Team bisimilarity, and its associated modal logic, for BPP nets

Published Version:
Gorrieri R. (2021). Team bisimilarity, and its associated modal logic, for BPP nets. ACTA INFORMATICA, 58(5), 529-569 [10.1007/s00236-020-00377-4].

Availability:
This version is available at: https://hdl.handle.net/11585/831107 since: 2021-09-03

Published:

DOI: http://doi.org/10.1007/s00236-020-00377-4

Gorrieri, R. Team bisimilarity, and its associated modal logic, for BPP nets. ***Acta Informatica*** **58, 529–569 (2021)**

The final published version is available online at: ***https://doi.org/10.1007/s00236-020-00377-4***

Acta Informatica manuscript No.
(will be inserted by the editor)

Team Bisimilarity, and its Associated Modal Logic, for BPP Nets

Roberto Gorrieri

Received: 21 June 2019 / Revised: 7 January 2020 / Accepted: date

Abstract BPP nets, a subclass of finite Place/Transition Petri nets, are equipped with an efficiently decidable, truly concurrent, bisimulation-based, behavioral equivalence, called *team bisimilarity*. This equivalence is a very intuitive extension of classic bisimulation equivalence (over labeled transition systems) to BPP nets and it is checked in a distributed manner, without necessarily building a global model of the overall behavior of the marked BPP net. An associated distributed modal logic, called *team modal logic* (TML, for short), is presented and shown to be coherent with team bisimilarity: two markings are team bisimilar if and only if they satisfy the same TML formulae. As the process algebra BPP (with guarded summation and guarded body of constants) is expressive enough to represent all and only the BPP nets, we provide algebraic laws for team bisimilarity as well as a finite, sound and complete, axiomatization.

Keywords Equivalence checking · bisimulation equivalence · Petri nets · BPP · Hennessy-Milner modal logic · axiomatization.

1 Introduction

A BPP net is a simple type of finite Place/Transition Petri net [52,56] whose transitions have singleton pre-set. Nonetheless, as a transition can produce more tokens than the only one consumed, the reachable markings of a BPP net can be infinitely many. BPP is the acronym of *Basic Parallel Processes* [16], a simple CCS [47,31] subcalculus (without the restriction operator) whose processes cannot communicate. In [32] a variant of BPP, which requires guarded summation (as in Simple BPP [20], SBPP [25] or BPP_g [16]) and also that the body of each process constant is guarded (i.e., guarded recursion; cf. Section 5.1), is actually shown to represent *all and only*

Dipartimento di Informatica — Scienza e Ingegneria
Università di Bologna,
Mura A. Zamboni, 7, 40127 Bologna, Italy
E-mail: roberto.gorrieri@unibo.it

the BPP nets, up to net isomorphism, and this explains the name of this class of nets. Hence, we can uniformly compare results achieved on BPP nets or on the BPP subcalculus with guarded summation and guarded recursion. For uniformity with [32], in the following the BPP variant we are using is often simply called BPP, while the original version in [16] is called full BPP; moreover, full BPP with guarded summation is called SBPP [25]. While these three calculi have the same expressive power w.r.t. interleaving semantics, in the true-concurrency world full BPP is strictly more expressive than SBPP [23–25], while SBPP and our BPP are equally expressive as it is possible to adapt the net semantics for SBPP in [20] to generate all (and only) the BPP nets as well.

Traditionally, bisimulation-based, behavioral equivalences over Petri nets have been defined as relations over the set of reachable markings (see, e.g., [53,7,32]); these relations are all undecidable [21,39]. However, the situation is strikingly different for BPP nets: although the set of reachable markings can be countably infinite, essentially all the behavioral bisimulation-based equivalences proposed in the literature are decidable for this class of nets. For instance, deciding if two markings of a BPP net are *interleaving bisimilar* (see Definition 4) is PSPACE-complete [40] (w.r.t. the size of the net, where the *size of a BPP net* is the sum of the number of its places and of its transitions).

Our goal is to define a bisimulation-based, behavioral semantics for BPP nets directly on the finite set of places, rather than on the, possibly infinite, set of reachable markings. In fact, BPP nets have an important property: since each transition has a singleton pre-set, the behavior of each token on the net is completely independent of any other token in the net. This observation suggests that the global behavior of a marking m can be decomposed into a multiset of independent local behaviors, corresponding to the behaviors of each single token in m, such that the global behavior of m can be reconstructed from such a collection.

This approach 'by decomposition' allows to perform equivalence checking in a distributed manner; in fact, the problem of checking whether two markings m_1 and m_2 are behaviorally related can be reduced to that of checking whether their token-based local behaviors can be bijectively related. Note that to check whether two markings are equivalent we need not construct an LTS describing the global behavior of the whole system, but only find a suitable, behavior-preserving match among the local, sequential states (i.e., the elements of the markings). In this way, the state-space explosion problem can be tackled efficiently for BPP nets.

Therefore, in this approach, equivalent markings must have the same size. We think that it is important to define equivalences which relate markings of the same size only, because a token in a place of a BPP net represents a sequential process, so that one processor is needed to implement it. So, the marking gives a precise information about the number of resources/processors that are needed to implement the system. Hence, an equivalence relation which relates markings of the same size only is *resource aware*, and so more useful from a practical point of view.

In a recent paper [33], we approached our problem of decomposing the net behavior for the simplest subclass of Petri nets, namely *finite-state machines* (FSMs, for short), a class of nets whose transitions not only have singleton pre-set (as for BPP nets), but also have singleton (or empty) post-set. Therefore, FSMs are very similar to finite-state, labeled transition systems (LTSs, for short) [42]. On this class of models, there is widespread agreement that a very natural and convenient equivalence relation is bisimulation equivalence [51,47], an equivalence relation that can be verified very efficiently for finite-state LTSs; more precisley, if m is the number of transitions and n is the number of states of the LTS, checking whether two states are bisimilar can be done in $O(m \cdot \log n)$ time [54]. As an FSM is so similar to an LTS, we have defined bisimulation equivalence directly over the set of places of the *unmarked* net, as a place in an FSM represents a strictly sequential process type (while the number of tokens in that place represents the number of currently available instances of that sequential process type). The advantage is that bisimulation equivalence is a relation on places, rather than on markings, and so much more easily computable; more precisely, if m are the net transitions and n are the places, checking whether two places of an FSM are bisimilar can be done in $O(m \cdot \log (n+1))$ time, by adapting the algorithm in [54]. Moreover, the resulting notion of bisimilarity enjoys the same properties of bisimulation over LTSs, i.e., it is coinductive and equipped with a fixpoint characterization [47,57,31]. After the bisimulation equivalence over the set of places has been computed once and for all, we have defined, in a purely structural way, that two markings m_1 and m_2 are *team equivalent* if they have the same size, say $|m_1| = k = |m_2|$, and there is a bisimulation-preserving, bijective mapping between the two markings, so that each of the k pairs of places (s_1, s_2), with $s_1 \in m_1$ and $s_2 \in m_2$, is such that s_1 and s_2 are bisimilar. We proved that team equivalent markings respect interleaving bisimilarity, and so team equivalence preserves the token game; actually, we proved that it coincides with *strong place bisimilarity* [4,5], a truly-concurrent equivalence respecting the causal global behavior of nets; hence, as a corollary, we have that if two nets are team equivalent, then they have the same *causal nets* [6,50].

In this paper, we extend the results above to the class of BPP nets. The extension is not obvious because of the more general form of net transitions: a BBP net transition consumes one token but produces a multiset of tokens. Therefore, we cannot use the simple definition of bisimulation over the places of an FSM net, rather we need to generalize it to *team bisimulation*: if two places s_1 and s_2 are related by a team bisimulation R, then if s_1 performs a and reaches the marking m_1, then s_2 may perform a reaching a marking m_2 such that m_1 and m_2 are element-wise, bijectively related by R (and vice versa if s_2 moves first). We show that team bisimilarity enjoys the same properties of bisimulation over LTSs, i.e., it is coinductive and equipped with a fixpoint characterization. Moreover, we argue that the optimal algorithm for bisimulation equivalence over LTSs [54] can be adapted to compute team bisimilarity in $O(m \cdot p^2 \cdot \log (n+1))$ time, where m is the number of net transitions, p is the size of the largest post-set (i.e., p is the least natural such that $|t^\bullet| \leq p$ for all t) and n is the number of places.

Team bisimulation equivalence can be extended to markings by *additive closure*: if place s_1 is team bisimilar to place s_2 and the marking m_1 is team bisimilar to m_2 (the base case relates the empty marking to itself), then also $s_1 \oplus m_1$ is team bisimilar to $s_2 \oplus m_2$, where $\_ \oplus \_$ is the operator of multiset union. Note that if we need to check whether other two markings of the same net, say m_1' and m_2', are team equivalent, we can reuse the already computed team bisimulation equivalence over places, and so such a verification will take only $O(k^2)$ time, if k is the size of m_1' and m_2'. Of course, we prove that team bisimilar markings respect the global behavior; in particular, the token game (actually, we prove that team bisimilarity implies interleaving bisimilarity) and the causal behavior (actually, we will prove that team bisimilarity coincides with strong place bisimilarity [4,5]).

The second part of the paper approaches the problem of finding a modal characterization of team bisimulation equivalence, in line of what Hennessy and Milner proved in [37] for standard bisimulation equivalence over LTSs. The modal logic we propose, called *Team Modal Logic* (TML, for short), is indeed a proper extension of (a variant of) Hennessy-Milner Logic [37,3] (HML, for short), with an additional operator of parallel composition $\_ \otimes \_$ of formulae. The semantics of TML formulae is given w.r.t. a BPP net; in particular, the semantics of a formula is the set of the net markings that satisfy it. We prove a *coherence theorem*: two markings are team bisimilar if and only if they satisfy the same TML formulae.

The third part of the paper is concerned with the BPP process algebra (with guarded summation and guarded constants). We recall from [32] the denotational net semantics for this calculus and, moreover, the so-called *representability theorem*: not only any BPP process term is given a BPP net semantics, but also for any BPP net N we can single out a BPP process term p such that the net semantics for p is a net isomorphic to N. Then, we prove that team bisimilarity is a congruence for the operators of the BPP process algebra, we study the algebraic properties of team bisimilarity and, finally, we provide a finite, sound and complete, axiomatization.

The paper is organized as follows. Section 2 introduces the basic definitions about BPP nets and two behavioral equivalences: interleaving bisimilarity and strong place bisimilarity [4,5]; the latter is quite interesting, as we will prove that team bisimilarity coincides with strong place bisimilarity for BPP nets. Section 3 copes with the distributed equivalence checking problem; first, we discuss the properties of the additive closure of a relation on places; then team bisimulation over places of an unmarked BPP net is defined, showing that the classic results of bisimulation over LTSs also hold in this case; in particular, we provide a fixed point characterization for it. Moreover, team bisimilarity is extended to markings by additive closure and a few examples discussing its pros and cons are presented; finally, minimization of BPP net w.r.t. team bisimilarity is defined. Section 4 describes TML, its syntax and semantics, and shows the coherence theorem: two markings are team bisimilar if and only if they satisfy the same set of TML formulae. Section 5 describes the BPP process algebra and its net semantics. Section 6 shows the proof that team bisimilarity is a congruence for the BPP operators, presents its algebraic properties and also a finite, sound and complete, axiomatization. Finally, Section 7 discusses related literature, some future research and open problems.

2 Basic Definitions

Definition 1 (Multiset) Let $\mathbb{N}$ be the set of natural numbers. Given a finite set S, a *multiset* over S is a function $m : S \to \mathbb{N}$. The *support* set $dom(m)$ of m is $\{s \in S \mid m(s) \neq 0\}$. The set of all multisets over S, denoted by $\mathscr{M}(S)$, is ranged over by m. We write $s \in m$ if $m(s) > 0$. The *multiplicity* of s in m is given by the number

$m(s)$. The *size* of m, denoted by $|m|$, is the number $\sum_{s \in S} m(s)$. A multiset m such that $dom(m) = \emptyset$ is called *empty* and is denoted by θ. We write $m \subseteq m'$ if $m(s) \leq m'(s)$ for all $s \in S$.

Multiset union $\_ \oplus \_$ is defined as follows: $(m \oplus m')(s) = m(s) + m'(s)$; the operation $\oplus$ is commutative, associative and has θ as neutral element. *Multiset difference* $\_ \ominus \_$ is defined as follows: $(m_1 \ominus m_2)(s) = max\{m_1(s) - m_2(s), 0\}$. The *scalar product* of a number j with m is the multiset $j \cdot m$ defined as $(j \cdot m)(s) = j \cdot (m(s))$.

By s_i we also denote the multiset with s_i as only element. Hence, a multiset m over $S = \{s_1, \dots, s_n\}$ can be represented as $k_1 \cdot s_1 \oplus k_2 \cdot s_2 \oplus \dots \oplus k_n \cdot s_n$, where $k_j = m(s_j) \geq 0$ for $j = 1, \dots, n$. $\square$

Definition 2 (BPP net) A labeled *BPP net* is a tuple $N = (S, A, T)$, where

- S is the finite set of *places*, ranged over by s (possibly indexed),
- A is the finite set of *labels*, ranged over by ℓ (possibly indexed), and
- $T \subseteq S \times A \times \mathscr{M}(S)$ is the finite set of *transitions*, ranged over by t (possibly indexed).

  Given a transition $t = (s, \ell, m)$, we use the notation:

- ${}^\bullet t$ to denote its *pre-set* s (which is a single place) of tokens to be consumed;
- $l(t)$ for its *label* ℓ, and
- $t^\bullet$ to denote its *post-set* m (which is a multiset, possibly even empty) of tokens to be produced.

Hence, transition t can be also represented as ${}^\bullet t \xrightarrow{l(t)} t^\bullet$. $\square$

Graphically, a place is represented by a little circle, a transition by a little box, which is connected by a directed arc from the place in its pre-set and to the places in its post-set (if any); the out-going arcs may be labeled with a number to denote the number of tokens produced by the transition (if the number is omitted, then the default value is 1).

Definition 3 (Marking, BPP net system, firing sequence, reachable place, dynamically reduced) A multiset over S is called a *marking*. Given a marking m and a place s, we say that the place s contains $m(s)$ *tokens*, graphically represented by $m(s)$ bullets inside place s. A *BPP net system* $N(m_0)$ is a tuple (S, A, T, m_0), where (S, A, T) is a BPP net and m_0 is a marking over S, called the *initial marking*. We also say that $N(m_0)$ is a *marked* net.

A transition t is *enabled* at marking m, denoted by $m[t\rangle$, if ${}^\bullet t \subseteq m$. The execution (or *firing*) of t enabled at m produces the marking $m' = (m \ominus {}^\bullet t) \oplus t^\bullet$. This is written $m[t\rangle m'$. This procedure is called the *token game*.

A *firing sequence* starting at m is defined inductively as follows:

- $m[\varepsilon\rangle m$ is a firing sequence (where ε denotes the empty sequence of transitions) and
- if $m[\sigma\rangle m'$ is a firing sequence and $m'[t\rangle m''$, then $m[\sigma t\rangle m''$ is a firing sequence.

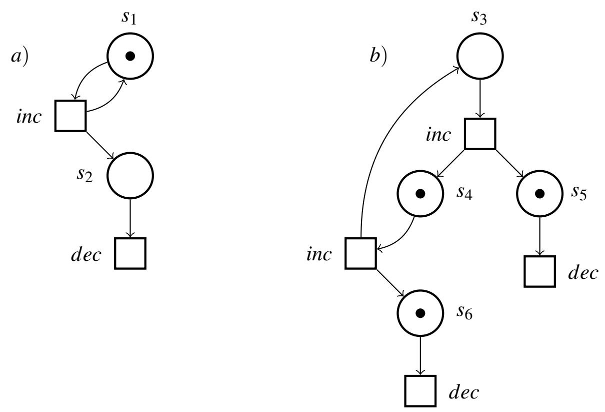

Fig. 1 The net representing a semi-counter in (a), and a variant in (b)

If $\sigma = t_1 \dots t_n$ (for $n \geq 0$) and $m[\sigma\rangle m'$ is a firing sequence, then there exist $m_1, \dots, m_{n+1}$ such that $m = m_1[t_1\rangle m_2[t_2\rangle \dots m_n[t_n\rangle m_{n+1} = m'$, and $\sigma = t_1 \dots t_n$ is called a *transition sequence* starting at m and ending at m'. The set of *reachable markings* from m is

$[m\rangle = \{m' \mid \exists \sigma. m[\sigma\rangle m'\}$.

Note that the reachable markings can be countably infinite. The set of *reachable places* from s is

$reach(s) = \bigcup_{m \in [s\rangle} dom(m)$.

Note that $reach(s)$ is always a finite set, even if $[s\rangle$ is infinite. A BPP net system $N(m_0) = (S, A, T, m_0)$ is *dynamically reduced* if $\forall s \in S\, \exists m \in [m_0\rangle . m(s) \geq 1$ and, moreover, $\forall t \in T\, \exists m, m' \in [m_0\rangle$ such that $m[t\rangle m'$. $\square$

Example 1 By using the drawing convention for Petri nets mentioned above, Figure 1 shows in (a) the simplest BPP net representing a semi-counter, i.e., a counter which cannot test for zero. Note that the number represented by this semi-counter is given by the number of tokens which are present in place s_2, i.e., in the place ready to perform *dec*; hence, Figure 1(a) represents a semi-counter holding number 0; note also that the number of tokens which can be accumulated in s_2 is unbounded. Indeed, the set of reachable markings for a BPP net can be countably infinite. In (b), a variant semi-counter is outlined, which holds number 2 (i.e., two tokens are ready to perform action *dec*). $\square$

Now we recall two well-known behavioral equivalences over Petri nets: interleaving bisimilarity and strong place bisimilarity.

Definition 4 (Interleaving Bisimulation) Let $N = (S, A, T)$ be a BPP net. An *interleaving bisimulation* is a relation $R \subseteq \mathscr{M}(S) \times \mathscr{M}(S)$ such that if $(m_1, m_2) \in R$ then

- $\forall t_1$ such that $m_1[t_1\rangle m_1'$, $\exists t_2$ such that $m_2[t_2\rangle m_2'$ with $l(t_1) = l(t_2)$ and $(m_1', m_2') \in R$,
- $\forall t_2$ such that $m_2[t_2\rangle m_2'$, $\exists t_1$ such that $m_1[t_1\rangle m_1'$ with $l(t_1) = l(t_2)$ and $(m_1', m_2') \in R$.

Two markings m_1 and m_2 are *interleaving bisimilar* (or *interleaving bisimulation equivalent*), denoted by $m_1 \sim_{int} m_2$, if there exists an interleaving bisimulation R such that $(m_1, m_2) \in R$. $\square$

Relation $\sim_{int}$, which is defined as the union of all the interleaving bisimulations, is the largest interleaving bisimulation and also an equivalence relation.

Remark 1 **(Interleaving bisimulation between two nets)** The definition above covers also the case of an interleaving bisimulation between two BPP nets, say, $N_1 = (S_1, A, T_1)$ and $N_2 = (S_2, A, T_2)$ with $S_1 \cap S_2 = \emptyset$, because we may consider just one single BPP net $N = (S_1 \cup S_2, A, T_1 \cup T_2)$: An interleaving bisimulation $R \subseteq \mathscr{M}(S_1) \times \mathscr{M}(S_2)$ is also an interleaving bisimulation on $\mathscr{M}(S_1 \cup S_2) \times \mathscr{M}(S_1 \cup S_2)$. Similar considerations hold for all the bisimulation-like definitions we propose in the following, which will be defined on a single net only. $\square$

Remark 2 **(Comparing two marked nets)** The definition above of interleaving bisimulation is defined over an *unmarked* BPP net, i.e., a net without the specification of an initial marking m_0. Of course, if one desires to compare two marked nets, then it is enough to find an interleaving bisimulation (over the union of the two nets, as discussed in the previous remark), containing the pair composed of the respective initial markings. This approach is also followed for the other bisimulation-like definitions we propose in the following. $\square$

Example 2 Continuing Example 1 about Figure 1, it is easy to realize that $R = \{(s_1 \oplus k \cdot s_2, s_3 \oplus k_1 \cdot s_5 \oplus k_2 \cdot s_6) \bigm| k = k_1 + k_2 \text{ and } k, k_1, k_2 \geq 0\} \cup \{(s_1 \oplus k \cdot s_2, s_4 \oplus k_1 \cdot s_5 \oplus k_2 \cdot s_6) \bigm| k = k_1 + k_2 \text{ and } k, k_1, k_2 \geq 0\}$ is an interleaving bisimulation. $\square$

We now introduce strong place bisimulation equivalence, introduced in [4,5] as an improvement of *strong bisimilarity*, a behavioral relation proposed by Olderog in [50] on safe nets which fails to be an equivalence relation. Our definition is formulated in a slightly different way, but it is coherent with the original one. First, an auxiliary definition, which will be further investigated in the next section.

Definition 5 (Additive closure) Given a BPP net $N = (S, A, T)$ and a *place relation* $R \subseteq S \times S$, we define a *marking relation* $R^{\oplus} \subseteq \mathscr{M}(S) \times \mathscr{M}(S)$, called the *additive closure* of R, as the least relation induced by the following axiom and rule.

$$\frac{}{(\theta, \theta) \in R^{\oplus}} \qquad \frac{(s_1, s_2) \in R \quad (m_1, m_2) \in R^{\oplus}}{(s_1 \oplus m_1, s_2 \oplus m_2) \in R^{\oplus}}$$

$\square$

Note that, by definition, two markings are related by $R^{\oplus}$ only if they have the same size; in fact, the axiom states that the empty marking is related to itself, while the rule, assuming by induction that m_1 and m_2 have the same size, ensures that $s_1 \oplus m_1$ and $s_2 \oplus m_2$ have the same size. Note also that there may be several proofs of $(m_1, m_2) \in R^{\oplus}$, depending on the chosen order of the elements of the two markings and on the definition of R. For instance, if $R = \{(s_1, s_3), (s_1, s_4), (s_2, s_3), (s_2, s_4)\}$, then $(s_1 \oplus s_2, s_3 \oplus s_4) \in R^{\oplus}$ can be proved by means of the pairs (s_1, s_3) and (s_2, s_4), as well as by means of $(s_1, s_4), (s_2, s_3)$. An alternative way to define that two markings m_1 and m_2 are related by $R^{\oplus}$ is to state that m_1 can be represented as $s_1 \oplus s_2 \oplus \ldots \oplus s_k$, m_2 can be represented as $s'_1 \oplus s'_2 \oplus \ldots \oplus s'_k$ and $(s_i, s'_i) \in R$ for $i = 1, \ldots, k$.

Definition 6 (Strong Place Bisimulation) Let $N = (S, A, T)$ be a BPP net. A *strong place bisimulation* is a relation $R \subseteq S \times S$ such that if $(m_1, m_2) \in R^{\oplus}$ then

- $\forall t_1$ such that $m_1[t_1\rangle m_1'$, $\exists t_2$ such that $m_2[t_2\rangle m_2'$ with $(^{\bullet}t_1, {}^{\bullet}t_2) \in R$, $l(t_1) = l(t_2)$, $(t_1^{\bullet}, t_2^{\bullet}) \in R^{\oplus}$ and $(m_1', m_2') \in R^{\oplus}$,
- $\forall t_2$ such that $m_2[t_2\rangle m_2'$, $\exists t_1$ such that $m_1[t_1\rangle m_1'$ with $(^{\bullet}t_1, {}^{\bullet}t_2) \in R$, $l(t_1) = l(t_2)$, $(t_1^{\bullet}, t_2^{\bullet}) \in R^{\oplus}$ and $(m_1', m_2') \in R^{\oplus}$.

Two markings m_1 and m_2 are *strong place bisimilar*, denoted by $m_1 \sim_p m_2$, if there exists a strong place bisimulation R such that $(m_1, m_2) \in R^{\oplus}$. □

Relation $\sim_p$ is an equivalence relation [4]. Its definition, however, is not completely coinductive, as the union of strong place bisimulations may be not a place bisimulation [4], at least for P/T nets. Nonetheless, $\sim_p$ has been characterized as the union of all the *reflexive* strong place bisimulations [4]. Of course, $\sim_p$ is finer than $\sim_{int}$, because a strong place bisimulation R is such that $R^{\oplus}$ is an interleaving bisimulation. This will be illustrated in the following, also by means of examples.

3 A Distributed Approach to Equivalence Checking

3.1 Additive Closure and its Properties

The additive closure $R^{\oplus}$ of a place relation R was defined in Definition 5. The sentence after that definition ensures the following.

Proposition 1 *For each BPP net $N = (S, A, T)$ and each place relation $R \subseteq S \times S$, if $(m_1, m_2) \in R^{\oplus}$, then $|m_1| = |m_2|$.* □

We list some obvious properties of an additively closed place relation.

Proposition 2 *For each BPP net $N = (S, A, T)$ and each place relation $R \subseteq S \times S$, the following hold:*

1. *If R is reflexive, then $R^{\oplus}$ is reflexive.*
2. *If R is symmetric, then $R^{\oplus}$ is symmetric.*
3. *If R is transitive, then $R^{\oplus}$ is transitive.*
4. *If $R_1 \subseteq R_2$, then $R_1^{\oplus} \subseteq R_2^{\oplus}$, i.e., the additive closure is monotone.* □

A consequence of the proposition above is that if R is an equivalence relation, then its additive closure $R^{\oplus}$ is also an equivalence relation. Another property of the additive closure $R^{\oplus}$ of a place relation R is that it is additive, indeed; moreover, it is also subtractive when R is an equivalence relation.

Proposition 3 (Additivity/Subtractivity) *Given a BPP net $N = (S, A, T)$ and a place relation R, the following hold:*

1. *If $(m_1, m_2) \in R^{\oplus}$ and $(m_1', m_2') \in R^{\oplus}$, then $(m_1 \oplus m_1', m_2 \oplus m_2') \in R^{\oplus}$.*
2. *If R is an equivalence relation, $(m_1 \oplus m_1', m_2 \oplus m_2') \in R^{\oplus}$ and $(m_1, m_2) \in R^{\oplus}$, then $(m_1', m_2') \in R^{\oplus}$.*

Proof By induction on the size of m_1. Case 1 is obvious. For case 2, if $|m_1| = 0$, then $m_1 = \theta = m_2$, so that the thesis follows trivially. Otherwise, $|m_1| = n+1$ (with $n \geq 0$). Since $(m_1, m_2) \in R^{\oplus}$, if $m_1 = s_1 \oplus \overline{m}_1$, then, by Definition 5, there exist s_2 and $\overline{m}_2$ such that $m_2 = s_2 \oplus \overline{m}_2$, $(s_1, s_2) \in R$ and $(\overline{m}_1, \overline{m}_2) \in R^{\oplus}$. Since $(s_1 \oplus \overline{m}_1 \oplus m_1', m_2 \oplus m_2') \in R^{\oplus}$ by hypothesis, then, by Definition 5, there exist s_2' and $\overline{m}$ such that $m_2 \oplus m_2' = s_2' \oplus \overline{m}$, $(s_1, s_2') \in R$ and $(\overline{m}_1 \oplus m_1', \overline{m}) \in R^{\oplus}$. Note that if $s_2 = s_2'$, then $\overline{m} = \overline{m}_2 \oplus m_2'$; hence, $(\overline{m}_1 \oplus m_1', \overline{m}_2 \oplus m_2') \in R^{\oplus}$, $(\overline{m}_1, \overline{m}_2) \in R^{\oplus}$ and $|\overline{m}_1| = n$, so that, by induction, we get the thesis $(m_1', m_2') \in R^{\oplus}$. Otherwise, since R is an equivalence relation, we have that s_2 and s_2' are R-related, and so when a place matches s_2, it also matches s_2' just as well. Note that $\overline{m}$ contains one occurrence of s_2, which is matched in the proof of $(\overline{m}_1 \oplus m_1', \overline{m}) \in R^{\oplus}$. If we replace this occurrence of s_2 by s_2' in this match, we get a proof for $(\overline{m}_1 \oplus m_1', \overline{m}_2 \oplus m_2') \in R^{\oplus}$, where $\overline{m}_2 \oplus m_2' = s_2' \oplus \overline{m} \ominus s_2$. So, we have that $(\overline{m}_1 \oplus m_1', \overline{m}_2 \oplus m_2') \in R^{\oplus}$, $(\overline{m}_1, \overline{m}_2) \in R^{\oplus}$ and $|\overline{m}_1| = n$, so that, by induction, we get the thesis $(m_1', m_2') \in R^{\oplus}$. □

Example 3 The requirement that R is an equivalence relation is strictly necessary for Propositon 3(2). As a counterexample, consider $R = \{(s_1, s_3), (s_1, s_4), (s_2, s_4)\}$. We have that $(s_1 \oplus s_2, s_3 \oplus s_4) \in R^{\oplus}$ and $(s_1, s_4) \in R^{\oplus}$, but $(s_2, s_3) \notin R^{\oplus}$. □

Algorithm 1 Checking the Additive Closure of an Equivalence Place Relation

Let $N = (S, A, T)$ be *BPP* net.
Let $R \subseteq S \times S$ be a place relation, which is an equivalence.
Let A be the adjacency matrix generated as follows: $A[s, s'] = 1$ if $(s, s') \in R$; otherwise $A[s, s'] = 0$.
Let $m_1 = k_1 \cdot s_{11} \oplus k_2 \cdot s_{12} \oplus \ldots \oplus k_{j_1} \cdot s_{1j_1}$ such that $k_i > 0$ for $i = 1, \ldots, j_1$, and $\sum_{i=1}^{j_1} k_i = k$. Let M_1 be an array of length j_1 such that $M_1[j] = k_j$, for $j = 1, \ldots, j_1$.
Let $m_2 = h_1 \cdot s_{21} \oplus h_2 \cdot s_{22} \oplus \ldots \oplus h_{j_2} \cdot s_{2j_2}$ such that $h_i > 0$ for $i = 1, \ldots, j_2$, and $\sum_{i=1}^{j_2} h_i = k$. Let M_2 be an array of length j_2 such that $M_2[j] = h_j$, for $j = 1, \ldots, j_2$.

```
Let P be the set of currently matched R-related places, initialized to ∅
for i = 1 to j1 do
    for j = 1 to M1[i] do
        h = 1
        b = true
        while (h ≤ j2 and b) do
            if M2[h] ≠ 0  and  A[s1i, s2h] == 1 then
                add (s1i, s2h) to P
                M2[h] = M2[h] − 1
                b = false
            else
                h = h + 1
            end if
        end while
        if h > j2 then
            return false
        end if
    end for
end for
return P
```

Remark 3 **(Complexity 1)** Given a place relation R, which is assumed to be an *equivalence* relation, the complexity of checking if two markings m_1 and m_2 of equal size are related by $R^{\oplus}$ is very low. In fact, if R is implemented as an adjacency matrix, then the complexity of checking if two markings m_1 and m_2 (represented as an array of places with multiplicities) are related by $R^{\oplus}$ is $O(k^2)$, where k is the size of the markings, since the problem is essentially that of finding for each element s_1 of m_1 a matching, R-related element s_2 of m_2, as described by Algorithm 1. Note that this algorithm is correct only if R is an equivalence relation, so that $R^{\oplus}$ is subtractive. In fact, assuming that $(m_1, m_2) \in R^{\oplus}$, when we match one place, say s_1, in m_1 with one place, say s_2, in m_2 such that $(s_1, s_2) \in R$, then we need that also $(m_1 \ominus s_1, m_2 \ominus s_2) \in R^{\oplus}$ (*cf.* Example 3). $\square$

Now we list some useful, and less obvious, properties of additively closed place relations.

Proposition 4 *For each BPP net $N = (S, A, T)$ and for each family of place relations $R_i \subseteq S \times S$ ($i \in I$), the following hold:*

1. $\emptyset^{\oplus} = \{(\theta, \theta)\}$*, i.e., the additive closure of the empty place relation is a singleton marking relation, relating the empty marking to itself.*
2. $(\mathscr{I}_S)^{\oplus} = \mathscr{I}_M$*, i.e., the additive closure of the identity relation on places* $\mathscr{I}_S = \{(s,s) \mid s \in S\}$ *is the identity relation on markings* $\mathscr{I}_M = \{(m,m) \mid m \in \mathscr{M}(S)\}$.
3. $(R^{\oplus})^{-1} = (R^{-1})^{\oplus}$*, i.e., the inverse of an additively closed relation R is the additive closure of its inverse R^{-1}.*
4. $(R_1 \circ R_2)^{\oplus} = (R_1^{\oplus}) \circ (R_2^{\oplus})$*, i.e., the additive closure of the composition of two place relations is the compositions of their additive closures.*
5. $\bigcup_{i \in I}(R_i^{\oplus}) \subseteq (\bigcup_{i \in I} R_i)^{\oplus}$*, i.e., the union of additively closed relations is included into the additive closure of their union.*

Proof The proof is by induction on the inductive definition of $R^{\oplus}$, as in Definition 5.

(1) The axiom ensures that $(\theta, \theta) \in \emptyset^{\oplus}$; no other pair can be added, as the rule can never be applied.

(2) The axiom ensures that the empty marking is related to itself. The rule, by assuming, by induction, that $m_1 = m_2$, ensures that $s_1 \oplus m_1 = s_2 \oplus m_2$, as $(s_1, s_2) \in \mathscr{I}_S$ ensures that $s_1 = s_2$.

(3) $(m_2, m_1) \in (R^{\oplus})^{-1}$ if and only if $(m_1, m_2) \in R^{\oplus}$. Then, $(m_1, m_2) \in R^{\oplus}$ if and only if either $m_1 = \theta = m_2$, or $m_1 = s_1 \oplus m_1'$, $m_2 = s_2 \oplus m_2'$, $(s_1, s_2) \in R$ and, moreover, $(m_1', m_2') \in R^{\oplus}$. In the former case, the pair $(\theta, \theta) \in (R^{-1})^{\oplus}$ by the axiom in Definition 5. In the latter case, by induction, we can assume that $(m_2', m_1') \in (R^{-1})^{\oplus}$; moreover, we have that $(s_2, s_1) \in R^{-1}$. Hence, by using the rule in Definition 5, we also have that $(s_2 \oplus m_2', s_1 \oplus m_1') \in (R^{-1})^{\oplus}$, i.e., $(m_2, m_1) \in (R^{-1})^{\oplus}$, as required.

(4) If $(m_1, m_3) \in (R_1^{\oplus}) \circ (R_2^{\oplus})$, then there exists m_2 such that $(m_1, m_2) \in R_1^{\oplus}$ and $(m_2, m_3) \in R_2^{\oplus}$. Now we proceed by induction on the size of the involved markings. If $m_1 = \theta$, then $m_2 = \theta = m_3$ and $(\theta, \theta) \in (R_1 \circ R_2)^{\oplus}$ by the axiom in Definition 5. If $m_1 = s_1 \oplus m_1'$, then by the rule in Definition 5, there exist s_2 and m_2' such that $m_2 = s_2 \oplus m_2'$, $(s_1, s_2) \in R_1$ and $(m_1', m_2') \in R_1^{\oplus}$. Since $(m_2, m_3) \in R_2^{\oplus}$, it follows that there exist s_3 and m_3' such that $m_3 = s_3 \oplus m_3'$, $(s_2, s_3) \in R_2$ and $(m_2', m_3') \in R_2^{\oplus}$. Therefore,

$(s_1, s_3) \in R_1 \circ R_2$ and, by induction, $(m'_1, m'_3) \in (R_1 \circ R_2)^{\oplus}$. Hence, as required, also $(m_1, m_3) \in (R_1 \circ R_2)^{\oplus}$, by the rule in Definition 5.

If $(m_1, m_3) \in (R_1 \circ R_2)^{\oplus}$, then we proceed by induction on the size of m_1. If $m_1 = \theta$, then $m_3 = \theta$ and $(\theta, \theta) \in (R_1^{\oplus}) \circ (R_2^{\oplus})$ because $(\theta, \theta) \in R_1^{\oplus}$ and $(\theta, \theta) \in R_2^{\oplus}$. If $m_1 = s_1 \oplus m'_1$, then by the rule in Definition 5, there exist s_3 and m'_3 such that $m_3 = s_3 \oplus m'_3$, $(s_1, s_3) \in R_1 \circ R_2$ and $(m'_1, m'_3) \in (R_1 \circ R_2)^{\oplus}$. Since $(s_1, s_3) \in R_1 \circ R_2$, there exists s_2 such that $(s_1, s_2) \in R_1$ and $(s_2, s_3) \in R_2$. Moreover, $(m'_1, m'_3) \in (R_1^{\oplus}) \circ (R_2^{\oplus})$ by induction, so that m'_2 exists such that $(m'_1, m'_2) \in R_1^{\oplus}$ and $(m'_2, m'_3) \in R_2^{\oplus}$. Therefore, by Definition 5 we have that $(s_1 \oplus m'_1, s_2 \oplus m'_2) \in R_1^{\oplus}$ and $(s_2 \oplus m'_2, s_3 \oplus m'_3) \in R_2^{\oplus}$, so that $(m_1, m_3) \in (R_1^{\oplus}) \circ (R_2^{\oplus})$.

(5) If $(m_1, m_2) \in \bigcup_{i \in I} (R_i^{\oplus})$, then there exists $j \in I$ such that $(m_1, m_2) \in R_j^{\oplus}$. Since $R_j \subseteq \bigcup_{i \in I} R_i$, then the proof that $(m_1, m_2) \in R_j^{\oplus}$ can be adapted to prove that $(m_1, m_2) \in (\bigcup_{i \in I} R_i)^{\oplus}$, too, by simply changing each occurrence of R_j (or $R_j^{\oplus}$) in the premise of the rule with $\bigcup_{i \in I} R_i$ (or $(\bigcup_{i \in I} R_i)^{\oplus}$). □

3.2 Team Bisimulation on Places

Definition 7 (Team bisimulation) Let $N = (S, A, T)$ be a BPP net. A *team bisimulation* is a place relation $R \subseteq S \times S$ such that if $(s_1, s_2) \in R$ then for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in R^{\oplus}$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $(m_1, m_2) \in R^{\oplus}$.

Two places s and s' are *team bisimilar* (or *team bisimulation equivalent*), denoted $s \sim s'$, if there exists a team bisimulation R such that $(s, s') \in R$. □

Remark 4 **(Conservative extension)** A team bisimulation over a finite-state machine (FSM, for short), i.e., a net whose transitions have singleton pre-set and singleton (or empty) post-set, is actually a conservative extension of the definition of bisimulation for FSMs, defined in [33]; indeed, the team bisimulation condition $(m_1, m_2) \in R^{\oplus}$ can be simplified to the equivalent FSM bisimulation condition: *either* $m_1 = \theta = m_2$ *or* $(m_1, m_2) \in R$. □

Example 4 Continuing Example 1 about the semi-counters in Figure 1, it is easy to see that relation $R = \{(s_1, s_3), (s_1, s_4), (s_2, s_5), (s_2, s_6)\}$ is a team bisimulation. In fact, the pair (s_1, s_3) is a team bisimulation pair because, to transition $s_1 \xrightarrow{inc} s_1 \oplus s_2$, s_3 can respond with $s_3 \xrightarrow{inc} s_4 \oplus s_5$, and $(s_1 \oplus s_2, s_4 \oplus s_5) \in R^{\oplus}$; symmetrically, if s_3 moves first. Also the pair (s_1, s_4) is a team bisimulation pair because, to transition $s_1 \xrightarrow{inc} s_1 \oplus s_2$, s_4 can respond with $s_4 \xrightarrow{inc} s_3 \oplus s_6$, and $(s_1 \oplus s_2, s_3 \oplus s_6) \in R^{\oplus}$; symmetrically, if s_4 moves first. Also the pair (s_2, s_5) is a team bisimulation pair: to transition $s_2 \xrightarrow{dec} \theta$, s_5 responds with $s_5 \xrightarrow{dec} \theta$, and $(\theta, \theta) \in R^{\oplus}$. Similarly for the pair (s_2, s_6). Hence, relation R is a team bisimulation, indeed. This example shows that team bisimilarity is compatible with the notion of net unfolding, as the net in (b) can be seen as a sort of partial unfolding of the net in (a).

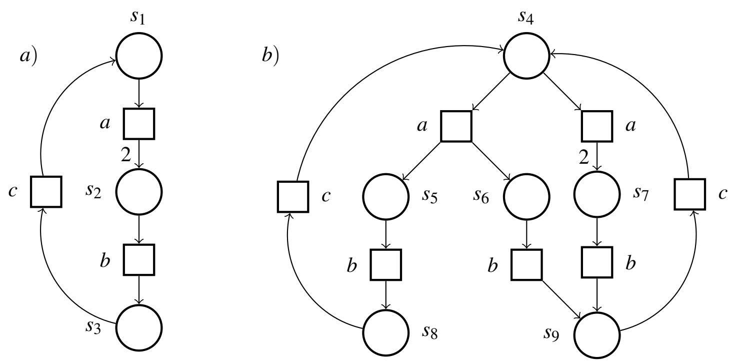

Fig. 2 Two team bisimilar BPP nets

The team bisimulation above is a very simple finite relation proving that s_1 and s_3 are team bisimulation equivalent. In Example 2, in order to show that s_1 and s_3 are interleaving bisimilar, we had to introduce a very complex relation, with infinitely many pairs. □

Example 5 Consider the nets in Figure 2. It is not difficult to realize that relation $R = \{(s_1,s_4),(s_2,s_5),(s_2,s_6),(s_2,s_7),(s_3,s_8),(s_3,s_9)\}$ is a team bisimulation. This example shows that team bisimulation is compatible with duplication of behavior and fusion of places. □

We now list some useful properties of team bisimulation relations.

Proposition 5 *For each BPP net $N = (S,A,T)$, the following hold:*

1. *The identity relation $\mathscr{I}_S = \{(s,s) \mid s \in S\}$ is a team bisimulation;*
2. *the inverse relation $R^{-1} = \{(s',s) \mid (s,s') \in R\}$ of a team bisimulation R is a team bisimulation;*
3. *the relational composition $R_1 \circ R_2 = \{(s,s'') \mid \exists s'.(s,s') \in R_1 \wedge (s',s'') \in R_2\}$ of two team bisimulations R_1 and R_2 is a team bisimulation;*
4. *the union $\bigcup_{i\in I} R_i$ of team bisimulations R_i is a team bisimulation.*

Proof The proof is almost standard, due to Proposition 4.

(1) $(s,s) \in \mathscr{I}_S$ is a team bisimulation pair because whatever transition s performs (say, $s \xrightarrow{\ell} m$), the other s in the pair does exactly the same transition $s \xrightarrow{\ell} m$ and $(m,m) \in \mathscr{I}_S^{\oplus}$, by Proposition 4(2), as required by the team bisimulation definition.

(2) Suppose $(s_2,s_1) \in R^{-1}$ and $s_2 \xrightarrow{\ell} m_2$. Since $(s_1,s_2) \in R$ and R is a team bisimulation, then second item of the team bisimulation game ensures that m_1 exists such that $s_1 \xrightarrow{\ell} m_1$, with $(m_1,m_2) \in R^{\oplus}$, i.e., $(m_2,m_1) \in (R^{\oplus})^{-1}$. By Proposition 4(3), $(m_2,m_1) \in (R^{\oplus})^{-1}$ iff $(m_2,m_1) \in (R^{-1})^{\oplus}$. Summing up, if $(s_2,s_1) \in R^{-1}$ and $s_2 \xrightarrow{\ell} m_2$, then a marking m_1 exists such that $s_1 \xrightarrow{\ell} m_1$, with $(m_2,m_1) \in (R^{-1})^{\oplus}$, as required. The case when s_1 moves first is symmetric and so omitted.

(3) Given a pair $(s,s'') \in R_1 \circ R_2$, there exists a place s' such that $(s,s') \in R_1$ and $(s',s'') \in R_2$; as $(s,s') \in R_1$, if $s \xrightarrow{\ell} m_1$, there exists m_2 such that $s' \xrightarrow{\ell} m_2$ with

$(m_1, m_2) \in R_1^{\oplus}$. But as $(s', s'') \in R_2$, we have also that there exists m_3 such that $s'' \xrightarrow{\ell} m_3$ with $(m_2, m_3) \in R_2^{\oplus}$. Hence, $(m_1, m_3) \in (R_1^{\oplus}) \circ (R_2^{\oplus})$ and, by Proposition 4(4), it follows that $(m_1, m_3) \in (R_1 \circ R_2)^{\oplus}$. Summing up, for any pair $(s, s'') \in R_1 \circ R_2$, if $s \xrightarrow{\ell} m_1$, then a marking m_3 exists such that $s'' \xrightarrow{\ell} m_3$ with $(m_1, m_3) \in (R_1 \circ R_2)^{\oplus}$, as required.

(4) Assume $(s, s') \in \bigcup_{i \in I} R_i$; then, there exists $j \in I$ such that (s, s') belongs to team bisimulation R_j. If $s \xrightarrow{\ell} m_1$, then there must exist a marking m_2 such that $s' \xrightarrow{\ell} m_2$ with $(m_1, m_2) \in R_j^{\oplus}$. By Proposition 4(5), $(m_1, m_2) \in (\bigcup_{i \in I} R_i)^{\oplus}$ as $R_j \subseteq \bigcup_{i \in I} R_i$. So $\bigcup_{i \in I} R_i$ is a team bisimulation, too. $\square$

Remember that $s \sim s'$ if there exists a team bisimulation containing the pair (s, s'). This means that $\sim$ is the union of all team bisimulations, i.e.,

$$\sim = \bigcup \{R \subseteq S \times S \mid R \text{ is a team bisimulation}\}.$$

By Proposition 5(4), $\sim$ is also a team bisimulation, hence the largest such relation.

Proposition 6 *For each BPP net $N = (S, A, T)$, relation $\sim \subseteq S \times S$ is the largest team bisimulation relation.* $\square$

Observe that a team bisimulation relation need not be reflexive, symmetric, or transitive. Nonetheless, the largest team bisimulation relation $\sim$ is an equivalence relation. As a matter of fact, as the identity relation $\mathscr{I}_S$ is a team bisimulation by Proposition 5(1), we have that $\mathscr{I}_S \subseteq \sim$, and so $\sim$ is reflexive. Symmetry derives from the following argument. For any $(s, s') \in \sim$, there exists a team bisimulation R such that $(s, s') \in R$; by Proposition 5(2), relation R^{-1} is a team bisimulation containing the pair (s', s); hence, $(s', s) \in \sim$ because $R^{-1} \subseteq \sim$. Transitivity also holds for $\sim$. Assume $(s, s') \in \sim$ and $(s', s'') \in \sim$; hence, there exist two team bisimulations R_1 and R_2 such that $(s, s') \in R_1$ and $(s', s'') \in R_2$; by Proposition 5(3), relation $R_1 \circ R_2$ is a team bisimulation containing the pair (s, s''); hence, $(s, s'') \in \sim$, because $R_1 \circ R_2 \subseteq \sim$. Summing up, we have the following.

Proposition 7 *For each BPP net $N = (S, A, T)$, relation $\sim \subseteq S \times S$ is an equivalence relation.* $\square$

It is sometimes convenient to write a team bisimulation compactly, by removing those pairs that differ from others only up to the use of team bisimulation equivalent alternatives. The resulting relation is *not* a team bisimulation, rather a team bisimulation up to $\sim$. We denote by $\sim R \sim$ the relational composition $\sim \circ R \circ \sim$; in other words, by $s \sim R \sim s'$ we mean that two places s_1 and s_2 exist such that $s \sim s_1$, $(s_1, s_2) \in R$ and $s_2 \sim s'$. By Proposition 4(4), $(\sim R \sim)^{\oplus}$ is the same as $\sim^{\oplus} \circ R^{\oplus} \circ \sim^{\oplus}$. Hence, by $m(\sim R \sim)^{\oplus} m'$ we mean that there exist two markings m_1 and m_2 such that $m \sim^{\oplus} m_1$, $(m_1, m_2) \in R^{\oplus}$ and $m_2 \sim^{\oplus} m'$.

Definition 8 (Team Bisimulation up to $\sim$) Given a BPP net $N = (S, A, T)$, a team bisimulation up to $\sim$ is a binary relation R on S such that if $(s_1, s_2) \in R$ then for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $m_1(\sim R \sim)^{\oplus} m_2$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $m_1(\sim R \sim)^{\oplus} m_2$. □

Lemma 1 *Given a BPP net $N = (S, A, T)$, if R is a team bisimulation up to $\sim$, then $\sim R \sim$ is a team bisimulation.*

Proof Assume $s \sim R \sim s'$, i.e., there exist s_1 and s_2 such that $s \sim s_1$, $(s_1, s_2) \in R$ and $s_2 \sim s'$. We have to prove that for each $s \xrightarrow{\ell} m^1$ there exists a transition $s' \xrightarrow{\ell} m^2$ such that $m^1(\sim R \sim)^{\oplus} m^2$ (the symmetric case when s' moves first is omitted). If $s \xrightarrow{\ell} m^1$, since $s \sim s_1$, then there exists m_1 such that $s_1 \xrightarrow{\ell} m_1$ with $m^1 \sim^{\oplus} m_1$. As $(s_1, s_2) \in R$, there exists m_2 such that $s_2 \xrightarrow{\ell} m_2$ with $m_1(\sim R \sim)^{\oplus} m_2$. Since $s_2 \sim s'$, there exists m^2 such that $s' \xrightarrow{\ell} m^2$ with $m_2 \sim^{\oplus} m^2$. Summing up, $m^1 \sim^{\oplus} m_1$ and $m_1(\sim R \sim)^{\oplus} m_2$ and $m_2 \sim^{\oplus} m^2$ can be shortened to $m^1(\sim R \sim)^{\oplus} m^2$, because $\sim^{\oplus} \circ (\sim R \sim)^{\oplus} \circ \sim^{\oplus}$ is the same as $(\sim \circ \sim R \sim \circ \sim)^{\oplus}$ by Proposition 4(4) and $\sim \circ \sim \subseteq \sim$ by Proposition 5(3) and Proposition 6. Hence, we have proved that if $s \sim R \sim s'$ then for each m^1 such that $s \xrightarrow{\ell} m^1$ there exists a marking m^2 such that $s' \xrightarrow{\ell} m^2$ with $m^1(\sim R \sim)^{\oplus} m^2$, as required by the definition of team bisimulation. □

Proposition 8 *If R is a team bisimulation up to $\sim$, then $R \subseteq \sim$.*

Proof By Lemma 1, $\sim R \sim$ is a team bisimulation, hence $\sim R \sim \subseteq \sim$ by definition of $\sim$. As the identity relation $\mathscr{I} \subseteq \sim$ by Proposition 5(1), we have that relation $R = \mathscr{I} \circ R \circ \mathscr{I} \subseteq \sim R \sim$, hence $R \subseteq \sim$ by transitivity. □

3.3 Team Bisimilarity over Markings

Starting from team bisimilarity $\sim$, which has been computed over the places of an *unmarked* BPP net, we can extend team bisimulation equivalence over its markings in a distributed way: m_1 is team bisimulation equivalent to m_2 if they are related by the additive closure of $\sim$, i.e., if $(m_1, m_2) \in \sim^{\oplus}$, usually denoted by $m_1 \sim^{\oplus} m_2$.

Proposition 9 *For each BPP net $N = (S, A, T)$, if $m_1 \sim^{\oplus} m_2$, then $|m_1| = |m_2|$.*

Proof Directly from Proposition 1. □

Proposition 10 *For each BPP net $N = (S, A, T)$, relation $\sim^{\oplus} \subseteq \mathscr{M}(S) \times \mathscr{M}(S)$ is an equivalence relation.*

Proof By Proposition 2: since $\sim$ is an equivalence relation (Proposition 7), its additive closure $\sim^{\oplus}$ is also an equivalence relation. □

Example 6 Continuing Example 4 about the semi-counters, the marking $s_1 \oplus 2 \cdot s_2$ is team bisimilar to the following markings of the net in (b): $s_3 \oplus 2 \cdot s_5$, or $s_3 \oplus s_5 \oplus s_6$, or $s_3 \oplus 2 \cdot s_6$, or $s_4 \oplus 2 \cdot s_5$, or $s_4 \oplus s_5 \oplus s_6$, or $s_4 \oplus 2 \cdot s_6$. □

Example 7 Continuing Example 5 about Figure 2, it is clear that, for instance, $s_1 \oplus 3 \cdot s_2$ is team bisimilar to any marking obtained with one token on place s_4 and three tokens distributed over the places s_5, s_6 and s_7; for instance, $s_1 \oplus 3 \cdot s_2 \sim^{\oplus} s_4 \oplus 2 \cdot s_5 \oplus s_7$ or $s_1 \oplus 3 \cdot s_2 \sim^{\oplus} s_4 \oplus s_6 \oplus 2 \cdot s_7$. □

Remark 5 **(Complexity 2)** In Section 3.4, we will argue that the complexity of computing $\sim$ is $O(m \cdot p^2 \cdot \log(n+1))$ time, where m is the number of the net transitions, n is the number of the net places and p is the size of the largest post-set of the net transitions (i.e., p is the least number such that $|t^\bullet| \leq p$, for all $t \in T$).

Once the place relation $\sim$ has been computed once and for all for the given net, Algorithm 1 checks whether two markings m_1 and m_2 are team bisimulation equivalent in $O(k^2)$ time, where k is the size of the markings. Moreover, if we want to check whether other two markings of the same net are team bisimilar, we can reuse the already computed $\sim$ relation, so that the time complexity is again quadratic on the size of the two markings. However, note that the time spent in creating the adjacency matrix A has not been considered: since n is the number of places, $O(n^2)$ time is needed to implement this matrix, so that the time spent for the first check is $O(n^2)$, while for subsequent checks it is only $O(k^2)$, where k is the size of the markings.

Algorithm 1 is not optimal. As a matter of fact, since the partition refinement algorithm does compute the equivalence classes of $\sim$, we can take advantage of this fact for checking whether $m_1 \sim^{\oplus} m_2$. The algorithm in [45] simply scans these equivalence classes and, for each class, it counts whether the number of tokens in the places of m_1 belonging to this class equals the number of tokens in the places of m_2 in the same class; if this holds for all the equivalence classes, then $m_1 \sim^{\oplus} m_2$. Of course, the complexity of this algorithm is $O(n)$, even for the first check; hence, this algorithm is usually more performant, even if, from the second check onwards, it may be slower when applied to small markings; in fact, if the number n of places is greater than k^2, then Algorithm 1 is better. □

Of course, two markings m_1 and m_2 are *not* team bisimilar if they have different size, or if Algorithm 1 fails by singling out a place s in the *residual* of m_1 (i.e., in the portion of m_1 which has not been scanned yet) which has no matching team bisimilar place in (the residual of) m_2. Or, equivalently, if for some equivalence class B of $\sim$, the number of all the tokens in the places of m_1 belonging to B is different from the number of all the tokens in the places of m_2 belonging to B.

Example 8 Team bisimulation equivalence is a truly concurrent equivalence. According to the semantics in Section 5, the BPP net in Figure 3(a) denotes (actually, it is isomorphic to) the net for the BPP process term $a.b.\mathbf{0} + b.a.\mathbf{0}$, which can perform the two actions a and b in either order. On the contrary, the BPP net in (b) denotes the net for the BPP process term $a.\mathbf{0}|b.\mathbf{0}$. Note that s_1 is not team equivalent to $s_4 \oplus s_5$, because the two markings have different size. Nonetheless, s_1 and $s_4 \oplus s_5$ are interleaving bisimilar. □

Example 9 If two markings m_1 and m_2 are interleaving bisimilar and have the same size, then they may be not team equivalent. For instance, consider Figure 3(c), which

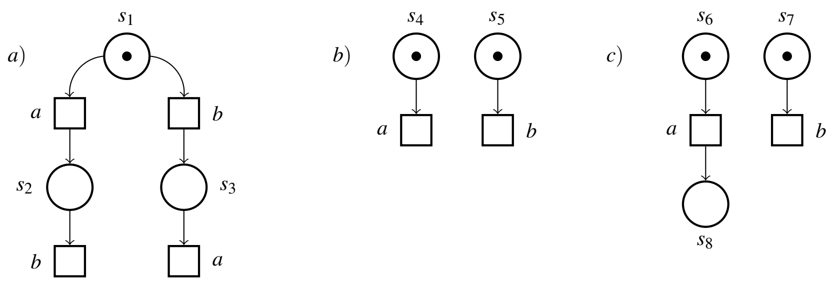

Fig. 3 Three non-team equivalent net systems: $a.b.\mathbf{0} + b.a.\mathbf{0}$, $a.\mathbf{0} \,|\, b.\mathbf{0}$ and $a.C \,|\, b.\mathbf{0}$ (with $C \doteq \mathbf{0}$)

denotes the net for the BPP process term $a.C \,|\, b.\mathbf{0}$, where C is a constant with empty body, i.e., $C \doteq \mathbf{0}$. Markings $s_4 \oplus s_5$ and $s_6 \oplus s_7$ have the same size, they are interleaving bisimilar (actually, they are even fully concurrent bisimilar [7]), but they are not team equivalent: even if $s_5 \sim s_7$, the residuals are not team bisimilar: $s_4 \nsim s_6$. As a matter of fact, $s_5 \sim s_7$ because both can only perform b and then stop successfully, reaching θ; however, $s_4 \xrightarrow{a} \theta$ while $s_6 \xrightarrow{a} s_8$ and the reached markings, having different size, are not team bisimilar. □

The following theorem provides a characterization of team bisimilarity $\sim^{\oplus}$ as a suitable bisimulation-like relation over markings. It is interesting to observe that this characterization gives a dynamic interpretation of team equivalence as a relation on the global model of the system under scrutiny, while Definition 5 gives a structural definition of team bisimulation equivalence $\sim^{\oplus}$ as the additive closure of the local relation $\sim$ on places.

Theorem 1 *Let $N = (S, A, T)$ be a BPP net. Two markings m_1 and m_2 are team bisimulation equivalent, $m_1 \sim^{\oplus} m_2$, if and only if $|m_1| = |m_2|$ and*

- *$\forall t_1$ such that $m_1[t_1\rangle m_1'$, $\exists t_2$ such that ${}^\bullet t_1 \sim {}^\bullet t_2$, $l(t_1) = l(t_2)$, $t_1^\bullet \sim^{\oplus} t_2^\bullet$, $m_2[t_2\rangle m_2'$ and $m_1' \sim^{\oplus} m_2'$, and symmetrically,*
- *$\forall t_2$ such that $m_2[t_2\rangle m_2'$, $\exists t_1$ such that ${}^\bullet t_1 \sim {}^\bullet t_2$, $l(t_1) = l(t_2)$, $t_1^\bullet \sim^{\oplus} t_2^\bullet$, $m_1[t_1\rangle m_1'$ and $m_1' \sim^{\oplus} m_2'$.*

Proof ($\Rightarrow$) If $m_1 \sim^{\oplus} m_2$, then $|m_1| = |m_2|$ by Proposition 9. Moreover, for any t_1 such that $m_1[t_1\rangle m_1'$, we have that $m_1 = s_1 \oplus \overline{m}_1$, where $s_1 = {}^\bullet t_1$. As $m_1 \sim^{\oplus} m_2$, by Definition 5, it follows that there exist s_2 and $\overline{m}_2$ such that $m_2 = s_2 \oplus \overline{m}_2$, $s_1 \sim s_2$ and $\overline{m}_1 \sim^{\oplus} \overline{m}_2$. Since $s_1 \sim s_2$, by Definition 7, there exists a transition t_2 such that ${}^\bullet t_2 = s_2$, $l(t_2) = l(t_1)$ and $t_1^\bullet \sim^{\oplus} t_2^\bullet$. Hence, $m_1' = t_1^\bullet \oplus \overline{m}_1$ and $m_2' = t_2^\bullet \oplus \overline{m}_2$, and so $m_1' \sim^{\oplus} m_2'$ by Proposition 3(1). The symmetric case when m_2 moves first is symmetric and hence omitted.

($\Leftarrow$) Let us assume that $|m_1| = |m_2|$ and that the two bisimulation-like conditions hold; then, we prove that $m_1 \sim^{\oplus} m_2$. First of all, assume that no transition t_1 is enabled at m_1; in such a case, also no transition t_2 can be enabled at m_2; in fact, if $m_2[t_2\rangle m_2'$, then, by the second condition, a transition t_1 must be executable at m_1, contradicting the assumption that no transition is enabled at m_1. If each place in m_1 is a deadlock, and similarly each place in m_2 is a deadlock, then all the places in m_1 and m_2 are pairwise team bisimilar; hence, the condition $|m_1| = |m_2|$ is enough to ensure that $m_1 \sim^{\oplus} m_2$. Now, assume that $m_1[t_1\rangle m_1'$ for some t_1; the first condition ensures that there exists t_2 such that ${}^\bullet t_1 \sim {}^\bullet t_2$, $l(t_1) = l(t_2)$, $t_1^\bullet \sim^{\oplus} t_2^\bullet$, $m_2[t_2\rangle m_2'$ and $m_1' \sim^{\oplus} m_2'$. Hence, we have that $m_1' = t_1^\bullet \oplus \overline{m}_1$, $m_2' = t_2^\bullet \oplus \overline{m}_2$, $m_1 = {}^\bullet t_1 \oplus \overline{m}_1$, $m_2 = {}^\bullet t_2 \oplus \overline{m}_2$. Since $m_1' \sim^{\oplus} m_2'$ and $t_1^\bullet \sim^{\oplus} t_2^\bullet$, by Proposition 3(2) it follows that $\overline{m}_1 \sim^{\oplus} \overline{m}_2$, and so, by Proposition 3(1), $m_1 \sim^{\oplus} m_2$, because ${}^\bullet t_1 \sim {}^\bullet t_2$. Symmetrically, if we start from a transition t_2 enabled at m_2. □

Corollary 1 (Strong place bisimilarity and team bisimilarity coincide) *Let $N = (S, A, T)$ be a BPP net. Two markings m_1 and m_2 are team bisimulation equivalent, $m_1 \sim^{\oplus} m_2$, if and only if they are strong place bisimilar, $m_1 \sim_p m_2$.*

Proof By Theorem 1, it is clear that team bisimulation equivalence $\sim$ over places is a strong place bisimulation. For the reverse implication, consider a strong place bisimulation R and some $(s_1, s_2) \in R$. By Definition 6, to transition $s_1 \xrightarrow{\ell} m_1$, s_2 can respond with $s_2 \xrightarrow{\ell} m_2$ such that $(m_1, m_2) \in R^{\oplus}$. This means that R is a team bisimulation. □

Hence, team bisimulation equivalence $\sim$ is the largest strong place bisimulation. Therefore, our characterization of strong place bisimilarity is quite appealing because it is based on the basic definition of team bisimulation on the places of the unmarked net, and, moreover, offers a very efficient algorithm to check if two markings are strong place bisimilar, as discussed in Remark 5 and in the next section.

Corollary 2 (Team bisimilarity is finer than interleaving bisimilarity) *Let $N = (S, A, T)$ be a BPP net. If $m_1 \sim^{\oplus} m_2$, then $m_1 \sim_{int} m_2$.*

Proof By Theorem 1, it is clear that $\sim^{\oplus}$ is an interleaving bisimulation. □

3.4 Team Bisimilarity over Places as a Fixed Point

Team bisimulation equivalence over places can be characterized nicely as the greatest fixed point of a suitable monotone relation transformer, essentially by extending the characterization developed for ordinary bisimulation over LTSs [47, 57, 31]. Not surpisingly, this has the interesting consequence of defining a natural, even if not optimal, algorithm for computing this equivalence.

Definition 9 Given a BPP net $N = (S, A, T)$, the functional $F : \wp(S \times S) \to \wp(S \times S)$ (i.e., a transformer of binary relations over S) is defined as follows. If $R \subseteq S \times S$, then $(s_1, s_2) \in F(R)$ if and only if for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in R^{\oplus}$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $(m_1, m_2) \in R^{\oplus}$. □

Proposition 11 *For each BPP net $N = (S, A, T)$, we have that:*

1. *The functional F is monotone, i.e., if $R_1 \subseteq R_2$ then $F(R_1) \subseteq F(R_2)$.*
2. *A relation $R \subseteq S \times S$ is a team bisimulation if and only if $R \subseteq F(R)$.*

Proof The proof of (1) derives easily form the definition of F: if $(s_1, s_2) \in F(R_1)$ then for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in R_1^{\oplus}$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\mu} m_1$ and $(m_1, m_2) \in R_1^{\oplus}$.

Since $R_1 \subseteq R_2$, by Proposition 2(4), it follows that $R_1^{\oplus} \subseteq R_2^{\oplus}$; hence, the above implies that for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in R_2^{\oplus}$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $(m_1, m_2) \in R_2^{\oplus}$,

which means that $(s_1, s_2) \in F(R_2)$. Summing up, $F(R_1) \subseteq F(R_2)$.

The proof of (2) is also easy: if R is a team bisimulation, then if $(s_1, s_2) \in R$ then for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in R^{\oplus}$,
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $(m_1, m_2) \in R^{\oplus}$.

and, by using the reverse implication, this means that $(s_1, s_2) \in F(R)$, i.e., $R \subseteq F(R)$. Similarly, if $R \subseteq F(R)$, then the condition holding for $F(R)$ holds also for all the elements of R, hence R is a team bisimulation. □

A *fixed point* for F is a relation R such that $R = F(R)$. Knaster-Tarski's fixed point theorem (see, e.g., [19]) ensures that the greatest fixed point of the monotone functional F is

$$\bigcup \{R \subseteq S \times S \mid R \subseteq F(R)\}.$$

We want to show that this greatest fixed point is $\sim$. A *post-fixed point* of F is a relation R such that $R \subseteq F(R)$. By Proposition 11(2), we know that the team bisimulations are the post-fixed points of F. Team bisimilarity $\sim$ is the union of all the team bisimulations:

$$\sim = \bigcup \{R \subseteq S \times S \mid R \text{ is a team bisimulation}\}.$$

Hence, we also conclude that $\sim$ is the greatest fixed point of F, i.e.:

$$\sim = \bigcup \{R \subseteq S \times S \mid R \subseteq F(R)\}.$$

Here we provide a direct proof of this fact.

Theorem 2 *Team bisimilarity $\sim$ is the greatest fixed point of F.*

Proof We first prove that $\sim$ is a fixed point, i.e., $\sim = F(\sim)$, by proving that $\sim \subseteq F(\sim)$ and that $F(\sim) \subseteq \sim$. Since $\sim$ is a team bisimulation, $\sim \subseteq F(\sim)$ by Proposition 11(2). As F is monotonic, by Proposition 11(1) we have that $F(\sim) \subseteq F(F(\sim))$, i.e., also $F(\sim)$ is a post-fixed point of F i.e., a team bisimulation. Since we know that $\sim$ is the union of all team bisimulation relations (as well as the greatest post-fixed point of F), it follows that $F(\sim) \subseteq \sim$.

Now we want to show that $\sim$ is the greatest fixed point. Assume T is another fixed point of F, i.e. $T = F(T)$. Then, in particular, we have that $T \subseteq F(T)$, i.e., T is a team bisimulation by Propostion 11(2), hence $T \subseteq \sim$. □

There is a natural iterative way of approximating $\sim$ by means of a descending (actually, initially descending, and then constant from a certain point onwards) chain of relations indexed on the natural numbers. We will see that there is a strict relation between this chain of relations and the functional F above.

Definition 10 Given a BPP net $N = (S, A, T)$, for each natural $i \in \mathbb{N}$, we define the binary relation $\sim_i$ over S as follows:

- $\sim_0 = S \times S$.
- $s_1 \sim_{i+1} s_2$ if and only if for all $\ell \in A$
  - $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $m_1 \sim_i^{\oplus} m_2$
  - $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $m_1 \sim_i^{\oplus} m_2$.

We denote by $\sim_\omega$ the relation $\bigcap_{i \in \mathbb{N}} \sim_i$. □

Intuitively, $s_1 \sim_i s_2$ if and only if the two places are team bisimilar up to paths of length at most i. Hence, all the places are in the relation $\sim_0$.

Proposition 12 *For each $i \in \mathbb{N}$:*

1. *relation $\sim_i$ is an equivalence relation,*
2. $\sim_i = F^i(S \times S)$
3. $\sim_{i+1} \subseteq \sim_i$,

Moreover, $\sim_\omega = \bigcap_{i \in \mathbb{N}} \sim_i$ is an equivalence relation.

Proof (1) The proof is by induction on i. The base case is obvious: $\sim_0$ is an equivalence relation because so is the universal relation. Assuming that $\sim_i$ is an equivalence relation, we show that also $\sim_{i+1}$ is an equivalence relation. Reflexivity is trivial: as for all $\ell \in A$

- $\forall m$ such that $s \xrightarrow{\ell} m$, $\exists m$ such that $s \xrightarrow{\ell} m$ and $m \sim_i^{\oplus} m$,

it follows that also $s \sim_{i+1} s$. Symmetry is also easy: we have to prove that if $s_1 \sim_{i+1} s_2$ then $s_2 \sim_{i+1} s_1$. We know that $s_1 \sim_{i+1} s_2$ if and only if for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $m_1 \sim_i^{\oplus} m_2$
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $m_1 \sim_i^{\oplus} m_2$.

We also know, by inductive hypothesis, that $\sim_i$ is symmetric, and so also $\sim_i^{\oplus}$ is symmetric by Proposition 2(2); hence, $m_2 \sim_i^{\oplus} m_1$, and, by reordering the two bisimulation-like conditions, we also have that

- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $m_2 \sim_i^{\oplus} m_1$.
- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $m_2 \sim_i^{\oplus} m_1$

which means that $s_2 \sim_{i+1} s_1$. Transitivity is similar, hence left to the reader.

(2) We first explain what we mean by the ith power of F: $F^0(R) = R$ and $F^{n+1}(R) = F(F^n(R))$. We prove that $\sim_i = F^i(S \times S)$ by induction on i. The base case is when $i = 0$. In such a case, $\sim_0 = S \times S = F^0(S \times S)$, as required. Now, by induction, we can assume that $\sim_i = F^i(S \times S)$; we want to prove that $\sim_{i+1} = F^{i+1}(S \times S)$. By definition, $s_1 \sim_{i+1} s_2$ if and only if for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $m_1 \sim_i^{\oplus} m_2$
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $m_1 \sim_i^{\oplus} m_2$.

By induction, this is equivalent to: for all $\ell \in A$

- $\forall m_1$ such that $s_1 \xrightarrow{\ell} m_1$, $\exists m_2$ such that $s_2 \xrightarrow{\ell} m_2$ and $(m_1, m_2) \in F^i(S \times S)^{\oplus}$
- $\forall m_2$ such that $s_2 \xrightarrow{\ell} m_2$, $\exists m_1$ such that $s_1 \xrightarrow{\ell} m_1$ and $(m_1, m_2) \in F^i(S \times S)^{\oplus}$.

So, by using the definition of F, it follows that $(s_1, s_2) \in F(F^i(S \times S)) = F^{i+1}(S \times S)$, as required.

(3) We prove that $\sim_{i+1} \subseteq \sim_i$ by induction on i. The base case is $\sim_1 \subseteq \sim_0$, which trivially holds because $\sim_0$ is the universal relation. Now, by induction, assume that $\sim_i \subseteq \sim_{i-1}$, which, by (2) above, is equivalent to $F^i(S \times S) \subseteq F^{i-1}(S \times S)$. We want to prove that $\sim_{i+1} \subseteq \sim_i$. By Proposition 11(1), we know that F is monotone. Hence, $\sim_{i+1} = F(F^i(S \times S)) \subseteq F(F^{i-1}(S \times S)) = \sim_i$.

Observe that $\sim_\omega = \bigcap_{i \in \mathbb{N}} \sim_i$ is an equivalence relation, as $\sim_i$ is an equivalence relation for all $i \in \mathbb{N}$. As a matter of fact, the identity relation $\mathscr{I}$ is a subset of all the $\sim_i$'s, hence $\mathscr{I} \subset \sim_\omega$, i.e., $\sim_\omega$ is reflexive. Relation $\sim_\omega$ is also symmetric because, if $(s_1, s_2) \in \sim_\omega$, then $(s_1, s_2) \in \sim_i$ for all $i \in \mathbb{N}$. Since each $\sim_i$ is symmetric, $(s_2, s_1) \in \sim_i$ for all $i \in \mathbb{N}$, so $(s_2, s_1) \in \sim_\omega$, hence $\sim_\omega$ is symmetric. Transitivity of $\sim_\omega$ can be proved similarly. □

Hence, we have a non-increasing chain of equivalence relations,

$$\sim_0 = F^0(S \times S) \supseteq \sim_1 = F^1(S \times S) \supseteq \ldots \supseteq \sim_i = F^i(S \times S) \supseteq \ldots \supseteq \sim_\omega,$$

with relation $\sim_\omega$ as its limit. Interestingly, this limit coincides with team bisimilarity $\sim$, as proved below. Some auxiliary lemmata are needed.

Lemma 2 *For each BPP net $N = (S, A, T)$, it holds that there exists an index k such that $\sim_k = \sim_{k+1} = \ldots = \sim_\omega$, i.e., the chain is initially decreasing, but becomes constant from index k onwards.*

Proof Since the BPP net is finite, the initial relation $\sim_0 = S \times S$ is finite as well. Therefore, it is not possible that $\sim_i = F^i(S \times S) \supset \sim_{i+1}$ for all $i \in \mathbb{N}$. This means that there exists an index k such that $\sim_k = F^k(S \times S) = F(F^k(S \times S)) = \sim_{k+1}$. Hence, $\sim_k = \sim_j$ for each $j > k$, and so $\sim_k = \sim_\omega$. □

Theorem 3 *For each BPP net $N = (S, A, T)$, it holds that $\sim = \sim_\omega$.*

Proof We prove first that $\sim \subseteq \sim_i$ for all i by induction on i. Indeed, $\sim \subseteq \sim_0$ (the universal relation); moreover, assuming $\sim \subseteq \sim_i$, by monotonicity of F and the fact that $\sim$ is a fixed point for F, we get $\sim = F(\sim) \subseteq F(\sim_i) = \sim_{i+1}$. Hence, $\sim \subseteq \sim_\omega$.

Now we prove that $\sim_\omega \subseteq \sim$, by showing that relation $\sim_\omega$ is a team bisimulation. Indeed, by Lemma 2, we know that $\sim_\omega = \sim_k$ for some $k \in \mathbb{N}$. As $\sim_{k+1} = F(\sim_k) = \sim_k$, we have that $\sim_k$, thanks to Definition 10, satisfies Definition 9, so that, by Proposition 11(2), $\sim_k$ is a team bisimulation. □

The proof of the theorem above can be also given by proving that the functional F is actually *cocontinuous* [57] for BPP nets, implying that

$$F(\sim_\omega) = F(\bigcap_{i\in\mathbb{N}} \sim_i) = \bigcap_{i\in\mathbb{N}} F(\sim_i) = \bigcap_{i\in\mathbb{N}} \sim_{i+1} = \sim_\omega$$

Therefore, $\sim_\omega$ is a fixed point of F. Actually, $\sim_\omega$ is the greatest fixed point of F, and so, by Theorem 2, it coincides with $\sim$. In fact, by the Cocontinuity Theorem [57], the greatest fixed point of F is computed by $\bigcap_{i\in\mathbb{N}} F^i(S\times S)$, which, by Proposition 12(2), coincides with $\bigcap_{i\in\mathbb{N}} \sim_i$, and so with $\sim_\omega$.

The characterization of $\sim$ as the limit of the non-increasing chain of relations $\sim_i$ offers an easy algorithm to compute team bisimilarity $\sim$ over BPP nets: just start from the universal relation $R_0 = S\times S$ and then iteratively apply functional F; when $R_{i+1} = F(R_i) = R_i$, then stop and take R_i as the team bisimilarity relation. Of course, this algorithm always terminates by the argument in Lemma 2: since S is finite, we are sure that an index k exists such that $R_{k+1} = F(R_k) = R_k$. The algorithm can be expressed by the following sequence of instructions.

$R_0 := S\times S$**;**
$R_1 := F(R_0)$**;**
while $R_0 \neq R_1$ **do** $\{R_0 := R_1$**;**$R_1 := F(R_0)\}$**;**
return R_0

As an example, take the BPP net in Figure 2(b). The set of places in $S = \{s_4, s_5, s_6, s_7, s_8, s_9\}$. The initial relation in $R_0 = S\times S$. Then, $R_1 = F(R_0)$ is $\mathscr{I} \cup R \cup R^{-1}$, where $\mathscr{I}$ is the identity relation and $R = \{(s_5,s_6),(s_5,s_7),(s_6,s_7),(s_8,s_9)\}$. Then, if we apply functional F to R_1 we get the same relation, i.e., R_1 is the greatest fixed point.

The algorithm above is not optimal. It is well-known that the optimal algorithm for computing bisimulation equivalence over a finite-state LTS with n states and m transitions has $O(m\cdot \log n)$ time complexity [54]; this very same partition refinement algorithm can be easily adapted also for team bisimilarity over BPP nets; it is enough to start from an initial partition composed of two blocks: S and $\{\theta\}$, and to consider the little additional cost due to the fact that the reached markings are to be related by the additive closure of the current partition; this extra cost is related to the size of the post-set of the net transitions; if p is the size of the largest post-set of the net transitions (i.e., p is the least number such that $|t^\bullet| \leq p$, for all $t \in T$), then the time complexity is $O(m\cdot p^2 \cdot \log (n+1))$, where m is the number of the net transitions and n is the number of the net places.

3.5 Minimizing Nets

In the theory of deterministic finite automata (DFAs, for short; see, e.g., [38]), two language equivalent states can be merged to obtain a language equivalent, smaller DFA; in fact, it is possible to get the least DFA, whose states are language equivalence classes of the states of the original DFA. Similarly, in the theory of labeled transition systems (LTSs, for short), two bisimilar states can be merged to get a smaller, behaviorally equivalent LTS; in fact, it is possible to get the least LTS, whose states are bisimulation equivalence classes of the states of the original LTS (see, e.g., [31]).

The situation is not very different for BPP nets, where the team bisimulation equivalence relation $\sim$ over places can be used to obtain minimized nets. Also in this case, two team bisimilar places can be safely merged; hence, given a BPP net N, we can obtain its behaviorally equivalent, reduced BPP net N', whose places are team bisimulation equivalence classes of places of N.

Definition 11 (Reduced net) Let $N = (S, A, T)$ be a BPP net and let $\sim$ be the team bisimulation equivalence relation over its places. The *reduced* net $N' = (S', A, T')$ is defined as follows:

- $S' = \{[s] \mid s \in S\}$, where $[s] = \{s' \in S \mid s \sim s'\}$;
- $T' = \{([s], a, [m]) \mid (s, a, m) \in T\}$,

where $[m]$ is defined as follows: $[\theta] = \theta$ and $[m_1 \oplus m_2] = [m_1] \oplus [m_2]$. If the net N has initial marking $m_0 = k_1 \cdot s_1 \oplus \ldots \oplus k_n \cdot s_n$, then N' as initial marking $[m_0] = k_1 \cdot [s_1] \oplus \ldots \oplus k_n \cdot [s_n]$. □

Lemma 3 *Let $N = (S, A, T)$ be a BPP net and let $N' = (S', A, T')$ be its reduced net w.r.t. $\sim$. The following holds: $m_1 \sim^{\oplus} m_2$ if and only if $[m_1] = [m_2]$.*

Proof By induction on the size of m_1. If $|m_1| = 0$, then $m_1 = \theta = m_2$ and $[m_1] = \theta = [m_2]$, as required. Otherwise, if $m_1 = s_1 \oplus m_1'$ and $m_1 \sim^{\oplus} m_2$, then, by Definition 5, there exist s_2 and m_2' such that $m_2 = s_2 \oplus m_2'$, $s_1 \sim s_2$ and $m_1' \sim^{\oplus} m_2'$. By Definition 11, we have that $[s_1] = [s_2]$ and, by induction, $[m_1'] = [m_2']$, so that the thesis $[m_1] = [m_2]$ follows trivially. Conversely, if $m_1 = s_1 \oplus m_1'$ and $[m_1] = [m_2]$, then $[m_1] = [s_1] \oplus [m_1']$ and there exist s_2 and m_2' such that $m_2 = s_2 \oplus m_2'$, $[s_1] = [s_2]$ and $[m_1'] = [m_2']$. Hence, by Definition 11, we have that $s_1 \sim s_2$ and, by induction, that $m_1' \sim^{\oplus} m_2'$, so that, by Definition 5, the thesis $m_1 \sim^{\oplus} m_2$ follows. □

Lemma 4 *Let $N = (S, A, T)$ be a BPP net and let $N' = (S', A, T')$ be its reduced net w.r.t. $\sim$. Relation $R = \{(s, [s]) \mid s \in S\}$ is a team bisimulation.*

Proof If $s \xrightarrow{a} m$, then also $[s] \xrightarrow{a} [m]$ by definition of T' with $(m, [m]) \in R^{\oplus}$, as required. The case when $[s]$ moves first is slightly more complex for the freedom in choosing the representative in an equivalence class. Transition $[s] \xrightarrow{a} [m]$ is possible, by Definition of T', if there exist $s' \in [s]$ and $m' \sim^{\oplus} m$ such that $s' \xrightarrow{a} m'$; as $s \sim s'$, there must exist a transition $s \xrightarrow{a} m''$ such that $m' \sim^{\oplus} m''$. Summing up, if $[s] \xrightarrow{a} [m]$, then $s \xrightarrow{a} m''$, with $(m'', [m'']) \in R^{\oplus}$, as required, because $[m] = [m'] = [m'']$ by Lemma 3. □

Theorem 4 *Let $N = (S, A, T)$ be a BPP net and let $N' = (S', A, T')$ be its reduced net w.r.t. $\sim$. For any $m \in \mathscr{M}(S)$, we have that $m \sim^{\oplus} [m]$.*

Proof By induction on the size of m. If $m = \theta$, then $[m] = \theta$ and the thesis follows trivially. If $m = s \oplus m'$, then $[m] = [s] \oplus [m']$; by Lemma 4, $s \sim [s]$ and, by induction, $m' \sim^{\oplus} [m']$; therefore, by the rule in Definition 5, $m \sim^{\oplus} [m]$. □

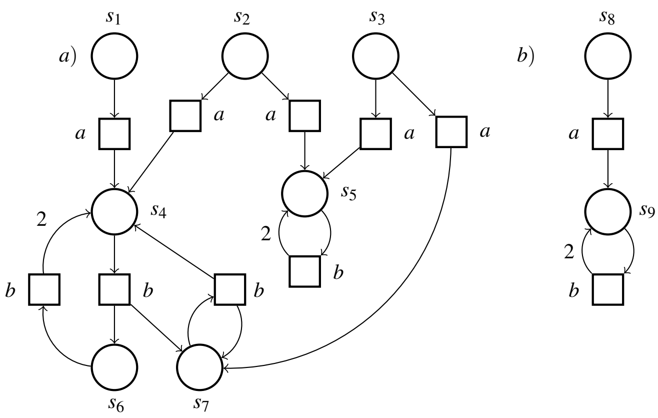

Fig. 4 A BPP net in (a) and its reduced net in (b)

As a consequence of this theorem, we would like to point out that the reduced net w.r.t. $\sim$ is indeed the *least* net offering the same team bisimilar behavior as the original net: no further fusion of places can be done, as there are not two places in the reduced net which are team bisimilar. As a consequence, in the reduced net, if two markings m_1 and m_2 are different, then they are not team bisimilar (*cf.* Lemma 3). Indeed, the reduced net is minimized.

Example 10 Let us consider the semi-counter nets in Figure 1, which are considered unmarked in this example. It is easy to see that the equivalence classes of $\sim$ are $\{s_1, s_3, s_4\}$ and $\{s_2, s_5, s_6\}$. Hence, the reduced net has just two places and is isomorphic to the unmarked net in (a). If we consider the current marking $s_4 \oplus s_5 \oplus s_6$ for the net in (b), then the corresponding team bisimilar marking in the reduced net in (a) is $s_1 \oplus 2 \cdot s_2$. □

Example 11 Consider the net in Figure 2(b), discussed in Examples 5 and 7. By fusing together the team bisimilar places, we get a net which is isomorphic to that in Figure 2(a). □

Example 12 Consider the net in Figure 4(a). It is not difficult to realize that the equivalence classes of $\sim$ are $\{s_1, s_2, s_3\}$ and $\{s_4, s_5, s_6, s_7\}$. Hence, the reduced net is isomorphic to the net in Figure 4(b). If the initial marking of the net in (a) is $s_1 \oplus s_3$, then the initial marking of the reduced net is $2 \cdot s_8$. Of course, the two initial markings are team bisimulation equivalent. □

4 Modal Logic

In this section we propose a new modal logic, called TML (acronym of *Team Modal Logic*), which extends conservatively Hennessy-Milner Logic (HML) [37, 3]. We will prove that model checking is coherent with equivalence checking: two markings are team bisimilar if and only if they satisfy the same TML formulae.

$[\![nn]\!] = S \qquad [\![vv]\!] = \{\theta\} \qquad [\![tt]\!] = \mathscr{M}(S) \qquad [\![\mathit{ff}]\!] = \emptyset$
$[\![F_1 \wedge F_2]\!] = [\![F_1]\!] \cap [\![F_2]\!] \qquad [\![F_1 \vee F_2]\!] = [\![F_1]\!] \cup [\![F_2]\!] \qquad [\![\neg F]\!] = \mathscr{M}(S) \setminus [\![F]\!]$
$[\![\langle a \rangle F]\!] = \{s \in S \mid \exists m. s \xrightarrow{a} m \text{ and } m \in [\![F]\!]\}$ $[\![[a]F]\!] = \{s \in S \mid \forall m(s \xrightarrow{a} m \text{ implies } m \in [\![F]\!])\}$
$[\![F_1 \otimes F_2]\!] = [\![F_1]\!] \otimes [\![F_2]\!]$ where $M_1 \otimes M_2 = \{m_1 \oplus m_2 \mid m_1 \in M_1, m_2 \in M_2\}$

Table 1 Denotational semantics

The TML *formulae* are generated from the finite set A of actions by the following abstract syntax:

$$F \ ::= \ nn \mid vv \mid tt \mid \mathit{ff} \mid F \wedge F \mid F \vee F \mid \neg F \mid \langle a \rangle F \mid [a]F \mid F \otimes F$$

where a is any action in A, nn and vv denote two atomic propositions (which are not tt, for *true*, and ff, for *false*), $\wedge$ is the operator of logical conjunction, $\vee$ is disjunction, $\langle a \rangle F$ denotes *possibility* (it is possible to do a and then reach a marking where F holds), $[a]F$ denotes *necessity* (by doing a, only markings where F holds can be reached), $\neg$ is logical negation and, finally, $\otimes$ is the operator of parallel composition of formulae.

We denote by $\mathscr{F}_A$ the set of all TML formulae, built from the actions in A. We sometimes use some useful abbreviations: if $B = \{a_1, a_2, \ldots, a_k\} \subseteq A$, $k \geq 1$, then $\langle B \rangle F$ stands for $\langle a_1 \rangle F \vee \langle a_2 \rangle F \vee \ldots \vee \langle a_k \rangle F$, and $[B]F$ stands for $[a_1]F \wedge [a_2]F \wedge \ldots \wedge [a_k]F$. In case $B = \emptyset$, $\langle B \rangle F = \mathit{ff}$ and $[B]F = nn$; the reason for the latter will be clear in the following.

The semantics of a formula F is the set of markings that satisfy it; hence, the semantic function is parametrized with respect to some given BPP net $N = (S, A, T)$. Let $[\![-]\!] : \mathscr{F}_A \to \wp(\mathscr{M}(S))$ be the denotational semantics function, defined in Table 1.

Definition 12 (TML satisfaction relation) We say that a marking m *satisfies* a formula F, written $m \models F$, if $m \in [\![F]\!]$. □

The semantics of any formula F is a set of markings. The semantics of nn is S: all the places satisfy nn. The semantics of vv is $\{\theta\}$: only the empty marking satisfies vv. The semantics of tt is $\mathscr{M}(S)$: every marking satisfies tt. The semantics of ff is $\emptyset$: no marking satisfies ff.

The logical operator of conjunction $\_ \wedge \_$ is interpreted as intersection $\_ \cap \_$ of the set of markings satisfied by the two subformulae; symmetrically, disjunction is interpreted as set union. The semantics of $\neg F$ is the set of all the markings that do not satisfy F, i.e., the complement of $[\![F]\!]$.

The semantics of $\langle a \rangle F$ is the set of all the places that can perform a and, in doing so, reach a marking that satisfies F. For instance, the formula $\langle a \rangle nn$ is satisfied by any place able to perform a, reaching another place.

The semantics of $[a]F$ is the set of all the places that, by performing a, can only reach a marking satisfying F. Note that a place s, which is unable to perform a altogether, satisfies $[a]F$, for any F, because the universal quantification in the semantic definition of $[a]\_$ is vacuously satisfied. For instance, the formula $[a]nn$ is satisfied by any place that, by performing a, can only reach markings composed of a single place. To explain why the auxiliary notation $[B]F$, when $B = \emptyset$, is to be interpreted as nn, we have to point out that the semantics of a box formula $[a]F$ is a set of places (not markings, in general), and so all the places, and only the places, satisfy this formula.

The semantics of $F_1 \otimes F_2$ is the set of markings of the form $m_1 \oplus m_2$ such that $m_1 \models F_1$ and $m_2 \models F_2$.

Example 13 Let us consider the net in Figure 2. It is not difficult to realize that formula $F_1 = [b]\langle c \rangle nn$ is such that all places, and in particular s_2, s_5, s_6 and s_7, satisfy F_1. Moreover, $F_2 = [a](F_1 \otimes F_1)$ is such that, e.g., $s_1 \models F_2$ and $s_4 \models F_2$. Finally, formula $G = F_2 \otimes F_1$ is such that, e.g., $s_1 \oplus s_2 \models G$ and $s_4 \oplus s_7 \models G$. □

We are now ready to prove the coherence theorem: two markings are team bisimilar if and only if they satisfy the same TML formulae. The proof is inspired by the analogous proof in [37,3] for HML and LTS bisimulation.

Proposition 13 *Let $N = (S, A, T)$ be a BPP net. If $m_1 \sim^{\oplus} m_2$, then m_1 and m_2 satisfy the same TML formulae, i.e., $\{F_1 \in \mathscr{F}_A \mid m_1 \models F_1\} = \{F_2 \in \mathscr{F}_A \mid m_2 \models F_2\}$.*

Proof Let us assume that $m_1 \sim^{\oplus} m_2$. We will prove that, for any $F \in \mathscr{F}_A$, if $m_1 \models F$ then also $m_2 \models F$. This is enough because, by a symmetric argument, we can also prove that if $m_2 \models F$ then also $m_1 \models F$, and so m_1 and m_2 satisfy the same TML formulae.

The proof is by induction on the structure of F, where the first four cases are the base cases of the induction.

- $F = nn$: if $m_1 \models nn$, then m_1 is a place; since two team bisimilar markings have the same size, also m_2 is a single place and so also $m_2 \models nn$.
- $F = vv$: if $m_1 \models vv$, then $m_1 = \theta$; since two team bisimilar markings have the same size, also $m_2 = \theta$ and so $m_2 \models vv$.
- $F = tt$: since all the markings satisfy true, also $m_2 \models tt$.
- $F = ff$: since no marking satisfies false, $m_1 \not\models ff$ and also $m_2 \not\models ff$.
- $F = F_1 \wedge F_2$: since $m_1 \models F_1 \wedge F_2$, it follows that $m_1 \models F_1$ and $m_1 \models F_2$; by induction, we can assume that also $m_2 \models F_1$ and $m_2 \models F_2$; hence, also $m_2 \models F_1 \wedge F_2$, as required.
- $F = F_1 \vee F_2$: since $m_1 \models F_1 \vee F_2$, it follows that $m_1 \models F_1$ or $m_1 \models F_2$; by induction, we can assume that also $m_2 \models F_1$ or $m_2 \models F_2$; hence, also $m_2 \models F_1 \vee F_2$, as required.
- $F = \langle a \rangle G$: since $m_1 \models \langle a \rangle G$, then $m_1 = s_1$ and there exists a marking m_1' such that $s_1 \xrightarrow{a} m_1'$ and $m_1' \models G$; as $m_1 \sim^{\oplus} m_2$, also m_2 must be a single place, say s_2, and

so $s_1 \sim s_2$ holds. By definition of $\sim$, there exists a marking m'_2 such that $s_2 \xrightarrow{a} m'_2$ and $m'_1 \sim^{\oplus} m'_2$. Since $m'_1 \sim^{\oplus} m'_2$ and $m'_1 \models G$, we can apply induction (because G is a subformula) and conclude that also $m'_2 \models G$; hence, $m_2 = s_2 \models \langle a \rangle G$, as required.

- $F = [a]G$: since $m_1 \models [a]G$, then $m_1 = s_1$ and, for all m'_1 such that $s_1 \xrightarrow{a} m'_1$, it follows that $m'_1 \models G$. As $m_1 \sim^{\oplus} m_2$, also m_2 must be a single place, say s_2, and so $s_1 \sim s_2$ holds. Since $s_1 \sim s_2$, for each m'_2 such that $s_2 \xrightarrow{a} m'_2$, there exists m'_1 such that $s_1 \xrightarrow{a} m'_1$ such that $m'_1 \sim^{\oplus} m'_2$. Now, since $m'_1 \sim^{\oplus} m'_2$ and $m'_1 \models G$, by induction, it follows also that $m'_2 \models G$. Hence, for all m'_2 such that $s_2 \xrightarrow{a} m'_2$, we have that $m'_2 \models G$; therefore, $m_2 = s_2 \models [a]G$, as required.
- $F = \neg F'$: since $m_1 \models \neg F'$, it follows that $m_1 \not\models F'$. By induction, as F' is a subformula, if m_1 does not satisfy F', then also m_2 does not satisfy F', and so $m_2 \models \neg F'$, as required.
- $F = F_1 \otimes F_2$: $m_1 \models F_1 \otimes F_2$ only if there exists m'_1 and m''_1 such that $m_1 = m'_1 \oplus m''_1$, $m'_1 \models F_1$ and $m''_1 \models F_2$. As $m_1 \sim^{\oplus} m_2$, there exists m'_2 and m''_2 such that $m_2 = m'_2 \oplus m''_2$ and $m'_1 \sim^{\oplus} m'_2$ and $m''_1 \sim^{\oplus} m''_2$. By induction, $m'_2 \models F_1$ and $m''_2 \models F_2$; therefore, also $m_2 \models F_1 \otimes F_2$, as required.

As no other cases are possible, the proof is complete. □

Lemma 5 *Let $N = (S, A, T)$ be a BPP net. If s_1 and s_2 satisfy the same TML formulae, i.e., $\{F_1 \in \mathscr{F}_A \mid s_1 \models F_1\} = \{F_2 \in \mathscr{F}_A \mid s_2 \models F_2\}$, then $s_1 \sim s_2$.*

Proof We want to prove that $R = \{(s, s') \mid s \text{ and } s' \text{ satisfy the same TML formulae}\}$ is a team bisimulation, hence proving that two places that satisfy the same formulae are team bisimilar.

Assume $(s_1, s_2) \in R$ and $s_1 \xrightarrow{a} m_1$. We will prove that there exists some m_2 such that $s_2 \xrightarrow{a} m_2$ and $(m_1, m_2) \in R^{\oplus}$. Since R is symmetric, this is enough for proving that R is a team bisimulation.

Assume, towards a contradiction, that there exists no m_2 such that $s_2 \xrightarrow{a} m_2$ and $(m_1, m_2) \in R^{\oplus}$. Since the net is finite, the set $\{m \in \mathscr{M}(S) \mid s_2 \xrightarrow{a} m\}$ is finite; let us denote such a set with $\{m'_1, m'_2, \ldots, m'_k\}$, with $k \geq 0$.

By assumption, for $j = 1, \ldots, k$, none of the m'_j is such that $(m_1, m'_j) \in R^{\oplus}$. Therefore, one of the following two cases is possible:

(a) $|m_1| \neq |m'_j|$ (i.e., the two markings are composed of a different number of tokens), or, by looking at Algorithm 1 (which is applicable as R is an equivalence relation),
(b) $|m_1| = |m'_j|$ but there is a place p_j in the residual of m_1 that has no R-match in the residual of m'_j.

In case (a), if $|m_1| = d \neq |m'_j|$, then the TML formula

$$nn^d = \underbrace{nn \otimes \ldots \otimes nn}_{d \text{ times}}$$

is such that $m_1 \models nn^d$, while $m'_j \not\models nn^d$.

In case (b), assume that $dom(m'_j)$ has $h_j \geq 1$ places which are not R-related to p_j, namely $\{s^j_1, \ldots, s^j_{h_j}\} \subseteq dom(m'_j)$. Hence, for each $s^j_i \in m'_j$, for $i = 1, \ldots, h_j$, there is a TML formula F^j_i such that $p_j \models F^j_i$ and $s^j_i \not\models F^j_i$. Let m' be the marking composed of all the elements s in m_1 such that $(s, p_j) \in R$; to be precise, any $s \in m'$ is such that $(s, p_j) \in R$, and any $s \in m_1 \ominus m'$ is such that $(s, p_j) \notin R$. Then,

$$m' \models G_j^{l_j} = \underbrace{G_j \otimes \ldots \otimes G_j}_{l_j \text{ times}},$$

where $G_j = F_1^j \wedge \ldots \wedge F_{h_j}^j$ and $l_j = |m'|$. By Definition 12, also $m_1 \models G_j^{l_j} \otimes nn^{d-l_j}$, where $d = |m_1|$. On the contrary, $m'_j \not\models G_j^{l_j} \otimes nn^{d-l_j}$ because in m'_j there are less than l_j elements which are R-related to p_j and any other s_i^j is such that $s_i^j \not\models F_i^j$ and so $s_i^j \not\models G_j$.

Finally, take the formula $G = \langle a \rangle (G'_1 \wedge G'_2 \wedge \ldots \wedge G'_k)$ where, for $j = 1, \ldots, k$, $G'_j = nn^d$ if m_1 and m'_j have different size, while G'_j is the formula $G_j^{l_j} \otimes nn^{d-l_j}$ above, otherwise. It is easy to see that $s_1 \models G$, because $m_1 \models G'_j$ for $j = 1, \ldots, k$; on the contrary, $s_2 \not\models G$, because, for $j = 1, \ldots, k$, $m'_j \not\models G'_j$, hence contradicting the previous assumption that s_1 and s_2 satisfy the same formulae. (In case $k = 0$, $G = \langle a \rangle tt$.) □

Proposition 14 *Let $N = (S, A, T)$ be a BPP net. If m_1 and m_2 satisfy the same TML formulae, i.e., $\{F_1 \in \mathscr{F}_A \mid m_1 \models F_1\} = \{F_2 \in \mathscr{F}_A \mid m_2 \models F_2\}$, then $m_1 \sim^\oplus m_2$.*

Proof We actually prove the contranominal: if two markings are not related by $\sim^\oplus$, then they do not satisfy the same TML formulae. Two markings are not related by $\sim^\oplus$ if they have not the same size or if Algorithm 1 fails. In the former case, assume that $|m_1| = k > |m_2|$ for some $k \geq 1$. Then, the TML formula

$$nn^k = \underbrace{nn \otimes \ldots \otimes nn}_{k \text{ times}}$$

is such that $m_1 \models nn^k$, while $m_2 \not\models nn^k$, and so m_1 and m_2 do not satisfy the same TML formulae.

In the latter case, looking at Algorithm 1, let s be the element of the residual of m_1 that has no team bisimilar match in the residual of m_2. Assume that $dom(m_2)$ has $h \geq 1$ places which are not team bisimilar to s, namely $\{s'_1, \ldots, s'_h\} \subseteq dom(m_2)$. Hence, by (the contranominal of) Lemma 5, for each $s'_j \in dom(m_2)$, there is a TML formula F_j such that $s \models F_j$ and $s'_j \not\models F_j$, for $j = 1, \ldots, h$. Let m'_1 be the marking composed of all the elements s' in m_1 such that $s' \sim s$; to be precise, any $s' \in m'_1$ is such that $s' \sim s$, and any $s' \in m_1 \ominus m'_1$ is such that $s' \nsim s$. Then,

$$m'_1 \models G^l = \underbrace{G \otimes \ldots \otimes G}_{l \text{ times}},$$

where $G = F_1 \wedge \ldots \wedge F_h$ and $l = |m'_1|$. By Definition 12, also $m_1 \models G^l \otimes nn^{k-l}$, with $k = |m_1|$; on the contrary, $m_2 \not\models G^l \otimes nn^{k-l}$ because in m_2 there are less than l elements which are team bisimilar to s and any other s'_j is such that $s'_j \not\models F_j$ and so $s'_j \not\models G$. In conclusion, m_1 and m_2 do not satisfy the same TML formulae. □

Theorem 5 (Coherence) *Let $N = (S, A, T)$ be a BPP net. It holds that $m_1 \sim^\oplus m_2$ if and only if $\{F_1 \in \mathscr{F}_A \mid m_1 \models F_1\} = \{F_2 \in \mathscr{F}_A \mid m_2 \models F_2\}$.*

Proof Direct consequence of Propositions 13 and 14. □

$dec(\mathbf{0}) = \theta$	$dec(\mu.p) = \{\mu.p\}$
$dec(p+p') = \{p+p'\}$	$dec(C) = \{C\}$
$dec(p\,|\,p') = dec(p) \oplus dec(p')$	

Table 2 Decomposition function

5 The BPP Process Algebra: Syntax and Net Semantics

Now we define a process algebra that truly represents BPP nets. It is called BPP as well (where BPP is the acronym of *Basic Parallel Processes*) and was originally studied in [16]. BPP is a simple CCS [47,31] subcalculus (without the restriction operator) whose processes cannot communicate. As mentioned in the Introduction, we actually study a variant which requires guarded summation and guarded recursion.

5.1 Syntax

Let *Act* be a finite set of *actions*, ranged over by μ, and let $\mathscr{C}$ be a finite set of *constants*, disjoint from *Act*, ranged over by $A, B, C, \ldots$. The size of the sets *Act* and $\mathscr{C}$ is not important: we assume that they can be chosen as large as needed. The BPP *terms* are generated from actions and constants by the following abstract syntax:

$$\begin{array}{lll} s ::= \mathbf{0} \mid \mu.p \mid s+s & & \textit{guarded processes} \\ q ::= s \mid \; C & & \textit{sequential processes} \\ p ::= q \mid p\,|\,p & & \textit{parallel processes} \end{array}$$

where $\mathbf{0}$ is the empty process, $\mu.p$ is a process where action $\mu \in Act$ prefixes the residual p ($\mu.-$ is the *action prefixing* operator), $s_1 + s_2$ denotes the alternative composition of s_1 and s_2 ($- + -$ is the *choice* operator), $p_1\,|\,p_2$ denotes the asynchronous parallel composition of p_1 and p_2 and C is a constant. A constant C may be equipped with a definition, but this must be a guarded process, i.e., in the syntactic category s: $C \doteq s$. A term p is a BPP *process* if each constant in $Const(p)$ (the set of constants used by p; see [32] for details) is equipped with a defining equation (in syntactic category s). The set of BPP processes is denoted by $\mathscr{P}_{BPP}$, the set of its sequential processes, i.e., of the processes in syntactic category q, by $\mathscr{P}_{BPP}^{seq}$, while the set of its guarded processes, i.e., of the processes in syntactic category s, by $\mathscr{P}_{BPP}^{grd}$,

5.2 Net Semantics

The net for the process algebra BPP (with guarded summation and guarded recursion), originally outlined in [32], is such that the set of places S_{BPP} is the set of the sequential BPP processes, without $\mathbf{0}$, i.e., $S_{BPP} = \mathscr{P}_{BPP}^{seq} \setminus \{\mathbf{0}\}$. The decomposition function $dec : \mathscr{P}_{BPP} \to \mathscr{M}(S_{BPP})$, mapping process terms to markings, is defined in Table 2. An easy induction proves that $dec(p)$ is a finite multiset of sequential processes for each $p \in \mathscr{P}_{BPP}$. Note that, if $C \doteq \mathbf{0}$, then $\theta = dec(\mathbf{0}) \neq dec(C) = \{C\}$.

$$
\begin{array}{rcll}
[\![\mathbf{0}]\!]_I & = & (\emptyset,\emptyset,\emptyset,\theta) & \\
[\![\mu.p]\!]_I & = & (S,A,T,\{\mu.p\}) & \text{given } [\![p]\!]_I = (S',A',T',dec(p)) \text{ and where} \\
 & & & S = \{\mu.p\} \cup S',\ A = \{\mu\} \cup A',\ T = \{(\{\mu.p\},\mu,dec(p))\} \cup T' \\
[\![p_1+p_2]\!]_I & = & (S,A,T,\{p_1+p_2\}) & \text{given } [\![p_i]\!]_I = (S_i,A_i,T_i,dec(p_i)) \text{ for } i=1,2, \text{ and where} \\
 & & & S = \{p_1+p_2\} \cup S'_1 \cup S'_2, \text{ with, for } i=1,2, \\
 & & & S'_i = \begin{cases} S_i & \exists t \in T_i \text{ such that } t^\bullet(p_i) > 0 \\ S_i \setminus \{p_i\} & \text{otherwise} \end{cases} \\
 & & & A = A_1 \cup A_2,\ T = T' \cup T'_1 \cup T'_2, \text{ with, for } i=1,2, \\
 & & & T'_i = \begin{cases} T_i & \exists t \in T_i . t^\bullet(p_i) > 0 \\ T_i \setminus \{t \in T_i \mid {}^\bullet t(p_i) > 0\} & \text{otherwise} \end{cases} \\
 & & & T' = \{(\{p_1+p_2\},\mu,m) \mid (\{p_i\},\mu,m) \in T_i, i=1,2\} \\
[\![C]\!]_I & = & (\{C\},\emptyset,\emptyset,\{C\}) & \text{if } C \in I \\
[\![C]\!]_I & = & (S,A,T,\{C\}) & \text{if } C \notin I, \text{ given } C \doteq p \text{ and } [\![p]\!]_{I\cup\{C\}} = (S',A',T',dec(p)) \\
 & & & A = A', S = \{C\} \cup S'', \text{ where} \\
 & & & S'' = \begin{cases} S' & \exists t \in T' . t^\bullet(p) > 0 \\ & \\ S' \setminus \{p\} & \text{otherwise} \end{cases} \\
 & & & T = \{(\{C\},\mu,m) \mid (\{p\},\mu,m) \in T'\} \cup T'' \text{ where} \\
 & & & T'' = \begin{cases} T' & \exists t \in T' . t^\bullet(p) > 0 \\ T' \setminus \{t \in T' \mid {}^\bullet t(p) > 0\} & \text{otherwise} \end{cases} \\
[\![p_1 \,|\, p_2]\!]_I & = & (S,A,T,m_0) & \text{given } [\![p_i]\!]_I = (S_i,A_i,T_i,m_i) \text{ for } i=1,2, \text{ and where} \\
 & & & S = S_1 \cup S_2,\ A = A_1 \cup A_2,\ T = T_1 \cup T_2,\ m_0 = m_1 \oplus m_2
\end{array}
$$

Table 3 Denotational net semantics

Now we provide a construction of the net system $[\![p]\!]_\emptyset$ associated with process p, which is compositional and denotational in style. The details of the construction are outlined in Table 3. The mapping is parametrized by a set of constants that have already been found while scanning p; such a set is initially empty and it is used to avoid looping on recursive constants. The definition is syntax driven and also the places of the constructed net are syntactic objects, i.e., BPP sequential process terms. For instance, the net system $[\![a.\mathbf{0}]\!]_\emptyset$ is a net composed of one single marked place, namely process $a.\mathbf{0}$, and one single transition $(\{a.\mathbf{0}\}, a, \theta)$. A bit of care is needed in the rule for choice: in order to include only strictly necessary places and transitions, the initial place p_1 (or p_2) of the subnet $[\![p_1]\!]_I$ (or $[\![p_2]\!]_I$) is to be kept in the net for $p_1 + p_2$ only if there exists a transition reaching place p_1 (or p_2) in $[\![p_1]\!]_I$ (or $[\![p_2]\!]_I$), otherwise p_1 (or p_2) can be safely removed in the new net. Similarly, for the rule for constants.

Example 14 Consider the BPP process SC for a semi-counter, whose definition is

$SC \doteq inc.(SC \,|\, dec.\mathbf{0})$.

We have that

$[\![SC]\!]_{\{SC\}} = (\{SC\}, \emptyset, \emptyset, \{SC\})$, and

$[\![dec.\mathbf{0}]\!]_{\{SC\}} = (\{dec.\mathbf{0}\}, \{dec\}, \{(\{dec.\mathbf{0}\}, dec, \theta)\}, \{dec.\mathbf{0}\})$.

Therefore, the net $[\![SC\,|\,dec.\mathbf{0}]\!]_{\{SC\}}$ is
$(\{SC, dec.\mathbf{0}\}, \{dec\}, \{(\{dec.\mathbf{0}\}, dec, \theta)\}, \{SC, dec.\mathbf{0}\})$.
The net $[\![inc.(SC\,|\,dec.\mathbf{0})]\!]_{\{SC\}}$ is
$(\{inc.(SC\,|\,dec.\mathbf{0}), SC, dec.\mathbf{0}\}, \{inc, dec\}, \{(\{inc.(SC\,|\,dec.\mathbf{0})\}, inc, \{SC, dec.\mathbf{0}\}),$
$(\{dec.\mathbf{0}\}, dec, \theta)\}, \{inc.(SC\,|\,dec.\mathbf{0})\})$.
Finally, the net $[\![SC]\!]_{\emptyset}$ is
$(\{SC, dec.\mathbf{0}\}, \{inc, dec\}, \{(\{SC\}, inc, \{SC, dec.\mathbf{0}\}), (\{dec.\mathbf{0}\}, dec, \theta)\}, \{SC\})$,
which is (isomorphic to) the net in Figure 1(a), where s_1 is SC and s_2 is $dec.\mathbf{0}$. $\square$

We now list some properties of the semantics, whose proofs are in [32], which state that the process algebra BPP (with guarded summation and guarded recursion) really represents the class of BPP nets.

Theorem 6 (Only BPP nets) *For each BPP process p, $[\![p]\!]_{\emptyset}$ is a BPP net.* $\square$

Definition 13 (Translating BPP Nets into BPP Terms) Let $N(m_0) = (S, A, T, m_0)$ — with $S = \{s_1, \dots, s_n\}$, $A \subseteq Act$, $T = \{t_1, \dots, t_k\}$, and $l(t_j) = \mu_j$ — be a BPP net. Function $\mathscr{T}_{BPP}(-)$, from BPP nets to BPP processes, is defined as

$$\mathscr{T}_{BPP}(N(m_0)) = \underbrace{C_1 | \cdots | C_1}_{m_0(s_1)} | \cdots | \underbrace{C_n | \cdots | C_n}_{m_0(s_n)}$$

where each C_i is equipped with a defining equation $C_i \doteq c_i^1 + \cdots + c_i^k$ (with $C_i \doteq \mathbf{0}$ if $k = 0$), and each summand c_i^j, for $j = 1, \dots, k$, is equal to

- $\mathbf{0}$, if $s_i \notin {}^\bullet t_j$;
- $\mu_j.\Pi_j$, if ${}^\bullet t_j = \{s_i\}$, where process Π_j is $\underbrace{C_1 | \cdots | C_1}_{t_j^\bullet(s_1)} | \cdots | \underbrace{C_n | \cdots | C_n}_{t_j^\bullet(s_n)}$, meaning that $\Pi_j = \mathbf{0}$ if $t_j^\bullet = \emptyset$. $\square$

Example 15 Consider the net in Figure 1(b), whose set of places is $\{s_3, s_4, s_5, s_6\}$. According to Definition 13, the associated BPP term is $p = C_4 \,|\, C_5 \,|\, C_6$, with the following constant definitions:
$C_3 \doteq inc.(C_4 \,|\, C_5) + \mathbf{0} + \mathbf{0} + \mathbf{0}$
$C_4 \doteq \mathbf{0} + inc.(C_3 \,|\, C_6) + \mathbf{0} + \mathbf{0}$
$C_5 \doteq \mathbf{0} + \mathbf{0} + dec.\mathbf{0} + \mathbf{0}$
$C_6 \doteq \mathbf{0} + \mathbf{0} + \mathbf{0} + dec.\mathbf{0}$.
Of course, the semantics associates to p a net isomorphic to the net in Figure 1(b). $\square$

Theorem 7 (All BPP nets: Representability Theorem) *Let $N(m_0) = (S, A, T, m_0)$ be a dynamically reduced BPP net such that $A \subseteq Act$, and let $p = \mathscr{T}_{BPP}(N(m_0))$. Then, $[\![p]\!]_{\emptyset}$ is isomorphic to $N(m_0)$.* $\square$

6 Process Algebraic Properties

Thanks to the theorems of the previous section, we can transfer the definition of team bisimilarity from BPP nets to BPP process terms in a simple way.

Definition 14 Two BPP processes p and q are team bisimilar, denoted $p \sim^{\oplus} q$, if, by considering the (union of the) nets $[\![p]\!]_{\emptyset}$ and $[\![q]\!]_{\emptyset}$, we have that $dec(p) \sim^{\oplus} dec(q)$. $\square$

Of course, for sequential BPP processes, team bisimulation equivalence $\sim^{\oplus}$ coincides with team bisimilarity on places $\sim$.

Thanks to Definition 14, we can now perform the usual process algebraic study of a behavioral equivalence: to prove that it is a congruence for the operators of the BPP process algebra, to study its algebraic properties and, finally, to define a (possibly finite) sound and complete, axiomatization for it. These will be the subject of the next subsections.

6.1 Congruence

Now we show that team equivalence is a congruence for all the BPP operators.

Proposition 15 *For each $p, q \in \mathscr{P}_{BPP}^{grd}$, if $p \sim q$ (or $p = q = \mathbf{0}$), then*
$p + r \sim q + r$ *for all* $r \in \mathscr{P}_{BPP}^{grd}$.

Proof *Assume R is a team bisimulation such that $(p,q) \in R$ (or $R = \emptyset$ in case $p = q = \mathbf{0}$). It is very easy to check that, for each r, the relation $R_r = \{(p+r, q+r)\} \cup R \cup \mathscr{I}_r$ is a team bisimulation, where $\mathscr{I}_r = \{(r', r') \mid r' \in reach(r), r' \neq \theta\}$ if $r \neq \mathbf{0}$, otherwise $\mathscr{I}_r = \emptyset$.* $\square$

Proposition 16 *For each $p, q \in \mathscr{P}_{BPP}$, if $p \sim^{\oplus} q$, then $\mu.p \sim \mu.q$ for all $\mu \in Act$.*

Proof *Assume R is a team bisimulation such that $(dec(p), dec(q)) \in R^{\oplus}$. Consider, for each $\mu \in Act$, relation $R_{\mu} = \{(\mu.p, \mu.q)\} \cup R$. It is very easy to check that R_{μ} is a team bisimulation on places.* $\square$

Proposition 17 *For every $p, q, r \in \mathscr{P}_{BPP}$, if $p \sim^{\oplus} q$, then $p \,|\, r \sim^{\oplus} q \,|\, r$.*

Proof *By induction on the size of $dec(p)$. If $|dec(p)| = 0 = |dec(q)|$, then $p = \mathbf{0} = q$. Hence, $dec(p\,|\,r) = dec(r) = dec(q\,|\,r)$ and the thesis follows trivially, because $\sim^{\oplus}$ is reflexive. Since $dec(p) \sim^{\oplus} dec(q)$, if $|dec(p)| = k+1$ for some $k \geq 0$, then by Definition 5, there exist p_1, p_2, q_1, q_2 such that $p_1 \sim q_1$, $dec(p_2) \sim^{\oplus} dec(q_2)$, $dec(p) = p_1 \oplus dec(p_2)$ and $dec(q) = q_1 \oplus dec(q_2)$. Since $|dec(p_2)| = k = |dec(q_2)|$ and $p_2 \sim^{\oplus} q_2$, by induction, we have that $p_2\,|\,r \sim^{\oplus} q_2\,|\,r$. Since $p_1 \sim q_1$, by Definition 5, we have that $dec(p\,|\,r) = p_1 \oplus dec(p_2\,|\,r) \sim^{\oplus} q_1 \oplus dec(q_2\,|\,r) = dec(q\,|\,r)$. Hence, $p\,|\,r \sim^{\oplus} q\,|\,r$.* $\square$

Note that the symmetric cases $r + p \sim r + q$ and $r\,|\,p \sim^{\oplus} r\,|\,q$ are implied by the fact that the operators of choice and parallelism are commutative w.r.t. $\sim$ and $\sim^{\oplus}$, respectively (see Proposition 18 and 20).

Still there is one construct missing: recursion, defined over guarded terms only. Consider an extension of BPP where terms can be constructed using variables, such as $x, y, \ldots$ (which are in syntactic category q): this defines an "open" BPP, where terms may be not given a complete semantics. For instance, $p_1(x) = a.(b.\mathbf{0} + c.x)$ and $p_2(x) = a.(c.x + b.\mathbf{0})$ are open guarded BPP terms.

Definition 15 (Open BPP) Let $Var = \{x, y, z, \ldots\}$ be a finite set of variables. The BPP *open terms* are generated from actions, constants and variables by the following abstract syntax:

$$\begin{array}{lll} s ::= \mathbf{0} \mid \mu.p \mid s+s & & \textit{guarded open processes} \\ q ::= s \mid \ C \ \mid \ x & & \textit{sequential open processes} \\ p ::= q \mid p|p & & \textit{parallel open processes} \end{array}$$

where x is any variable taken from Var. The *open* net semantics for open BPP extends the net semantics in Table 3 with $[\![x]\!]_I = (\{x\}, \emptyset, \emptyset, \{x\})$, so that, e.g., the semantics of $a.x$ is the net $(\{a.x, x\}, \{a\}, \{(a.x, a, x)\}, a.x)$. □

However, a place x is not equivalent to $\mathbf{0} + \mathbf{0}$, even if both are stuck, because x is intended to be a placeholder for a sequential BPP term. Team bisimulation equivalence can be extended to open terms, by considering the variables in a proper way. For instance, $a.x + a.\mathbf{0}$ is team bisimilar to $a.x + a.\mathbf{0} + a.x$, because $R = \{(a.x + a.\mathbf{0}, a.x + a.\mathbf{0} + a.x), (x, x)\}$ is an *open* team bisimulation. However, more formally, we can extend team bisimilarity to open terms as follows.

An open term $p(x_1, \ldots, x_n)$ can be *closed* by means of a substitution as follows:

$$p(x_1, \ldots, x_n)\{r_1/x_1, \ldots, r_n/x_n\}$$

with the effect that each occurrence of the variable x_i (within p and the body of each constant in $Const(p)$) is replaced by the *closed* BPP sequential process r_i, for $i = 1, \ldots, n$. For instance, $p_1(x)\{d.\mathbf{0}/x\} = a.(b.\mathbf{0} + c.d.\mathbf{0})$.

A natural extension of team bisimulation equivalence $\sim$ over open *guarded* terms is as follows: $p(x_1, \ldots, x_n) \sim q(x_1, \ldots, x_n)$ if for all tuples of (closed) BPP sequential terms $(r_1, \ldots, r_n)$, $p(x_1, \ldots, x_n)\{r_1/x_1, \ldots, r_n/x_n\} \sim q(x_1, \ldots, x_n)\{r_1/x_1, \ldots, r_n/x_n\}$. E.g., it is easy to see that $p_1(x) \sim p_2(x)$. As a matter of fact, for all r, $p_1(x)\{r/x\} = a.(b.\mathbf{0} + c.r) \sim a.(c.r + b.\mathbf{0}) = p_2(x)\{r/x\}$, which can be easily proved by means of the algebraic properties (discussed in the next subsection) and the congruence ones of $\sim$.

For simplicity's sake, let us now restrict our attention to open guarded terms using a single undefined variable. We can *recursively close* an open term $p(x)$ by means of a recursively defined constant. For instance, $A \doteq p(x)\{A/x\}$. The resulting process constant A is a closed BPP sequential process. By saying that team bisimilarity is a congruence for recursion we mean the following: If $p(x) \sim q(x)$ and $A \doteq p(x)\{A/x\}$ and $B \doteq q(x)\{B/x\}$, then $A \sim B$. The following theorem states this fact.

Theorem 8 *Let p and q be two open guarded BPP terms, with one variable x at most. Let $A \doteq p\{A/x\}$, $B \doteq q\{B/x\}$ and $p \sim q$. Then $A \sim B$.*

Proof *Consider $R = \{(r\{A/x\}, r\{B/x\}) \mid r \in reach(p) \cup reach(q), r \neq \theta\}$. Note that when r is x, we get $(A,B) \in R$. The proof that R is a team bisimulation up to $\sim$ (cf. Definition 8) is not difficult. By symmetry, it is enough to prove that if $r\{A/x\} \xrightarrow{\mu} m_1$, then $r\{B/x\} \xrightarrow{\mu} m_2$ such that $m_1 (\sim R \sim)^{\oplus} m_2$. The proof proceeds by induction on the definition of the net for $r\{A/x\}$. We examine the possible shapes of r, which is an open sequential process.*

- *$r = \mu.r'$. In this case, $r\{A/x\} = \mu.r'\{A/x\} \xrightarrow{\mu} dec(r')\{A/x\}$. Similarly, $r\{B/x\} = \mu.r'\{B/x\} \xrightarrow{\mu} dec(r')\{B/x\}$, and $(dec(r')\{A/x\}, dec(r')\{B/x\}) \in R^{\oplus}$.*
- *$r = D$, with $D \doteq s$. So, $r\{A/x\} \doteq s\{A/x\}$ and $r\{B/x\} \doteq s\{B/x\}$. If $r\{A/x\} \xrightarrow{\mu} m_1$, then this is possible only if $s\{A/x\} \xrightarrow{\mu} m_1$. Since s is guarded, $s\{A/x\} \xrightarrow{\mu} m_1$ is possible only if $s \xrightarrow{\mu} m$ with $m_1 = m\{A/x\}$. Therefore, also $s\{B/x\} \xrightarrow{\mu} m\{B/x\}$ is derivable, and also $r\{B/x\} \xrightarrow{\mu} m\{B/x\}$, with $(m\{A/x\}, m\{B/x\}) \in R^{\oplus}$.*
- *$r = r_1 + r_2$. In this case, $r\{A/x\} = r_1\{A/x\} + r_2\{A/x\}$. A transition from $r\{A/x\}$, e.g., $r_1\{A/x\} + r_2\{A/x\} \xrightarrow{\mu} m_1$, is derivable only if $r_i\{A/x\} \xrightarrow{\mu} m_1$ for some $i = 1,2$. Without loss of generality, assume the transition is due to $r_1\{A/x\} \xrightarrow{\mu} m_1$. Since r_1 is guarded, transition $r_1\{A/x\} \xrightarrow{\mu} m_1$ is derivable because $r_1 \xrightarrow{\mu} m$, with $m_1 = m\{A/x\}$. Therefore, also $r_1\{B/x\} \xrightarrow{\mu} m\{B/x\}$ is derivable, as well $r\{B/x\} = r_1\{B/x\} + r_2\{B/x\} \xrightarrow{\mu} m\{B/x\}$, with $(m\{A/x\}, m\{B/x\}) \in R^{\oplus}$.*
- *$r = x$. Then, we have $r\{A/x\} = A$ and $r\{B/x\} = B$. We want to prove that for each $A \xrightarrow{\mu} m_1$, there exists m_2 such that $B \xrightarrow{\mu} m_2$ with $m_1 (\sim R \sim)^{\oplus} m_2$. By hypothesis, $A \doteq p\{A/x\}$, hence also $p\{A/x\} \xrightarrow{\mu} m_1$ is a transition in the net for $p\{A/x\}$; since p is guarded, $p\{A/x\} \xrightarrow{\mu} m_1$ is possible only if $p \xrightarrow{\mu} m$ with $m_1 = m\{A/x\}$. Therefore, also $p\{B/x\} \xrightarrow{\mu} m\{B/x\}$ is derivable. But we also have that $p \sim q$, so $p \xrightarrow{\mu} m$ can be matched by $q \xrightarrow{\mu} m'$ with $m \sim^{\oplus} m'$. Hence, $q\{B/x\} \xrightarrow{\mu} m'\{B/x\}$ is derivable with $m\{B/x\} \sim^{\oplus} m'\{B/x\}$. Since $B \doteq q\{B/x\}$, also $B \xrightarrow{\mu} m'\{B/x\}$ is a transition with $m_1 \sim^{\oplus} m\{A/x\} R^{\oplus} m\{B/x\} \sim^{\oplus} m'\{B/x\}$, as required.* □

The extension to the case of open terms with multiple undefined constants, e.g., $p(x_1, \ldots, x_n)$ can be obtained in a standard way [47,31].

6.2 Algebraic Laws

On guarded/sequential processes we have the following algebraic laws.

Proposition 18 (Laws of the choice operator) *For each $p, q, r \in \mathscr{P}_{BPP}^{grd}$, the following hold:*

$p + (q + r) \sim (p + q) + r$		*(associativity)*
$p + q \sim q + p$		*(commutativity)*
$p + \mathbf{0} \sim p$	*if* $p \neq \mathbf{0}$	*(identity)*
$p + p \sim p$	*if* $p \neq \mathbf{0}$	*(idempotency)*

Proof *For each law, it is enough to exhibit a suitable team bisimulation relation on places, where each place is actually a process term, according to the net semantics. E.g., for idempotency, for each p guarded ($p \neq \mathbf{0}$), take relation $R_p = \{(p+p,p)\} \cup \mathscr{I}_p$ where $\mathscr{I}_p = \{(q,q) \mid q \in reach(p), q \neq \theta\}$ is the identity relation. It is an easy exercise to check that R_p is a team bisimulation on the places of $[\![p+p]\!]_\emptyset$ and $[\![p]\!]_\emptyset$. In fact, if $p \xrightarrow{\mu} m$, then (according to the semantics for $p+p$) also $p+p \xrightarrow{\mu} m$ and $(m,m) \in \mathscr{I}_p^{\oplus}$, and so $(m,m) \in R_p^{\oplus}$. Symmetrically, if $p+p \xrightarrow{\mu} m$, then (according to the semantics for $p+p$) this is possible only if $p \xrightarrow{\mu} m$ is derivable and the condition $(m,m) \in R_p^{\oplus}$ is trivially satisfied. As a further example, for the associativity law, the candidate team bisimulation relation is $R_{(p,q,r)} = \{(p+(q+r),(p+q)+r)\} \cup \mathscr{I}_{(p,q,r)}$, where $\mathscr{I}_{(p,q,r)} = \{(v,v) \mid v \in reach(p+(q+r)), v \neq \theta\}$ is the identity relation.* □

Proposition 19 (Laws of the constant) *For each $p \in \mathscr{P}_{BPP}^{grd}$, and each $C \in \mathscr{C}$, the following hold:*

if $C \doteq \mathbf{0}$, then	$C \sim \mathbf{0}+\mathbf{0}$	*(stuck)*
if $C \doteq p$ and $p \neq \mathbf{0}$, then	$C \sim p$	*(unfolding)*
if $C \doteq p\{C/x\}$ and $q \sim p\{q/x\}$ then	$C \sim q$	*(folding)*

where, in the third law, p is actually open on x (while q is closed).

Proof *The stuck property is trivial: since the decomposition of a constant is a place, if the body is stuck, it corresponds to a stuck place, such as $\mathbf{0}+\mathbf{0}$.*

The required team bisimulation on places proving the unfolding property is $R_{C,p} = \{(C,p)\} \cup \mathscr{I}_C$, where $\mathscr{I}_C = \{(q,q) \mid q \in reach(C), q \neq \theta\}$ is the identity relation. In fact, if $C \xrightarrow{\mu} m$, then (according to the net semantics for $C \doteq p$) this means that also $p \xrightarrow{\mu} m$, with $(m,m) \in \mathscr{I}_C^{\oplus}$, and so $(m,m) \in R_{C,p}^{\oplus}$ as required. Symmetrically if p moves first.

For the folding property, observe that the statement is implied by the following: if $q_1 \sim p\{q_1/x\}$ and $q_2 \sim p\{q_2/x\}$ then $q_1 \sim q_2$. In fact, if we choose $q_1 = C$, then $C = q_1 \sim p\{q_1/x\} = p\{C/x\}$ (which holds by hypothesis, due to the unfolding property) and $C = q_1 \sim q_2$, which is the thesis. This statement can be proven by showing that

$R = \{(r\{q_1/x\}, r\{q_2/x\}) \mid r \in reach(p), r \neq \theta\}$

is a team bisimulation up to $\sim$ (cf. Definition 8). Clearly, when $r = x$, we have that $(q_1,q_2) \in R$. So, it remains to prove the team bisimulation (up to) conditions.

If $r\{q_1/x\} \xrightarrow{\mu} t$, then this can be due to one of the following:

- *$r \xrightarrow{\mu} m$ and so $t = m\{q_1/x\}$, where the substitution is applied element-wise to each place in m. In this case, also $r\{q_2/x\} \xrightarrow{\mu} m\{q_2/x\}$ is derivable such that $(m\{q_1/x\}, m\{q_2/x\}) \in R^{\oplus}$.*
- *$r = x$ and $q_1 \xrightarrow{\mu} m_1$, and so $t = m_1$. Since $q_1 \sim p\{q_1/x\}$ and p is guarded, we have that there exists m such that $p \xrightarrow{\mu} m$ and $p\{q_1/x\} \xrightarrow{\mu} m\{q_1/x\}$ with $m_1 \sim^{\oplus} m\{q_1/x\}$. Therefore, $p\{q_2/x\} \xrightarrow{\mu} m\{q_2/x\}$ is derivable, too. Since $q_2 \sim p\{q_2/x\}$, it follows that there exists a marking m_2 such that $q_2 \xrightarrow{\mu} m_2$ with $m_2 \sim^{\oplus} m\{q_2/x\}$. Summing up, if $x\{q_1/x\} = q_1 \xrightarrow{\mu} m_1$, then $x\{q_2/x\} = q_2 \xrightarrow{\mu} m_2$ such that $m_1 \sim^{\oplus}$*

$m\{q_1/x\}$, $(m\{q_1/x\}, m\{q_2/x\}) \in R^\oplus$ and, moreover, $m\{q_2/x\} \sim^\oplus m_2$, as required by the team bisimulation up to condition.

Simmetrically, if $r\{q_2/x\}$ moves first. Hence, R is a team bisimulation up to $\sim$. $\square$

Proposition 20 (Laws of the parallel operator) *For each $p, q, r \in \mathscr{P}_{BPP}$, the following hold:*

$$\begin{array}{rcll} p\,|\,(q\,|\,r) & \sim^\oplus & (p\,|\,q)\,|\,r & \textit{(associativity)} \\ p\,|\,q & \sim^\oplus & q\,|\,p & \textit{(commutativity)} \\ p\,|\,\mathbf{0} & \sim^\oplus & p & \textit{(identity)} \end{array}$$

Proof *To prove that each law is sound, it is enough to observe that the net for the process in the left-hand-side is exactly the same as the net for the process in the right-hand-side. For instance, $[\![p\,|\,q]\!]_\emptyset = [\![q\,|\,p]\!]_\emptyset$. In fact, $dec(p\,|\,q) = dec(p) \oplus dec(q) = dec(q) \oplus dec(p) = dec(q\,|\,p)$ and the resulting net is obtained by simply joining the net for p with the net for q. Therefore, the identity relation on places, which is a team bisimulation, is enough to prove that $dec(p\,|\,q) \sim^\oplus dec(q\,|\,p)$.* $\square$

6.3 Axiomatization

In this section we provide a sound and complete, finite axiomatization of team bisimulation equivalence over BPP. For simplicity's sake, the syntactic definition of open BPP (*cf.* Definition 15) is assumed here flattened, with only one syntactic category, but we require that each ground instantiation of an axiom must respect the syntactic definition of (closed) BPP given in Section 5.1. This means that we can write the axiom $x + (y + z) = (x + y) + z$ (these terms cannot be written in open BPP according to Definition 15), but it is invalid to instantiate it to $C + (a.\mathbf{0} + b.\mathbf{0}\,|\,\mathbf{0}) = (C + a.\mathbf{0}) + (b.\mathbf{0}\,|\,\mathbf{0})$ because these are not legal BPP processes (the constant C and the parallel process $b.\mathbf{0}\,|\,\mathbf{0}$ cannot be used as summands).

The set of axioms are outlined in Table 4. We call E the set of axioms $\{$**A1, A2, A3, A4, R1, R2, R3, P1, P2, P3**$\}$. By the notation $E \vdash p = q$ we mean that there exists an equational deduction proof of the equality $p = q$, by using the axioms in E. Besides the usual equational deduction rules of reflexivity, symmetry, transitivity, substitutivity and instantiation (see, e.g., [31]), in order to deal with constants we need also the following recursion congruence rule:

$$\frac{p = q \ \wedge\ A \doteq p\{A/x\} \ \wedge\ B \doteq q\{B/x\}}{A = B}$$

The axioms **A1-A4** are the usual axioms for choice where, however, **A3-A4** have the side condition $x \neq \mathbf{0}$; hence, it is not possible to prove $E \vdash \mathbf{0} + \mathbf{0} = \mathbf{0}$, as expected, because these two terms have a completely different semantics; in fact, no other *sequential* process p can be equated to $\mathbf{0}$. The *conditional* axioms (or inference rules) **R1-R3** are about process constants. Note that **R2** requires that p is not (equal to) $\mathbf{0}$ (condition $p \neq \mathbf{0}$). Note also that these conditional axioms are actually a finite collection of axioms, one for each constant definition: since the set $\mathscr{C}$ of process constants is finite, the instances of **R1-R3** are finitely many. Finally, we have axioms **P1-P3** for parallel composition.

A1	Associativity	$x+(y+z) = (x+y)+z$	
A2	Commutativity	$x+y = y+x$	
A3	Identity	$x+\mathbf{0} = x$	if $x \neq \mathbf{0}$
A4	Idempotence	$x+x = x$	if $x \neq \mathbf{0}$
R1	Stuck	if $C \doteq \mathbf{0}$, then $C = \mathbf{0}+\mathbf{0}$	
R2	Unfolding	if $C \doteq p \wedge p \neq \mathbf{0}$, then $C = p$	
R3	Folding	if $C \doteq p\{C/x\} \wedge q = p\{q/x\}$, then $C = q$	
P1	Associativity	$x \,\vert\, (y \,\vert\, z) = (x \,\vert\, y) \,\vert\, z$	
P2	Commutativity	$x \,\vert\, y = y \,\vert\, x$	
P3	Identity	$x \,\vert\, \mathbf{0} = x$	

Table 4 Axioms for team bisimulation equivalence

Theorem 9 (Soundness) *For every $p,q \in \mathscr{P}_{BPP}$, if $E \vdash p = q$, then $p \sim^{\oplus} q$.*

Proof The proof is by induction on the proof of $E \vdash p = q$. The thesis follows by observing that all the axioms in E are sound by Propositions 18, 19 and 20 and that $\sim^{\oplus}$ is a congruence. □

Proposition 21 (Unique solution) *Let $\widetilde{X} = (x_1, x_2, \dots, x_n)$ be a tuple of variables and let $\widetilde{p} = (p_1, p_2, \dots, p_n)$ be a tuple of open guarded BPP terms ($p_i \neq \mathbf{0}$ for $i = 1, \dots, n$), using the variables in $\widetilde{X}$. Then, there exists a tuple $\widetilde{q} = (q_1, q_2, \dots, q_n)$ of closed sequential BPP terms such that*

$E \vdash q_i = p_i\{\widetilde{q}/\widetilde{X}\}$ *for $i = 1, \dots, n$.*

Moreover, if the same property holds for $\widetilde{q'} = (q'_1, q'_2, \dots, q'_n)$, then

$E \vdash q'_i = q_i$ *for $i = 1, \dots, n$.*

Proof By induction on n. We assume that there exists a tuple of constants $\widetilde{C} = (C_1, C_2, \dots, C_n)$ that do not occur in $\widetilde{p} = (p_1, p_2, \dots, p_n)$.

For $n = 1$, we choose $q_1 = C_1$, and we close this new constant with the definition $C_1 \doteq p_1\{C_1/x_1\}$, and so the result follows immediately using axiom **R2**. This solution is unique: if $E \vdash r_1 = p_1\{r_1/x_1\}$, since $C_1 \doteq p_1\{C_1/x_1\}$, by axiom **R3** we get $E \vdash C_1 = r_1$.

Now assume a tuple $\widetilde{p} = (p_1, p_2, \dots, p_n)$ and the term p_{n+1}, so that they are all open on $\widetilde{X} = (x_1, x_2, \dots, x_n)$ and the additional x_{n+1}. Assume, w.l.o.g., that x_{n+1} occurs in p_{n+1}. First, define $C_{n+1} \doteq p_{n+1}\{C_{n+1}/x_{n+1}\}$, so that this new constant C_{n+1} is now open on $\widetilde{X}$ only. Therefore, also for $i = 1, \dots, n$, each $p_i\{C_{n+1}/x_{n+1}\}$ is now open on $\widetilde{X}$ only. Thus, we are now able to use induction on $\widetilde{X}$ and $(p_1\{C_{n+1}/x_{n+1}\}, \dots, p_n\{C_{n+1}/x_{n+1}\})$, to conclude that there exists a tuple $\widetilde{q} = (q_1, q_2, \dots, q_n)$ of closed sequential BPP terms such that

$E \vdash q_i = (p_i\{C_{n+1}/x_{n+1}\})\{\widetilde{q}/\widetilde{X}\} = p_i\{\widetilde{q}/\widetilde{X}, C_{n+1}\{\widetilde{q}/\widetilde{X}\}/x_{n+1}\}$ for $i = 1, \dots, n$.

Note that above by $C_{n+1}\{\widetilde{q}/\widetilde{X}\}$ we have implicitly closed the definition of C_{n+1} as

$C_{n+1} \doteq p_{n+1}\{C_{n+1}/x_{n+1}\}\{\widetilde{q}/\widetilde{X}\} = p_{n+1}\{\widetilde{q}/\widetilde{X}\}\{C_{n+1}/x_{n+1}\}$,
so that C_{n+1} can be chosen as q_{n+1}. By axiom **R2**, $E \vdash C_{n+1} = p_{n+1}\{\widetilde{q}/\widetilde{X}\}\{C_{n+1}/x_{n+1}\}$, as required.

Unicity of the tuple $(\widetilde{q}, q_{n+1})$ can be proved by using axiom **R3**. Assume to have another solution tuple $(\widetilde{q'}, q'_{n+1})$. This means that

$E \vdash q'_i = p_i\{\widetilde{q'}/\widetilde{X}, q'_{n+1}/x_{n+1}\} \quad$ for $i = 1, \dots, n+1$.

By induction, we can assume that $E \vdash q_i = q'_i$, for $i = 1, \dots, n$.

Since $E \vdash C_{n+1} = p_{n+1}\{\widetilde{q}/\widetilde{X}\}\{C_{n+1}/x_{n+1}\}$, by substitutivity, we get

$E \vdash C_{n+1} = p_{n+1}\{\widetilde{q'}/\widetilde{X}\}\{C_{n+1}/x_{n+1}\}$.

Let F be a constant defined as $F \doteq p_{n+1}\{\widetilde{q'}/\widetilde{X}\}\{F/x_{n+1}\}$. Then, by axiom **R3**, $E \vdash F = C_{n+1}$. Hence, since $E \vdash q'_{n+1} = p_{n+1}\{\widetilde{q'}/\widetilde{X}\}\{q'_{n+1}/x_{n+1}\}$, by axiom **R3**, we get $E \vdash F = q'_{n+1}$; so the thesis $E \vdash C_{n+1} = q'_{n+1}$ by transitivity. □

Proposition 22 (Equational characterization) *For each $p \in \mathscr{P}^{seq}_{BPP}$ ($p \neq \mathbf{0}$), there exists a set $\{p_1, p_2, \dots, p_k\} \subseteq \mathscr{P}^{seq}_{BPP}$ such that $k \geq 1$, $E \vdash p = p_1$ and, for $i = 1, \dots, k$, $E \vdash p_i = p'_i$ where p'_i can be either $\mathbf{0}+\mathbf{0}$ or $\sum_{j=1}^{n(i)} a_{ij}.q_{ij}$ (with $n(i) \geq 1$) such that $dom(dec(q_{ij})) \subseteq \{p_1, p_2, \dots, p_k\}$.*

Proof By induction on the syntactic definition of p. Actually, induction is on the pair (p, I), where I is a set of constants, initially empty: the proof starts with the pair $(p, \emptyset)$. In all the cases, except for the case of process constants, the parameter I is omitted for the sake of simplicity.

If $p = \mu.q$, then let $dec(q) = k_1 \cdot r_1 \oplus k_2 \cdot r_2 \oplus \dots \oplus k_h \cdot r_h$. If $h = 0$, then $E \vdash q = \mathbf{0}$ and the thesis follows trivially by choosing $p_1 = \mu.\mathbf{0} = p'_1$. Otherwise, by induction, for each $j = 1, \dots, h$, there exist $\{r^j_1, \dots r^j_{l_j}\} \subseteq \mathscr{P}^{seq}_{BPP}$ such that $E \vdash r_j = r^j_1$, $E \vdash r^j_i = s^j_i$ for $i = 1, \dots, l_j$, where s^j_i can be either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{h=1}^{n_j(i)} a_{ih}.t^j_{ih}$ such that $dom(dec(t^j_{ih})) \subseteq \{r^j_1, \dots r^j_{l_j}\}$. We can choose $p_1 = \mu.t$, where t is a term such that $dec(t) = k_1 \cdot r^1_1 \oplus k_2 \cdot r^2_1 \oplus \dots \oplus k_h \cdot r^h_1$; indeed, $E \vdash p = p_1$ via axioms **P1-P3** and by substitutivity. Moreover, the set of sequential processes is $\{p_1\} \cup \{r^1_1, \dots r^1_{l_1}\} \cup \dots \cup \{r^h_1, \dots r^h_{l_h}\}$. Since for each r^j_i there is already a suitable s^j_i, it remains to define p'_1, which is $p'_1 = \mu.t$.

If $p = r_1 + r_2$, then, in case $r_1 = \mathbf{0} = r_2$, take $p_1 = \mathbf{0}+\mathbf{0} = p'_1$. In case $r_1 \neq \mathbf{0} = r_2$, then, by induction there exist $\{r^1_1, \dots r^1_{k_1}\}$ such that $E \vdash r_1 = r^1_1$, and for $i = 1, \dots, k_1$, $E \vdash r^1_i = s^1_i$, where s^1_i can be either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{h=1}^{n_1(i)} a_{ih}.t^1_{ih}$ such that $dom(dec(t^1_{ih})) \subseteq \{r^1_1, \dots r^1_{k_1}\}$. We can take $p_1 = r^1_1 + \mathbf{0}$ and $p'_1 = s^1_1$, with $E \vdash p = p_1$ by substitutivity, and $E \vdash p_1 = p'_1$ by substitutivity and axiom **A3**; the other terms are $\{r^1_2, \dots r^1_{k_1}\}$, with their corresponding $\{s^1_2, \dots s^1_{k_1}\}$. Symmetrically in case $r_1 = \mathbf{0} \neq r_2$. Otherwise (i.e., when $r_1 \neq \mathbf{0} \neq r_2$), by induction there exist $\{r^1_1, \dots r^1_{k_1}\}$ and $\{r^2_1, \dots r^2_{k_2}\}$, such that $E \vdash r_1 = r^1_1$, $E \vdash r_2 = r^2_1$, and (for $j = 1, 2$) $E \vdash r^j_i = s^j_i$, where s^j_i can be either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{h=1}^{n_j(i)} a_{ih}.t^j_{ih}$ such that $dom(dec(t^j_{ih})) \subseteq \{r^j_1, \dots r^j_{k_j}\}$. We can take $p_1 = r^1_1 + r^2_1$ so that the set is $\{p_1\} \cup \{r^1_1, \dots r^1_{k_1}\} \cup \{r^2_1, \dots r^2_{k_2}\}$. Since for each r^j_i there is already a suitable s^j_i, it remains to define p'_1. If s^1_1 is $\mathbf{0}+\mathbf{0}$ and

s_1^2 is $\mathbf{0}+\mathbf{0}$, then $p_1' = \mathbf{0}+\mathbf{0}$ (by axiom **A4**). If s_1^1 is $\mathbf{0}+\mathbf{0}$ and $s_1^2 = \sum_{j=1}^{n_2(1)} a_{1j}.t_{1j}^2$, then $p_1' = s_1^2$ (by axioms **A1-A3**). Symmetrically, if s_1^2 is $\mathbf{0}+\mathbf{0}$ and $s_1^1 = \sum_{j=1}^{n_1(1)} a_{1j}.t_{1j}^1$, then $p_1' = s_1^1$ (by axioms **A1-A3**). If $s_1^1 = \sum_{j=1}^{n_1(1)} a_{1j}.t_{1j}^1$ and $s_1^2 = \sum_{j=1}^{n_2(1)} a'_{1j}.t_{1j}^2$, then $p_1' = \sum_{j=1}^{n_1(1)} a_{1j}.t_{1j}^1 + \sum_{j=1}^{n_2(1)} a'_{1j}.t_{1j}^2$.

In case $p = C$, we have to consider the second parameter I: if (C,I) is such that $C \in I$, then we have that $p_1 = C$ and $p_1' = \mathbf{0}+\mathbf{0}$. If $C \notin I$ and $C \doteq r$, then we have to distinguish two subcases:

(i) If $r = \mathbf{0}$, then $p_1 = C$ and $p_1' = \mathbf{0}+\mathbf{0}$, with $E \vdash p_1 = p_1'$ by axiom **R1**.

(ii) If $r \neq \mathbf{0}$, by induction on $(r, I \cup \{C\})$, we know that for r there exist $k \geq 1$ and $\{r_1, \dots, r_k\}$ such that $E \vdash r = r_1$ and for $i = 1, \dots, k$, $E \vdash r_i = r_i'$ where r_i' can be either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{j=1}^{n(i)} a_{ij}.t_{ij}$ such that $dom(dec(t_{ij})) \subseteq \{r_1, r_2, \dots, r_k\}$. Note that $E \vdash C = r_1'$, because $E \vdash r = r_1$, $E \vdash r_1 = r_1'$ and, by axiom **R2**, $E \vdash C = r$. If r_1' is $\mathbf{0}+\mathbf{0}$, then $k = 1$ and $p_1 = C$ and $p_1' = \mathbf{0}+\mathbf{0}$. If r_1' is $\sum_{j=1}^{n(1)} a_{1j}.t_{ij}$, then, $p_1 = C$ and for $i = 1, \dots, k$, $p_{i+1} = r_i$ and, moreover, $p_1' = r_1'$ and $p_{i+1}' = r_i'$ if $r_i \neq C$, while $p_{i+1}' = r_1'$ otherwise. □

Example 16 To illustrate how induction works in the proof of the proposition above, consider the constant $C \doteq a.C + \mathbf{0}$. We have to start with $(C, \emptyset)$, whose solution requires to consider $(a.C + \mathbf{0}, \{C\})$, in turn requiring to consider $(a.C, \{C\})$, in turn $(C, \{C\})$, which is the base of the induction. As $C \in \{C\}$, we have that for $(C, \{C\})$ the required terms are $p_1 = C$ and $p_1' = \mathbf{0}+\mathbf{0}$. Now we can compute the terms associated with $(a.C, \{C\})$, which are: $p_1 = a.C$, $p_1' = a.C$, $p_2 = C$, $p_2' = \mathbf{0}+\mathbf{0}$. So, now we can compute the terms for $(a.C + \mathbf{0}, \{C\})$, which are: $p_1 = a.C + \mathbf{0}$, $p_1' = a.C$, $p_2 = C$, $p_2' = \mathbf{0}+\mathbf{0}$. Finally, $(C, \emptyset)$ has these terms: $p_1 = C$, $p_1' = a.C$, $p_2 = a.C + \mathbf{0}$, $p_2' = a.C$, $p_3 = C$ and $p_3' = a.C$. Of course, p_3 is redundant and the required set is $\{C, a.C + \mathbf{0}\}$, with their associated $a.C$. □

Proposition 23 (Completeness for sequential terms)
For each $p, p' \in \mathscr{P}_{BPP}^{seq}$, if $p \sim p'$ (or $p = \mathbf{0} = p'$), then $E \vdash p = p'$.

Proof If $p = \mathbf{0} = p'$, then $E \vdash p = p'$ by reflexivity. Otherwise, by Proposition 22, we have that there exists a set $\{p_1, p_2, \dots, p_k\}$ of sequential processes such that $E \vdash p = p_1$, and there exist $r_1, r_2, \dots, r_k$ such that, for $i = 1, \dots, k$, $E \vdash p_i = r_i$ and r_i is either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{j=1}^{n(i)} a_{ij}.t_{ij}$ such that $dom(dec(t_{ij})) \subseteq \{p_1, p_2, \dots, p_k\}$.

Similarly, there exists a set $\{p_1', p_2', \dots, p_h'\}$ of sequential processes such that $E \vdash p' = p_1'$, and there exist $r_1', r_2', \dots, r_h'$ such that, for $i = 1, \dots, h$, $E \vdash p_i' = r_i'$ and r_i' is either $\mathbf{0}+\mathbf{0}$ or a sumform $\sum_{j=1}^{n'(i)} a'_{ij}.t'_{ij}$ such that $dom(dec(t'_{ij})) \subseteq \{p_1', p_2', \dots, p_h'\}$.

By Theorem 9, we have that $p \sim p_1 \sim r_1$ and $p' \sim p_1' \sim r_1'$; as by hypothesis $p \sim p'$, by transitivity we have that $p_1 \sim p_1'$ and $r_1 \sim r_1'$.

If $r_1 = \mathbf{0}+\mathbf{0}$, then also $r_1' = \mathbf{0}+\mathbf{0}$, and so $E \vdash r_1 = r_1'$ by reflexivity and $E \vdash p = p'$ by transitivity. Otherwise, let $r_1 = \sum_{j=1}^{n(1)} a_{1j}.t_{1j}$ and $r_1' = \sum_{j=1}^{n'(1)} a'_{1j}.t'_{1j}$.

Now, let $I = \{(i, i') \mid p_i \sim p'_{i'}\}$. Clearly, $(1,1) \in I$. Since p_i and $p'_{i'}$ are team bisimilar when $(i, i') \in I$, the following hold: for each $(i, i') \in I$, there exists a total surjective relation $J_{ii'}$ between $\{1, 2, \dots n(i)\}$ and $\{1, 2, \dots n'(i')\}$ given by $J_{ii'} = \{(j, j') \mid a_{ij} = a'_{i'j'} \wedge (dec(t_{ij}), dec(t'_{i'j'})) \in I^{\oplus}\}$, where $(dec(t_{ij}), dec(t'_{i'j'})) \in I^{\oplus}$ if

- $dec(t_{ij}) = p_{d_{(i,j,1)}} \oplus p_{d_{(i,j,2)}} \oplus \ldots \oplus p_{d_{(i,j,n)}}$ such that $1 \leq d_{(i,j,l)} \leq k$ for $l = 1, \ldots, n$;
- $dec(t'_{i'j'}) = p'_{d'_{(i',j',1)}} \oplus p'_{d'_{(i',j',2)}} \oplus \ldots \oplus p'_{d'_{(i',j',n)}}$, such that $1 \leq d'_{(i',j',l)} \leq h$ for $l = 1, \ldots, n$; and
- $(d_{(i,j,l)}, d'_{(i',j',l)}) \in I$ for $l = 1, \ldots, n$. (If $n = 0$, then $(\theta, \theta) \in I^{\oplus}$).

Now, for each $(i,i') \in I$, let us consider the variables $x_{ii'}$ and the open term

$$v_{ii'} = \sum_{(j,j') \in J_{ii'}} a_{ij}.(x_{d_{(i,j,1)}d'_{(i',j',1)}} \,|\, x_{d_{(i,j,2)}d'_{(i',j',2)}} \,|\, \ldots \,|\, x_{d_{(i,j,n)}d'_{(i',j',n)}})$$

where, if $J_{ii'} = \emptyset$, then $v_{ii'} = \mathbf{0} + \mathbf{0}$, while in case $dec(t_{ij}) = dec(t'_{i'j'}) = \theta$, the open parallel process
$x_{d_{(i,j,1)}d'_{(i',j',1)}} \,|\, x_{d_{(i,j,2)}d'_{(i',j',2)}} \,|\, \ldots \,|\, x_{d_{(i,j,n)}d'_{(i',j',n)}}$
is actually $\mathbf{0}$. By Proposition 21, for each $(i,i') \in I$, there exists $s_{ii'}$ such that $E \vdash s_{ii'} = v_{ii'}\{\widetilde{s}/\widetilde{X}\}$, where $\widetilde{s}$ denotes the tuple of terms of the form $s_{ii'}$ for each $(i,i') \in I$, and $\widetilde{X}$ denotes the tuple of variables $x_{ii'}$ for each $(i,i') \in I$.

If we close each $v_{ii'}$ by replacing each $x_{d_{(i,j,l)}d'_{(i',j',l)}}$ with $p_{d_{(i,j,l)}}$, we get

$$\sum_{(j,j') \in J_{ii'}} a_{ij}.(p_{d_{(i,j,1)}} \,|\, p_{d_{(i,j,2)}} \,|\, \ldots \,|\, p_{d_{(i,j,n)}})$$

which is equal, up to axioms **P1-P3**, to $\sum_{(j,j') \in J_{ii'}} a_{ij}.t_{ij}$, in turn equal, via axioms **A1-A4**, to r_i: in fact, $J_{ii'}$ is surjective so that the two summations differ only for possible repeated summands. Since $E \vdash p_i = r_i$ for $i = 1, \ldots, k$, we get that
$E \vdash r_i = \sum_{(j,j') \in J_{ii'}} a_{ij}.(r_{d_{(i,j,1)}} \,|\, r_{d_{(i,j,2)}} \,|\, \ldots \,|\, r_{d_{(i,j,n)}})$.
Therefore, we note that r_i is such that $E \vdash r_i = v_{ii'}\{\widetilde{r}/\widetilde{X}\}$ and so, by Proposition 21, we have that $E \vdash s_{ii'} = r_i$. Since $(1,1) \in I$, we have that $E \vdash s_{11} = r_1$.

Similarly, if we close each $v_{ii'}$ by replacing each $x_{d_{(i,j,l)}d'_{(i',j',l)}}$ with $p'_{d'_{(i',j',l)}}$, we get

$$\sum_{(j,j') \in J_{ii'}} a_{ij}.(p'_{d'_{(i',j',1)}} \,|\, p'_{d'_{(i',j',2)}} \,|\, \ldots \,|\, p'_{d'_{(i',j',n)}})$$

which is equal, up to axioms **P1-P3**, to $\sum_{(j,j') \in J_{ii'}} a_{ij}.t'_{i'j'}$, in turn equal, via axioms **A1-A4**, to $r'_{i'}$: in fact, $J_{ii'}$ is surjective so that the two summations differ only for possible repeated summands. Since $E \vdash p'_i = r'_i$ for $i = 1, \ldots, h$, we get that
$E \vdash r'_{i'} = \sum_{(j,j') \in J_{ii'}} a_{ij}.(r'_{d'_{(i',j',1)}} \,|\, r'_{d'_{(i',j',2)}} \,|\, \ldots \,|\, r'_{d'_{(i',j',n)}})$.
Thus, we note that $r'_{i'}$ is such that $E \vdash r'_{i'} = v_{ii'}\{\widetilde{r'}/\widetilde{X}\}$ and so, by Proposition 21, we have that $E \vdash s_{ii'} = r'_{i'}$. Since $(1,1) \in I$, we have that $E \vdash s_{11} = r'_1$; by transitivity, it follows that $E \vdash r_1 = r'_1$, and so that $E \vdash p = p'$. □

Theorem 10 (Completeness) *For every $p,q \in \mathscr{P}_{BPP}$, if $p \sim^{\oplus} q$, then $E \vdash p = q$.*

Proof The proof is by induction on the size of $dec(p)$. If $|dec(p)| = 0$, then $dec(p) = \theta$; as $p \sim^{\oplus} q$, necessarily also $dec(q) = \theta$. By observing the definition of the decomposition function in Table 2, this is possible only if p and q are either $\mathbf{0}$ or a parallel composition of $\mathbf{0}$'s, e.g., $\mathbf{0}\,|\,\mathbf{0}$; hence, $E \vdash p = \mathbf{0}$ and $E \vdash q = \mathbf{0}$, possibly using axioms **P1-P3**; hence, by transitivity we get $E \vdash p = q$. If $|dec(p)| = k+1$, then there exist p_1 and p_2 such that $dec(p) = p_1 \oplus dec(p_2)$. As $p \sim^{\oplus} q$, there exist q_1, q_2 such that $dec(q) = q_1 \oplus dec(q_2)$, $p_1 \sim q_1$ and $dec(p_2) \sim^{\oplus} dec(q_2)$. By the definition of the decomposition function and by axioms **P1-P3**, this means that $E \vdash p = p_1\,|\,p_2$ and $E \vdash q = q_1\,|\,q_2$. By Proposition 23 we have that $E \vdash p_1 = q_1$. By induction, we have that $E \vdash p_2 = q_2$. By substitutivity we get $E \vdash p_1\,|\,p_2 = q_1\,|\,q_2$ and so the thesis follows by transitivity. $\square$

7 Conclusion, Related Literature and Future Research

Team bisimulation equivalence is a truly concurrent equivalence which seems the most natural, intuitive and simple extension of LTS bisimulation equivalence to BPP nets. It also has a very low complexity: indeed, $\sim$ can be computed in $O(m \cdot p^2 \cdot \log{(n+1)})$ time, where m is the number of net transitions, p is the size of the largest post-set (i.e., p is the least natural such that $|t^{\bullet}| \leq p$ for all t) and n is the number of places; after having computed $\sim$, checking whether two markings of size k are team bisimilar can be done in $O(k^2)$ (according to Algorithm 1) or in $O(n)$ (according to the algorithm in [45], *cf.* Remark 5). As, in order to perform team bisimulation equivalence checking, there is no need to compute the LTSs of the global behavior of the systems under scrutiny, our proposal seems a natural solution to solving the state-space explosion problem for BPP nets.

Moreover, team bisimilarity is intuitively appealing as it coincides with *strong place bisimilarity* [4,5], which on BPP nets is coarser than the branching-time semantics of *isomorphism of (nondeterministic) occurrence nets* (or unfoldings) [26] and finer than the linear-time semantics of *isomorphism of causal (or deterministic occurrence) nets* [6,50]. Moreover, in the companion paper [35] we prove that for BPP nets team bisimulation equivalence coincides with a notion of bisimulation on causal nets, called *causal-net bisimulation* (inspired to [29]), and with a slight strengthening of *history-preserving bisimilarity* [55,18,27] (which on nets takes the form of so-called *fully concurrent bisimilarity* [7]) which requires additionally that whenever two markings are related they must have the same size. Hence, team bisimulation equivalence does respect the causal behavior of BPP nets.

Concrete semantics, such as team bisimilarity, which observe also the structure of the distributed state are *resource aware*, and so potentially more useful in practical applications. For instance, the security paper [36] shows that an illegal information flow cannot be detected if the low user cannot observe the structure of the state.

Our approach is based on an intuition very similar to [23,24], where Fröschle extends the *unique decomposition* idea, originally proposed in [48] for interleaving bisimilarity over finite BPP (i.e., BPP without constants/recursion), to a truly-concurrent setting. She proposes to consider the decomposition of a truly-concurrent semantic model (e.g., 1-safe P/T net, or asynchronous transition system [59]) so that it can be dissected into independent 'chunks of behaviour'. She introduces the corresponding concept of 'decomposition into independent components': in order to check whether two systems P and Q are equivalent, one can check whether there is a *one-*

to-one correspondence between the independent components of P and those of Q such that related components are equivalent. By means of this approach, she was able to prove that *history-preserving bisimilarity* [55, 18, 27] and *hereditary history-preserving bisimilarity* [10, 41] coincide on SBPP (while this is not the case on full BPP). Moreover, on SBPP she proved, in joint work with her co-workers, in [25], by using an event structure [58] semantics, that *history-preserving bisimilarity* (which on nets takes the form of *fully concurrent bisimilarity* [7]) is decidable with time complexity $O(n^3 \cdot \log n)$, where n is a measure of the *size of the involved SBPP terms*; therefore, the time complexity for hpb on SBPP is roughly comparable with the complexity of team bisimilarity on BPP nets.[1] In the companion paper [35] we show that our definition of team bisimulation can be weakened to *h-team bisimilarity*, an equivalence which coincides with history-preserving bisimilarity on BPP, with a time complexity similar to that of team bisimilarity.

Since BPP (with guarded summation and guarded constants) is the process algebra representing, up to net isomorphism, all the possible BPP nets [32], it would also be interesting to compare team bisimulation equivalence with other non-interleaving equivalences proposed on process algebras, such as *distributed bisimilarity* [14], *performance bisimilarity* [30], *location bisimilarity* [15] or *causal bisimilarity* [17] (the latter being equivalent to *history-preserving bisimilarity* [55, 18, 27]; hpb, for short). All these non-interleaving equivalences do coincide on SBPP [2, 22, 44]. As discussed above, team bisimulation equivalence coincides with strong place bisimilarity, which is finer than than hpb. Hence, team bisimulation equivalence is finer than all of these non-interleaving behavioral equivalences; the inclusion is strict: for instance, let C be a process constant with empty body, $C \doteq \mathbf{0}$; then the two terms $a.\mathbf{0}$ and $a.\mathbf{0}\,|\,C$ are causal bisimilar, but they generate two markings of different size, and so they are not related by team bisimulation equivalence (see also Example 9).

This research is a generalization of our previous paper [33], where we approached the same problem for a simpler class of finite Petri nets, called *finite-state machines* [52, 32], a class of nets whose transitions have singleton pre-set (as for BPP) and singleton post-set, so that the set of reachable markings is always finite. The main problem we had to face here was related to the more complex nature of post-sets for BPP nets, so that *classic* bisimulation, which is enough to deal with finite-state machines, has to be properly generalized to *team* bisimulation. In this paper, we have defined the theory of team bisimulation, including its fixpoint characterization and the definition of the "up-to" proof technique.

Team bisimulation equivalence is characterized by a very simple and natural modal logic, namely TML, extending conservatively Hennessy-Milner Logic (HML) [37, 3]. TML is also an extension of BTML (Basic TML), we defined in [33] in order to characterize team equivalence over finite-state machines. TML parallel composition operator $\_ \otimes \_$ on formulae reminds the spatial operator of Caires' and Cardelli's *spatial logic* [13], also used on spatial transition systems in, e.g., [1]. More complex modal logics characterizing some non-interleaving equivalences have been proposed

[1] However, we note that this value n is strictly larger than the size of the corresponding BPP net. In fact, in [25] the size of a BPP term p is defined as "the total number of occurrences of symbols (including parentheses)", where p is defined by means of a non-ambiguous, concrete syntax. For instance, $p = (a.\mathbf{0})\,|\,(a.\mathbf{0})$ has size 11, while the net semantics for p generates one place and one transition (and 2 tokens).

in, e.g., [11,9]. A possible future work is to extend TML to become a temporal logic, with least and greatest fixpoint operators, as in Kozen's modal mu-calculus [43].

The set of axioms in Table 4 we have provided for axiomatizing team bisimilarity over BPP (with guarded summation and guarded recursion) is the same set we first outlined in our previous paper [34], where we proved that exactly this set constitutes also a finite, sound and complete, axiomatization for team equivalence of the process algebra CFM, which truly represents the class of finite-state machines [32]. This is not surprising, because BPP and CFM have exactly the same algebraic operators, the only difference being that, in the action prefixing operator, an action may prefix only a sequential process in CFM, while it may prefix a parallel process in BPP. Our axiomatization, and the proof techniques we adopted to prove its completeness, are based on [46], where Milner provided a finite axiomatization of interleaving bisimilarity for finite-state CCS; nonetheless, our technical treatment, based on constants defined over guarded processes (e.g., $C \doteq a.C$) rather than on the recursive operator (with possible unguarded variables; e.g. $fixX.(a.X+X)$), is simpler than that.

In the literature there are only few examples of finite axiomatizations for truly-concurrent equivalences. In [16] *distributed bisimilarity* [14] is axiomatized for SBPP, and in [24] *hereditary history-preserving bisimilarity* [10,41] is axiomatized for full BPP with a sequent-based approach. These two finite axiomatizations are actually sound and complete also for *history-preserving bisimilarity* (hpb, for short), as on SBPP these two behavioral equivalences coincide with hpb [2,22]. Our axiomatization can be adapted to offer an alternative finite axiomatization of hpb for BPP (with guarded summation and guarded recursion): it is enough to remove the side-condition $x \neq \mathbf{0}$ in axioms **A3-A4**, to drop axiom **R1** and to remove the premise $p \neq \mathbf{0}$ in the conditional axiom **R2**. A formal proof of this fact is described in the forthcoming paper [35]. This explains the essence of the difference between these two equivalences: hpb is slightly coarser than team bisimulation equivalence because only the former may relate markings with different size where, however, the possible additional components are all stuck. Moreover, to get the axiomatization for the linear-time versions of team bisimulation equivalence and history-preserving bisimilarity for BPP with guarded summation, we conjecture that it is enough to add the distributivity axiom: $\mu.(x+y) = \mu.x+\mu.y$.

Finally, a possible extension of this work is about BPP nets with silent moves: *weak* team bisimulation and *branching* team bisimulation can be defined, taking inspiration from the definitions of weak bisimulation [47] and branching bisimulation [28] over LTSs.

Acknowledgements The two anonymous referees are thanked for their careful proof-reading, for their very useful comments and suggestions, and for having noticed a mistake in Proposition 3 of the submitted version of this paper.

References

1. L. Acciai, M. Boreale, G. Zavattaro, On the Relationship between Spatial Logics and Behavioral Simulations, in Procs. Fossacs'10, LNCS 6014, 146-160, Springer (2010)

2. L. Aceto, Relating Distributed, Temporal and Causal Observations of Simple Processes, *Fundamenta Informaticae* 17(4):319-331 (1992)
3. L. Aceto, A. Ingólfsdóttir, K. Larsen, J. Srba, *Reactive Systems: Modelling, Specification and Verification*, Cambridge University Press (2007)
4. C. Autant, Z. Belmesk, Ph. Schnoebelen, Strong Bisimilarity on Nets Revisited, in Procs. PARLE'91, vol. II: Parallel Languages, LNCS 506, 295-312, Springer (1991)
5. C. Autant, Ph. Schnoebelen, Place Bisimulations in Petri Nets, in Procs. Application and Theory of Petri Nets 1992, LNCS 616, 45-61, Springer (1992)
6. E. Best, R. Devillers, Sequential and Concurrent Behavior in Petri Net Theory, *Theoretical Computer Science* 55(1):87-136 (1987)
7. E. Best, R. Devillers, A. Kiehn, L. Pomello, Concurrent Bisimulations in Petri Nets, *Acta Informatica* 28(3): 231-264 (1991)
8. J.A. Bergstra, A. Ponse, S.A. Smolka (eds.), *Handbook of Process Algebra*, Elsevier (2001)
9. P. Baldan, S. Crafa, A Logic for True Concurrency, *Journal of the ACM* 61(4): 24:1-24:36 (2014)
10. M. Bednarczyk, Hereditary history preserving bisimulation or what is the power of the future perfect in program logics, Technical report, Polish Academy of Sciences, Gdansk (1991).
11. J.C. Bradfield, S.B. Fröschle, Independence-Friendly Modal Logic and True Concurrency, *Nord. J. Comput.* 9(1): 102-117 (2002)
12. J. W. de Bakker, W. P. de Roever, G. Rozenberg (Eds.), *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency*, REX School/Workshop, Noordwijkerhout, The Netherlands, LNCS 354, Springer-Verlag (1989)
13. L. Caires, L. Cardelli, A Spatial Logic for Concurrency (part I), *Information and Computation* 186(2): 194-235 (2003)
14. I. Castellani, M. Hennessy, Distributed Bisimulations, *Journal of the ACM* 36(4):887-911 (1989)
15. I. Castellani, Process Algebras with Localities, Chapter 15 in [8], 945-1046, Elsevier (2001)
16. S. Christensen, *Decidability and Decomposition in Process Algebra*, Ph.D. Thesis, University of Edinburgh (1993)
17. Ph. Darondeau, P. Degano, Causal Trees, in Procs. ICALP'89, LNCS 372, 234-248, Springer (1989)
18. P. Degano, R. De Nicola, U. Montanari, Partial Ordering Descriptions and Observations of Nondeterministic Concurrent Systems, in [12], 438-466 (1989)
19. B.A. Davey, H.A. Priestley, *Introduction to Lattices and Order* (second edition), Cambridge University Press (2002)
20. J. Esparza, A. Kiehn, On the Model Checking Problem for Branching-time Logics and Basic Parallel Processes, in Procs. CAV'95, LNCS 939,Springer, 353-366 (1995)
21. J. Esparza, Decidability and Complexity of Petri Net Problems: An Introduction, *Lectures on Petri Nets I: Basic Models*, LNCS1491, 374-428, Springer-Verlag, 1998.
22. S. Fröschle, Decidability of Plain and Hereditary History-preserving Bisimulation for BPP, in Procs. EXPRESS'99, ENTCS 27 (1999)
23. S. Fröschle, Composition and Decomposition in True-Concurrency, in Procs. Fossacs'05, LNCS 3441, 333–347, Springer (2005)
24. S. Fröschle, S. Lasota, Decomposition and Complexity of Hereditary History Preserving Bisimulation on BPP, in Procs. CONCUR'05, LNCS 3656, 263–277, Springer (2005)
25. S. Fröschle, P. Jančar, S. Lasota, Z. Sawa, Non-interleaving Bisimulation Equivalences on Basic Parallel Processes, *Information and Computation* 208(1):42-62 (2010)
26. R.J. van Glabbeek, F.W. Vaandrager, Petri Net Models for Algebraic Theories of Concurrency, In Proc. PARLE'87, LNCS 259, 224-242, Springer (1987)
27. R.J. van Glabbeek, U. Goltz, Equivalence Notions for Concurrent Systems and Refinement of Actions, in Procs. MFCS'89, LNCS 379, 237-248, Springer (1989)
28. R.J. van Glabbeek, W.P. Weijland, Branching Time and Abstraction in Bisimulation Semantics, *Journal of the ACM* 43(3):555-600 (1996)
29. R.J. van Glabbeek, Structure Preserving Bisimilarity, Supporting an Operational Petri Net Semantics of CCSP, in [49], 99-130 (2015)
30. R. Gorrieri, M. Roccetti, E. Stancampiano, A Theory of Processes with Durational Actions, *Theoretical Computer Science* 140(1):73-94 (1995)
31. R. Gorrieri, C. Versari, *Introduction to Concurrency Theory: Transition Systems and CCS*, EATCS Texts in Theoretical Computer Science, Springer-Verlag (2015)
32. R. Gorrieri, *Process Algebras for Petri Nets: The Alphabetization of Distributed Systems*, EATCS Monographs in Theoretical Computer Science, Springer-Verlag (2017)

33. R. Gorrieri, Verification of Finite-State Machines: A Distributed Approach, *Journal of Logic and Algebraic Methods in Programming* 96:65-80 (2018)
34. R. Gorrieri, Axiomatizing Team Equivalence for Finite-State Machines, *The Art of Modeling Computational Systems: A Journey from Logic and Concurrency to Security and Privacy: Catuscia Palamidessi's Festschrift*, LNCS 11760, pages 14-32, Springer (2019)
35. R. Gorrieri, A Study on Team Bisimulations for BPP nets, in preparation, 2019.
36. R. Gorrieri, Interleaving vs True Concurrency: Some Instructive Security Examples, in preparation, 2019.
37. M. Hennessy, R. Milner, Algebraic Laws for Nondeterminism and Concurrency, *Journal of the ACM* 32(1):137-161 (1985)
38. J.E. Hopcroft, R. Motwani, J.D. Ullman, *Introduction to Automata Theory, Languages and Computation*, 2nd ed., Addison-Wesley (2001)
39. P. Jančar, Undecidability of Bisimilarity for Petri Nets and Some Related Problems, *Theoretical Computer Science* 148(2):281-301, 1995.
40. P. Jančar, Strong Bisimilarity on Basic Parallel Processes is PSPACE-complete, in Procs. of the 18th Annual IEEE Symposium on Logic in Computer Science (LICS'03), 218-227, IEEE Computer Society Press (2003)
41. A. Joyal, M. Nielsen, G. Winskel, Bisimulation from open maps, *Information and Computation* 127:164-185 (1996)
42. R. Keller, Formal Verification of Parallel Programs, *Comm. of the ACM* 19(7):561-572 (1976)
43. D. Kozen, Results on the Propositional mu-Calculus, *Theoretical Computer Science* 27:333-354 (1983)
44. S. Lasota, Decidability of Performance Equivalence for Basic Parallel Processes, *Theoretical Computer Science* 360(1-3):172-192 (2006)
45. A. Liberato, *A Study on Bisimulation Equivalence and Team Equivalence*, Master Thesis of the University of Bologna (supervisor R. Gorrieri), October 2019.
46. R. Milner. A Complete Inference Systems for a Class of Regular Behaviors, *J. Comput. System Sci.* 28: 439-466 (1984)
47. R. Milner. *Communication and Concurrency*, Prentice-Hall (1989)
48. R. Milner, F. Moller, Unique Decomposition of Processes, *Theoretical Computer Science* 107(2):357-363 (1993)
49. R. Meyer, A. Platzer, H. Wehrheim (Eds.), *Correct System Design* – Symposium in Honor of Ernst-Rüdiger Olderog on the Occasion of His 60th Birthday, LNCS 9360, Springer (2015)
50. E.R. Olderog, *Nets, Terms and Formulas*, Cambridge Tracts in Theoretical Computer Science 23, Cambridge University Press (1991)
51. D.M.R. Park, Concurrency and Automata on Infinite Sequences, In Proc. 5th GI-Conference on Theoretical Computer Science, LNCS 104, 167-183, Springer (1981)
52. J.L. Peterson, *Petri Net Theory and the Modeling of Systems*, Prentice-Hall (1981)
53. L. Pomello, G. Rozenberg, C. Simone, A Survey of Equivalence Notions for Net Based Systems, in *Advances in Petri Nets: The DEMON Project*, LNCS 609, 410-472, Springer (1992)
54. R. Paige, R.E. Tarjan, Three Partition Refinement Algorithms, *SIAM Journal of Computing* 16(6):973-989 (1987)
55. A. Rabinovich, B.A. Trakhtenbrot, Behavior Structures and Nets, *Fundamenta Informaticae* 11(4):357-404 (1988)
56. W. Reisig, *Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies*, Springer (2013)
57. D. Sangiorgi, *An Introduction to Bisimulation and Coinduction*, Cambridge University Press (2012)
58. G. Winskel, Event Structures, *Advances in Petri Nets, Part II*, Proceedings of an Advanced Course, Bad Honnef, 1986, LNCS 255, 325-392, Springer (1987)
59. G. Winskel, M. Nielsen, Models for Concurrency, *Handbook of Logic in Computer Science*, Vol. 4, 1-148, Oxford Univ. Press (1995)