Decidability of Strong Bisimilarity for Timed BPP

Sławomir Lasota*

LSV, CNRS UMR 8643 & ENS Cachan, France
Institute of Informatics, Warsaw University, Poland
sl@mimuw.edu.pl

Abstract. We investigate a timed extension of the class of Basic Parallel Processes (BPP), in which actions are durational and urgent and parallel components have independent local clocks. The main result is decidability of strong bisimilarity, known also as performance equivalence, in this class. This extends the earlier decidability result for plain BPP [8] as well as decidability for timed BPP with strictly positive durations of actions [3]. Both ill-timed and well-timed semantics are treated. Our decision procedure is based on decidability of the validity problem for Presburger arithmetic.

1 Introduction

One important problem in verification of concurrent systems is to check whether two given systems P and Q are equivalent under a given notion of equivalence. For process algebras generating infinite-state systems the equivalence checking problem is not decidable in general, hence restricted classes of processes have been defined and investigated. We study here Basic Parallel Processes (BPP) [7], an extension of recursively defined finite-state systems by an operator of parallel composition. Strong bisimilarity [19] is a well accepted behavioural equivalence which often remains decidable for infinite-state systems. An elegant proof of decidability of bisimilarity for BPP was given in [8], while all other equivalences in van Glabbeek's spectrum are undecidable in BPP [15, 17].

In order to deal with timing aspects of systems, process algebras were extended with an appropriate notion of time (see eg. [20], [1], [14], [10], [9], [2]). A substantial effort have also been directed toward defining a robust notion of equivalence taking performance into consideration, e.g., equivalence that relates only those processes that exhibit the same behavior at the same speed. The bisimilarity investigated in [14, 10, 9] applies to discrete-time process algebras where parallel components have their own local clocks and actions are *durational* and *urgent*. Urgent actions take place as soon as possible and can only be delayed in order to wait for a communication with another process. The semantics allows for runs which are *ill-timed*, i.e. the observed time of an action need not to increase along a run, but *well-caused*, i.e., causally-related local clock values do increase along a run.

* This work has been supported by the fellowship of the Foundation for Polish Science, during the author's post-doc stay at Laboratoire Specification et Verification, ENS Cachan. Partially supported by the KBN grant 7 T11C 002 21 and by the Research Training Network *Games*.

Equivalence checking of finite-state timed processes attracted some attention [6, 11]. For instance, in [6] bisimilarity was proved decidable for finite timed transition systems. In contrast to this, there are not many results on verification of infinite-state timed systems. We can mention here [22], where the tableau-based technique of [8] was adapted to timed-arc BPP nets. One relevant result [3] is the decidability and surprisingly polynomial-time complexity of strong bisimilarity for timed BPP when durations of actions are strictly positive. Due to this latter restriction, process algebra does not subsume plain BPP. Only when the restriction is omitted, the process algebra obtained is a timed extension of BPP, since BPP embeds naturally into it as the case when all durations equal zero and no time delay is used in process definitions.

As the main result of this paper, we prove decidability of bisimilarity for the timed extension of BPP, called TBPP in the sequel. In fact, our formulation of the process algebra is simpler and more general than in [14, 10, 3], as durations are implicitly assigned to each rewrite rule rather than to each action. In particular, duration of an action can be chosen in a nondeterministic way.

Outline: After basic definitions in Section 2, we prove in Section 3 a series of decomposition results. They allow us to overcome in Section 4 the main difficulty: while the set of plain BPP processes, up to a structural congruence, forms a finitely-generated commutative semigroup, this is not the case in TBPP, since parallel components of a process can be arbitrarily spread out over time. Then Section 5 contains the decision procedure. We exploit decidability of the validity problem for Presburger arithmetic [21] in the way suggested by Jančar in [18] for untimed systems. In Section 6 we extend our decidability result to TBPP with communication and to TBPP under *well-timed* semantics. The last section contains a few remarks.

2 Definitions

Let $Const$ be a finite set of *process constants*, ranged over by X, Y, etc. The set of TBPP process expressions (processes) over $Const$, ranged over by P, Q, P_1, etc. is given by

$$P \ ::= \ \mathbf{0} \ | \ X \ | \ P \| P \ | \ 1 \triangleright P$$

where $\mathbf{0}$ denotes the *empty* process, $\_\|\_$ stands for a parallel composition and $1 \triangleright P$ denotes a process which behaves like P but delayed by one time unit. Processes not involving the delay operator $1 \triangleright \_$ are called *untimed* processes or BPP processes [7]. For $t \in \mathbb{N}$, a natural number, we write $t \triangleright P$ as a shorthand for $\underbrace{1 \triangleright (1 \triangleright (\ldots (1 \triangleright P) \ldots))}_{t \text{ times}}$.

We do not distinguish between processes related by a *structural congruence*, i.e., the smallest congruence induced by:

- associativity and commutativity of parallel composition,
- nilpotency and time cancellation for $\mathbf{0}$: $\mathbf{0} \| P = P$, $1 \triangleright \mathbf{0} = \mathbf{0}$,
- clock distribution law [10]: $1 \triangleright (P_1 \| P_2) = 1 \triangleright P_1 \| 1 \triangleright P_2$.

From now on, equality of processes in understood up to the structural congruence. This allows us to consider only processes in the following *timed normal form*:

$$t_1 \triangleright \alpha_1 \| t_2 \triangleright \alpha_2 \| \ \ldots \ \| t_n \triangleright \alpha_n \tag{1}$$

where $n \geq 0$, $0 \leq t_1 < t_2 < \ldots < t_n$ and $\alpha_1 \ldots \alpha_n$ are non-empty untimed processes, i.e., not involving the delay operator and $\mathbf{0}$. When $n = 0$, (1) denotes the empty process $\mathbf{0}$. We assume here and later that delay operator binds stronger than parallel composition.

Due to associativity, commutativity and nilpotency laws, each untimed process can be seen as a finite multiset of process constants, or equivalently as an element of the free commutative semigroup over $Const$. This observation was a key point in the decision procedure for strong bisimilarity of BPP [8]. Unfortunately, in TBPP we lose this finite-generation property: TBPP processes can be rather seen as elements of the free commutative semigroup over $\mathbb{N} \times Const$.

A TBPP process definition consists of a finite set $Const$ of process constants, a finite set Act of *actions* and a finite set of process rewrite rules of the form $X \xrightarrow{a} P$, where P is a process over $Const$ in timed normal form and $a \in Act$. We assume at least one rewrite rule for each $X \in Const$. Each process definition Δ induces a labeled transition system, whose states are processes, and whose transitions are labeled by pairs (t, a), with t a natural number and $a \in Act$. The transitions are derived from the following SOS rules (recall that _‖_ is commutative):

$$\frac{(X \xrightarrow{a} P) \in \Delta}{X \xrightarrow{0,a} P} \qquad \frac{P \xrightarrow{t,a} P'}{P \| Q \xrightarrow{t,a} P' \| Q} \qquad \frac{P \xrightarrow{t,a} P'}{1 \triangleright P \xrightarrow{t+1,a} 1 \triangleright P'} \tag{2}$$

A transition $P \xrightarrow{t,a} Q$ is to mean "P does a at time t and becomes Q". The transition system obtained is finitely-branching despite that it has infinite labeling set. According to the right-most rule, delay operator is preserved during the transition and thus is used to record the value of a clock associated to a process. Following [10],[14] and [3], we allow for *ill-timed runs*, i.e., we *do not* exclude $P \xrightarrow{t_1,a_1} P' \xrightarrow{t_2,a_2} P''$ when $t_1 > t_2$. This is due to the middle rule, which allows for a transition in one component independently from values of clocks in other components. It was argued convincingly in [1] that ill-timed semantics brings no semantical problems since the ill-timed runs are well-caused and on the other hand greatly simplifies the technical treatment. While we agree with the first point, in Section 6 we demonstrate that deciding strong bisimilarity is even easier for well-timed semantics.

If we restrict ourselves to BPP processes only, and if all right-hand sides of rewrite rules in Δ are BPP processes, each transition derived from rules (2) is performed at time $t = 0$. Hence the time becomes irrelevant and we essentially obtain a BPP process definition.

The rules (2) require some further comment. In [3] the transition system for TBPP was defined for an arbitrary *duration function* $f : Act \to \mathbb{N} \setminus \{0\}$, assigning a positive duration to each action. Consequently, instead of the first rule in (2) a rule

$$\frac{(X \xrightarrow{a} P) \in \Delta}{X \xrightarrow{0,a} f(a) \triangleright P}$$

was used, following [10, 14]. A rewrite rule $X \xrightarrow{a} P$ of [3] can be obviously expressed in our framework by the rule $X \xrightarrow{a} t \triangleright P$, where $t = f(a)$. On the other hand, our rules are more general in at least two respects: first, zero-durations are allowed; second, a duration is assigned to each rewrite rule separately rather then to each action (a rule $X \xrightarrow{a} t_1 \triangleright \alpha_1 \| \ldots \| t_n \triangleright \alpha_n$ is implicitly assigned duration $t_1 \geq 0$, in fact).

Given a process definition Δ, two processes P_1 and P_2 are *(strongly) bisimilar* w.r.t. Δ, denoted $P_1 \sim_\Delta P_2$ (or $P_1 \sim P_2$ when Δ is understood in the context), if they are related by some bisimulation R, that is a binary relation over processes such that whenever $(P, Q) \in R$, for each $a \in Act$ and $t \geq 0$,

– if $P \xrightarrow{t,a} P'$ then $Q \xrightarrow{t,a} Q'$ for some Q' such that $(P', Q') \in R$
– if $Q \xrightarrow{t,a} Q'$ then $P \xrightarrow{t,a} P'$ for some P' such that $(P', Q') \in R$.

In [3] and [10], such relations were called *performance relations* and *R-bisimulations*, respectively, and the induced equivalence was called *performance equivalence*. The equivalence was also called *timed bisimilarity*, eg. in [6]. We prefer to call it *bisimilarity* to avoid introducing unnecessary names and to emphasize that we generalize bisimilarity on BPP. In the following sections we show the decidability of the following problem:

Problem:	STRONG BISIMILARITY FOR TBPP
Instance:	A TBPP process definition Δ and two process constants X, Y
Question:	$X \sim_\Delta Y$?

We do not lose generality by only restricting to constants, since checking $P \sim Q$, for arbitrary P, Q is equivalent to checking $X_P \sim X_Q$, where X_P and X_Q are new fresh constants and their only rewrite rules are: $X_P \xrightarrow{a} P$ and $X_Q \xrightarrow{a} Q$. The rest of this section is devoted to an example illustrating the main difficulty of the problem.

Example 1. Assume that two processes $X \| 3 \triangleright (Y \| Z)$ and $(Y \| Y) \| 3 \triangleright (Z \| Z \| Z)$ are bisimilar. Decomposition results in Lemmas 1 and 2 in Section 3 imply that $3 \triangleright (Y \| Z)$ and $3 \triangleright (Z \| Z \| Z)$ are then bisimilar too. So we can substitute one for another and consider $X \| 3 \triangleright (Y \| Z)$ and $(Y \| Y) \| 3 \triangleright (Y \| Z)$, which are *almost identical* according to terminology from the next section, i.e., differ only at time 0. At first sight, this short deduction suggests that we are not very far from the bisimilarity checking problem for BPP, as we only need to consider BPP processes X and $Y \| Y$ in a context $\_ \| 3 \triangleright (Y \| Z)$, roughly. But this is not really true. Assume that Z has a rewrite rule $Z \xrightarrow{a} Y \| Z$. According to Lemma 5 in the next section, a transition in the context part can always be matched by identical transition in the context part, e.g., $X \| 3 \triangleright (Y \| Z) \xrightarrow{3,a} X \| 3 \triangleright (Y \| Y \| Z)$ is matched by $(Y \| Y) \| 3 \triangleright (Y \| Z) \xrightarrow{3,a} (Y \| Y) \| 3 \triangleright (Y \| Y \| Z)$. But nevertheless we must take into account an infinite number of contexts: $\_ \| 3 \triangleright (Y^i \| Z)$, for $i > 0$. Moreover, if Y has for instance a rule $Y \xrightarrow{b} Y \| 1 \triangleright Y$, contexts that are arbitrarily spread out through the time domain $\mathbb{N}$ must be considered. $\square$

In particular, all proof methods for decidability of bisimilarity for BPP work no more for TBPP, since they are based on the fact that the set of BPP processes is a finitely-generated commutative semigroup. This difficulty is overcome in Lemma 7 in Section 4, where we prove that one can restrict to a sufficiently large initial fragment of the time domain and ignore transitions which lead out of this fragment.

3 Decomposition and Almost-Identical Pairs

From now on we assume that Δ is an arbitrary fixed process definition and that all processes are in timed normal form (1). For such $P = t_1 \triangleright \alpha_1 \| t_2 \triangleright \alpha_2 \| \ \ldots \ \| t_n \triangleright \alpha_n$, which is *non-empty*, i.e., $n > 0$, we put:

- $\mathrm{head}(P) := t_1 \triangleright \alpha_1$,
- $\mathrm{tail}(P) := t_2 \triangleright \alpha_2 \| \ \ldots \ \| t_n \triangleright \alpha_n$,
- $\mathrm{minclock}(P) := t_1$,
- $\mathrm{now}(P) := \alpha_1 \| (t_2 - t_1) \triangleright \alpha_2 \| \ \ldots \ \| (t_n - t_1) \triangleright \alpha_n$.

Note that $\mathrm{head}(P)$, $\mathrm{tail}(P)$ and $\mathrm{now}(P)$ are in timed normal form again, $\mathrm{head}(\mathrm{now}(P))$ is untimed, $P = \mathrm{head}(P) \| \mathrm{tail}(P)$ and $P = \mathrm{minclock}(P) \triangleright \mathrm{now}(P)$.

Bisimilarity is not only compositional (i.e. a congruence) w.r.t. parallel composition and the delay operator but it is also partially decompositional:

Lemma 1 (Decomposition w.r.t. $1 \triangleright \_$**).** *For non-empty P and Q, $P \sim Q$ implies*

1. *minclock*$(P) =$ *minclock*(Q) *and*
2. *now*$(P) \sim$ *now*(Q).

Proof. 1. $\mathrm{minclock}(P) = \mathrm{minclock}(Q)$ as the earliest action of P must be matched in Q and vice versa. 2. The relation $\{(t \triangleright \mathrm{now}(P), t \triangleright \mathrm{now}(Q)) : t \geq 0,\ P \sim Q\}$ is a bisimulation. □

Lemma 2 (Partial decomposition w.r.t. $\_\|\_$**).** *For non-empty P and Q, $P \sim Q$ implies tail*$(P) \sim$ *tail*(Q).

Proof. The relation $\{(\mathrm{tail}(P), \mathrm{tail}(Q)) : P \sim Q,\ \ P$ and Q are non-empty$\}$ is a bisimulation, since any action in $\mathrm{tail}(P)$ at time $t > \mathrm{minclock}(P) = \mathrm{minclock}(Q)$ must be matched in $\mathrm{tail}(Q)$. □

Example 2. Strong bisimilarity is not fully decompositional w.r.t. $\_\|\_$, $P \sim Q$ does not imply $\mathrm{head}(P) \sim \mathrm{head}(Q)$. As a counterexample, consider a process definition:

$$A \xrightarrow{a} 1 \triangleright B \qquad A' \xrightarrow{a} 1 \triangleright C \qquad B \xrightarrow{b} B \qquad C \xrightarrow{c} C$$

We have $A \| 1 \triangleright (B \| C) \sim A' \| 1 \triangleright (B \| C)$ but $A \not\sim A'$. □

By Lemmas 1 and 2, if two processes are bisimilar, say $t_1 \triangleright \alpha_1 \| \ \ldots \ \| t_n \triangleright \alpha_n$ and $u_1 \triangleright \beta_1 \| \ \ldots \ \| u_n \triangleright \beta_m$, then necessarily they involve the same time stages, that is to say $n = m$ and $t_i = u_i$, for $i \leq n$.

Assumption. In Sections 3, 4 and 5, whenever we consider a pair (P, Q) of processes, we silently assume that $P = t_1 \triangleright \alpha_1 \| \ldots \| t_n \triangleright \alpha_n$ and $Q = t_1 \triangleright \beta_1 \| \ldots \| t_n \triangleright \beta_n$, for some $n \geq 0$, $0 \leq t_1 < t_2 < \ldots < t_n$ and non-empty untimed processes $\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n$. We omit routine verification that all pairs mentioned throughout the following sections are of this form.

Lemma 2 suggests to study pairs of processes in a particularly simple form:

Definition 1. (almost-identical pairs) *Processes P and Q are called* almost-identical *if $\mathit{minclock}(P) = \mathit{minclock}(Q) = 0$ and $\mathit{tail}(P) = \mathit{tail}(Q)$.*

We will apply Lemma 2 to decompose a pair of processes into a number of almost-identical pairs. As a consequence, we will be able to ignore all pairs which are not almost-identical (Lemma 3 below). Before the formal definition of decomposition, we give an example to explain the idea.

Example 3. Consider a pair:

$$\begin{array}{rcrcrcrcr} P & = & 2 \triangleright (X \| Y) & \| & 7 \triangleright (X \| X \| Y) & \| & 10 \triangleright (Z \| Z) & \| & 16 \triangleright Y \\ Q & = & 2 \triangleright (X \| X \| Z) & \| & 7 \triangleright (X \| Z) & \| & 10 \triangleright (X \| X) & \| & 16 \triangleright Y \end{array}$$

(P, Q) can be decomposed into four pairs (P_i, Q_i), $i \leq 4$. The rule is as follows: $\mathrm{head}(P_i)$ and $\mathrm{head}(Q_i)$ is the i-th component of P and Q, respectively, while $\mathrm{tail}(P_i) = \mathrm{tail}(Q_i)$ contains the components of P from all later time stages.

$$\begin{array}{rcrcrcrcr} P_1 & = & (X \| Y) & \| & 5 \triangleright (X \| X \| Y) & \| & 8 \triangleright (Z \| Z) & \| & 14 \triangleright Y \\ Q_1 & = & (X \| X \| Z) & \| & 5 \triangleright (X \| X \| Y) & \| & 8 \triangleright (Z \| Z) & \| & 14 \triangleright Y \\ P_2 & = & & & (X \| X \| Y) & \| & 3 \triangleright (Z \| Z) & \| & 9 \triangleright Y \\ Q_2 & = & & & (X \| Z) & \| & 3 \triangleright (Z \| Z) & \| & 9 \triangleright Y \\ P_3 & = & & & & & (Z \| Z) & \| & 6 \triangleright Y \\ Q_3 & = & & & & & (X \| X) & \| & 6 \triangleright Y \\ P_4 & = & & & & & & & Y \\ Q_4 & = & & & & & & & Y. \end{array}$$

It can be shown using Lemmas 1 and 2 that $P \sim Q \iff \forall i \leq 4. P_i \sim Q_i$. The last pair is identical and hence can be ignored, therefore to show $P \sim Q$ it is sufficient to consider only the first three: $P \sim Q \iff \forall i \leq 3. P_i \sim Q_i$. The formal definition follows. $\square$

Definition 2. (decomposition set) *A* decomposition set $\mathbb{DS}(P, Q)$ *of a pair (P, Q) is defined recursively by (for brevity, $\bar{P}$ stands for $\mathit{now}(P)$ and $\bar{Q}$ stands for $\mathit{now}(Q)$)*

$$\mathbb{DS}(P, Q) := \begin{cases} \{(\bar{P}, \mathit{head}(\bar{Q}) \| \mathit{tail}(\bar{P}))\} \cup \mathbb{DS}(\mathit{tail}(\bar{P}), \mathit{tail}(\bar{Q})) & \text{if } P \neq Q \\ \emptyset & \text{if } P = Q. \end{cases}$$

Note that all pairs in $\mathbb{DS}(P, Q)$ are almost-identical by definition.

Lemma 3. $P \sim Q$ *iff* $\mathbb{DS}(P,Q) \subseteq\ \sim$.

Proof. By induction on the size of P, Q. As for $P = Q$ we are immediately done, assume that P, Q are non-identical, hence non-empty and that the lemma holds for a pair $(\text{tail}(\text{now}(P)), \text{tail}(\text{now}(Q)))$, i.e.,

$$\text{tail}(\text{now}(P)) \sim \text{tail}(\text{now}(Q)) \iff \mathbb{DS}(\text{tail}(\text{now}(P)), \text{tail}(\text{now}(Q))) \subseteq\ \sim . \quad (3)$$

IF-IMPLICATION: If $\mathbb{DS}(P,Q) \subseteq\ \sim$, then also $\mathbb{DS}(\text{tail}(\text{now}(P)), \text{tail}(\text{now}(Q))) \subseteq\ \sim$, as a subset of $\mathbb{DS}(P,Q)$. Hence, by induction assumption (3),

$$\text{tail}(\text{now}(P)) \sim \text{tail}(\text{now}(Q)). \quad (4)$$

Moreover, from $\mathbb{DS}(P,Q) \subseteq\ \sim$ we derive

$$\text{now}(P) \sim \text{head}(\text{now}(Q)) \| \text{tail}(\text{now}(P)), \quad (5)$$

since this pair is in $\mathbb{DS}(P,Q)$. By (4) and (5) and the congruence property of $\sim$ we conclude $\text{now}(P) \sim \text{now}(Q)$ hence $P \sim Q$.
ONLY-IF-IMPLICATION: Assume $P \sim Q$. By Lemma 1 and 2, (4) holds again. By (3), $\mathbb{DS}(\text{tail}(\text{now}(P)), \text{tail}(\text{now}(Q))) \subseteq\ \sim$. Now, we only need to show (5), which follows by (4) and by $\sim$ being a congruence:
$\text{now}(P) \sim\ \text{now}(Q) = \text{head}(\text{now}(Q)) \| \text{tail}(\text{now}(Q)) \sim\ \text{head}(\text{now}(Q)) \| \text{tail}(\text{now}(P))$. $\square$

In Definitions 3, 5 and 6 below we introduce a series of computationally more and more tractable notions of bisimulation.

Definition 3. (decompositional bisimulation) *A binary relation R over processes is a* decompositional bisimulation *if whenever $(P,Q) \in R$, for each $a \in Act$ and $t \geq 0$,*

- *if $P \xrightarrow{t,a} P'$ then $Q \xrightarrow{t,a} Q'$ for some Q' such that $\mathbb{DS}(P',Q') \subseteq R$*
- *if $Q \xrightarrow{t,a} Q'$ then $P \xrightarrow{t,a} P'$ for some P' such that $\mathbb{DS}(P',Q') \subseteq R$.*

Lemma 4. $P \sim Q$ *iff* P *and* Q *are related by a decompositional bisimulation.*

Before the proof, we need a definition of a *bisimulation-base*. It is an adaptation of the well-know notion due to Caucal [4, 5, 7].

Definition 4. *For a binary relation R, let $\overset{R}{\equiv}$ denote the smallest congruence w.r.t. parallel composition and delay operator containing R. We call R a* bisimulation-base *if whenever $(P,Q) \in R$, for each $a \in Act$ and $t \geq 0$,*

- *if $P \xrightarrow{t,a} P'$ then $Q \xrightarrow{t,a} Q'$ for some Q' such that $(P',Q') \in \overset{R}{\equiv}$*
- *if $Q \xrightarrow{t,a} Q'$ then $P \xrightarrow{t,a} P'$ for some P' such that $(P',Q') \in \overset{R}{\equiv}$.*

Proof of Lemma 4. By Lemma 3, $\sim$ is a decompositional bisimulation. Hence we only need to show the IF-direction. This is shown in two steps:

1. each decompositional bisimulation is a bisimulation-base,

2. if R is a bisimulation-base, then $\stackrel{R}{\equiv}$ is a bisimulation.

Point 1. is immediate, since each pair (P', Q') belongs to the congruence generated by its decomposition set, $(P', Q') \in \stackrel{\mathbb{DS}(P',Q')}{\equiv}$. The proof of 2. amounts to showing that the two clauses in the definition of bisimulation hold for $\stackrel{R}{\equiv}$. This can be done by a straightforward induction on the depth of inference of $P \stackrel{R}{\equiv} Q$. $\square$

Bisimilarity is closed under identical transitions performed in identical parts of almost-identical processes:

Lemma 5. *Whenever P and Q are almost-identical, $P \sim Q$ and $\mathit{tail}(P) \xrightarrow{t,a} P'$ then $\mathit{head}(P) \| P' \sim \mathit{head}(Q) \| P'$.*

Proof. Note that, by assumption, t is necessarily strictly positive. Thus the answer of Q to the transition $P \xrightarrow{t,a} \mathsf{head}(P) \| P'$ is performed in $\mathsf{tail}(Q)$, since the time t must be matched exactly. Hence for some Q', $\mathsf{tail}(Q) \xrightarrow{t,a} Q'$ and $\mathsf{head}(P) \| P' \sim \mathsf{head}(Q) \| Q'$. By Lemma 2 and the congruence property of $\sim$, $\mathsf{head}(Q) \| Q' \sim \mathsf{head}(Q) \| P'$, which completes the proof. $\square$

To make our notation more succinct, we introduce a closure operator on binary relations over processes as follows: let $R^{\rightarrow}$ be the smallest relation containing R such that whenever an almost-identical pair (P, Q) is in $R^{\rightarrow}$ and $\mathsf{tail}(P) \xrightarrow{t,a} P'$ then $(\mathsf{head}(P) \| P', \mathsf{head}(Q) \| P') \in R^{\rightarrow}$. Lemma 5 can be then shortly rephrased by $\sim^{\rightarrow} \subseteq \sim$.

Due to Lemma 3, it is sufficient to consider almost-identical pairs. Lemma 5 allows us to refine the notion of decompositional bisimulation for almost-identical pairs in Definition 5 below and to strengthen Lemma 4 in Lemma 6 below.

Definition 5. *A binary relation R over processes is an* almost-identical decompositional bisimulation *if it only contains almost-identical pairs and*

1. whenever $(P, Q) \in R$, for each $a \in Act$,
   - *if $P \xrightarrow{0,a} P'$ then $Q \xrightarrow{0,a} Q'$ for some Q' such that $\mathbb{DS}(P', Q') \subseteq R$*
   - *if $Q \xrightarrow{0,a} Q'$ then $P \xrightarrow{0,a} P'$ for some P' such that $\mathbb{DS}(P', Q') \subseteq R$*

2. $R^{\rightarrow} \subseteq R$.

Roughly, the Defender's invention in the Bisimulation Game is restricted only to moves at time 0, since all transitions performed at time $t > 0$ are matched identically.

Lemma 6. *Let P and Q be almost-identical. $P \sim Q$ iff P and Q are related by an almost-identical decompositional bisimulation.*

Proof. By Lemma 5, $\sim$ restricted to only almost-identical pairs is an almost-identical decompositional bisimulation. On the other hand, each almost-identical decompositional bisimulation is obviously a decompositional bisimulation. $\square$

4 Bounded Bisimulations

Having elaborated an appropriate notion of bisimulation, we are ready to face the main difficulty of the problem. It turns out that we can restrict ourselves to only a finite initial fragment of the time domain $\mathbb{N}$, without losing the completeness of almost-identical decompositional bisimulations w.r.t. $\sim$ as stated in Lemma 6. The restriction is imposed by a "horizon" $2{\cdot}H$, where H is defined as

$$H := \max\{\text{maxclock}(P) : X \xrightarrow{a} P \in \Delta\}$$

and $\text{maxclock}(P)$ of a non-empty process P in timed normal form (1) is to denote t_n and $\text{maxclock}(\mathbf{0}) = 0$. Intuitively, H is the maximal *scope* of a single transition: in effect of a transition performed at time t, say, only process components at time between t and $t{+}H$ can be modified.

Definition 6. *A binary relation R over processes is a* bounded almost-identical decompositional bisimulation *(*bounded bisimulation *in short) if it only contains almost-identical pairs (P, Q) with $\mathit{maxclock}(P) \leq 2{\cdot}H$ and $\mathit{maxclock}(Q) \leq 2{\cdot}H$, and whenever $(P, Q) \in R$, for each $a \in Act$,*

1. – *if $P \xrightarrow{0,a} P'$ then $Q \xrightarrow{0,a} Q'$ for some Q' such that $\mathbb{DS}(P', Q') \subseteq R$*
   – *if $Q \xrightarrow{0,a} Q'$ then $P \xrightarrow{0,a} P'$ for some P' such that $\mathbb{DS}(P', Q') \subseteq R$*
2. *for $0 < t \leq H$, if $\mathit{tail}(P) \xrightarrow{t,a} P'$ then $(\mathit{head}(P)\|P', \mathit{head}(Q)\|P') \in R$.*

Having restricted in Lemma 6 non-identical responses of the Defender to transitions at $t{=}0$, we restrict ourselves further and take into consideration only transitions at time $t{\leq}H$. Intuitively, H is chosen sufficiently large to guarantee that the additional restriction has no impact on the relevant part of the Bisimulation Game played at time 0. If one reminds that the scope of transitions performed at time 0 does not go beyond $0 \ldots H$, then all transitions (and the identically-matching responses) that are performed at time $t > 0$ belonging to that scope are clearly taken into consideration. This guarantees that each bounded bisimulation can be extended to an almost-identical decompositional bisimulation, by closing under identically-matched transitions performed at time $t{>}H$.

Lemma 7. *Let P and Q be almost-identical with $\mathit{maxclock}(P) \leq 2{\cdot}H$. Then $P \sim Q$ iff P and Q are related by a bounded bisimulation.*

Proof. ONLY-IF-IMPLICATION: $\sim$ restricted to only those almost-identical pairs (P, Q) with $\text{maxclock}(P) \leq 2{\cdot}H$ is a bounded bisimulation.
IF-IMPLICATION: Relying on Lemma 6, we will show that if R is a bounded bisimulation then $R^{\rightarrow}$ is an almost-identical decompositional bisimulation. Point 2. in Definition 5 is immediate, i.e., $(R^{\rightarrow})^{\rightarrow} \subseteq R^{\rightarrow}$. For the proof of Point 1. we will need the following notation. For $P = t_1{\triangleright}\alpha_1\| \ \ldots \ \|t_n{\triangleright}\alpha_n$ and $t \geq 0$, let

$$\text{head}_t(P) = t_1{\triangleright}\alpha_1\| \ \ldots \ \|t_i{\triangleright}\alpha_i,$$
$$\text{tail}_t(P) = t_{i+1}{\triangleright}\alpha_{i+1}\| \ \ldots \ \|t_n{\triangleright}\alpha_n,$$

where i is chosen so that $t_j <= t$ for all $1 \leq j \leq i$ and $t_j > t$ for all $i + 1 \leq j \leq n$. In particular $\text{head}(P) = \text{head}_0(P)$ and $\text{tail}(P) = \text{tail}_0(P)$, when $\text{minclock}(P) = 0$.

We need to show Point 1. in Definition 5 only for pairs in $R^{\rightarrow} \setminus R$. Let (P, Q) be any pair in $R^{\rightarrow} \setminus R$. So there exists a pair $(P_0, Q_0) \in R$ such that P is obtained from P_0 and Q is obtained from Q_0 by a finite sequence of transitions performed in $\mathsf{tail}(P_0) = \mathsf{tail}(Q_0)$. W.l.o.g. we can assume that the transitions performed at time less or equal to H precede in the sequence those performed at time strictly greater than H. Thus we can further assume that all transitions in the sequence are performed at time greater than H, since all earlier transitions performed at time less or equal H do not lead out of R. To be more precise, there is $(P_0, Q_0) \in R$ such that

$$\mathsf{head}_H(P_0) = \mathsf{head}_H(P), \qquad\qquad \mathsf{head}_H(Q_0) = \mathsf{head}_H(Q), \tag{6}$$

and there exists $m > 0$, $t_1, \ldots, t_m > H$ and $a_1, \ldots, a_m \in Act$ such that

$$\mathsf{tail}_H(P_0) \xrightarrow{t_1, a_1} \ldots \xrightarrow{t_m, a_m} \mathsf{tail}_H(P). \tag{7}$$

Recall that both (P_0, Q_0) and (P, Q) are almost-identical pairs, hence

$$Q_0 = \mathsf{head}_H(Q_0) \| \mathsf{tail}_H(P_0) \qquad \text{and} \qquad Q = \mathsf{head}_H(Q) \| \mathsf{tail}_H(P).$$

Now we will show only the first clause in Point 1. – the second clause follows by the identical reasoning. Here is the simple idea underlying the pedantic analysis to follow: all transitions performed at time 0 by P or Q have its scope inside $\mathsf{head}_H(P)$ or $\mathsf{head}_H(Q)$, respectively, hence by (6) the matching of a transition of P_0 by Q_0 can be re-used for P and Q. Assume that P has a transition labelled by $(0, a)$. This means that $\mathsf{head}_H(P) \xrightarrow{0, a} P'$ for some P' with $\mathsf{maxclock}(P') \leq H$ and $P \xrightarrow{0, a} P' \| \mathsf{tail}_H(P)$. Hence $\mathsf{head}_H(P) \| \mathsf{tail}_H(P_0) = P_0 \xrightarrow{0, a} P' \| \mathsf{tail}_H(P_0)$.

$$\begin{tikzcd}
\overbrace{\mathsf{head}_H(P) \| \mathsf{tail}_H(P_0)}^{P_0} \arrow[r, "t_1, a_1"] \arrow[d, "0, a"] & \cdots \arrow[r, "t_m, a_m"] & \overbrace{\mathsf{head}_H(P) \| \mathsf{tail}_H(P)}^{P} \arrow[d, "0, a"] \\
P' \| \mathsf{tail}_H(P_0) \arrow[r, "t_1, a_1"] & \cdots \arrow[r, "t_m, a_m"] & P' \| \mathsf{tail}_H(P)
\end{tikzcd}$$

$(P_0, Q_0) \in R$ and R is a bounded bisimulation, hence there exists a Q' such that $\mathsf{maxclock}(Q') \leq H$, $\mathsf{head}_H(Q_0) \xrightarrow{0, a} Q'$ and

$$\mathbb{DS}(P' \| \mathsf{tail}_H(P_0), Q' \| \mathsf{tail}_H(P_0)) \subseteq R. \tag{8}$$

This means that $Q \xrightarrow{0, a} Q' \| \mathsf{tail}_H(P)$ and we have the following picture again:

$$\begin{tikzcd}
Q' \| \mathsf{tail}_H(P_0) \arrow[r, "t_1, a_1"] & \cdots \arrow[r, "t_m, a_m"] & Q' \| \mathsf{tail}_H(P) \\
\underbrace{\mathsf{head}_H(Q) \| \mathsf{tail}_H(P_0)}_{Q_0} \arrow[u, "0, a"] \arrow[r, "t_1, a_1"] & \cdots \arrow[r, "t_m, a_m"] & \underbrace{\mathsf{head}_H(Q) \| \mathsf{tail}_H(P)}_{Q} \arrow[u, "0, a"]
\end{tikzcd}$$

We only need to show that

$$\mathbb{DS}(P' \| \text{tail}_H(P), Q' \| \text{tail}_H(P)) \subseteq R^{\rightarrow}.$$

Consider any pair $(\bar{P}, \bar{Q}) \in \mathbb{DS}(P' \| \text{tail}_H(P), Q' \| \text{tail}_H(P))$. By (8) it is sufficient to show that there exists a pair $(\bar{P_0}, \bar{Q_0}) \in \mathbb{DS}(P' \| \text{tail}_H(P_0), Q' \| \text{tail}_H(P_0))$ such that

$$\text{tail}(\bar{P_0}) = \text{tail}(\bar{Q_0}) \xrightarrow{\bar{t_1}, \bar{a_1}} \ldots \xrightarrow{\bar{t_m}, \bar{a_m}} \text{tail}(\bar{P}) = \text{tail}(\bar{Q}), \tag{9}$$

for some sequences $\bar{t_1}, \ldots, \bar{t_m} > 0$, $\bar{a_1}, \ldots, \bar{a_m} \in Act$. Let $(\text{tail}_H(P))^{-u}$ denote process $\text{tail}_H(P)$ in which all time prefixes are decreased by u; analogously $(\text{tail}_H(P_0))^{-u}$. It is crucial now to observe that by the very definition of $\mathbb{DS}(\_,\_)$ we have

$$\bar{P} = \bar{P}' \| (\text{tail}_H(P))^{-u} \quad \text{and} \quad \bar{Q} = \bar{Q}' \| (\text{tail}_H(P))^{-u},$$

for some u such that $0 \leq u \leq H$ and some $(\bar{P}', \bar{Q}') \in \mathbb{DS}(P', Q')$, Moreover, if we put:

$$\bar{P_0} := \bar{P}' \| (\text{tail}_H(P_0))^{-u} \quad \text{and} \quad \bar{Q_0} := \bar{Q}' \| (\text{tail}_H(P_0))^{-u},$$

then we have $(\bar{P_0}, \bar{Q_0}) \in \mathbb{DS}(P' \| \text{tail}_H(P_0), Q' \| \text{tail}_H(P_0))$, by the very definition of $\mathbb{DS}(\_,\_)$ again. Now if we put $\bar{t_i} := t_i - u$ and $\bar{a_i} := a_i$, by (7) it is routine to check that (9) holds. $\square$

5 Decision Procedure

Our algorithm is composed of two semi-decision procedures, analogously as in the case of BPP. The TBPP processes are *finitely branching*, hence bisimulation inequivalence $\not\sim$ is semi-decidable. The semi-decision procedure is essentially the same as for BPP [4] and is based on the fact that $\sim$ is the intersection of a countable chain of decidable approximations. Furthermore, our development in Section 4 enables us to prove also semi-decidability of $\sim$ essentially in the same way as it was done for BPP. The semi-decision procedure for $\sim$ proposed below is motivated by the idea of Jančar [18] to use Presburger arithmetic.

For BPP, the crucial observation was that BPP processes can be seen as elements of the free commutative semigroup over $Const$ with the semigroup operation $\_\|\_$, isomorphic to $\mathbb{N}^{Const}$ with vector addition $\_+\_$. The TBPP processes can be rather seen as elements of $\mathbb{N}^{\mathbb{N} \times Const}$. But due to Lemma 7, we can essentially restrict ourselves to only processes P with $\text{maxclock}(P) \leq 2{\cdot}H$, i.e., to $\mathbb{N}^{\{0,\ldots,2 \cdot H\} \times Const}$. This allows us to apply Theorem 1 below.

Definition 7. *A* linear *subset of* $\mathbb{N}^n$, $n > 0$, *is a set of the form*

$$v + span\{w_1, \ldots, w_k\} = \{v + n_1 w_i + \ldots + n_k w_k : n_1, \ldots, n_k \in \mathbb{N}\}.$$

I.e., each linear set is determined by a base $v \in \mathbb{N}^n$ *and* periods $\{w_1, \ldots, w_k\} \subset \mathbb{N}^n$. *A* semi-linear *set is a finite union of linear sets.*

Every binary relation over $\mathbb{N}^n$, i.e., a subset of $\mathbb{N}^n \times \mathbb{N}^n$, can be seen naturally as a subset of $\mathbb{N}^{2\cdot n}$. In the sequel we silently identify $\mathbb{N}^n \times \mathbb{N}^n$ and $\mathbb{N}^{2\cdot n}$.

Theorem 1 ([12, 16, 18]). *Each congruence in* $(\mathbb{N}^n, \_+\_)$*,* $n > 0$*, is semi-linear.*

By Theorem 1, $\sim$ restricted to the almost-identical pairs (P, Q) with $\mathrm{maxclock}(P) \leq 2{\cdot}H$ is semi-linear, as it is obviously a congruence w.r.t. $\_\|\_$. Hence we refine Lemma 7 as follows:

Corollary 1. *Let P and Q be almost-identical with $\mathrm{maxclock}(P) \leq 2{\cdot}H$. Then $P \sim Q$ iff P and Q are related by a semi-linear bounded bisimulation.*

Lemma 8. *For a semi-linear binary relation R over $\mathbb{N}^{\{0,\ldots,2\cdot H\}\times Const}$, given by a finite set of base-periods pairs, it is decidable whether R is a bounded bisimulation.*

Proof. Given R, we can effectively construct a closed formula in Presburger arithmetic ϕ_R such that ϕ_R is valid iff R is a bounded bisimulation. The validity problem for Presburger arithmetic is decidable (see for instance [21]), hence this gives a decision procedure required.

Let $Const = \{X_1, \ldots, X_N\}$. The formula ϕ_R involves several tuples of variables of the form $\{x_{t,i}\}_{0\leq t\leq 2\cdot H, 1\leq i\leq N}$, denoted in short $\bar{x}$. A valuation of each such tuple corresponds to a TBPP process.

The structure of ϕ_R follows directly the two points in Definition 6 (we omit the second clause in the first point of Definition 6, which is symmetric to the first clause):

$$\forall \bar{x}, \bar{y}.\ (\bar{x}, \bar{y}) \in R \implies \ (\bar{x}, \bar{y}) \text{ is an almost-identical pair } \ \wedge$$
$$\big(\wedge_{r_x\in\Delta}\ \forall \bar{x'}.\ \bar{x} \xrightarrow{0,r_x} \bar{x'} \implies \exists \bar{y'}.\ \vee_{r_y\in\Delta \text{ labelled as } r_x} (\bar{y} \xrightarrow{0,r_y} \bar{y'} \wedge \mathbb{DS}(\bar{x'}, \bar{y'}) \subseteq R)\big) \ \wedge$$
$$\big(\wedge_{r\in\Delta}\ \wedge_{1\leq t\leq H} \forall \bar{x'}.\ \bar{x} \xrightarrow{t,r} \bar{x'} \implies \exists \bar{y'}.\ (\bar{y} \xrightarrow{t,r} \bar{y'} \wedge (\bar{x'}, \bar{y'}) \in R)\big)$$

We argue that all ingredients of ϕ_R are expressible in Presburger arithmetic. First, it is well-known that semi-linear sets are expressible in Presburger arithmetic [13]. Hence there exists a formula to denote "$(\bar{x}, \bar{y}) \in R$", with free variables $\bar{x}, \bar{y}$. Further, "$(\bar{x}, \bar{y})$ is an almost-identical pair" is easily expressible by a conjunction of $2{\cdot}H \times N$ equations. "$\mathbb{DS}(\bar{x'}, \bar{y'}) \subseteq R$" is expressible by a conjunction of $2{\cdot}H + 1$ formulas of the form "$\bar{x} \neq \bar{y} \Rightarrow (\bar{x}, \bar{y}) \in R$". Finally, let

$$r = (X_i \xrightarrow{a} 0{\triangleright}P_0\|\ldots\|H{\triangleright}P_H) \in \Delta,$$

for some $1 \leq i \leq N$, where $P_u = X_1^{p_{u,1}}\|\ldots\|X_N^{p_{u,N}}$, $p_{u,j} \geq 0$ for $0 \leq u \leq H$ and $1 \leq j \leq N$. Now "$\bar{x} \xrightarrow{t,r} \bar{x'}$", $0 \leq t \leq H$, is a shorthand for

$$x_{t,i} > 0 \ \wedge \ x_{t,i} + p_{0,i} = x'_{t,i} + 1 \ \wedge$$
$$\wedge_{1\leq j\leq N,\ 0\leq u\leq H,\ (u,j)\neq(0,i)}\ x'_{t+u,j} = x_{t+u,j} + p_{u,j}.$$

□

Corollary 1 and Lemma 8 form the core of the semi-decidability proof. The semi-decision procedure for $\sim$ consists of enumerating base-periods representations of all the semi-linear binary relations over $\mathbb{N}^{\{0,\ldots,2\cdot H\}\times Const}$ and checking whether any of them is a bounded bisimulation. Hence we have proved:

Theorem 2. *Strong bisimilarity is decidable for TBPP.*

6 Extensions

The proposed method of bisimilarity checking is quite robust and can be easily extended and adapted to other frameworks. Below we briefly sketch the proof of decidability of $\sim$ when communication is allowed and when well-timed semantics is considered instead of ill-timed one.

Communication. Assume that Act contains a distinguished silent action τ. Moreover, assume that for each $a \neq \tau$, there exists a complementary action $\bar{a} \in Act$ such that $\bar{\bar{a}} = a$. Two different synchronization rules have been proposed for timed processes, for $a \neq \tau$:

$$\frac{P \xrightarrow{t_1,a} P' \quad Q \xrightarrow{t_2,\bar{a}} Q' \quad t = \max\{t_1, t_2\}}{P \| Q \xrightarrow{t,\tau} P' \| Q'} \quad (10) \qquad \frac{P \xrightarrow{t,a} P' \quad Q \xrightarrow{t,\bar{a}} Q'}{P \| Q \xrightarrow{t,\tau} P' \| Q'} \quad (11)$$

According to the rule (10), proposed in [14], the busy waiting is allowed, when one of components is able to execute an action before the other. The second rule (11), studied in [1], prevents from any waiting and two processes can synchronize only if they are ready to perform complementary actions at the same time (see [10] for a detailed discussion).

We consider the second rule first. It is routine to verify that all the facts proved in Sections 3 and 4 are still valid; in particular, $\sim$ is still a congruence. Moreover, Lemma 8 from Section 5 still holds, since the Presburger formula constructed in the proof can be easily adapted for τ-moves arising from communication. Hence the decision procedure is exactly the same as in Section 5 and we conclude:

Theorem 3. *Strong bisimilarity is decidable for TBPP with rule (11).*

When the rule (10) is chosen, even Lemma 2 from Section 3 does not hold and it is not clear how to adapt our proof for this case.

Well-timed semantics. In well-timed semantics a transition $P \xrightarrow{t,a} P'$ is allowed exclusively at time $t = \text{minclock}(P)$. We lose now all the decomposition properties from Section 3 except from Lemma 1. Hence we can consider only pairs (P, Q) with

$$\text{minclock}(P) = 0 = \text{minclock}(Q).$$

We put additionally $\text{minclock}(\mathbf{0}) = 0$, so that the empty process is not ruled out. Two such P, Q are bisimilar iff they are related by some 0-*bisimulation*, that is a binary relation R over processes such that whenever $(P, Q) \in R$, for each $a \in Act$,

- if $P \xrightarrow{0,a} P'$ then $Q \xrightarrow{0,a} Q'$ for some Q' such that $(\text{now}(P'), \text{now}(Q')) \in R$
- if $Q \xrightarrow{0,a} Q'$ then $P \xrightarrow{0,a} P'$ for some P' such that $(\text{now}(P'), \text{now}(Q')) \in R$.

Moreover, due to the bound H on $\text{maxclock}(\_)$ of all right-hand sides of rewrite rules, there always exists a 0-bisimulation containing only pairs (P, Q) with $\text{maxclock}(P) \leq H$ and $\text{maxclock}(Q) \leq H$, called a *bounded* 0-*bisimulation* below (note that $\text{maxclock}(P)$ needs not be equal to $\text{maxclock}(Q)$, even when P and Q are bisimilar). Strong bisimilarity is a congruence again, hence by Theorem 1 we derive:

Lemma 9. *Assume* $\mathit{maxclock}(P) = 0 = \mathit{maxclock}(Q)$. $P \sim Q$ *iff* P *and* Q *are related by a semi-linear bounded* 0*-bisimulation.*

Lemma 10. *For a semi-linear binary relation* R *over* $\mathbb{N}^{\{0,\ldots,H\}\times Const}$*, given by a finite set of base-periods pairs, it is decidable whether* R *is a bounded* 0*-bisimulation.*

Proof. Similarly as in Lemma 8, given R, we can effectively construct a formula in Presburger arithmetic ϕ_R such that ϕ_R is valid iff R is a bounded 0-bisimulation. □

Similarly as in Section 5, Lemmas 9 and 10 form a core of the semi-decision procedure: it consists of enumerating (base-periods representations of) all the semi-linear binary relations over $\mathbb{N}^{\{0,\ldots,H\}\times Const}$ and checking whether any of them is a bounded 0-bisimulation. Hence we have proved:

Theorem 4. *Strong bisimilarity is decidable for TBPP with rule (11), under well-timed semantics.*

7 Final Remarks

The main result of our paper is decidability of strong bisimilarity, also known as performance equivalence, for timed BPP. The decision procedure is based on decidability of validity problem for Presburger arithmetic. Nevertheless we believe that after the crucial development of Sections 3 and 4, also other proof methods known for plain BPP could be used here. For instance, the decision procedure could be also based on searching for a finite bisimulation base [4] or for a successful tableau [8, 7]. In the latter case, we suppose that the method proposed in [22], being a generalization of the tableau method of [8], could be further generalized to capture our setting.

Exact complexity of the problem rests still unknown, similarly as in the case of BPP. Since our problem subsumes bisimilarity checking for BPP, the recent PSPACE lower bound of Srba [23] applies as well.

Following many papers in the area, we considered only processes in *normal form*, i.e., without action prefix and summation. While each BPP process can be effectively transformed into normal form [7], this is not the case for timed BPP. However, one more advantage of our new formulation is that the minor extension of the process algebra satisfies the *expansion law* [19], in contrast to the process algebra as considered in [14, 10, 9, 3]. Therefore we hope that only a slight effort is needed to extend our result to the language with action prefix and summation.

Acknowledgments The author is very grateful to Philippe Schnoebelen for many fruitful discussions and to anonymous reviewers for constructive suggestions helpful in improving the presentation.

References

1. L. Aceto and D. Murphy. Timing and causality in process algebra. *Acta Informatica*, 33(4):317–350, 1996.

2. J. Baeten and C. Middelburg. Process algebra with timing: real time and discrete time. In *J. Bergstra, A. Ponse, S. Smolka, eds., Handbook of Process Algebra, chapter 10*, pages 627–684, 2001.
3. B. Bérard, A. Labroue, and P. Schnoebelen. Verifying performance equivalence for timed Basic Parallel Processes. In *Proc. 3rd Int. Conf. Foundations of Software Science and Computation Structures (FOSSACS'2000), LNCS 1784*, pages 35–47, 2000.
4. O. Burkart, D. Caucal, F. Moller, and B. Steffen. Verification of infinite structures. In *J. Bergstra, A. Ponse, S. Smolka, eds., Handbook of Process Algebra, chapter 9*, pages 545–623, 2001.
5. D. Caucal. Graphes canoniques des graphes algebraiques. *Informatique Theoretique et Applications (RAIRO)*, 24(4):339–352, 1990.
6. K. Čerāns. Decidability of bisimulation equivalence for parallel timer processes. In *Proc. CAV'92, LNCS 663*, 1992.
7. S. Christensen. *Decidability and Decomposition in process algebras*. PhD thesis, Dept. of Computer Science, University of Edinbourg, UK, 1993. PhD Thesis CST-105-93.
8. S. Christensen, Y. Hirshfeld, and F. Moller. Bisimulation equivalence is decidable for Basic Parallel Processes. In *Proc. 4th Int. Conf. Concurrency Theory (CONCUR'93), LNCS 713*, pages 143–157, 1993.
9. F. Corradini. On performance congruences for process algebras. *Information and Computation*, 145(2):191–230, 1998.
10. F. Corradini, R. Gorrieri, and M. Roccetti. Performance preorder and competitive equivalence. *Acta Informatica*, 34(11):805–835, 1997.
11. F. Corradini and M. Pistore. Specification and verification of lazy timed systems. In *Proc. 21th Int. Symp. math. Found. Comp. Sci. (MFCS'96), LNCS 1113*, pages 279–290, 1996.
12. S. Eilenberg and M. Schuetzenberger. Rational sets in commutative monoids. *J. of Algebra*, 13:173–191, 1969.
13. S. Ginsburg and E. Spanier. Semigroups, Presburger formulas, and languages. *Pacific J. of Mathematics*, 16(2):285–296, 1966.
14. R. Gorrieri, M. Roccetti, and E. Stancampiano. A theory of processes with durational actions. *Theoretical Computer Science*, 140(1):73–94, 1995.
15. Y. Hirshfeld. Petri nets and the equivalence problem. In *Proc. Computer Science Logic (CSL'93), LNCS 832*, pages 165–174, 1993.
16. Y. Hirshfeld. Congruences in commutative semigroups. LFCS report ECS-LFCS-94-291, Laboratory for Foundations of Computer Science, University of Edinbourg, 1994.
17. H. Hüttel. Undecidable equivalences for basic parallel processes. In *Proc. TACS'94, LNCS 789*, pages 454–464, 1994.
18. P. Jančar. Decidability questions for bisimilarity of Petri nets and some related problems. In *Proc. 11th International Symposium on Theoretical Aspects of Computer Science (STACS'94), Caen, France, LNCS 775*, pages 581–592, 1994.
19. R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.
20. F. Moller and C. Tofts. Relating processes with respect to speed. In *Proc. 2nd Int Conf. Concurrency Theory (CONCUR'91), LNCS 527*, pages 424–438, 1991.
21. D. Oppen. A $2^{2^{2^{pn}}}$ upper bound on the complexity of Presburger arithmetic. *J. of Comp. and System Sciences*, 16:323–332, 1978.
22. J. Srba. Note on the tableau technique for commutative transition systems. In *Proc. 5th Foundations of Software Science and Computation Structures (FOSSACS'02), LNCS 2303*, pages 387–401, 2002.
23. J. Srba. Strong bisimilarity and regularity of Basic Parallel Processes is PSPACE-hard. In *Proc. 19th International Symposium on Theoretical Aspects of Computer Science (STACS'02), LNCS 2285*, 2002.