Bull. Nov. Comp. Center, Comp. Science, 24 (2006), 1–25

Action refinement and equivalence notions for timed event structures

M. V. Andreeva

Abstract. The paper is contributed to study an operator for refinement of actions to be used in the design of concurrent real time systems. The refinement operator replaces actions on a given level of abstraction by more complicated processes on a lower level. We define this operator on a causality based, event-oriented timed model. Then we investigate the interplay of action refinement with abstractions in terms of equivalence notions of the linear time - branching time spectrum for concurrent systems in terms of timed partial orders. As a result, we propose variations of these equivalences with additional timing requirements sufficient for preserving equivalences under refinement. Furthermore, when dealing with particular subclasses of the model under consideration, these additional requirements are not still necessary for preservation of the equivalences under refinement.

1. Introduction

In the design of parallel systems, the operation of refinement of actions is widely used, which allows us to represent behaviour of a system on higher or lower levels of abstraction. We consider parallel systems for which the basic building blocks are the actions which occur in the system. By an action we mean here any activity of the system which can be considered as a conceptual entity on the given level of abstraction. This allows us to represent systems in a hierarchical way, changing the level of abstraction by interpreting actions on a higher level by more complicated processes on a lower level. The operation is constructed in such a way that the behaviour of the refined system can be derived compositionally from the behaviour of the initial system and the behaviour of the processes which refine actions. For the purposes of specification and verification of the systems behaviour, different equivalence notions are used in order to be able to choose the simplest possible view of the system. In this paper we study the interplay between the refinement and certain equivalence notions or, in more detail, the problem of preserving the behaviour equivalences under the refinement.

Recently, a variety of equivalences of parallel systems have been introduced and the relations between them are well-studied (see, for instance, [12, 13]). There are two important aspects of equivalences used in their classification: the preserved level of detail in system runs and the preserved level of the choice structure between system runs. Concerning the first aspect, there are two opposite approaches based on the interleaving semantics and the causal semantics. In the interleaving semantics, a run consists of a sequence of actions, therewith we abstract from the causal relations between these actions. In the causal semantics, all causal dependencies between actions are preserved. We will consider only the partial order semantics which is a causal semantics where all causal dependencies are represented by a partial order [13]. The choice is motivated by the fact that the interleaving semantics and other semantics lying in between in the spectrum of semantics are not preserved under refinement [13, 14]. For the second aspect concerning the choice structure between system runs, the simplest notion is the trace semantics (linear time) [17] when the system behaviour is defined by the sets of its possible runs and the choice structure is neglected. At the opposite end of the spectrum, there is the bisimulation semantics (branching time) [15] which preserves the information on the points of choices between system runs. In between there are test semantics [11]. The third aspect of the equivalence classification consists in the treatment of internal or invisible actions (strong versus weak equivalences). We will consider only strong equivalences which do not distinguish internal actions.

The equivalence notions mentioned above were introduced for formal models of systems without time delays. A growing interest in modeling of real-time systems observed recently motivates a necessity for a formal representation of the lapse of time. Several formal methods for specifying and confirming such systems were recently introduced [4, 5]. In papers [9, 22, 25], the questions related to time-sensitive equivalences were studied. In these investigations, real-time systems are represented by timed interleaving models which are parallel timer processes or timed automata containing clocks by which one means fictitious time measuring elements.

In this paper, we study the relations between the action refinement and behaviour equivalences in the context of the timed partial order model. In particular, we consider a family of equivalences based on the partial order semantics from the linear-time/branching time spectrum in the setting of event structures with a dense time domain [7, 23].

In the paper, we show that the timed extensions of those equivalence notions, which are preserved under refinement in the event structure theory [13, 14], are not preserved under refinement in the model under consideration. We propose some variations of these equivalences with additional timing requirements sufficient for preserving equivalences under refinement. Moreover, we distinguish several subclasses of timed event structures on which the original equivalences are invariant under refinement without these additional requirements.

The remained part of the paper is structured as follows. In Section 2, we introduce the main definitions and notation related to the timed event structures theory. In the next section, we recall trace, test and history preserving bisimulation equivalences of timed event structures in terms of the timed partial order. In Section 4, we define the refinement operator of timed event structures. Further in Section 5, we introduce additional timing requirements for equivalences from Section 3 such that the new equivalences we get are invariant under action refinement. In Section 6, we distinguish some subclasses of timed event structures on which the original equivalences from Section 3 are invariant under refinement. Section 7 contains some conclusions and remarks on the future works.

2. Timed event structures

In this section, we introduce some basic notions and notations concerning timed event structures.

First, we recall a notion of event structures [24] which constitute a major branch of partial order models. The main idea behind event structures is to view distributed computations as action occurrences, called events, together with a notion of causality dependency between events (which is reasonably characterized via a partial order). Moreover, in order to model nondeterminism, there is a notion of conflicting (mutually incompatible) events. A labelling function records which action an event corresponds to.

We consider a dense time extension of event structures, called *timed event structures* [7, 23]. The time incorporated into event structures can be characterized in the following way. A global real-valued clock [8, 16, 18, 19, 21] is assumed. Events are associated with timed constraints restricting the times at which the events can occur. The occurrences of enable events themselves take no time. We assume an implicit passage of time [8, 16, 18, 19]. We do not force events to occur once they are ready, i.e. their causal predecessors have occurred and their timing constraints are respected, since the concept of urgent events [3, 19] is sometimes quite constraining in the timing actions.

Let $\mathbb{R}$ be the set of nonnegative real numbers, $\mathbb{N}$ be the set of integer numbers. Denote the set of closed intervals (segments) in $\mathbb{R}$ as

$$Interv = \{ [d_1, d_2] \subset \mathbb{R} \mid d_1 \leq d_2\}.$$

Definition 1.

- *Let Act be a finite set of action. A (labelled) event structure is a tuple* $S = (E, \leq, \#, l)$*, where*
  - E *is a denumerable set of events,*
  - $\leq \subseteq E \times E$ *is a partial order (the causality relation), satisfying the principle of finite courses:* $\forall e \in E \ \diamond \ \downarrow e = \{e' \in E \mid e' \leq e\}$ *is finite,*

– $\# \subseteq E \times E$ *is a symmetric and irreflexive relation (the conflict relation) satisfying the principle of conflict heredity:* $\forall e, e', e'' \in E \ \diamond e \# e' \leq e'' \ \Rightarrow \ e \# e''$;

– $l : E \rightarrow Act$ *is a labelling.*

- *A (labelled) timed event structure is a tuple* $TS = (S, D)$*, where* S *is a (labelled) timed event structure and* $D : E \rightarrow Interv$ *is a timing function.*

We will use $\mathcal{S}$ to denote the set of event structures. Let $\mathcal{O} = (\emptyset, \emptyset, \emptyset, \emptyset)$ be the empty event structure. For $C \subseteq E$, we define the restriction S to C as $S\lceil C = (C, \leq \cap (C \times C), \# \cap (C \times C), l\,|_C)$. Also, we will use $\mathcal{TS}$ to denote the set of timed event structures.

Timed event structures $TS, TS' \in \mathcal{TS}$ *are isomorphic* (denoted as $TS \simeq TS'$) if there exists a bijection $\varphi : E_{TS} \longrightarrow E_{TS'}$ such that $e \leq_{TS} e' \iff \varphi(e) \leq_{TS'} \varphi(e')$, $e \#_{TS} e' \iff \varphi(e) \#_{TS'} \varphi(e')$, $l_{TS}(e) = l_{TS'}(\varphi(e))$ and $D_{TS}(e) = D_{TS'}(\varphi(e))$ for all $e, e' \in E_{TS}$.

For depicting timed event structures, we use the following conventions. The action labels and timing constraints associated with events are drawn near the events. If no confusion arises, we will often use action labels rather than event identities to denote events. The <-relation is depicted by arcs (omitting those derivable by transitivity), and conflicts are also drawn (omitting those derivable by conflict heredity).

$$TS_0 : \quad \begin{array}{ccc} [\frac{2}{3}, 4] & & [2, 6] \\ a : e_1 & \longrightarrow & b : e_2 \\ & & \# \\ & & c : e_3 \\ & & [1, 1] \end{array}$$

Figure 1. An example of a labelled timed event structure

An execution of a timed event structure is a *timed configuration* which consists of a configuration and a timing function, recording global time moments at which events occur, and satisfies some additional requirements.

Definition 2. *Let* $TS = (S, D) \in \mathcal{TS}$*, then*

- *a finite set of events* $C \subseteq E_{TS}$ *is a configuration in* S *(in* TS*) if*

  – C *is left-closed:* $\forall e \in C \ \diamond \ \downarrow e \subseteq C$;

  – C *is conflict-free:* $\forall e, \ e' \in C \diamond \neg(e \# e')$;

- *a tuple* $TC = (C, T)$ *consisting of a configuration* C *and a timing function* $T : C \longrightarrow \mathbb{R}$ *is a timed configuration in* TS*, if*

– $\forall\, e \in C \;\diamond\; T(e) \in D_{TS}(e)$;
– $\forall\, e, e' \in C \;\diamond\; e \leq_{TS} e' \;\Rightarrow\; T(e) \leq T(e')$.

Informally speaking, each event can occur at a time when its timing constraints are met, and, for any two events e and e' occurred, if e causally precedes e' then e should temporally precede e'. The *initial timed configuration* of TS is $(\emptyset, \emptyset)$. We use $\mathcal{C}(S)$ to denote the set of configurations of S and $\mathcal{TC}(TS)$ to denote the set of timed configurations of TS. The *restriction* of TS to TC, denoted as $TS\lceil TC$, is defined as $(S\lceil C, T)$.

We define the leading relation on a timed configuration in the following way. Let TS be a timed event structure and $TC = (C, T), TC' = (C', T') \in \mathcal{TC}(TS)$. We will write $TC \longrightarrow TC'$ iff $C \subseteq C'$ and $T'|_C = T$.

A timed event structure $TS \in \mathcal{TS}$ is said to *have a valid timing* if $e \leq_{TS} e'$ implies $\min D_{TS}(e) \leq \min D_{TS}(e')$ and $\max D_{TS}(e) \leq \max D_{TS}(e')$. The valid timing property of the timed event structure guarantees that if for some event its causal predecessors have occurred and no conflicting events have occured, then the event can occur respectively with its timing constraints.

Lemma 1. *Let $TS = (S, D)$ has a valid timing, then if $TC = (C, T) \in \mathcal{TC}(TS)$ and $C \subseteq C_1 \in \mathcal{C}(S)$, then there exists $T_1 : C_1 \longrightarrow \mathbb{R}$ such that $TC_1 = (C_1, T_1) \in \mathcal{TC}(TS)$ and $TC \longrightarrow TC_1$.*

In the following, we will consider only timed event structures having a valid timing and call them simply timed event structures.

3. Equivalence notions

In this section, we recall those equivalence notions, based on timed partial orders from [7, 23], which are the timed restrictions of the equivalences from [13, 14] preserved under action refinement.

The partial order semantics of timed event structures is defined by means of timed posets. A *timed poset* is a (labelled) timed event structure $TP = ((E, \leq, \#, l), D)$ with $\# = \emptyset$ and $D(e) = [d, d]$ (or simply $D(e) = d$), where $d \in \mathbb{R}$, for all $e \in E$. We use $\mathcal{TP}$ to indicate the set of timed finite posets. For two timed posets TP and TP', TP is a *direct prefix* of TP' (denoted as $TP \prec TP'$) if $E_{TP} \subseteq E_{TP'}$, $E_{TP'} \setminus E_{TP} = \{e\}$, e is a maximal element of $E_{TP'}$ with respect to $\leq_{TP'}$, $\leq_{TP'}|_{E_{TP} \times E_{TP}} = \leq_{TP}$, $l_{TP'}|_{E_{TP}} = l_{TP}$, and $D_{TP'}|_{E_{TP}} = D_{TP}$.

The first equivalence we consider is a trace equivalence defined in terms of systems languages.

Definition 3. *Let TS and TS' be timed event structures.*

- *The set $L(TS) = \{TP \in \mathcal{TP} \mid TP \simeq TS\lceil TC$ for some $TC \in \mathcal{TC}(TS)\}$ is the language of $TS \in \mathcal{TS}$.*
- *TS and TS' are trace equivalent, denoted as $TS \approx_{trace} TS'$, iff $L(TS) = L(TS')$.*

Example 1. *As an illustration, consider the language of the timed event structure TS_0 from Figure 1:*

$$L(TS_0) = \{(\mathcal{O}, \emptyset),\ \overset{[d_1]}{a},\ \overset{[d_1]}{a} \rightarrow \overset{[d_2]}{b},\ \overset{[1]}{c},\ \begin{array}{l} a\,[d_1] \\ c\,[1] \end{array} \mid d_1 \in [\tfrac{2}{3}, 4],\ d_2 \in [2, 6],$$
$$d_1 \leq d_2\}.$$

The next equivalence we consider is *testing* which is a timed extension of causal testing defined in [14]. Two systems are consider to be testing equivalent, if they pass the same number of tests. A *test* consists of a timed poset $TP \in \mathcal{TP}$ and a set of timed posets $Q \subset \mathcal{TP}$ such that $TP \prec TP_1$ for all $TP_1 \in Q$.

Definition 4. *Let TS and TS' be timed event structures.*

- *Let $TP \in \mathcal{TP}$ and $Q \subset \mathcal{TP}$ such that $\forall TP_1 \in Q \ \diamond\ TP \prec TP_1$.
  Then TS after TP MUST Q iff for all $TC \in \mathcal{TC}(TS)$ such that $TS\lceil TC \simeq TP$ and for all isomorphisms $f : TS\lceil TC \longrightarrow TP$ there exist $TP_1 \in Q$, $TC_1 \in \mathcal{TC}(TS)$ and an isomorphism $f_1 : TS\lceil TC_1 \longrightarrow TP_1$ such that $f \subset f_1$.*
- *TS and TS' are testing equivalent, denoted as $TS \approx_{test} TS'$, iff for all $TP \in \mathcal{TP}$ and $Q \subset \mathcal{TP}$ it holds: TS after TP MUST $Q \iff TS'$ after TP MUST Q.*

Further, we consider the timed extensions of the history preserving bisimulation from [13] which is the culminating point of the poset bisimulation approach.

Definition 5. *Let TS and TS' be timed event structures.*

- *A relation $\mathcal{B}$ consisting of triples (TC, f, TC'), where TC is a timed configuration of TS, TC' is a timed configuration of TS', and $f : TS\lceil TC \rightarrow TS'\lceil TC'$ is an isomorphism, is a history preserving bisimulation between TS and TS' iff $((\emptyset, \emptyset), \emptyset, (\emptyset, \emptyset)) \in \mathcal{B}$ and for all $(TC, f, TC') \in \mathcal{B}$ it holds:*

  (a) if $TC \longrightarrow TC_1$ in TS, then $TC' \longrightarrow TC'_1$ in TS' and $(TC_1, f_1, TC'_1) \in \mathcal{B}$ with $f \subseteq f_1$, for some TC'_1 and f_1,

(b) *if* $TC' \longrightarrow TC'_1$ *in* TS', *then* $TC \longrightarrow TC_1$ *in* TS *and* $(TC_1, f_1, TC'_1) \in \mathcal{B}$ *with* $f \subseteq f_1$, *for some* TC_1 *and* f_1,

- TS *and* TS' *are history preserving equivalent, denoted as* $TS \approx_{hpb} TS'$, *iff there exists a history preserving bisimulation* $\mathcal{B}$ *between them.*

The relationships of the observational equivalences defined above are shown in the following theorem.

Theorem 1. *Let* TS *and* TS' *be timed event structures, then*

$$TS \approx_{trace} TS' \;\Leftarrow\; TS \approx_{test} TS' \;\Leftarrow\; TS \approx_{hpb} TS'.$$

Proof Sketch. Immediately follows from the definitions of the equivalences. ■

The following example shows that the converse implications of the above theorem do not hold and that the three equivalences are all different.

Example 2. *Consider timed event structures on Figure 2. First, we have* $TS_1 \approx_{trace} TS_2$, *while* $TS_1 \not\approx_{test} TS_2$, *since, for example,* TS_1 *after* $a^{[0]}$ *MUST* $a^{[0]} \to b^{[3]}$ *and* $\neg(TS_2$ *after* $a^{[0]}$ *MUST* $a^{[0]} \to b^{[3]})$. *Second,* $TS_2 \approx_{test} TS_3$, *but* $TS_2 \not\approx_{hpb} TS_3$, *because, for instance, the timed configurations of* TS_2 *obtained by the execution of the medium timed action* $(a, 1)$ *can be related neither to the timed configuration of* TS_3 *obtained by the execution of the left timed action* $(a, 1)$, *because the execution of the timed action* $(b, 2)$ *is not further possible in* TC_3, *nor to the timed configuration of* TS_3 *obtained by the execution of the right timed action* $(a, 1)$, *because the execution of the timed action* $(b, 3)$ *is further possible in* TC_3. *Finally,* $TS_3 \approx_{hpb} TS_4$.

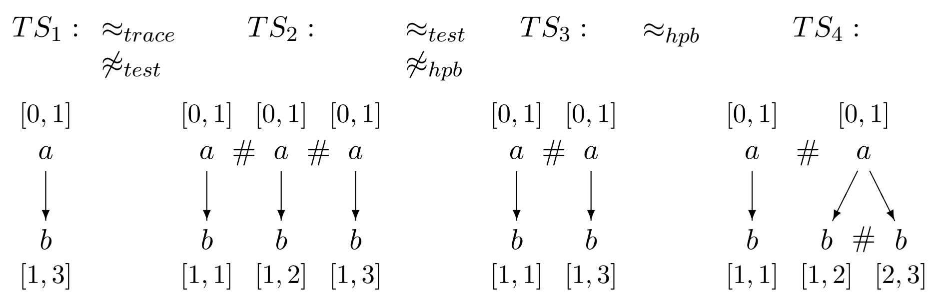

Figure 2

4. Refinement

In this section, the operator of action refinement for timed event structures is proposed. In the theory of event structures, action refinement means substituting single events by complex event structures. A refinement function maps actions (and thereby all events labelled with this action) to finite non-empty conflict-free event structures [13, 14]. In timed event structures, action refinement means also that all timed constraints of single events are inherited by the substituted events.

Definition 6.

- *A function $r : Act \longrightarrow \mathcal{S} \setminus \{\mathcal{O}\}$ is called a refinement function if $r(a)$ is finite and conflict-free for all $a \in Act$.*
- *Let $TS \in \mathcal{TS}$ and r be a refinement function. Then $r(TS)$ is defined in the following way: ($S_{r(TS)}$ is defined as in [13, 14])*
  - $E_{r(TS)} = \{(e, e') \mid e \in E_{TS}, e' \in E_{r(l_{TS}(e))}\}$;
  - $(e, e') \leq_{r(TS)} (e_1, e'_1) \iff e <_{TS} e_1$ *or* $(e = e_1 \wedge e' \leq_{r(l_{TS}(e))} e'_1)$;
  - $(e, e') \#_{r(TS)} (e_1, e'_1) \iff e \#_{TS} e_1$;
  - $l_{r(TS)}(e, e') = l_{r(l_{TS}(e))}(e')$;
  - $D_{r(TS)}(e, e') = D_{TS}(e)$.

By the definition above, if TS has a valid timing then $r(TS)$ has a valid timing too.

The behaviour of the refined timed event structure $r(TS)$ can be compositionally derived from the behaviour of TS and the behaviour of the substituted event structures. For a timed configuration $TC = (C, T)$ of TS, we denote a timed configuration, which is a refinement of TC, as $\widetilde{TC} \in \mathcal{TC}(r(TS))$. A refinement of a timed configuration may be represented as a composition of a refined timed configuration and timed configurations substituted the events in C. For $e \in C$, we define $TS_e = (r(l_{TS}(e)), D_{TS_e})$, where $D_{TS_e}(e') = D_{TS}(e) \cap [\max_{e_1 <_{TS} e} T(e_1), T(e)]$ for all $e' \in E_{r(l_{TS}(e))}$. The timed constraints of TS_e guarantee that those events in $\widetilde{C}$ which substituted e can occur neither earlier than causal predecessors of e in TS, nor later than e. Moreover, we require that if all events in $E_{r(l_{TS}(e))}$ have occurred, then there is an event $e' \in E_{r(l_{TS}(e))}$ which occurred in the time moment $T(e)$. We insist on this requirement to bind times of action occurrences with the ends of executions of the substituting processes.

Proposition 1. *Let TS be a timed event structure and r be a refinement function. $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ is called a refinement of a timed configuration $TC = (C, T) \in \mathcal{TC}(TS)$ by r iff*

- $\widetilde{C} = \bigcup\limits_{e \in C} \{e\} \times C_e$ *and* $\widetilde{T}(e, e') = T_e(e')$, *where* $(C_e, T_e) \in \mathcal{TC}(TS_e) \setminus \{(\emptyset, \emptyset)\}$, $TS_e = (r(l(e)), D_{TS_e}) \in \mathcal{TS}$, *and* $D_{TS_e}(e') = D_{TS}(e) \cap [\max\limits_{e_1 < e} T(e_1),\, T(e)]$;
- $e \in busy(\widetilde{C}) \;\Rightarrow\; e$ *is maximal in* C, $e \notin busy(\widetilde{C}) \;\Rightarrow\; T(e) = \widetilde{T}((e, e'))$ *for some* $e' \in C_e$, *where* $busy(\widetilde{C}) = \{e \in C \mid C_e \neq E_{TS_e}\}$.

Then $\mathcal{TC}(r(TS)) = \{\widetilde{TC} \mid \widetilde{TC}$ *is a refinement of* $TC \in \mathcal{TC}(TS)\}$.

Proof.
"$(\subseteq)$"
Assume that $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ is a timed configuration of $r(TS)$.
For a configuration $\widetilde{C}$, construct the projection functions

$$C = \{e \mid (e, e') \in \widetilde{C}\} \text{ and } C_e = \{e' \mid (e, e') \in \widetilde{C}\},$$

and timed functions $T : C \longrightarrow \mathbb{R}$ and $T_e : C_e \longrightarrow \mathbb{R}$, where

$$T(e) = \max T_e \text{ and } T_e(e') = \widetilde{T}((e, e')) \text{ for all } e \in C \text{ and } e' \in C_e.$$

Now $\widetilde{C}$ and $\widetilde{T}$ can be represented as

$$\widetilde{C} = \bigcup_{e \in C} \{e\} \times C_e \text{ and } \widetilde{T}((e, e')) = T_e(e') \text{ for all } (e, e') \in \widetilde{C}.$$

Further, for all $e \in C$ construct $TS_e = (r(l_{TS}(e)), D_{TS_e})$, where $D_{TS_e}(e') = D(e) \cap [\max\limits_{e_1 < e} T(e_1),\, T(e)]$.

We have to show that $\widetilde{TC}$ is a refinement of $TC = (C, T) \in \mathcal{TC}(TS)$ by r.

1. $TC = (C, T)$ is a timed configuration of TS.

   Since $\widetilde{C}$ is finite, left-closed and conflict-free in $r(TS)$, then $C \subseteq E_{TS}$ is finite, left-closed and conflict-free in TS, therefore it is a configuration.

   Since $\widetilde{TC} \in \mathcal{TC}(r(TS))$, then $\widetilde{T}((e, e')) \in D_{TS}(e)$ for all $(e, e') \in \widetilde{C}$. Hence, $T(e) \in D_{TS}(e)$ for all $e \in C$. Moreover, if $e <_{TS} e_1 \in C$ then $(e, e') <_{r(TS)} (e_1, e'_1)$ for all $e' \in C_e$ and $e'_1 \in C_{e_1}$, which implies $\max(T_e) \leq \max(T_{e_1})$, i.e. $T(e) \leq T(e_1)$.
2. $TS_e = (r(l_{TS}(e)), D_{TS_e})$ is a timed event structure for all $e \in C$.

   By definition, $r(l_{TS}(e)) \in \mathcal{S}$. Besides, $D_{TS_e}(e') \in Interv$ for all $e' \in E_{TS_e}$, since $\widetilde{T}(e, e') \in D_{TS}(e) \cap [\max_{e_1 <_{TS} e} T(e_1),\, T(e)]$ for all $e' \in C_e \neq \emptyset$. Obviously, TS_e has a valid timing.
3. (C_e, T_e) is a nonempty configuration of TS_e for all $e \in C$.

   Since $\widetilde{C}$ is finite, left-closed and conflict-free in $r(TS)$, C_e is finite, left-closed and conflict-free in $r(l_{TS}(e))$, i.e. C_e is a configuration. $C_e \neq \emptyset$, since $(e, e') \in \widetilde{C}$ for some $e' \in C_e$.

   By construction, $T_e(e') \in D_e(e')$ for all $e' \in C_e$. For $e' <_{TS_e} e''$, we get $(e, e') <_{r(TS)} (e, e'')$ which implies $\widetilde{T}((e, e')) \leq \widetilde{T}((e, e''))$ and $T_e(e') \leq T_e(e'')$.

4. $e \in busy(\widetilde{C}) \Rightarrow e$ is maximal in C according to $\leq_{TS}$.

   Suppose $e <_{TS} e_1 \in C$. Then $(e, e') <_{r(TS)} (e_1, e'_1)$ for some $e' \in E_{TS_e}$ and $e'_1 \in C_{e_1} \neq \emptyset$, which implies $C_e = E_{TS_e}$, since $\widetilde{C}$ is left-closed. Hence, $e \notin busy(\widetilde{C})$.

5. By construction of T, for all $e \in C$ it holds: $T(e) = \widetilde{T}((e, e'))$ for some $e' \in C_e$.

"$(\supseteq)$"
Assume that $TC = (C, T) \in \mathcal{TC}(TS)$ and $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ is a refinement of TC. We will show that $\widetilde{TC} \in \mathcal{TC}(r(TS))$.

1. $\widetilde{C}$ is a configuration of $r(TS)$.

   By our assumption, $\widetilde{C} = \bigcup\limits_{e \in C} \{e\} \times C_e$, where $\forall e \in C \ \diamond\ C_e \in \mathcal{C}(r(l_{TS}(e)) \setminus \{\emptyset\}$, and $e \in busy(\widetilde{C}) \Rightarrow e$ is maximal in C. Hence, $\widetilde{C} \subseteq E_{r(TS)}$. Since C and C_e are finite and conflict-free for all $e \in C$, then $\widetilde{C}$ is finite and conflict-free. Let us check that $\widetilde{C}$ is left-closed. Suppose $(e_1, e'_1) <_{r(TS)} (e, e') \in \widetilde{C}$, and consider the following cases:

   (a) if $e_1 = e$ then $e'_1 <_{r(l_{TS}(e))} e' \in C_e$. Since C_e is left-closed, we get $e'_1 \in C_e$ and $(e_1, e'_1) \in \widetilde{C}$;
   (b) if $e_1 <_{TS} e$ then $e_1 \in C$ since C is left-closed, and e_1 is not maximal in C. Then $e_1 \notin busy(\widetilde{C})$, i.e. $C_{e_1} = E_{r(l_{TS}(e_1))}$ which implies $e'_1 \in C_{e_1}$ and $(e_1, e'_1) \in \widetilde{C}$.

2. $\widetilde{T}$ satisfies the Definition of a timed configuration.

   By our assumption, $\widetilde{T}((e, e')) = T_e(e') \in D_{TS}(e)$ for all $(e, e') \in \widetilde{C}$. For $(e_1, e'_1) \leq_{r(TS)} (e, e') \in \widetilde{C}$, the following cases are possible:

   (a) if $e_1 = e$ then $e'_1, e' \in C_e$ and $e'_1 \leq_{r(l_{TS}(e))} e'$, which implies $T_e(e'_1) \leq T_e(e')$, or $\widetilde{T}((e_1, e'_1)) \leq \widetilde{T}((e, e'))$;
   (b) if $e_1 <_{TS} e$ then $T(e_1) \leq T(e)$, that, by definition of TS_{e_1} and TS_e, implies $T_{e_1}(e'_1) \leq T_e(e')$, or $\widetilde{T}((e_1, e'_1)) \leq \widetilde{T}((e, e'))$.

Thus, $\widetilde{TC} \in \mathcal{TC}(r(TS))$. ■

The next lemma illustrates the interplay between the leading relation on a timed configuration and a refinement operator.

Lemma 2. *Let $TS \in \mathcal{TS}$ and r, r' be refinement functions, then we have*

1) refinement transitivity: if $\widetilde{TC} \in \mathcal{TC}(r(TS))$ is a refinement of $TC \in \mathcal{TC}(TS)$ by r and $\widetilde{\widetilde{TC}} \in \mathcal{TC}(r' \circ r(TS))$ is a refinement of $\widetilde{TC}$ by r', then $\widetilde{\widetilde{TC}}$ is a refinement of TC by $r \circ r'$;

2) if $\widetilde{TC} \longrightarrow \widetilde{TC_1}$ in $r(TS)$ and $\widetilde{TC_1}$ is a refinement of $TC_1 \in \mathfrak{TC}(TS)$, then $\widetilde{TC}$ is a refinement of TC such that $TC \longrightarrow TC_1$ in TS;

3) if $\widetilde{TC} \longrightarrow \widetilde{TC_1}$ in $r(TS)$, $\widetilde{TC}$ is a refinement of $TC \in \mathfrak{TC}(TS)$, and $\forall (e, e') \in \widetilde{C_1} \setminus \widetilde{C} \diamond e \notin C$, then $\widetilde{TC_1}$ is a refinement of $TC_1 \in \mathfrak{TC}(TS)$ such that $TC \longrightarrow TC_1$ in TS.

Proof Sketch. Immediately follows from the definitions of the equivalences. ∎

As was shown in [13, 14], the behaviour equivalences of event structures whose timed extensions were exposed above are invariant under refinement. However, by the following example, their considered timed extensions are not preserved under refinement of timed event structures.

Example 3. *Consider timed event structures*

$$TS_5 : \overset{[0,2]}{a}\ , \qquad\qquad TS_6 : \overset{[0,1]}{a}\ \#\ \overset{[1,2]}{a}\ ,$$

and a refinement function $r(a) = a_1 \to a_2$. We have $TS_5 \approx_{hpb} TS_6$, but $r(TS_5) \not\approx_{trace} r(TS_6)$, since $a_1^{[0]} \to a_2^{[2]} \in L(r(TS_5)) \setminus L(r(TS_6))$:

$$r(TS_5) : \overset{[0,2]}{a_1} \to \overset{[0,2]}{a_2}\ , \qquad\qquad r(TS_6) : \begin{array}{c} [0,1]\ a_1 \to a_2\ [0,1] \\ \# \\ [1,2]\ a_1 \to a_2\ [1,2] \end{array} .$$

Roughly speaking, action implementations after refinement could acquire some positive durations reflecting time periods during which the substituted events occur. Since time restrictions for the corresponding events from equivalent event structures may not coincide, the resulted refinements may be non-equivalent.

Further we introduce two possible solutions to the posed problem of preservation of equivalences under refinement of timed event structures. One of them consists in incorporating additional time requirements into the definitions of equivalences. However, another solution is based on restricting the considering model of timed event structures onto certain subclasses.

5. Equivalence notions with additional timing requirements

We expose here the equivalence notions strengthened by additional requirements on the *time neighborhood* of timed configuration sufficient to preserve them under refinement of timed event structures.

Definition 7. *Let $TS \in \mathcal{TS}$ and $TC = (C, T) \in \mathcal{TC}(TS)$. A function $\delta_{TC} : C \longrightarrow Interv$ is called a time neighborhood of TC, if*

$$\delta_{TC}(e) = \begin{cases} D_{TS}(e) \bigcap [\max\limits_{e_1 <_{TS} e} T(e_1), \infty], & \text{if } e \text{ is maximal in } C; \\ D_{TS}(e) \bigcap [\max\limits_{e_1 <_{TS} e} T(e_1), T(e)], & \text{otherwise.} \end{cases}$$

Roughly speaking, a time neighborhood δ_{TC} of a timed configuration $TC = (C, T) \in \mathcal{TC}(TS)$ defines a timed segment $\delta_{TC}(e)$ for all $e \in C$, in which the constraints events $(e, e') \in \widetilde{C_1}$ can occur, for all timed configurations $\widetilde{TC}, \widetilde{TC_1}$ of $r(TS)$ such that $\widetilde{TC}$ is a refinement of TC by r and $\widetilde{TC} \longrightarrow \widetilde{TC_1}$ in $r(TS)$.

Further, consider the equivalence notions for timed event structures with additional requirements concerning the notion of a time neighborhood of a timed configuration.

First, consider the δ-trace equivalence.

Definition 8. *Let TS and TS' be timed event structures.*

- *The set $L_\delta(TS) = \{(TP, \delta) \in \mathcal{TP} \mid$ there exist $TC \in \mathcal{TC}(TS)$ and an isomorphism $f : TS\lceil TC \longrightarrow TP$ such that $\delta_{TC} = \delta \circ f\}$ is the δ-language of $TS \in \mathcal{TS}$.*
- *TS and TS' are δ-trace equivalent, denoted as $TS \approx_{\delta\text{-}trace} TS'$, iff $L_\delta(TS) = L_\delta(TS')$.*

It is obvious that δ-trace equivalent timed event structures are trace equivalent.

The next equivalence notion we consider is δ-testing, where δ-*test* consists of a timed poset $TP \in \mathcal{TP}$, a function $\delta : E_{TP} \longrightarrow Interv$, and a set $Q \subseteq \mathcal{TP}$ such that $TP \prec TP_1$ for all $TP_1 \in Q$.

Definition 9. *Let TS and TS' are timed event structures.*

- *Let $TP \in \mathcal{TP}$, $\delta : E_{TP} \longrightarrow Interv$ and $Q \subset \mathcal{TP}$ such that $\forall TP_1 \in Q \diamond TP \prec TP_1$.*
  TS after (TP, δ) MUST Q iff for all $TC \in \mathcal{TC}(TS)$ such that $TS\lceil TC \simeq TP$, and for all isomorphisms $f : TS\lceil TC \longrightarrow TP$ satisfying the condition $\delta_{TC} = \delta \circ f$, there exist $TP_1 \in Q$, $TC_1 \in \mathcal{TC}(TS)$ and an isomorphism $f_1 : TS\lceil TC_1 \longrightarrow TP_1$ with $f \subset f_1$.
- *TS and TS' are δ-testing equivalent, denoted as $TS \approx_{\delta\text{-}test} TS'$, iff for all $TP \in \mathcal{TP}$, $\delta : E_{TP} \longrightarrow Interv$ and $Q \subset \mathcal{TP}$ it holds:*
  TS after (TP, δ) MUST $Q \iff TS'$ after (TP, δ) MUST Q.

It is easy to check that δ-testing equivalent timed event structures are testing equivalent.

Finally, consider the δ-history preserving equivalence.

Definition 10. *Let TS and TS' be timed event structures.*

- *A history preserving bisimulation $\mathcal{B}$ between TS and TS' is called a δ-history preserving bisimulation iff $\forall (TC, f, TC') \in \mathcal{B} \ \diamond \ \delta_{TC} = \delta_{TC'} \circ f$.*
- *TS and TS' are δ-history preserving equivalent, denoted as $TS \approx_{\delta\cdot hpb} TS'$ iff there exists a δ-history preserving bisimulation between them.*

The interrelationships between behavioural equivalences and δ-equivalences of timed event structures are shown in the following theorems.

Theorem 2. *Let TS and TS' be timed event structures and*

$$\alpha \in \{trace, test, hpb\},$$

then

$$TS \approx_{\alpha} TS' \ \Leftarrow \ TS \approx_{\delta\cdot\alpha} TS'.$$

Proof Sketch. Immediately follows from the definitions of the equivalences. ■

The converse implications of the above theorem do not hold, because, for example, for TS_3 and TS_4 shown on Figure 2 we have $TS_3 \approx_{hpb} TS_4$ and $TS_3 \not\approx_{\delta\cdot trace} TS_4$, since $(a^{[0]}, [0,1]) \to (b^{[1]}, [1,2]) \in L_\delta(TS_4) \setminus L_\delta(TS_4)$.

Theorem 3. *Let TS and TS' be timed event structures, then*

$$TS \approx_{\delta\cdot trace} TS' \ \Leftarrow \ TS \approx_{\delta\cdot test} TS' \ \Leftarrow \ TS \approx_{\delta\cdot hpb} TS'.$$

Proof Sketch. Immediately follows from the definitions of the equivalences. ■

The following example shows that the converse implications of the above theorem do not hold and that the three δ-equivalences are all different.

Example 4. *Consider timed event structures shown on Figure 3. First, we have $TS_7 \approx_{\delta\cdot trace} TS_8$, while $TS_7 \not\approx_{\delta\cdot test} TS_8$, since, for example, TS_8 after $(a^{[0]}, [0,1])$ $MUST\, a^{[0]} \to b^{[2]}$ and $\neg(TS_7$ after $(a^{[0]}, [0,1])$ $MUST\, a^{[0]} \to b^{[2]})$. Second, $TS_8 \approx_{\delta\cdot test} TS_9$, but $TS_8 \not\approx_{\delta\cdot hpb} TS_9$, because, for instance, the timed configurations of TS_8 obtained by the execution of the right timed action $(a, 1)$ can't be related to the correspondent timed configuration of TS_9, because the execution of the timed action $(b, 1)$ with the time neighborhood $[1, 1]$ is not further possible in TC_8, that is not the case in TC_9. Finally, $TS_9 \approx_{\delta\cdot hpb} TS_{10}$.*

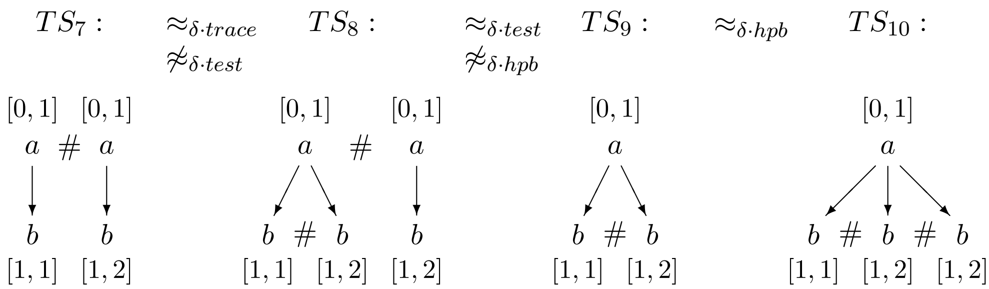

Figure 3

Finally, we will prove that δ-equivalences indeed are invariant under refinement. If two systems are δ-equivalent, then, after refining actions in both systems in the same way, the resulting systems are still δ-equivalent. For this purpose we need the following lemma.

Lemma 3.

Let $TC \in \mathfrak{TC}(TS)$, $TC' \in \mathfrak{TC}(TS')$, $f : TS\lceil TC \longrightarrow TS'\lceil TC'$ be an isomorphism, and $\delta_{TC} = \delta_{TC'} \circ f$. Then if $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ is a refinement of TC by r, then $\widetilde{TC'} = (\widetilde{f}(\widetilde{C}), \widetilde{T} \circ \widetilde{f}^{-1})$ is a refinement of TC' by r, where $\widetilde{f}((e, e')) = (f(e), e')$ is an isomorphism between $r(TS)\lceil\widetilde{TC}$ and $r(TS')\lceil\widetilde{TC'}$ with $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{f}$.

Proof.

Let $TC = (T, C) \in \mathfrak{TC}(TS)$, $TC' = (C', T') \in \mathfrak{TC}(TS')$, $f : TS\lceil TC \longrightarrow TS'\lceil TC'$ be an isomorphism, $\delta_{TC} = \delta_{TC'} \circ f$, and $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ be a refinement of TC, i.e.

$$\widetilde{C} = \bigcup_{e \in C} \{e\} \times C_e \ \text{ and } \ \widetilde{T}((e, e')) = T_e(e') \text{ for all } (e, e') \in \widetilde{C},$$

where $(C_e, T_e) \in \mathfrak{TC}(TS_e) \setminus \{(\emptyset, \emptyset)\}$, $TS_e = (r(l(e)), D_{TS_e}) \in \mathfrak{TS}$. Since $f : TS\lceil TC \longrightarrow TS'\lceil TC'$ is an isomorphism, every refinement of TC' has a form $\widetilde{TC'} = (\widetilde{C'}, \widetilde{T'})$, where

$$\widetilde{C'} = \bigcup_{e \in C} \{f(e)\} \times C_{f(e)} \ \text{ and } \ \widetilde{T'}(f(e), e') = T_{f(e)}(e') \text{ for all } (e, e') \in \widetilde{C},$$

where $(C_{f(e)}, T_{f(e)}) \in \mathfrak{TC}(TS_{f(e)}) \setminus \{(\emptyset, \emptyset)\}$, $TS_{f(e)} = (r(l(e)), D_{TS_{f(e)}}) \in \mathfrak{TS}$.

Since $\delta_{TC} = \delta_{TC'} \circ f$, we have $D_{TS_e} = D_{TS_{f(e)}}$ for all $e \in C$, which implies $TS_e = TS_{f(e)}$. Hence, $(C_e, T_e) \in \mathfrak{TC}(TS_{f(e)})$, and therefore $\widetilde{TC'} = (\widetilde{C'}, \widetilde{T'})$ is a refinement of TC', where

$$\widetilde{C'} = \textstyle\bigcup_{e \in C} \{f(e)\} \times C_e \text{ and } \widetilde{T'}(f(e), e') = T_e(e') \text{ for all } (e, e') \in \widetilde{C}.$$

By Definition 6 of the refinement of TS, a bijection $\widetilde{f} : r(TS)\lceil\widetilde{TC} \longrightarrow r(TS')\lceil\widetilde{TC'}$, where $\widetilde{f}(e, e') = (f(e), e)$, is an isomorphism because it preserves labels, partial order and the timing function. Thus, $\widetilde{TC'} = (\widetilde{f}(\widetilde{C}), \widetilde{T} \circ \widetilde{f}^{-1})$. Moreover,

$$\delta_{\widetilde{TC}}((e,e')) = \begin{cases} \delta_{TC}(e) \bigcap [\max_{e'_1 <_{TS_e} e'} \widetilde{T}((e,e'_1)), \infty], \text{ if } (e,e') \text{ is maximal in } \widetilde{C}; \\ \delta_{TC}(e) \bigcap [\max_{e'_1 <_{TS_e} e'} \widetilde{T}((e,e'_1)), \widetilde{T}((e,e'))], \text{ otherwise.} \end{cases}$$

Since $\widetilde{T} = \widetilde{T}' \circ \widetilde{f}$ and $\delta_{TC} = \delta_{TC'} \circ f$, we have $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{f}$. ■

Theorem 4. *Let TS and TS' be timed event structures, r be a refinement function. Then*

(i) $TS \approx_{\delta\text{-}trace} TS' \;\Rightarrow\; r(TS) \approx_{\delta\text{-}trace} r(TS')$,

(ii) $TS \approx_{\delta\text{-}test} TS' \;\Rightarrow\; r(TS) \approx_{\delta\text{-}test} r(TS')$,

(iii) $TS \approx_{\delta\text{-}hpb} TS' \;\Rightarrow\; r(TS) \approx_{\delta\text{-}hpb} r(TS')$.

Proof.

(i)

Assume $TS \approx_{\delta\text{-}trace} TS'$.

Take arbitrary $(\widetilde{TP}, \widetilde{\delta}) \in L_\delta(r(TS))$. By definition, there exist $\widetilde{TC} \in \mathcal{TC}(r(TS))$ and an isomorphism $f : r(TS)\lceil\widetilde{TC} \longrightarrow \widetilde{TP}$ such that $\delta_{\widetilde{TC}} = \widetilde{\delta} \circ f$. Hence by Proposition 1, $\widetilde{TC}$ is a refinement of some $TC = (C, T) \in \mathcal{TC}(TS)$. Let $TP = TS\lceil TC$ and $\delta = \delta_{TC}$. Then $(TP, \delta) \in L_\delta(TS)$, which by our assumption implies $(TP, \delta) \in L_\delta(TS')$. This means that there exist $TC' = (C', T') \in \mathcal{TC}(TS')$ and an isomorphism $\varphi : TS\lceil TC \longrightarrow TS'\lceil TC'$ such that $\delta_{TC} = \delta_{TC'} \circ \varphi$. By Lemma 3, we get that $\widetilde{TC'} = (\widetilde{\varphi}(\widetilde{C}), \widetilde{T} \circ \widetilde{\varphi}^{-1})$ is a refinement of TC', where $\widetilde{\varphi} : r(TS)\lceil\widetilde{TC} \longrightarrow r(TS')\lceil\widetilde{TC'}$ is an isomorphism, $\widetilde{\varphi}((e,e')) = (\varphi(e), e')$, and $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{\varphi}$. Hence, $(\widetilde{TP}, \widetilde{\delta}) \in L_\delta(r(TS))$.

The arbitrary choice of $(\widetilde{TP}, \widetilde{\delta}) \in L_\delta(r(TS))$ guarantees $r(TS) \approx_{\delta\text{-}trace} r(TS')$.

(ii)

Assume $TS \approx_{\delta\text{-}test} TS'$.

Suppose $r(TS')$ *after* $(\widetilde{TP}, \widetilde{\delta})$ *MUST* $\widetilde{Q}$ for some $\widetilde{TP} \in \mathcal{TP}$, $\widetilde{\delta} : E_{\widetilde{TP}} \longrightarrow$ *Interv* and $\widetilde{Q} \subset \mathcal{TP}$. We will show that $r(TS)$ *after* $(\widetilde{TP}, \widetilde{\delta})$ *MUST* $\widetilde{Q}$.

By Theorem 3 and Item (i) of the Theorem, we have $r(TS) \approx_{\delta\text{-}trace} r(TS')$, that impies $(\widetilde{TP}, \widetilde{\delta}) \in L_\delta(r(TS))$. Take arbitrary $\widetilde{TC} = (\widetilde{C}, \widetilde{T}) \in \mathcal{TC}(r(TS))$ and an isomorphism $f : r(TS)\lceil\widetilde{TC} \longrightarrow \widetilde{TP}$ such that $\delta_{\widetilde{TC}} = \widetilde{\delta} \circ f$. We have to show that there exist $\widetilde{TP_1} \in \widetilde{Q}$, $\widetilde{TC_1} \in \mathcal{TC}(r(TS))$ and an isomorphism $f_1 : r(TS)\lceil\widetilde{TC_1} \longrightarrow \widetilde{TP_1}$ such that $f \subset f_1$.

By Proposition 1, $\widetilde{TC}$ is a refinement of some $TC = (C, T) \in \mathcal{TC}(TS)$.

Denote $TP = TS\lceil TC$ and $\delta = \delta_{TC}$. Then $(TP, \delta) \in L_\delta(TS)$, that by our assumption and Theorem 3 implies $(TP, \delta) \in L_\delta(TS')$. So, there exist timed configurations of TS' isomorphic to $TP = TS\lceil TC$ such that the correspondent isomorphisms preserve the time neighborhood $\delta = \delta_{TC}$. By Lemma 3, for each such timed configuration with an isomorphism, there is a refinement of it which is a timed configuration in $r(TS')$ isomorphic to $r(TS)\lceil\widetilde{TC}$ (therefore to $\widetilde{TP}$) with the correspondent isomorphism preserving the time neighborhood $\delta_{\widetilde{TC}}$.

Following our assumption, we have $r(TS')$ *after* $(\widetilde{TP}, \widetilde{\delta})$ *MUST* $\widetilde{Q}$. So, for all $\widetilde{TC'} \in \mathcal{TC}(r(TS'))$ such that $r(TS')\lceil\widetilde{TC'} \simeq \widetilde{TP}$ and for al isomorphisms $g : r(TS')\lceil\widetilde{TC'} \longrightarrow \widetilde{TP}$ satisfying the condition $\delta_{\widetilde{TC'}} = \widetilde{\delta} \circ g$, there exist $\widetilde{TP_1} \in \widetilde{Q}$, $\widetilde{TC'_1} \in \mathcal{TC}(r(TS))$ and an isomorphism $g_1 : r(TS)\lceil\widetilde{TC'_1} \longrightarrow \widetilde{TP_1}$ such that $g \subseteq g_1$. Note that $\widetilde{TC'} \longrightarrow \widetilde{TC'_1}$ in $r(TS')$ and $\widetilde{C'_1} = \widetilde{C'} \uplus \{(e, e')\}$ for some $(e, e') \in \widetilde{C'}$, since $\widetilde{TP} \prec \widetilde{TP_1}$.

Further, consider two possible cases.

1. *There exist $TC' \in \mathcal{TC}(TS')$ and an isomorphism $\varphi : TS\lceil TC \longrightarrow TS'\lceil TC'$ with $\delta_{TC} = \delta_{TC'} \circ \varphi$ such that for a refinement $\widetilde{TC'} = (\widetilde{\varphi}(\widetilde{C}), \widetilde{T} \circ \widetilde{\varphi}^{-1})$ of TC' and for an isomorphism $\widetilde{\varphi}$, which exist by Lemma 3 (where $\widetilde{\varphi}((e, e')) = (\varphi(e), e')$ for all $(e, e') \in \widetilde{C}$ is an isomorphism between $r(TS)\lceil\widetilde{TC}$ and $r(TS')\lceil\widetilde{TC'}$ with $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{\varphi}$), there are a timed configuration $\widetilde{TC'_1} \in \mathcal{TC}(r(TS'))$, a timed poset $\widetilde{TP_1} \in \widetilde{Q}$ and an isomorphism $g_1 : r(TS')\lceil\widetilde{TC'_1} \longrightarrow \widetilde{TP_1}$ satisfying the δ-test we have chosen, i.e. $g \subset g_1$ for $g = f \circ \widetilde{\varphi}^{-1} : r(TS')\lceil\widetilde{TC'} \longrightarrow \widetilde{TP}$, such that:*

$$\widetilde{C'_1} = \widetilde{C'} \uplus \{(\mathbf{e}, \mathbf{e}')\} \textit{ with } \mathbf{e} \in C'.$$

   By Proposition 1, $\widetilde{TC'_1}$ is a refinement of some $TC'_1 = (C'_1, T'_1) \in \mathcal{TC}(TS')$. Since $\mathbf{e} \in C'$ we get $C' = C'_1$. Since $\widetilde{TC} \longrightarrow \widetilde{TC'_1}$, we have $\widetilde{T'}((e, e')) = \widetilde{T'_1}((e, e'))$ for all $(e, e') \in \widetilde{C'}$. By the definition of a refinement of a timed configuration, we get $T'(e) = T'_1(e) = \max_{(e,e') \in \widetilde{C'}} \widetilde{T'}((e, e'))$ for all $e \notin busy(\widetilde{C})$. Hence, by the definition of a time neighborhood, we get $\delta_{TC'} = \delta_{TC'_1}$ that implies $T'_1(e) \in \delta_{TC'}(e)$ for all $e \in C'$.

   Construct $T_1 = T'_1 \circ \psi$. Since $\delta_{TC} = \delta_{TC'} \circ \varphi$, we get $T_1(e) \in \delta_{TC}(e) \subseteq D_{TS}(e)$ for all $e \in C$. By the definition of a timed configuration, $TC_1 = (C, T_1) \in \mathcal{TC}(TS)$, since $\varphi : TS\lceil TC_1 \longrightarrow TS'\lceil TC'_1$ is an isomorphism. Moreover, φ is an isomorphism between $TS\lceil TC_1$ and $TS'\lceil TC'_1$. Consider the inverse isomorphism $\psi = \varphi^{-1} : TS'\lceil TC'_1 \longrightarrow$

$TS\lceil TC_1$. Applying Lemma 3, we get that $\widetilde{TC_1} = (\widetilde{\psi}(\widetilde{C_1'}), \widetilde{T_1'} \circ \widetilde{\psi}^{-1})$ is a refinement of TC_1, where $\widetilde{\psi}((e,e')) = (\psi(e), e')$ for all $(e,e') \in \widetilde{C_1'}$ is an isomorphism between $r(TS')\lceil\widetilde{TC_1'}$ and $r(TS)\lceil\widetilde{TC_1}$.

Since $\widetilde{C'} \subset \widetilde{C_1'}$, we have $\widetilde{\varphi} \subset \widetilde{\psi}$. Thus, $f_1 = g_1^{-1} \circ \widetilde{\psi}^{-1} : r(TS)\lceil\widetilde{TC_1} \longrightarrow \widetilde{TP_1}$ and $f \subset f_1$, which we had to show.

2. *For all $TC' \in \mathcal{TC}(TS')$ and for all isomorphisms $\varphi : TS\lceil TC \longrightarrow TS'\lceil TC'$ with $\delta_{TC} = \delta_{TC'} \circ \varphi$, for a refinement $\widetilde{TC'} = (\widetilde{\varphi}(\widetilde{C}), \widetilde{T} \circ \widetilde{\varphi}^{-1})$ of TS' and for isomorphism $\widetilde{\varphi}$ which exist by Lemma 3 (where $\widetilde{\varphi}((e,e')) = (\varphi(e), e')$ for all $(e,e') \in \widetilde{C}$ is an isomorphism between $r(TS)\lceil\widetilde{TC}$ and $r(TS')\lceil\widetilde{TC'}$, $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{\varphi}$), it holds that all timed configurations $\widetilde{TC_1'} \in \mathcal{TC}(r(TS'))$, all timed posets $\widetilde{TP_1} \in \widetilde{Q}$ and all isomorphisms $g_1 : r(TS')\lceil\widetilde{TC_1'} \longrightarrow \widetilde{TP_1}$, satisfying the δ-test we have chosen, i.e. $g \subset g_1$ for $g = f \circ \widetilde{\varphi}^{-1} : r(TS')\lceil\widetilde{TC'} \longrightarrow \widetilde{TP}$, also satisfying*

$$\widetilde{C_1'} = \widetilde{C'} \uplus \{(\mathbf{e}, \mathbf{e'})\},\ \textit{where } \mathbf{e} \notin C'.$$

For $TP = TS\lceil TC$ and $\delta = \delta_{TC}$, we construct a δ-test with the set $Q \subset \mathcal{TP}$ using $\widetilde{Q}$ and all of TC' and φ such as we explained above:

- suppose $TC' \in \mathcal{TC}(TS')$ and isomorphism $\varphi : TS\lceil TC \longrightarrow TS'\lceil TC'$ such that $\delta_{TC} = \delta_{TC'} \circ \varphi$. Then, by Lemma 3, there exist $\widetilde{TC'} = (\widetilde{\varphi}(\widetilde{C}), \widetilde{T} \circ \widetilde{\varphi}^{-1})$, a refinement of TC', where $\widetilde{\varphi}((e,e')) = (\varphi(e), e')$ for all $(e,e') \in \widetilde{C'}$ is an isomorphism between $r(TS)\lceil\widetilde{TC}$ and $r(TS')\lceil\widetilde{TC'}$ with $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{\varphi}$. Hence, the mapping $g = f \circ \widetilde{\varphi}^{-1} : r(TS')\lceil\widetilde{TC'} \longrightarrow \widetilde{TP}$ is an isomorphism with $\delta_{\widetilde{TC'}} = \widetilde{\delta} \circ g$. By our assumption, there exist $\widetilde{TC_1'} \in \mathcal{TC}(r(TS'))$, $\widetilde{TP_1} \in \widetilde{Q}$ and an isomorphism $g_1 : r(TS')\lceil\widetilde{TC_1'} \longrightarrow \widetilde{TP_1}$ such that $g \subset g_1$. According to the case we consider, we have $\widetilde{C_1'} = \widetilde{C'} \uplus \{(\mathbf{e}, \mathbf{e'})\}$, where $\mathbf{e} \notin C'$. By Lemma 2(3), $\widetilde{TC_1'}$ is a refinement of $TC_1' \in \mathcal{TC}(TS)$ such that $TC' \longrightarrow TC_1'$ in TS'. It is obvious that $C_1' = C' \uplus \{\mathbf{e}\}$. W.l.o.g., suppose that $T_1'(\mathbf{e}) = \widetilde{T_1'}((\mathbf{e}, \mathbf{e'}))$. Construct a timed poset $TP_{TC',\varphi} = (E, \leq, l, D)$, where
  - $E = E_{TP} \cup \{\mathbf{e}\}$;
  - $\leq\ = \ \leq_{TP} \cup \{(e, \mathbf{e}) \mid e \in E_{TP} \text{ and } \varphi(e) <_{TS'} \mathbf{e}\} \cup \{(\mathbf{e}, \mathbf{e})\}$;
  - $l\,|_{E_{TP}} = l_{TP}$, $l(\mathbf{e}) = l_{TS'}(\mathbf{e})$;
  - $D\,|_{E_{TP}} = D_{TP}$, $D(\mathbf{e}) = T_1'(\mathbf{e})$.

  By construction, $TP \prec TP_{TC',\varphi}$ and the mapping $\psi : TS'\lceil TC_1' \longrightarrow TP_{TC',\varphi}$ is an isomorphism, where $\psi|_{C'} = \varphi^{-1}$ and $\psi(\mathbf{e}) = \mathbf{e}$.

Let Q consist of all such $TP_{TC',\varphi}$. Then TS' *after* (TP, δ) *MUST* Q and by our assumption we get TS *after* (TP, δ) *MUST* Q.

Hence, there exist $TC_1 = (C_1, T_1) \in \mathcal{TC}(TS)$, $TP_{TC',\varphi} \in Q$ and an isomorphism $\iota_1 : TC_1 \longrightarrow TP_{TC',\varphi}$ such that $\iota \subset \iota_1$, where $\iota : TS\lceil TC \longrightarrow TP$ is the identity mapping. Then, $\varphi_1 = \psi^{-1} \circ \iota_1 : TS\lceil TC_1 \longrightarrow TS'\lceil TC'_1$ is an isomorphism and $\varphi \subset \varphi_1$. Since δ_{TC_1} should not be equal to $\delta_{TC'_1 \circ f}$, we can't apply Lemma 3.

Consider a set $\widetilde{C_1} = \widetilde{C} \uplus \{(\varphi^{-1}(\mathbf{e}), \mathbf{e}')\}$ and a timing function $\widetilde{T_1} : \widetilde{C_1} \longrightarrow \mathbb{R}$, where $\widetilde{T_1}|_{\widetilde{C}} = \widetilde{T}$ and $\widetilde{T_1}((\varphi^{-1}(\mathbf{e}), \mathbf{e}')) = T_1(\varphi^{-1}(\mathbf{e}))$. Since $\varphi^{-1}(e) \notin C$ and $\{\mathbf{e}'\}$ is a configuration of $r(l_{TC'}(\mathbf{e}))$, then, by the definition of a refinement of a timed configuration, we get that $\widetilde{TC_1} = (\widetilde{C_1}, \widetilde{T_1})$ is a refinement of TC_1. Moreover, the mapping $\widetilde{\varphi_1}((e, e')) = (\varphi_1(e), e')$ for all $(e, e') \in \widetilde{C'_1}$ is an isomorphism between $r(TS)\lceil\widetilde{TC_1}$ and $r(TS')\lceil\widetilde{TC'_1}$ such that $\widetilde{\varphi} \subset \widetilde{\varphi_1}$. Hence, the mapping $f_1 = g_1 \circ \widetilde{\varphi_1} : r(TS)\lceil\widetilde{TC_1} \longrightarrow TP_1$ is an isomorphism such that $f \subset f_1$, that we had to show.

The arbitrary choice of $\widetilde{TP}$, $\widetilde{\delta}$ and $\widetilde{Q}$ guarantees $r(TS) \approx_{\delta\text{-}test} r(TS')$.

(iii)

Assume $TS \approx_{\delta\text{-}hpb} TS'$. Then there exists a δ-history preserving bisimulation $\mathcal{B}$ between TS and TS'.

Further we will need an additional auxiliary notation. Let μC be the timed configuration of TS containing the configuration C and the minimal possible timing function T, i.e. $\mu C = (C, T)$, where $T(e) = \min D_{TS}(e)$ for all $e \in C$. This notation is correct, since TS has a valid timing.

Construct a new relation $\mathcal{B}^{\mu}$ in the following way:

$$\mathcal{B}^{\mu} = \{(TC, f, TC') \mid TC \in \mathcal{TC}(TS),\ TC' \in \mathcal{TC}(TS'),\ T = T' \circ f \text{ and } (\mu C, f, \mu C') \in \mathcal{B}\}.$$

We have to check that $\mathcal{B}^{\mu}$ is a δ-history preserving bisimulation between TS and TS'.

1. $((\emptyset, \emptyset), \emptyset, (\emptyset, \emptyset)) = (\mu\emptyset, \emptyset, \mu\emptyset) \in \mathcal{B}^{\mu}$.
2. Let $(TC, f, TC') \in \mathcal{B}^{\delta}$.
   (a) Since f preserves a partial order and labels, and $T = T' \circ f$, we resume that f is an isomorphism between $TS\lceil TC$ and $TS'\lceil TC'$.
   (b) Since $\delta_{\mu C} = \delta_{\mu C'}$, we have $\min D_{TS}(e) = \min D_{TS'}(f(e))$ for all $e \in C$ and $D_{TS}(e) = D_{TS'}(f(e))$ for all maximal $e \in C$. Hence, by the definition of δ, we get $\delta_{TC} = \delta_{TC'} \circ f$.

(c) Suppose $TC \longrightarrow TC_1$ in TS. By the construction of $\mathcal{B}^{\mu}$, we have $(\mu C, f, \mu C') \in \mathcal{B}$. Since $C \subseteq C_1$ and TS has a valid timing, $\mu C \longrightarrow \mu C_1$ in TS. Then there exist $TC'_1 \in \mathcal{TC}(TS')$ and an isomorphism $f_1 : TS\lceil \mu C_1 \longrightarrow TS'\lceil TC'_1$ such that $\mu C' \longrightarrow TC'_1$, $f \subseteq f_1$ and $(\mu C_1, f_1, TC'_1) \in \mathcal{B}$. By the definition of $\mathcal{B}$, we have $\delta_{\mu C_1} = \delta_{TC'_1}$ which implies $TC'_1 = \mu C'_1$. By the construction of $\mathcal{B}^{\mu}$, we get $(TC_1, f_1, TC''_1,) \in \mathcal{B}^{\mu}$ for $TC''_1 = (C'_1, T_1 \circ f_1)$. Since $C' \subseteq C'_1$ and $T' = T'_1 \mid_{C'}$, we get $TC' \longrightarrow TC'_1$ in TS'.

(d) Symmetric to item (c).

Further, we construct a relation $\widetilde{\mathcal{B}}$ between $r(TS)$ and $r(TS')$ using $\mathcal{B}^{\mu}$.

$\widetilde{\mathcal{B}} = \{(\widetilde{TC}, \widetilde{f}, \widetilde{TC'}) \mid (TC, f, TC') \in \mathcal{B}^{\mu}$ such that:

– $\widetilde{TC} = (\widetilde{C}, \widetilde{T})$ is a refinement of TC by r, where $\widetilde{C} = \bigcup_{e \in C} \{e\} \times C_e$ and $\widetilde{T}(e, e') = T_e(e')$, $(C_e, T_e) \in \mathcal{TC}(TS_e) \setminus \{(\emptyset, \emptyset)\}$ for all $e \in C$,

– $\widetilde{TC'} = (\widetilde{C'}, \widetilde{T'})$ is a refinement of TC' by r, where $\widetilde{C'} = \bigcup_{e \in C} \{f(e)\} \times C_e$ and $\widetilde{T'}(f(e), e') = T_e(e')$,

– $\widetilde{f} : \widetilde{C} \longrightarrow \widetilde{C'}$ is a bijection satisfying $\widetilde{f}(e, e') = (f(e), e')$.

We have to show that $\widetilde{\mathcal{B}}$ is a δ-history preserving bisimulation between $r(TS)$ and $r(TS')$.

1. Since $((\emptyset, \emptyset), \emptyset, (\emptyset, \emptyset)) \in \mathcal{B}^{\mu}$, we get $((\emptyset, \emptyset), \emptyset, (\emptyset, \emptyset)) \in \widetilde{\mathcal{B}}$.
2. Suppose $(\widetilde{TC}, \widetilde{f}, \widetilde{TC'}) \in \widetilde{\mathcal{B}}$ for some $(TC, f, TC') \in \mathcal{B}^{\mu}$, which implies $(\mu C, f, \mu C') \in \mathcal{B}$.

(a) By Lemma 3 we have that $\widetilde{f} : r(TS)\lceil \widetilde{TC} \longrightarrow r(TS')\lceil \widetilde{TC'}$ is an isomorphism and $\delta_{\widetilde{TC}} = \delta_{\widetilde{TC'}} \circ \widetilde{f}$.

(b) Suppose $TC \longrightarrow \widetilde{TC_1}$ in $r(TS)$.

Let $\widetilde{TC_1}$ be a refinement of $TC_1 \in \mathcal{TC}(TS)$. Then, by Lemma 2(2), $\widetilde{TC}$ is a refinement of some $TC_2 \in \mathcal{TC}(TS)$ such that $TC_2 \longrightarrow TC_1$ in TS. It is clear that $TC_2 = (C, T_2)$. By the definition of a refinement of a timed configuration, if $T(e) \neq T_2(e)$ then $e \in busy(\widetilde{C})$, therefore e is maximal in C. Hence, $\delta_{TC_2} = \delta_{TC}$ by the definition of δ, which implies $T_2(e) \in \delta_{TC'}(f(e)) \subseteq D_{TS'}(f(e))$ for all $e \in C$. This means that $TC'_2 = (C', T_2 \circ f^{-1}) \in \mathcal{TC}(TS')$. Further, following the similar way, we get $\delta_{TC'_2} = \delta_{TC'}$ by the definition of δ, thus we get $\delta_{TC_2} = \delta_{TC'_2} \circ f$.

Since $\widetilde{TC}$ is a refinement of TC_2, by the construction of $\mathcal{B}$ and Lemma 3, we get that $\widetilde{TC'}$ is a refinement of TC'_2. Since

$(\mu C, f, \mu C') \in \mathcal{B}$, then $(TC_2, f, TC_2') \in \mathcal{B}^{\mu}$ by the construction of $\mathcal{B}^{\mu}$. Then, $TC_2' \longrightarrow TC_1'$ in TS' implies $(TC_1, f_1, TC_1') \in \mathcal{B}^{\mu}$ for some TC_1' and f_1 with $f \subseteq f_1$ and $\delta_{TC_1} = \delta_{TC_1'} \circ f_1$.

Applying Lemma 3, we get that $\widetilde{TC_1'} = (\widetilde{f_1}(\widetilde{C_1}), \widetilde{T_1} \circ \widetilde{f_1}^{-1})$ is a refinement of TC_1'. By the construction of $\widetilde{\mathcal{B}}$, we have $(\widetilde{TC_1}, \widetilde{f_1}, \widetilde{TC_1'}) \in \widetilde{\mathcal{B}}$. Since $\widetilde{TC} \longrightarrow \widetilde{TC_1}$ in $r(TS)$ and $\widetilde{f} \subseteq \widetilde{f_1}$, we get $\widetilde{TC'} \longrightarrow \widetilde{TC_1'}$ in $r(TS')$.

(c) Symmetric to item (b).

Thus, $r(TS) \approx_{\delta\text{-}hpb} r(TS')$. ■

6. Subclasses of timed event structures and action refinement

In this section, we introduce several subclasses of timed event structures constructed by some additional requirements on the timed constraints of events. We show that equivalences considered in Section 3, when restricted onto these subclasses, are invariant under refinement of timed event structures.

Definition 11. *Let TS be a timed event structure, then*

- *TS is said to have a* deterministic timing, *if for all TC, $TC' \in \mathcal{TC}(TS)$ and for all isomorphisms $f : TS\lceil TC \longrightarrow TS\lceil TC'$ it holds: $D_{TS}|_C = D_{TS}|_{C'} \circ f$,*
- *TS is called* discrete *if $\forall e \in E \diamond D(e) = [d, d]$ for some $d \in \mathbb{R}$;*
- *TS is called* segmentary *if $\forall e \in E \diamond D(e) = [n, n+1]$ for some $n \in \mathbb{N}$.*

By the above definition, discrete timed event structures have a deterministic timing, so further we will not distinguish discrete ones. The next theorem shows that δ-equivalences and the corresponding original equivalences coincide when dealing with the defined subclasses of timed event structures. For these purposes, we will need the following lemma.

Lemma 4. *Let TS and TS' be timed event structures having a deterministic timing such that $TS \approx_{trace} TS'$. Then for all $TC \in \mathcal{TC}(TS)$, for all $TC' \in \mathcal{TC}(TS')$ and for all isomorphisms $f : TS\lceil TC \longrightarrow TS'\lceil TC'$ it holds: $D_{TS}|_C = D_{TS'}|_{C'} \circ f$.*

Proof Sketch. Immediately follows from the mathematical fact that if $s \in Interv$ is a closed interval and s can be represented as a denumerable union of closed intervals $s = \bigcup_{i \in I \subseteq N} s_i$ such that $s_i \cap s_j = \emptyset$ for all $i \neq j$, then $I = \{i\}$ and $s = s_i$. ■

Theorem 5. *Let TS and TS' be timed event structures which are segmentary or have a deterministic timing. Then*

(i) $TS \approx_{trace} TS' \iff TS \approx_{\delta\cdot trace} TS'$,

(ii) $TS \approx_{test} TS' \iff TS \approx_{\delta\cdot test} TS'$,

(iii) $TS \approx_{hpb} TS' \iff TS \approx_{\delta\cdot hpb} TS'$.

Proof. In the case when TS and TS' have a deterministic timing, the proofs of all these items immediately follows from the definitions and Lemma 4.

Now consider the case when TS and TS' are segmentary timed event structures.

Further we will need an additional auxiliary notation. Let λC be the timed configuration of TS containing the configuration C and the medium timing function T, i.e. $\lambda C = (C, T)$, where $T(e) = \min D_{TS}(e) + \frac{1}{2}$ for all $e \in C$. This notation is correct, since TS have a valid timing and it is segmentary.

(i)

Assume $TS \approx_{trace} TS'$.

Suppose $(TP, \delta) \in L_\delta(TS)$. Then there exists $TC = (C, T) \in \mathcal{TC}(TS)$ and an isomorphism $f : TS\lceil TC \longrightarrow TP$ such that $\delta_{TC} = \delta \circ f$. Since $\lambda C \in \mathcal{TC}(TS)$, according to our assumption, there exist $TC' = (C', T') \in \mathcal{TC}(TS')$ and an isomorphism $f : TS\lceil \lambda C \longrightarrow TS'\lceil TC'$. By the definition of a segmentary timed event structure, we get $D_{TS}|_C = D_{TS'}|_{C'} \circ f$. Hence, $TC' = (C', T'') \in \mathcal{TC}(TS')$, where $T'' = T \circ f^{-1}$ with $\delta_{TC} = \delta_{TC''} \circ f$. Thus, $(TP, \delta) \in L_\delta(TS')$.

The arbitrary choice of $(TP, \delta) \in L_\delta(TS)$ guarantees $TS \approx_{\delta\cdot trace} TS'$.

(ii)

Assume $TS \approx_{test} TS'$.

Suppose TS *after* (TP, δ) *MUST* Q for some $TP \in \mathcal{TP}$, $\delta : E_{TP} \longrightarrow Interv$ and $Q \subset \mathcal{TP}$. We will show that TS' *after* (TP, δ) *MUST* Q.

In the case when $(TP, \delta) \notin L_\delta(TS')$, we immediately get TS' *after* (TP, δ) *MUST* Q. Further we consider the case when $(TP, \delta) \in L_\delta(TS')$.

Take an arbitrary $TC' = (C', T') \in \mathcal{TC}(TS')$ and an isomorphism $f : TS'\lceil TC' \longrightarrow TP$ such that $\delta_{TC'} = \delta \circ f$. We have to show that there exist $TP_1 \in Q$, $TC'_1 \in \mathcal{TC}(TS')$ and an isomorphism $f_1 : TS'\lceil TC'_1 \longrightarrow TP_1$ such that $f \subset f_1$.

For $\lambda C' = (C', \lambda T') \in \mathcal{TC}(TS')$, construct a timed poset

$$TP^\lambda = (E_{TP}, \leq_{TP}, l_{TP}, \lambda T' \circ f^{-1}).$$

Then $f : TS'\lceil \lambda C' \longrightarrow TP^\lambda$ is an isomorphism, therefore $TP^\lambda \in L(TS')$. By our assumption and Theorem 1, we have $TP^\lambda \in L(TS)$. Further we will form a test consisting of TP^λ and a set $Q^\lambda \subset \mathcal{TP}$ which we construct using Q as follows:

– since $TP^\lambda \in L(TS)$, take an arbitrary configuration C of TS and an isomorphism $g : TS\lceil\lambda C \longrightarrow TP^\lambda$. Then $\varphi = f^{-1} \circ g : TS\lceil\lambda C \longrightarrow TS'\lceil\lambda C'$ is an isomorphism, therefore $D_{TS}|_C = D_{TS'}|_{C'} \circ \varphi$ because TS and TS' are segmentary. Hence, $TC = (C, T' \circ \varphi) \in \mathcal{TC}(TS)$, $\varphi : TS\lceil TC \longrightarrow TS'\lceil TC'$ is an isomorphism and $\delta_{TC} = \delta_{TC'} \circ \varphi$, which implies $g : TS\lceil TC \longrightarrow TP$ is an isomorphism and $\delta_{TC} = \delta \circ g$. By our assumption, there exist $TP_1 \in Q$, $TC_1 = (C_1, T_1) \in \mathcal{TC}(TS)$ and an isomorphism $g_1 : TS\lceil TC_1 \longrightarrow TP_1$ such that $g \subset g_1$. We have $\lambda C_1 = (C_1, \lambda T_1) \in \mathcal{TC}(TS)$, because TS is segmentary. Construct a timed poset $TP_{C,g} = (E_{TP_1}, \leq_{TP_1}, l_{TP_1}, \lambda T_1 \circ g_1^{-1}) \in \mathcal{TP}$. By the construction of $TP_{C,g}$, we get $TP^\lambda \prec TP_{C,g}$ and $g_1 : TS\lceil\lambda C_1 \longrightarrow TP_{C,g}$ is an isomorphism.

Let a set Q^λ consist of all such $TP_{C,g}$. By the construction of Q^λ, we have TS *after* TP^λ *MUST* Q^λ. According to our assumption, we get TS' *after* TP^λ *MUST* Q^λ.

Recall that $f : TS'\lceil\lambda C' \longrightarrow TP^\lambda$ is an isomorphism, there exist a configuration C_1' of TS, $TP_{C,g} \in Q^\lambda$ and an isomorphism $f_1 : TS'\lceil\lambda C_1' \longrightarrow TP_{C,g}$ such that $f \subset f_1$. By the construction of $TP_{C,g}$, a mapping $\varphi_1 = f_1^{-1} \circ g_1 : TS\lceil\lambda C_1 \longrightarrow TS'\lceil\lambda C_1'$ is an isomorphism for some C_1 and $\varphi \subset \varphi 1$. Hence $D_{TS}|_{C_1} = D_{TS'}|_{C_1'} \circ \varphi_1$, because TS and TS' are segmentary. Therefore, $TC_1' = (C_1', T_1 \circ \varphi_1) \in \mathcal{TC}(TS)$ and $\delta_{TC_1} = \delta_{TC_1'} \circ \varphi_1$. Thus, $f_1 : TS'\lceil TC_1' \longrightarrow TP_1$ is an isomorphism and $f \subset f_1$, which we had to show.

The arbitrary choice of TP, δ and Q guarantees $TS \approx_{\delta\text{-}test} TS'$.

(iii)

Assume that $TS \approx_{hpb} TS'$ and $\mathcal{B}$ is a history preserving bisimulation between TS and TS'.

Construct a new relation $\mathcal{B}^\lambda$ as follows.

$$\mathcal{B}^\lambda = \{(TC, f, TC') \mid TC \in \mathcal{TC}(TS),\ TC' \in \mathcal{TC}(TS'),\ T = T' \circ f \text{ and } (\lambda C, f, \lambda C') \in \mathcal{B}\}.$$

By the construction of $\mathcal{B}^\lambda$, for all $(TC, f, TC') \in \mathcal{B}^\lambda$ it holds $D_{TS}|_C = D_{TS'}|_{C'} \circ f$, because TS and TS' are segmentary. We will show that $\mathcal{B}^\lambda$ is a δ-history preserving bisimulation between TS and TS'.

1. $((\emptyset, \emptyset), \emptyset, (\emptyset, \emptyset)) \in \mathcal{B}^\lambda$, since $(\lambda\emptyset, \emptyset, \lambda\emptyset) \in \mathcal{B}$.
2. Suppose $(TC, f, TC') \in \mathcal{B}^\lambda$.
   (a) Since f preserved the partial order and labels, and $T = T' \circ f$, we resume that f is an isomorphism between $TS\lceil TC$ and $TS'\lceil TC'$.

(b) Since $D_{TS}|_C = D_{TS'}|_{C'} \circ f$ and $T = T' \circ f$, we get $\delta_{TC} = \delta_{TC'} \circ f$.

(c) Suppose $TC \longrightarrow TC_1$ in TS. By the construction of $\mathcal{B}^\lambda$, we have $(\lambda C, f, \lambda C') \in \mathcal{B}$. Since $\lambda C \longrightarrow \lambda C_1$ in TS, there exist $\lambda C_1' \in \mathcal{TC}(TS')$ and an isomorphism $f_1 : TS\lceil\lambda C_1 \longrightarrow TS'\lceil\lambda C_1'$ such that $f \subseteq f_1$ with $(\lambda C_1, f_1, TC_1') \in \mathcal{B}$. Hence $D_{TS}|_{C_1} = D_{TS'}|_{C_1'} \circ f_1$ which implies $TC_1' = (C_1', T_1 \circ f_1^{-1}) \in \mathcal{TC}(TS')$. By the construction of $\mathcal{B}^\lambda$, we have $(TC_1, f_1, TC_1') \in \mathcal{B}^\lambda$. Since $C' \subseteq C_1'$ and $T' = T_1' \mid_{C'}$, we get $TC' \longrightarrow TC_1'$ in TS'.

(d) Symmetric to item (c).

Thus $TS \approx_{\delta\text{-}hpb} TS'$. ∎

As a corollary of the theorems, we resume that the trace, testing and history preserving equivalences of segmentary timed event structures or timed event structures with a deterministic timing are invariant under refinement.

Corollary 1. *Let TS and TS' be timed event structures which are segmentary or have a deterministic timing, r be a refinement function. Then*

(i) $TS \approx_{trace} TS' \;\Rightarrow\; r(TS) \approx_{trace} r(TS')$,

(ii) $TS \approx_{test} TS' \;\Rightarrow\; r(TS) \approx_{test} r(TS')$,

(iii) $TS \approx_{hpb} TS' \;\Rightarrow\; r(TS) \approx_{hpb} r(TS')$.

7. Conclusion

In this paper, we have studied an operator of action refinement and the interplay between the operator and the behavioural equivalences of the linear-time/branching-time spectrum in the framework of event structures with the dense time domain. We have shown that in general these equivalences are not preserved under refinement in contrast to their counterparts in the theory of event structures [13, 14]. As a compromise we have constructed certain equivalences strengthened by additional requirements sufficient for their preservation under refinement. Moreover, we have presented some subclasses of timed event structures on which these equivalences are preserved by refinement without any additional assumptions.

In the future, we plan to extend these results onto other classes of timed event structures (e.g., timed stable event structures, timed local event structures, etc.).

References

[1] Aceto L. History preserving, causal and mixed-ordering equivalence over stable event structures // Fundamenta Informaticae. — 1992. — Vol. 17(4). — P. 319–331.

[2] Aceto L., De Nicola R., Fantechi A. Testing Equivalences for Event Structures // Lect. Notes Comput. Sci. — 1987. — Vol. 280. — P. 1–20.

[3] Aceto L., Murphi D. Timing and causality in process algebra // Acta Informatica. — 1996. — Vol. 33 (4). — P. 317–350.

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

[5] Alur R., Henzinger T.A. Logics and models of real time: a survey // Lect. Notes Comput. Sci. — 1992. — Vol. 600. — P. 74–106.

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

[7] Andreeva M.V., Virbitskaite I.B. Timed Equivalences for Timed Event Structures // Lect. Notes Comput. Sci. — 2005. — Vol. 3606. — P. 16–26.

[8] Baier C., Katoen J.-P., Latella D. Metric semantics for true concurrent real time // Proc. 25th Internat. Colloquium, ICALP'98, Aalborg, Denmark, 1998. — P. 568–579.

[9] Čerāns K. Decidability of bisimulation equivalences for parallel timer processes // Lect. Notes Comput. Sci. — 1993. — Vol. 663. — P. 302–315.

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

[11] De Nicola R., Hennessy M. Testing Equiavalence for Processes // Theor. Comput. Sci. — 1984. — Vol. 34. — P. 83–133.

[12] 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.

[13] van Glabbeek R.J., Goltz U. Equivalence notions for concurrent systems and refinement of actions // Lect. Notes Comput. Sci. — 1989. — Vol. 379. — P. 237–248.

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

[15] Hennessy M., Milner R. Algebraic laws for nondeterminism and concurrency // J. of ACM. — 1985. — Vol. 32. — P. 137–162.

[16] Henzinger T.A., Manna Z., Pnueli A. Timed transition systems // Lect. Notes Comput. Sci. — 1991. — Vol. 600. — P. 226–251.

[17] Hoare C.A.R. Communicating sequential processes. — London: Prentice-Hall, 1985.

[18] 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.

[19] Maggiolo-Schettini A., Winkowski J. Towards an algebra for timed behaviours // Theor. Comput. Sci. — 1992. — Vol. 103. — P. 335–363.

[20] Milner R. Communication and concurrency. — Prentice-Hall, 1989.

[21] Murphy D. Time and duration in noninterleaving concurrency // Fundamenta Informaticae. — 1993. — Vol. 19. — P. 403–416.

[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] Virbitskaite I.B., Gribovskaya N.S. Open maps and observational equivalences for timed partial order models // Fundamenta Informaticae. — 2004. — Vol. 60(1-4). — P. 383–399.

[24] Winskel G. An introduction to event structures // Lect. Notes Comput. Sci. — 1988. — Vol. 354. — P. 364–397.

[25] Weise C., Lenzkes D. Efficient scaling-invariant checking of timed bisimulation // Lect. Notes Comput. Sci. — 1997. — Vol. 1200. — P. 176–188.