarrow{d} (\widetilde{C}_k,\widetilde{\delta}_k)\big] \ \wedge \ \big[\exists\pi(n_{R'}) \ . \ (C'_j,\delta'_j) = (\widetilde{C}_{\pi(n_{R'})(k)}, \widetilde{\sigma}_{\pi(n_{R'})(k)}) \in \mu_{R'}\big]\}$.

We also need a time assignment of our counters, so, into all common states $\mu \in R$, we include $RI_\mu = RC(R)$ and the time assignment $\Delta_\mu : RI_\mu \to \mathbf{R}_0^+$. At first, suppose $\Delta_\mu \equiv 0$. We will omit the subscript μ if it will be clear. The leading relation on common states is modified as follows:

- $(\mu, RI, \Delta) \xrightarrow{d} (\mu', RI', \Delta')$ $(d \in \mathbf{R}^+)$ iff $\mu \xrightarrow{d} \mu'$ and $\Delta' \mid_{RI} = \Delta \mid_{RI} + d$;
- $(\mu, RI, \Delta) \xrightarrow{a} (\mu', RI', \Delta')$ $(a \in Act)$ iff $\mu \xrightarrow{a} \mu'$.

It is clear that additional pieces of information have no influence on the leading relations on common states and regions. In the following, we will use a simple notation R and μ instead of $(R, RC(R))$ and $(\mu,\ RI,\ \Delta)$.

4.3. Class graph

We next define in a usual way the notions of a class (the τ-closure of regions) and the class graph of a timed event structure [4].

Let $RG(TS) = (V_{RG}, E_{RG}, l_{RG})$ 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 TS. Define $Q_0 = \{R_0\}^\tau$, 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 $Q_1 = (Der(Q,\ z))^\tau$.

We need the following notations.

$S(Q) = \{z \in Act \cup \{\chi\} \mid \ Q \xrightarrow{z}\}$, $QC(Q) = \bigcup_{R\in Q} RC(R)$.

Definition 8. The *class graph* of TS is the labelled directed graph $CG(TS) = (V_{CG}, E_{CG}, l_{CG})$. The set of vertices V_{CG} is the set of reachable classes of TS and E_{CG} is the leading relation on classes of the set V_{CG}, the labelling function $l_{CG} : E_{CG} \longrightarrow (Act \cup \{\chi\})$.

To prove the main result, we need a notion that connects the notions of a common state and a class.

Definition 9. Let $\langle w, d\rangle \in L(TS)$ and $CG(TS) = (V_{CG},\ E_{CG},\ l_{CG})$. Let $p = Q_0 \ \ldots \ Q$ be a path in $CG(TS)$. Then $\mu \in STC(TS)$ is *class-reachable by* $\langle w, d\rangle$ *consistent with* p iff $[\mu] \in Q$ and either

- $p = Q_0$ and $\langle w, d\rangle = \langle \epsilon, 0\rangle$, or
- $p = p' \xrightarrow{z} Q$ and there exists $\mu' \in STC(TS)$ class-reachable by $\langle w', d'\rangle$ consistent with p', and either
  - $z = a \in Act$, $\mu' \overset{a}{\Rightarrow}\overset{d''}{\rightarrow} \mu$, and $\langle w, d\rangle = \langle w'a(d' - \Delta(w')), d' + d''\rangle$, for some $d'' \in \mathbf{R}_0^+$, or
  - $z = \chi$, $\mu' \xrightarrow{d''} \mu$, and $\langle w, d\rangle = \langle w', d' + d''\rangle$, for some $d'' \in \mathbf{R}^+$.

Now, let us consider an example mentioned above. Let us return to the model with internal actions that have a dense timed interval. Then in $CG(TS'_2)$ (see Figure 2) there exist vertices Q and Q_1 with regions $[\mu'] \in Q$ ($[\mu'_1] \in Q_1$) such that μ'(μ'_1, respectively) is reachable by $\langle \epsilon(1), 1\rangle$ consistent with a path from Q_0 to Q (to Q_1, respectively).

We get such a situation because in TS'_2 we can execute an action τ and get a new state both at time 0 and at time $0 < d < 1$. If τ is executed at time 0, then the set of values of the function δ in a new state consists of 0. If τ is executed at time $0 < d < 1$, then the set of values of function δ in a new state consists of 0 and d. So, there exist several paths in the region graph $RG(TS'_2)$ from $R_{TS'_2}$ to regions that include the states reachable by $\langle \epsilon(1), 1\rangle$. Namely, one of them consists of sequence of τ-, and two χ-transitions, the other consists of sequence of τ- and three χ-transitions. Thus, according to construction of the class graph, these regions belong to different classes.

Since we wish to avoid these cases, we impose a restriction that internal actions have only discrete time.

Lemma 2. *Let $\langle w, d\rangle \in L(TS)$ and $\mu_0 \overset{\langle w,d\rangle}{\Longrightarrow} \mu$. Then there exists the only path p in $CG(TS)$ such that μ is reachable by $\langle w, d\rangle$ consistent with p.*

5. Formula construction

Now we can construct a formula for each class Q. In the formula, we use the notations $Q \xrightarrow{a} Q_a$ and $Q \xrightarrow{\chi} Q_\chi$ and write its optional parts between $\langle\!\langle$ and $\rangle\!\rangle$. In addition, we suppose $\hat{R} \in Q$ such that $\hat{R} \overset{\tau}{\not\rightarrow}$.

$$
\begin{aligned}
F_Q = \;& \forall\!\!\!\forall \beta(Q) \Rightarrow \; \psi_Q; \\
\psi_Q = \;& \langle\!\langle \forall\!\!\!\forall \beta^{>}(Q) \Rightarrow F_{nil}\rangle\!\rangle \;\wedge\; \langle\!\langle F_{Q_\chi}\rangle\!\rangle \;\wedge\; \textstyle\bigwedge_{a \notin S(Q)|_{Act}} [a]\mathit{ff} \;\wedge \\
& \textstyle\bigwedge_{a \in S(Q)|_{Act}} [a](\langle\!\langle X Q_a \; in\rangle\!\rangle \; \widehat{F}_{Q_a}) \;\wedge\; (ACC(Q) \;\vee\; \langle\tau\rangle \mathit{tt}); \\
\widehat{F}_Q = \;& \begin{cases} F_Q, & \text{if } Q = \{R\} \;\wedge\; \exists \mu \in R \; \exists d \in \mathbf{R}^+ \,.\, \mu_R \xrightarrow{d} \mu, \\ \psi_Q, & \text{otherwise.} \end{cases}
\end{aligned}
$$

Here the conditions $\beta(Q)$ that hold for the time assignment of states only from $\hat{R}$ are constructed in the following way:

1. $\beta(Q) = tt$;
2. for all $x_i, x_j (x_i \neq x_j) \in RC(\hat{R})$ let $\lfloor \Delta_{\mu_{\hat{R}}}(x_i) \rfloor = a$, $\lfloor \Delta_{\mu_{\hat{R}}}(x_j) \rfloor = b$, then

$$\beta(Q) = \beta(Q) \wedge \begin{cases} x_i = a, & \text{if } \Delta_{\mu_{\hat{R}}}(x_i) = \lfloor \Delta_{\mu_{\hat{R}}}(x_i) \rfloor, \\ a < x_i < a+1, & \text{otherwise;} \end{cases}$$

3.

$$\beta(Q) = \beta(Q) \wedge \begin{cases} x_i + b = x_j + a, & \text{if } \{\Delta_{\mu_{\hat{R}}}(x_i)\} = \{\Delta_{\mu_{\hat{R}}}(x_j)\}, \\ x_i + b < x_j + a, & \text{if } \{\Delta_{\mu_{\hat{R}}}(x_i)\} < \{\Delta_{\mu_{\hat{R}}}(x_j)\}, \\ x_i + b > x_j + b, & \text{if } \{\Delta_{\mu_{\hat{R}}}(x_j)\} < \{\Delta_{\mu_{\hat{R}}}(x_i)\}. \end{cases}$$

The conditions $\beta^{>}(Q)$, which mean that the values of counters are larger than the appropriate time assignments in the states from $\hat{R}$, are constructed as follows:

$$\beta^{>}(Q) = \begin{cases} \beta(Q) \ \vee \ \bigvee_{x_i \in RC(\hat{R})} x_i \geq \lceil \Delta_{\mu_{\hat{R}}}(x_i) \rceil & \text{if all}(C,\delta) \in \mu_{\hat{R}} \\ & \text{are terminated,} \\ \bigvee_{\{x_i \in RC(\hat{R}) \mid \{\Delta_{\mu_{\hat{R}}}(x_i)\} = 0\}} x_i > \lceil \Delta_{\mu_{\hat{R}}}(x_i) \rceil & \\ \vee \ \bigvee_{\{x_i \in RC(\hat{R}) \mid \{\Delta_{\mu_{\hat{R}}}(x_i)\} \neq 0\}} x_i \geq \lceil \Delta_{\mu_{\hat{R}}}(x_i) \rceil, & \text{otherwise.} \end{cases}$$

Below we give the subformulas of ψ_Q and conditions on including them into ψ_Q.

- $XQ_a = \{x \mid x \in QC(Q_a) \setminus QC(Q)\}$ is added if it is not empty;
- $\forall\!\!\!\forall \beta^{>}(Q) \Rightarrow F_{nil}$ is added into ψ_Q if the class Q_χ does not exist;
- F_{Q_χ} is added into ψ_Q if the class Q_χ exists;
- $ACC(Q) = \bigvee_{(C,\delta) \in \mu_{\hat{R}}, (C,\delta) \stackrel{\tau}{\not\rightarrow}} \big((\bigwedge_{a \in S((C,\delta))} \langle a \rangle tt) \ \wedge \ \langle\!\langle \chi_{(C,\delta)} \rangle\!\rangle \ \wedge \ \langle\!\langle F_{nil} \rangle\!\rangle \big)$;
- $F_{nil} = \bigwedge_{a \in Act} [a] ff$ is added into $ACC(Q)$ for all $(C,\delta) \in \mu_{\hat{R}}$ such that $S((C,\delta)) \mid_{Act} = \emptyset$;
- $\chi_{(C,\delta)} = \begin{cases} \exists \beta(Q_\chi) \Rightarrow (\bigwedge_{a \in S((C,\delta))} \langle a \rangle tt), & \text{if } S(C,\delta) \mid_{Act} \neq \emptyset, \\ \exists \beta^{>}(Q) \Rightarrow (\bigvee_{a \in Act_\tau} \langle a \rangle tt), & \text{otherwise;} \end{cases}$
- $\chi_{(C,\delta)}$ is added into $ACC(Q)$ for all $(C,\delta) \in \mu_{\hat{R}}$ such that $S((C,\delta)) \mid_{\mathbf{R}^+} \neq \emptyset$.

Note that we use the symbol of implication ($\Rightarrow$) for simplicity. But it is easy to transform our formula into a correct formula from L_ν, because negation of $\beta(Q)$ and $\beta^{>}(Q)$ can be expressed in L_ν. Also, $XQ_a \ in \ F$ means $(x_1 \ in \ (x_2 \ in \ (\ldots \ (x_n \ in \ F)))$ for $XQ_a = \{x_1, \ x_2, \ldots, \ x_n\}$. The formula ψ_Q

contains three obligatory groups. The first group of conjunctions contains an $[a]$-formula for any action that cannot be executed in Q. The second group of conjunctions contains an $[a]$-formula for any action that can be executed in Q. The third group is a group of disjunctions over all states in $\mu_{\hat{R}}$ and each disjunction part contains conjunctions of $\langle a\rangle$-formulas for each action that can be executed in some state, and an optional part which characterizes the possibility of some amount of time to pass in this state. The optional group of ψ_Q is included into the formula, if there is no region Q_χ.

For a timed event structure TS, *a characteristic must-formula* is defined as $F_{TS}^{must} = x_0 \; in \; F_{Q_0}$.

TS_4 $\qquad\qquad\qquad$ $CG(TS_4)$

$$\begin{tikzcd}[row sep=small] {[1,1]} & {[1,1]} \\ a & a \\ \# & \# \\ \tau \arrow[r] & \tau \\ {[1,1]} & {[1,1]} \end{tikzcd} \qquad\qquad Q_0 \xrightarrow{\chi} Q_1 \xrightarrow{\chi} Q_2 \xrightarrow{a} Q_3 \xrightarrow{a} Q_4$$

Figure 3. A timed event structure and its class graph

Let us construct a characteristic *must*-formula for a timed event structure shown in Figure 3. In this figure, the class graph of TS_4 is also shown. For simplicity, we suppose here $Act = \{a\}$. So, we get

$$\begin{aligned}
F_{TS_4}^{must} &= x_0 \; in \; \Big(\forall\!\!\!\forall\; x_0 = 0 \Rightarrow \big[F_{Q_1} \wedge [a]\mathit{ff} \wedge (ACC(Q_0) \vee \langle\tau\rangle\mathit{tt})\big]\Big),\\
F_{Q_1} &= \forall\!\!\!\forall\; 0 < x_0 < 1 \Rightarrow \big[F_{Q_2} \wedge [a]\mathit{ff} \wedge (ACC(Q_1) \vee \langle\tau\rangle\mathit{tt})\big],\\
F_{Q_2} &= \forall\!\!\!\forall\; x_0 = 1 \Rightarrow \big[(\forall\!\!\!\forall\; x_0 > 1 \Rightarrow F_{nil}) \wedge [a]F_{Q_3} \wedge (ACC(Q_2) \vee \langle\tau\rangle\mathit{tt})\big],\\
F_{Q_3} &= \forall\!\!\!\forall\; x_0 = 1 \Rightarrow \big[(\forall\!\!\!\forall\; x_0 > 1 \Rightarrow F_{nil}) \wedge [a]F_{Q_4} \wedge (ACC(Q_3) \vee \langle\tau\rangle\mathit{tt})\big],\\
F_{Q_4} &= \forall\!\!\!\forall\; x_0 = 1 \Rightarrow \big[(\forall\!\!\!\forall\; x_0 > 1 \Rightarrow F_{nil}) \wedge [a]\mathit{ff} \wedge (ACC(Q_4) \vee \langle\tau\rangle\mathit{tt})\big],\\
ACC(Q_0) &= F_{nil} \wedge \exists\!\!\!\exists\; x_0 > 0 \Rightarrow (\langle a\rangle\mathit{tt} \vee \langle\tau\rangle),\\
ACC(Q_1) &= F_{nil} \wedge \exists\!\!\!\exists\; x_0 \geq 0 \Rightarrow (\langle a\rangle\mathit{tt} \vee \langle\tau\rangle),\\
ACC(Q_2) &= F_{nil},\\
ACC(Q_3) &= F_{nil} \vee (\langle a\rangle\mathit{tt} \wedge F_{nil}),\\
ACC(Q_4) &= F_{nil},\\
F_{nil} &= [a]\mathit{ff}.
\end{aligned}$$

Lemma 3. *Let* $(C_0', \delta_0' \; u) \models_D F_{TS}^{must}$, *where* $(C_0', \delta_0') = M_{TS'}$, $u \equiv 0$. *For all* $\langle w,\; d\rangle \in L(TS) \cap L(TS')$ *and* $(C_0', \delta_0') \stackrel{\langle w,\; d\rangle}{\Longrightarrow} (C', \delta')$ *it holds that* $(C', \delta' \; u') \models_D \psi_Q$, *where* Q *and* u' *are such that there exists* μ *which is reachable by* $\langle w,\; d\rangle$ *consistent with a path from* Q_0 *to* Q, *and* $u' \mid_{RI_\mu} = \Delta_\mu$.

Lemma 4. *Let* $(C_0', \delta_0' \; u) \models_D F_{TS}^{must}$, *where* $(C_0', \delta_0') = M_{TS'}$, $u \equiv 0$. *Then* $L(TS') \subseteq L(TS)$.

Now, we are ready to prove the following theorem.

Theorem. $TS \leq_{must} TS' \iff TS' \models_D F_{TS}^{must}$, where D corresponds to the previous definition of F_Q for each Q from $V_{CG(TS)}$.

Proof.
($\Leftarrow$) Take an arbitrary $\langle w,\ d\rangle \in L(TS')$ and (C', δ') such that $(C'_0, \delta'_0) \stackrel{\langle w,\ d\rangle}{\Longrightarrow} (C', \delta')$. According to Definition 3, we will show that there exists $(C, \delta) \in ST(TS)$ such that $(C_0, \delta_0) \stackrel{\langle w,\ d\rangle}{\Longrightarrow} (C, \delta)$ and $S((C, \delta)) \mid_{Act} \subseteq S((C', \delta')) \mid_{Act}$, $S((C', \delta')) \mid_{\mathbf{R}^+} = \emptyset \Rightarrow S((C, \delta)) \mid_{\mathbf{R}^+} = \emptyset$.

By Lemma 4, $\langle w,\ d\rangle \in L(TS)$. By Lemmas 2 and 3, we can find Q and u' such that p is a path from Q_0 to Q, $u' \mid_{RI_\mu} = \Delta_\mu$ and $(C', \delta'\ u') \models_D \psi_Q$. By construction of the formula ψ_Q and Lemma 1, there exists $(C, \delta) \in \mu$ for which $S((C, \delta)) \mid_{Act} \subseteq S((C', \delta')) \mid_{Act} \ \wedge S((C', \delta')) \mid_{\mathbf{R}^+} \Rightarrow S((C, \delta)) \mid_{\mathbf{R}^+}$.
($\Rightarrow$) Follows from construction of the formula F_{TS}^{must}. □

6. Conclusion

This article is concentrated on constructing a characteristic formula for timed event structures with discrete internal actions. This formula allows us to decide the problem of recognizing the timed *must*-equivalences by reducing it to the model-checking one. In general, we can use internal actions with discrete-timed intervals. Then the obtained formula for each class can be easily modified by including the parts for each region of the class that cannot execute a τ-action.

Additionally, we hope that the results could be extended onto a model with dense-timed internal actions. The way of construction of the characteristic formula may be applied to other timed testing equivalences, for example, to *may*-equivalences, that can lead to decision of the problem of inclusion of timed languages of the model under consideration.

References

[1] Aceto L., De Nicola R., Fantechi A. Testing equivalences for event structures // Lect. Notes Comput. Sci. — 1987. — Vol. 280. — P. 1–20.

[2] Alur R., Courcoubetis C., Dill D. Model checking in dense real time // Inform. and Comput. — 1993. — Vol. 104. — P. 2–34.

[3] Alur R., Dill D. The theory of timed automata // Theor. Comput. Sci. — 1994. — Vol. 126. — P. 183–235.

[4] Andreeva M.V., Bozhenkova E.N., Virbitskaite I.B. Analysis of timed concurrent models based on testing equivalence // Fundamenta Informaticae. — 2000. — Vol. 43. — P. 1–20.

[5] Baier C., Katoen J.-P., Latella D. Metric semantics for true concurrent real time // Lect. Notes Comput. Sci. — 1998. — Vol. 1443. — P. 568–579.

[6] Bozhenkova E.N. Towards decidability of timed testing // Joint NCC& IIS Bull. Ser.: Comput. Sci. — 2001. — N 15. — P. 17–29.

[7] Castellani I., Hennessy M. Testing theories for asynchromous langugages // Lect. Notes Comput. Sci. — 1998. — Vol. 1530. — P. 90–101.

[8] Cleaveland R., Hennessy M. Testing equivalence as a bisimulation equivalence // Lect. Notes Comput. Sci. — 1989. — Vol. 407. — P. 11–23.

[9] Cleaveland R., Lee I., Lewis P.M., Smolka S.A. A theory of testing for soft real time processes // Proc. 8th Intern. Conf. Software Engineering and Knowledge Engineering, SEKE'96, Lake Tahoe, Nevada, USA, June 1996. — P. 474–479.

[10] Cleaveland R., Zwarico A.E. A theory of testing for real-time // Proc. 6th IEEE Symp. on Logic in Comput. Sci. (LICS 91), Amsterdam, The Netherlands, 1991. — P. 110–119.

[11] Corradini F., Vogler W, Jenner L. Comparing the Worst-Case Efficiency of Asynchronous Systems with PAFAS // Augsburg, 2000. — (Tech. Rep. / Inst. fur Informatik / Univ. of Augsburg; N 2000-6).

[12] De Nicola R., Hennessy M. Testing equivalence for processes // Theor. Comput. Sci. — 1984. — Vol. 34. — P. 83–133.

[13] van Glabbeek R.J. The linear time — branching time spectrum II: the semantics of sequential systems with silent moves. Extended abstract // Lect. Notes Comput. Sci. — 1993. — Vol. 715. — P. 66–81.

[14] Goltz U., Wehrheim H. Causal testing // Lect. Notes Comput. Sci. — 1996. — Vol. 1113. — P. 394–406.

[15] Hennessy M., Regan T. A process algebra for timed systems// Inform. and Comput. — 1995. — Vol. 117. — P. 221–239.

[16] Katoen J.-P., Langerak R., Latella D., Brinksma E. On specifying real-time systems in a causality-based setting// Lect. Notes Comput. Sci. — 1996. — Vol. 1135. — P. 385–404.

[17] Kumar K.N., Cleaveland R., Smolka S.A. Infinite probabilistic and nonprobabolostic testing.// Lect. Notes Comput. Sci. — 1998. — Vol. 1530. — P. 209–220.

[18] Llana L., de Frutos D. Denotational semantics for timed testing// Lect. Notes Comput. Sci. — 1997. — Vol. 1231. — P. 368–382.

[19] Laroussinie F., Larsen K.L., Weise C. From timed automata to logic — and back // Århus, 1995. — (Tech. Rep. / BRICS, Dept. Comput. Sci., Univ. of Århus; N RS-95-2) (available at `http://www.brics.aau.dk/BRICS`).

[20] López N., Núñez M. A testing theory for generally distributed stochastic processes // Lect. Notes Comput. Sci. — 2001. — Vol. 2154. — P. 321–335.

[21] Murphy D. Testing, betting and timed true concurrency // Lect. Notes Comput. Sci. — 1991. — Vol. 575. — P. 439–453.

[22] Steffen B., Weise C. Deciding testing equivalence for real-time processes with dense time // Lect. Notes Comput. Sci. — 1993. — Vol. 711. — P. 703–713.

[23] Winskel G. An introduction to event structures // Lect. Notes Comput. Sci. — 1989. — Vol. 354. — P. 364–397.