Normed Processes, Unique Decomposition, and Complexity of Bisimulation Equivalences ⋆

Sibylle Fröschle and Sławomir Lasota ⋆⋆

Institute of Informatics, Warsaw University, Poland

Abstract. We propose a decision procedure for a general class of normed commutative process rewrite systems and their induced bisimulation equivalences. Our technique is inspired by the polynomial-time algorithm for strong bisimilarity on normed Basic Parallel Processes (BPP), developed by Hirshfeld, Jerrum and Moller. We apply our general technique to derive polynomial-time algorithms for strong bisimilarity on normed BPP with communication and for distributed bisimilarity on all BPP with communication. Moreover, our technique yields the first elementary upper bound for weak and branching bisimilarity on totally normed BPP.

1 Introduction

The idea of unique prime decomposition is relevant in process theory: given a behavioural equivalence one asks whether the commutative monoid of the equivalence classes of processes, with parallel composition as operation, has unique decomposition. This means that each element can be uniquely expressed as a composition of primes; an element is prime if it is not the composition of two non-trivial elements.

Unique decomposition results in process theory have helped to construct decision procedures for bisimulation equivalences on various process classes. A classical example is the algorithm of Hirshfeld et al. [6, 2], which proves that strong bisimilarity on normed Basic Parallel Processes (BPP) is polynomial-time decidable. Using this algorithm as a subroutine, Lasota has shown that distributed bisimilarity on BPP is also polynomial-time decidable [11]. Hirshfeld et al.'s technique does not extend to strong bisimilarity on the full BPP class. Indeed, it has turned out that this problem is PSPACE-complete [14, 8]. The PSPACE decision procedure uses a special technique developed by Jančar, which can also be applied to obtain polynomial-time algorithms for strong bisimilarity on normed BPP and distributed bisimilarity on BPP [9, 10].

Weak bisimilarity on BPP is PSPACE-hard [13], and it is not even known whether the notion is decidable; this is a long-standing open problem. For normed BPP, weak bisimilarity remains PSPACE-hard but decidability has been achieved for slight restrictions thereof [5, 15]; one of them is the class totally

⋆ This work is supported by the EU Research Training Network Games.

⋆⋆ Partially supported by Polish Kbn grant No. 4 T11C 042 25.

normed BPP. However, both algorithms seem to have non-elementary complexity. Branching bisimilarity, which like weak bisimilarity reflects a notion of observable behaviour, is also decidable on totally normed BPP [5].

Two directions seem important to gear up the techniques towards realistic systems: one is to cover process classes with communication; the second is to obtain better techniques for observational equivalences. For BPP_τ, which extends BPP with CCS-style communication, strong bisimilarity restricted to the normed case as well as distributed bisimilarity have been proved decidable [3]. The algorithms are based on unique decomposition results but they both run in exponential time. Jančar's technique is not suited to processes with communciation, and it appears not be applicable to observable semantics either. Thus, we were motivated to take another look at unique decomposition. In this paper we will present a general technique, based on unique decomposition, that applies to a wide range of process semantics.

Our contributions are as follows. (1) We define a general class of *commutative rewrite systems* (*CRS'*) and a general definition of when a CRS is *normed*. With each CRS we associate a notion of bisimilarity. (2) We demonstrate that a wide range of semantics can be captured within our framework. In particular, we show that each of the following equivalences can be expressed as the bisimilarity of a normed CRS: strong bisimilarity on normed BPP_τ, distributed bisimilarity on all BPP_τ, weak and branching bisimilarity on totally normed BPP. (3) We prove that the notions of bisimilarity induced by normed CRS', and hence our four concrete equivalences, satisfy unique decomposition. To our knowledge, unique decomposition of weak and branching bisimilarity on BPP has not been known before. (4) We exhibit a general procedure which, given an *effective* normed CRS, decides its associated notion of bisimilarity. (5) We instantiate this general technique to obtain the following concrete results: strong bisimilarity on normed BPP_τ and distributed bisimilarity on all BPP_τ are polynomial-time decidable; weak and branching bisimilarity on totally normed BPP are decidable in elementary time (we conjecture that even PSPACE upper bound can be obtained within our framework for these two equivalences). To our knowledge all of these results are new. The result on weak bisimilarity is particularly interesting: no elementary upper bound has been achieved before for this equivalence on a reasonably large infinite-state class.

It is standard to define classes of infinite-state processes in terms of rewrite systems [2]. For example, BPP are captured by what is commonly referred to as $(1, \mathcal{P})$-PRS. Our notion of CRS generalizes $(1, \mathcal{P})$-PRS in two ways. Firstly, we allow processes to communicate; the type of communication will be specified by means of a synchronization algebra, a concept that originates from the semantics of concurrency [16]. Secondly, we allow a process to evolve into a tuple of processes rather than only one. This ensures we can capture generalizations of transition systems such as that used by distributed bisimilarity but it is also crucial to capture branching bisimilarity.

Our unique decomposition result is achieved with the help of an important tool of Luttik and Oostrom [12]: a partial commutative monoid has unique decomposition iff it has a *decomposition order*. The concept of decomposition order captures the algebraic properties that typically underly unique decomposition proofs in process theory. Each normed CRS induces a notion of *norm-reducing* bisimulation. We will show that congruences which are norm-reducing bisimulations have a natural decomposition order, and hence by [12] they satisfy unique decomposition. Not only will this result imply unique decomposition for bisimilarity but it will also provide the backbone of our general decision procedure.

Our decision procedure is an extension of Hirshfeld et al.'s technique for strong bisimilarity on normed BPP [6]. In their algorithm bisimilarity is approximated from above, in a small number of steps, by a sequence of norm-preserving congruences which have unique decomposition. We will make prominent what is underlying but hidden in [6]: each approximation is obtained as the greatest norm-reducing bisimulation that contains a refinement of the previous approximation. Combined with our general unique decomposition result this makes the technique applicable to a wide range of processes. Furthermore, we generalize [6] in that our notion of norm-decreasing transition is non-strict.

It is interesting that we can capture the four concrete cases as instantiations of the same type of normed CRS. On the one hand, this makes concrete the connection between strong bisimilarity on normed BPP and distributed bisimilarity on all BPP that has already become apparent from the similarity in their decision procedures [11, 9, 10]. On the other hand, it shows that this connection extends to BPP_τ. Finally, and perhaps most interestingly, it highlights a new connection between observable transitions and communication. Altogether, our framework brings to light structural similarities between a priori very different notions. Thus, we hope our general technique will be applicable to a wide range of process semantics beyond those studied in this paper.

The remainder of the paper is structured according to our contributions. All the missing proofs can be found in the full version of the paper.

2 Normed Commutative Rewrite Systems

(Weighted) Synchronization Algebras. For technical convenience we assume a special action $*$, the *idle action*, which can always happen. A *synchronization algebra* is a pair $\mathrm{Act} = (\mathrm{Act}, \bullet)$ where Act is a set of actions such that $* \notin \mathrm{Act}$, and $\bullet : (\mathrm{Act} \cup \{*\})^2 \to \mathrm{Act} \cup \{*\}$ is a binary, commutative, associative partial operation which satisfies:

(S1) $a \bullet b = *$ iff $a = * = b$;
(S2) $a \bullet * = * \bullet a = a$.

Axiom (S1) is standard. By imposing (S2) we restrict our attention to communication schemes where all actions can occur asynchronously. Typically, we let $a, b, \ldots$ range over actions including $*$.

A *weighted synchronization algebra* is a triple $(Act, \bullet, W)$ where $(Act, \bullet)$ is a synchronization algebra and $W : Act \cup \{*\} \to \mathbb{N}_0$ is a *weight function* such that:

(S3) $W(*) = 0$;

(S4) $a \bullet b = c$ implies $W(c) = W(a) + W(b)$.

Axiom (S3) ensures the idle action has weight 0. (S4) expresses that the weight of an action does not depend on whether or how it arises from synchronization. The weight of a sequence $w = a_1 \ldots a_k$ is $W(w) = W(a_1) + \ldots + W(a_k)$.

CRS'. A *commutative rewrite system* (*CRS*) is a triple $P = (V, \mathrm{Act}, \Gamma)$ where

- V is a finite set of variables,
- Act is a finite synchronization algebra, and
- Γ is a finitely presentable set of rewrite rules of the form $(X, a, \alpha_1, \ldots, \alpha_m)$ where $X \in V$, $a \in \mathrm{Act}$, and $\alpha_i \in V^{\otimes}$. $m \geq 1$ is fixed throughout Γ.

Note that Γ is assumed to be finitely presentable but not necessarily finite; in Section 5 we will further require Γ to be *effective*.

$V^{\otimes}$ is the free commutative monoid over V. Its elements are multisets of variables from V. Variables are thought to represent *elementary processes*, and multisets of variables are interpreted as parallel compositions of elementary processes, where parallel composition is considered modulo associativity, commutativity, and the empty process as unit. The elements of $V^{\otimes}$ are thus thought to represent the *processes* of P. Let X, $Y, \ldots$ range over V, and α, $\beta, \ldots$ over $V^{\otimes}$. We write $\alpha\beta$ for the composition of α and β. The empty process is denoted by ϵ.

The rewrite rules of Γ specify the behaviour of elementary processes: if $(X, a, \alpha_1, \ldots, \alpha_m) \in \Gamma$ then X can perform action a and thereby evolve into the tuple of processes $(\alpha_1, \ldots, \alpha_m)$. We assume the following, for all $X \in V$:

(P1) $(X, *, \alpha_1, \ldots, \alpha_m) \in \Gamma$, for some $\alpha_1, \ldots, \alpha_m$ with $\alpha_m = X$.

We define a *transition relation* for all processes inductively as follows:

1. if $(X, a, \alpha_1, \ldots, \alpha_m) \in \Gamma$ then $X \overset{a}{\twoheadrightarrow} (\alpha_1, \ldots \alpha_m)$;
2. if $\alpha \overset{a}{\twoheadrightarrow} (\alpha_1, \ldots, \alpha_m)$, $\beta \overset{b}{\twoheadrightarrow} (\beta_1, \ldots, \beta_m)$, and $a \bullet b$ is defined then we have $\alpha\beta \overset{a \bullet b}{\twoheadrightarrow} (\alpha_1\beta_1, \ldots, \alpha_m\beta_m)$.

From the m-ary steps we derive single-derivative steps as follows: $\alpha \xrightarrow{a} \alpha'$ iff $\alpha \overset{a}{\twoheadrightarrow} \alpha_1, \ldots, \alpha_m$ with $\alpha_m = \alpha'$. This means we single out the mth derivate to play a special role. The transition relation $\alpha \xrightarrow{a} \alpha'$ is extended to finite words $w \in \mathrm{Act}^*$, written $\alpha \xrightarrow{w} \alpha'$, in a standard way.

Fact 1. *Let a, b, and c range over* $\mathrm{Act} \cup \{*\}$.

1. $\alpha \xrightarrow{*} \alpha$.
2. *If* $\alpha \xrightarrow{a} \alpha'$, $\beta \xrightarrow{b} \beta'$, *and* $a \bullet b$ *is defined then* $\alpha\beta \xrightarrow{a \bullet b} \alpha'\beta'$.
3. *If* $\alpha\beta \xrightarrow{c} \gamma$ *then there are some* a, b, α', β' *such that* $c = a \bullet b$, $\gamma = \alpha'\beta'$, $\alpha \xrightarrow{a} \alpha'$, *and* $\beta \xrightarrow{b} \beta'$.

The *norm* of a process α, denoted by $|\alpha|$, is the smallest weight necessary to reach from α the empty process by a sequence of transitions. Formally, we define:

$$\begin{array}{ll} |\alpha| = \min\{W(w) : \alpha \xrightarrow{w} \epsilon,\ w \in \mathrm{Act}^*\} & \text{if } \alpha \xrightarrow{w} \epsilon \text{ for some } w \in \mathrm{Act}^*, \\ |\alpha| = \infty & \text{otherwise.} \end{array}$$

Norm is additive: if α and β have finite norms then $|\alpha\beta| = |\alpha| + |\beta|$.

Normed CRS'. A *normed CRS* is a triple $(V, \mathrm{Act}, \Gamma)$ such that Act is a weighted synchronization algebra, $(V, \mathrm{Act}', \Gamma)$ is a CRS where Act' is the synchronization algebra underlying Act, and the following holds:

(N1) each process $\alpha \neq \epsilon$ has finite and positive norm: $0 < |\alpha| < \infty$.

We will impose a second condition (N2). For its formulation we need some more definitions. Let $(V, \mathrm{Act}, \Gamma)$ be a normed CRS as defined so far. A transition $\alpha \xrightarrow{a} \alpha'$ is *norm-reducing*, denoted by $\alpha \xrightarrow{a}_{\mathrm{n-r}} \alpha'$, iff it can be extended to a sequence from α to ϵ of minimal weight: $\alpha \xrightarrow{a}_{\mathrm{n-r}} \alpha'$ iff $\alpha \xrightarrow{a} \alpha' \xrightarrow{w} \epsilon$ for some $w \in \mathrm{Act}^*$ with $W(aw) = |\alpha|$. We extend norm-reducing transitions to sequences $v \in \mathrm{Act}^*$ as one could expect: $\alpha \xrightarrow{v}_{\mathrm{n-r}} \alpha'$ iff $\alpha \xrightarrow{v} \alpha' \xrightarrow{w} \epsilon$ for some $w \in \mathrm{Act}^*$ with $W(vw) = |\alpha|$. Due to additivity of norm we obtain the analogue of Fact 1 for $\longrightarrow_{\mathrm{n-r}}$.

Without loss of generality we may assume that the variables of V are given as a sequence $X_1, \ldots, X_k$, ordered by non-decreasing norm: $|X_1| \leq |X_2| \leq \ldots \leq |X_k|$. Norm-reducing transitions are non-strict in that they do not lead to a strict decrease in norm: if $\alpha \xrightarrow{a}_{\mathrm{n-r}} \alpha'$ then $|\alpha'| \leq |\alpha|$. If norm-reducing transitions were strict then a norm-reducing transition from a variable X_i would always lead to a multiset of variables of index strictly smaller than i. We will assume that every normed CRS satisfies a weakening of this property:

(N2) $X_i \xrightarrow{a}_{\mathrm{n-r}} \alpha$ implies $\alpha \in \{X_1, \ldots, X_i\}^{\otimes}$.

(N2) cannot be derived from (N1) in general, but it is satisfied immediately or modulo bisimilarity by all the concrete normed CRS' we will study. With (N2) we obtain the crucial properties of norm-reducing transitions.

Proposition 1. *If* $X_i \xrightarrow{a}_{\mathrm{n-r}} \alpha$ *then* $\alpha \in \{X_1, \ldots, X_{i-1}\}^{\otimes}$ *or* $\alpha = X_i$.

This is easily deduced from (N2) because $|\alpha| \leq |X_i|$.

2.1 Bisimilarity and Norm-Reducing Bisimulations

In this section we work relative to a CRS $P = (V, \mathrm{Act}, \Gamma)$. We associate a notion of bisimilarity with P, defined as follows.

Definition 1. *Given a relation* $R \subseteq V^{\otimes} \times V^{\otimes}$*, we say a pair* $(\alpha, \beta) \in V^{\otimes} \times V^{\otimes}$ satisfies expansion in R *iff:*

1. *whenever* $\alpha \overset{a}{\twoheadrightarrow} (\alpha_1, \ldots, \alpha_m)$ *for some* $a, \alpha_1, \ldots, \alpha_m$ *then there exist* $\beta_1, \ldots, \beta_m$ *such that* $\beta \overset{a}{\twoheadrightarrow} (\beta_1, \ldots, \beta_m)$ *and all pairs* $(\alpha_1, \beta_1), \ldots, (\alpha_m, \beta_m)$ *are in* R*;*
2. *the symmetric condition holds.*

We denote the set of pairs (α, β) *that satisfy expansion in* R *by* $exp(R)$*. A relation* $R \subseteq V^{\otimes} \times V^{\otimes}$ *is a* bisimulation *iff each pair of* R *satisfies expansion in* R*, in other words, iff* $R \subseteq exp(R)$*. We denote the greatest bisimulation by* $\approx$ *and refer to it as* bisimilarity*. If* P *is not clear from the context we also use the term* P*-bisimilarity and write* $\approx_P$*.*

A *congruence* is any equivalence $\equiv$ which is compositional: if $\alpha \equiv \alpha'$ and $\beta \equiv \beta'$ then $\alpha\beta \equiv \alpha'\beta'$. We show that bisimilarity is a congruence; and that if $\equiv$ is a congruence then the relation $\exp(\equiv)$ is a congruence too.

if P is normed, then there is also a notion of *norm-reducing* bisimulation associated with P.

Definition 2. *Given a relation* $R \subseteq V^{\otimes} \times V^{\otimes}$*, we say a pair* $(\alpha, \beta) \in V^{\otimes} \times V^{\otimes}$ *satisfies* norm-reducing expansion *in* R*, written* $(\alpha, \beta) \in$ *n-r-exp*(R)*, iff:*

1. *whenever* $\alpha \xrightarrow{a}_{\mathrm{n-r}} \alpha'$ *for some* a*,* α' *then there exists* β' *such that* $\beta \xrightarrow{a} \beta'$ *and* $(\alpha', \beta') \in R$*;*
2. *the symmetric condition holds.*

A relation $R \subseteq V^{\otimes} \times V^{\otimes}$ *is a* norm-reducing bisimulation *iff* $R \subseteq$ *n-r-exp*(R)*.*

By definition each bisimulation is a norm-reducing bisimulation. We say that a relation $R \subseteq V^{\otimes} \times V^{\otimes}$ is *norm-preserving* iff $|\alpha| = |\beta|$ for each $(\alpha, \beta) \in R$. We prove that each norm-reducing bisimulation, and thus each bisimulation, is norm-preserving. As a consequence we obtain: in each norm-reducing bisimulation, every (necessarily norm-reducing) transition will be matched by a transition that is *norm-reducing* itself. We will assume this property in the sequel without further mentioning.

Given a congruence $\equiv$, the greatest norm-reducing bisimulation contained in $\equiv$ always exists, and is given by the set-theoretic union of all norm-reducing bisimulations contained in $\equiv$. We denote this relation by $\mathrm{gnrb}(\equiv)$.

Proposition 2. *Given a congruence* $\equiv$*,* $gnrb(\equiv)$ *is a congruence.*

3 Concrete Normed Commutative Rewrite Systems

Finally, we will get to know concrete CRS', and see how their associated notions of bisimilarity coincide with well-known equivalences. We cannot be completely self-contained here; in particular we refer the reader to standard literature for the definitions of the well-known equivalences [3, 2, 5].

In the following, assume an infinite set of variables $X, Y, \ldots$. We will use definitions Δ contaning a finite number of rules of the form $X \xrightarrow{a} \alpha$; in short we write $\Delta = \{X \xrightarrow{a} \alpha\}$. We always assume that if a variable occurs on the right side of a rule it will also appear on the left side of some rule.

Synchronization Algebra for BPP$_\tau$. Assume an infinite set of actions $Act = a, b, c, \ldots$, and an infinite set of complementary actions $\overline{Act} = \bar{a}, \bar{b}, \bar{c}, \ldots$. Let τ be a special element not contained in Act, the *silent* action. BPP$_\tau$ uses CCS-style communication where two complementary actions can synchronize forming τ. The silent action can also occur by itself and any action can occur asynchronously. The corresponding synchronization algebra, Act_{CCS}, is then $(Act \cup \overline{Act} \cup \{\tau\}, \bullet)$ where $\bullet$ is defined by the table on the right. We extend Act_{CCS} by a weight function W, defined by: $W(*) = 0$, $W(a) = 1$ for all $a \in Act \cup \overline{Act}$, and $W(\tau) = 2$.

	$*$	a	$\bar{a}$	b	$\bar{b}$	$\ldots$	τ
$*$	$*$	a	$\bar{a}$	b	$\bar{b}$	$\ldots$	τ
a	a		τ			$\ldots$	
$\bar{a}$	$\bar{a}$	τ				$\ldots$	
b	b				τ	$\ldots$	
.	.	.	.	.	.	$\ldots$	.
τ	τ					$\ldots$	

BPP$_\tau$ under Interleaving Semantics. A BPP$_\tau$ definition in normal form is a finite set $\Delta = \{X \xrightarrow{a} \alpha\}$ where $a \in Act_{CCS}$. Δ induces a CRS $P = (V, \mathrm{Act}, \Gamma)$ as follows: V and Act are derived from Δ and Act_{CCS} in the obvious way; Γ is Δ with the additional rule $(X, *, X)$ for all $X \in V$. The transition relation $\twoheadrightarrow_P$ coincides with that obtained from the standard operational semantics for BPP$_\tau$ definition Δ. This implies:

Fact 2. *$\approx_P$ coincides with classical bisimilarity on processes of Δ.*

Our notion of norm for processes of P coincides with that typically associated with processes of Δ [2]. Δ is normed iff every non-empty process has finite norm.

Fact 3. *If Δ is normed then P is a normed CRS.*

BPP$_\tau$ under Distributed Semantics. Distributed bisimilarity reflects that whenever a process performs an action it can be thought to evolve into a *local remainder* and a *concurrent remainder*. Any guarded BPP$_\tau$ definition can be transformed (in polynomial-time) into a BPP$_\tau$ definition in distributed normal form [3], that is a finite set $\Delta = \{X \xrightarrow{a} (\alpha_l, \alpha_c)\}$ where $a \in Act_{CCS}$ and the following relation $\prec$ on the variables of Δ is irreflexive: $\prec$ is the transitive closure of the relation $\prec_1$ where $Y \prec_1 X$ iff there is a rule $(X, a, \alpha_l, \alpha_c)$ in Δ such that $Y \in \alpha_c$.

Δ induces a CRS $P = (V, \mathrm{Act}, \Gamma)$ as follows: V and Act are derived from Δ and Act_{CCS} in the obvious way; Γ is Δ with the additional rule $(X, *, 0, X) \in \Gamma$ for all $X \in V$. The transition relation $\twoheadrightarrow_P$ coincides with that obtained from the distributed operational semantics for Δ. Thus, we have:

Fact 4. *$\approx_P$ coincides with distributed bisimilarity on processes of Δ.*

We assume that the variables of P are ordered according to $\prec$: $X_i \prec X_j$ implies $i < j$. Then it is immdediate:

Fact 5. *P is a normed CRS.*

BPP under Weak Semantics. Let Δ be a BPP definition, i.e., a BPP_τ definition that does not contain any actions of $\overline{Act}$. By $\longrightarrow$ denote the transition relation associated with Δ under the standard operational semantics, or equivalently, the transition relation of the CRS induced by Δ.

Given two processes α, β, we write $\alpha\ (\xrightarrow{\tau})^*\ \beta$ if β can be reached from α by an arbitrary number of τ transitions. Weak bisimilarity abstracts away from silent actions by reflecting the following weak transition relation:

$$\alpha \overset{a}{\Longrightarrow} \beta \text{ iff } \begin{cases} \alpha\ (\xrightarrow{\tau})^*\ \beta & \text{if } a = \tau, \\ \alpha\ (\xrightarrow{\tau})^*\ \beta' \ \xrightarrow{a} \beta''\ (\xrightarrow{\tau})^*\ \beta \text{ for some } \beta', \beta'' & \text{otherwise.} \end{cases}$$

To capture weak bisimilarity as a notion of CRS bisimilarity we need to exhibit a CRS P such that $\twoheadrightarrow_P$ coincides with $\Longrightarrow$. We achieve this by defining the *synchronization algebra of weak actions*, $Act_W = (Act_W, \bullet)$ where Act_W and $\bullet$ are as follows.

Act_W is the set of *weak actions* given by $\{\tau^* a \tau^* : a \in Act\} \cup \{\tau^*\}$. The action names indicate the type of weak transition a weak action is thought to represent; but note that we could have chosen other names just as well. When a process α performs a weak transition then several of α's elementary processes may contribute to it. To reflect this we allow weak actions to occur as synchronized actions: two processes may jointly perform action $\tau^* a \tau^*$ if one process performs $\tau^* a \tau^*$ and the other process performs τ^*. Similarly, two processes may jointly perform action τ^* if they both perform τ^*. Formally, $\bullet$ is given by the following table:

	$*$	τ^*	$\tau^* a \tau^*$	$\tau^* b \tau^*$	$\ldots$
$*$	$*$	τ^*	$\tau^* a \tau^*$	$\tau^* b \tau^*$	$\ldots$
τ^*	τ^*	τ^*	$\tau^* a \tau^*$	$\tau^* b \tau^*$	$\ldots$
$\tau^* a \tau^*$	$\tau^* a \tau^*$	$\tau^* a \tau^*$			$\ldots$
$\tau^* b \tau^*$	$\tau^* b \tau^*$	$\tau^* b \tau^*$			$\ldots$
.	.	.	.	.	$\ldots$

We associate a weight function W with Act_W, defined by: $W(*) = 0$, $W(\tau^*) = 0$, and $W(\tau^* a \tau^*) = 1$ for all $a \in Act$.

Δ induces a CRS $P = (V, \mathrm{Act}, \Gamma)$ as follows: V and Act are derived from V and Act_W in the obvious way; Γ is defined by: $(X, \tau^*, \alpha) \in \Gamma$ iff $X \overset{\tau}{\Longrightarrow} \alpha$; $(X, \tau^* a \tau^*, \alpha) \in \Gamma$ iff $X \overset{a}{\Longrightarrow} \alpha$; $(X, *, X) \in \Gamma$.

The transition relation $\twoheadrightarrow_P$ indeed captures the weak transition relation of Δ: $\alpha \overset{\tau^*}{\twoheadrightarrow} \beta$ iff $\alpha \overset{\tau}{\Longrightarrow} \beta$; and, $\alpha \overset{\tau^* a \tau^*}{\twoheadrightarrow} \beta$ iff $\alpha \overset{a}{\Longrightarrow} \beta$. Thus, we obtain:

Fact 6. *$\approx_P$ coincides with weak bisimilarity on processes of Δ.*

Our notion of norm for processes of P coincides with that typically associated with processes of Δ in the context of weak bisimilarity. Δ is totally normed if each process has finite and positive norm. Assume P is induced from a totally normed Δ. Then it is immediate that P satisfies (N1) of the definition of normed CRS. However, condition (N2) is not necessarily obtainable: simply consider two variables X, Y such that $X \stackrel{\tau}{\Longrightarrow} Y \stackrel{\tau}{\Longrightarrow} X$. On the other hand, it is easy to check that such X and Y will be weakly bisimilar ($|X| = |Y|$ in particular). Thus, we can transform P into a weakly bisimilar CRS P' by removing one of X or Y, say Y, whenevever $X \stackrel{\tau}{\Longrightarrow} Y \stackrel{\tau}{\Longrightarrow} X$. This also involves substituting X for Y whenever Y appears on the right side of a rule in Γ. Then the following relation $\preceq$ is a partial order: $X \preceq Y$ iff $|X| < |Y|$ or $Y \stackrel{\tau}{\Longrightarrow} X$. We assume that the variables of P' are ordered according to $\prec$: $X_i \prec Y_j$ implies $i < j$. Then P' will indeed satisfy (N2): whenever $X_i \xrightarrow{a}_{\mathrm{n-r}} \alpha$ then $\alpha \in \{X_1, \ldots, X_{i-1}\}^{\otimes}$.

Fact 7. *P' is a normed CRS.*

BPP under Branching Semantics. Branching bisimilarity abstracts away from silent actions while preserving the branching structure in more detail than weak bisimilarity. It can be captured by considering the following type of observable transitions:

$$\alpha \stackrel{a}{\Longrightarrow} (\beta_1, \beta_2) \text{ iff } \begin{cases} \alpha \ (\xrightarrow{\tau})^* \ \beta_1 \xrightarrow{a} \beta_2, \text{ or} \\ \alpha \ (\xrightarrow{\tau})^* \ \beta_1 = \beta_2 \text{ and } a = \tau. \end{cases} \tag{1}$$

(In fact this is based on the notion of *semi-branching bisimulation* [1]). Accordingly, the set of branching-observable actions is now $\{\tau^* a : a \in \mathit{Act} \cup \{\tau\}\} \cup \{\tau^*\}$. Note that we distinguish between τ^* and $\tau^*\tau$, which correspond to the two different cases in (1). By similar considerations as for weak bisimilarity we arrive at the following synchronization algebra.

	$*$	τ^*	$\tau^*\tau$	τ^*a	τ^*b	$\ldots$
$*$	$*$	τ^*	$\tau^*\tau$	τ^*a	τ^*b	$\ldots$
τ^*	τ^*	τ^*	$\tau^*\tau$	τ^*a	τ^*b	$\ldots$
$\tau^*\tau$	$\tau^*\tau$	$\tau^*\tau$				$\ldots$
τ^*a	τ^*a	τ^*a				$\ldots$
τ^*b	τ^*b	τ^*b				$\ldots$
.	.	.	.	.	.	$\ldots$

Clearly $\tau^* \bullet \tau^* = \tau^*$ and $\tau^* \bullet \tau^* a = \tau^* a$, but $\tau^*\tau \bullet \tau^* a$ is undefined. Hence necessarily $\tau^* \neq \tau^*\tau$. On the other hand, branching bisimilarity allows to match a τ^* step by $\tau^*\tau$ and vice versa. This motivates the following definition of Γ:

$$\begin{aligned} (X, \tau^*, \alpha_1, \alpha_2) \in \Gamma \quad &\text{iff} \quad X \ (\xrightarrow{\tau})^* \ \alpha_1 = \alpha_2, \\ (X, \tau^* a, \alpha_1, \alpha_2) \in \Gamma \quad &\text{iff} \quad X \ (\xrightarrow{\tau})^* \ \alpha_1 \xrightarrow{(a)} \alpha_2, \end{aligned}$$

where $\alpha_1 \xrightarrow{(a)} \alpha_2$ means $\alpha_1 \xrightarrow{a} \alpha_2$ or $a = \tau$ and $\alpha_1 = \alpha_2$. Note that if $(X, \tau^*, \alpha_1, \alpha_2) \in \Gamma$ then $(X, \tau^*\tau, \alpha_1, \alpha_2) \in \Gamma$ as well. Additionally we put $(X, *, X, X) \in \Gamma$. This induces a CRS P and we obtain:

Fact 8. *$\approx_P$ coincides with branching bisimilarity on processes of Δ.*

If Γ is totally normed, then P can be easily made normed, similarly as before.

4 Unique Decomposition

We now prove our unique decomposition results. As explained in the introduction we build on a theorem by Luttik and Oostrom that characterizes when a partial commutative monoid has unique decomposition. We first introduce this tool, specialized to the case of commutative monoids.

Let M be a commutative monoid with identity element e. An element p of M is called *indecomposable* if $p \neq e$ and $p = xy$ implies $x = e$ or $y = e$. A *decomposition* of x in M is a finite multiset $p_1^{a_1} \dots p_r^{a_r}$ of indecomposable elements of M such that $x = p_1^{a_1} \dots p_r^{a_r}$. If every element of M has a unique decomposition (up to multiset equality) then we say that M has *unique decomposition.*

A partial order $\preceq \subseteq M \times M$ is a *decomposition order* if

(i) it is *well-founded*: every non-empty subset of M has a $\preceq$-minimal element;
(ii) e is the least element of M wrt. $\preceq$: $e \preceq x$ for all $x \in M$;
(iii) it is *strictly compatible*: for all x, y, $z \in M$, if $x \prec y$ then $xz \prec yz$;
(iv) it is *strictly precompositional*: for all x, y, $z \in M$, $x \preceq yz$ implies $x = y'z'$ for some $y' \preceq y$ and $z' \preceq z$; and
(v) it is *Archimedean*: for all x, $y \in M$, $x^n \prec y$ for all $n \in \mathbb{N}$ implies $x = e$.

Corollary 1 (by Corollary 3.15 from [12]). *A commutative monoid has unique decomposition iff it has a decomposition order.*

With the help of this tool we now prove our unique decomposition results. Let $P = (V, \mathrm{Act}, \mathit{\Gamma})$ be a normed CRS, and $\sim \subseteq V^{\otimes} \times V^{\otimes}$ be a congruence. $V^{\otimes}/{\sim}$ is the quotient of $V^{\otimes}$ by $\sim$, i.e., the set of equivalence classes of $V^{\otimes}$ wrt. $\sim$. We write $[\alpha]_\sim$ for the equivalence class containing α. Since $\sim$ is a congruence we can define on $V^{\otimes}/{\sim}$ a binary operation as follows: $[\alpha]_\sim[\beta]_\sim = [\alpha\beta]_\sim$. It is easy to verify that $V^{\otimes}/{\sim}$ is a commutative monoid under this operation with identity element $[\epsilon]_\sim$.

Theorem 1. *Let $\sim$ be a congruence which is a norm-reducing bisimulation. The commutative monoid $V^{\otimes}/{\sim}$ has unique decomposition.*

Assume $\sim$ to be a congruence which is a norm-reducing bisimulation. We can exploit the properties of norm-reducing transitions to exhibit a decomposition order on $V^{\otimes}/{\sim}$. Define a relation $\preccurlyeq$ on $V^{\otimes}/{\sim}$ as follows:

$$[\beta]_\sim \preccurlyeq [\alpha]_\sim \iff \alpha \sim \alpha' \xrightarrow{w}_{\mathrm{n-r}} \beta' \sim \beta \text{ for some } \alpha',\ \beta',\ \text{and } w \in \mathrm{Act}^*.$$

The following proposition establishes that $\preccurlyeq$ is a decomposition order. Theorem 1 is then an immediate consequence of Corollary 1.

Proposition 3. *1. $\preccurlyeq$ is a partial order.*
2. $\preccurlyeq$ is a decomposition order.

Since $\approx_P$ is a congruence and every bisimulation is a norm-reducing bisimulation we immediately obtain:

Corollary 2. *The commutative monoid $V^{\otimes}/{\approx_P}$ has unique decomposition.*

Finally, in view of our algorithm we derive a more compact formulation of our unique decomposition results. The formulation is in the style of [6], and makes use of the assumption that decompositions preserve norm, and that variables are given in order of non-decreasing norm.

Assume $\sim$ to be a norm-preserving congruence. An elementary process X_i is *decomposable* wrt. $\sim$ if there exists $\alpha \in \{X_1, \ldots, X_{i-1}\}^{\otimes}$ such that $X_i \sim \alpha$. Otherwise X_i is *prime* wrt. $\sim$. Let $P_1, \ldots, P_r$ be the primes of V wrt. $\sim$, ordered consistently with V itself. We say that $\sim$ has *unique decomposition* if for any processes $\alpha = P_1^{a_1} \ldots P_r^{a_r}$ and $\beta = P_1^{b_1} \ldots P_r^{b_r}$, $\alpha \sim \beta$ implies $a_1 = b_1, \ldots, a_r = b_r$.

Corollary 3.

1. *Let $\sim$ be a congruence which is a norm-reducing bisimulation. Then $\sim$ has unique decomposition.*
2. *$\approx_P$ has unique decomposition.*

5 Decision Procedure

In this section we provide our general decision procedure. As explained in Section 1 it is an extension of [6]. Let $P = (V, \mathrm{Act}, \Gamma)$ be a normed CRS. Assume that P is presented in a finitary way, e.g., by a BPP process definition Δ (c.f. Section 3). Throughout this section, denote the size of the presentation by n. We consider the following bisimilarity problem: for a given normed CRS P and two variables X_i, X_j, decide whether $X_i \approx_P X_j$.

For the purpose of generality, we work in this section with an arbitrary complexity class $\mathcal{T}$. For convenience we need to assume that $\mathcal{T}$ is induced by some function $T(n)$ satisfying $T(n) \geq n$, in the following way. Let $\mathcal{T}_k$ denote the complexity class determined by a bound $T(n^k)$, $k > 0$. We leave unspeficied whether the bound concerns space or time. Then we assume:

$$\mathcal{T} = \bigcup_{k>0} \mathcal{T}_k.$$

In particular, $\mathcal{T}$ always includes P (polynomial time).

We prove in this section the main result of this paper (c.f. Theorem 3 below) that states that the bisimilarity problem is in $\mathcal{T}$ whenever P is $\mathcal{T}$-effective (c.f. Definition 4 below).

The initial insight behind the procedure is this: since $\approx_P$ satisfies unique decomposition it can be represented by a *unique decomposition base* [6].

Definition 3. *A* (norm-preserving) unique decomposition base *for the variables $V_i = \{X_1, \ldots, X_i\}$ of V is a pair $D = (\Pi, \Gamma)$, where $\Pi \subseteq V_i$ is a set of* primes, *and Γ is a set of pairs (X, α) such that $\alpha \in \Pi^{\otimes}$, $|X| = |\alpha|$, and there is one pair for each variable $X \in V_i \setminus \Pi$. Γ is thought to specify for each non-prime variable*

X a decomposition into primes. The primes of a decomposition base are, a priori, arbitrarily chosen and not to be confused with the primes wrt. a given congruence. D induces a relation $\equiv_D$ on $V_i^\otimes$: $\alpha \equiv_D \beta$ iff the prime decompositions of α and β are equal. For $V_i = V$ we simply call D a decomposition base.

A process α is called *small* if $|\alpha| \leq |X|$ for some variable X. In all concrete cases considered in Section 6 the norm of a small process will be at most exponential wrt. n. In such case a small process $X_1^{a_1} \ldots X_k^{a_k}$ can be represented in space polynomial wrt. n, by binary encodings of exponents $a_1, \ldots, a_k$.

Definition 4. *We say P is $\mathcal{T}$-*effective *iff the following problems are in $\mathcal{T}$ (the presentation of P is a part of input in each case):*

(a) Given an action a, and a small process α, enumerate all processes from the set n-r-succ$_a(\alpha) = \{\beta : \alpha \xrightarrow{a}_{\text{n-r}} \beta\}$.
(b) Given a transition $X \xrightarrow{a}_{\text{n-r}} \alpha$, and a unique decomposition base D, enumerate all processes from the set n-r-pred$_{D,a}(\alpha) = \{\beta : \beta \xrightarrow{a}_{\text{n-r}} \alpha'$ for some $\alpha' \equiv_D \alpha\}$.
(c) Given small processes α, β, and a unique decomposition base D, decide whether $(\alpha, \beta) \in exp(\equiv_D)$.

In points (a)–(c) we extracted the relevant sub-problems encountered in our algorithm. Observe that in (a) and (b) we do not require to compute the whole sets n-r-succ$_a(\alpha)$ and n-r-pred$_{D,a}(\alpha)$; we only need to *enumerate* the processes from these sets, i.e, to generate them in a systematic way, for further processing.

Proposition 4. *The norms of all variables are computable in polynomial time.*

Let $\equiv$ be a norm-preserving congruence and recall from Section 2.1 that the greatest norm-reducing bisimulation contained in $\equiv$ exists and is denoted by gnrb($\equiv$). By Prop. 2 and Corollary 3(1) gnrb($\equiv$) has unique decomposition, and is thus representable by a unique decomposition base D. Our core insight is that due to Prop. 1 we can construct D inductively, and thus efficiently.

Using this insight we can approximate bisimilarity from above by a sequence of norm-preserving congruences. We start with the greatest such relation, say $\equiv$, and compute the decomposition base D that represents gnrb($\equiv$). If $\equiv_D$ is a bisimulation then we have already reached bisimilarity. Otherwise we can move strictly closer to bisimilarity by intersecting $\equiv_D$ with exp($\equiv_D$). That is, we consider a new congruence $\equiv'$ defined by $\alpha \equiv' \beta$ iff $\alpha \equiv_D \beta$ and (α, β) satisfies expansion in $\equiv_D$. We continue by computing the unique decomposition base that represents gnrb($\equiv'$), and proceed in this fashion until finally we will hit bisimilarity. First of all, we prove the core insight.

Theorem 2. *Let $\equiv$ be a norm-preserving congruence decidable in $\mathcal{T}$. Then a unique decomposition base D can be computed in $\mathcal{T}$ such that $\equiv_D = gnrb(\equiv)$.*

Proof. For each $i \in [1, \ldots, k]$ we define, by induction, a unique decomposition base $D_i = (\Pi_i, \Gamma_i)$ for $\{X_1, \ldots, X_i\}^\otimes$, with $\Pi_i = \{P_1, \ldots, P_r\}$, such that:

(1) each P_j is prime wrt. gnrb($\equiv$); and
(2) $(X_j, P_1^{x_1} \dots P_r^{x_r}) \in \Gamma_i$ implies $(X_j, P_1^{x_1} \dots P_r^{x_r}) \in$ gnrb($\equiv$).

Assume there exists a unique decomposition base D_i for $\{X_1, \dots, X_i\}^{\otimes}$ that satisfies these two properties. Then the following three claims will hold:

Claim 1. $\equiv_{D_i} =$ gnrb($\equiv$) holds on the set $\{X_1, \dots, X_i\}^{\otimes}$.

Claim 2. Let α be a composition $P_1^{x_1} \dots P_r^{x_r}$. Then $(X_{i+1}, \alpha) \in$ gnrb($\equiv$) iff $X_{i+1} \equiv \alpha$ and $(X_{i+1}, \alpha) \in$ n-r-exp($\equiv_{D_i} \cup \{(X_{i+1}, \alpha)\}$).

Claim 3. It can be decided in $\mathcal{T}$ whether there is $\alpha = P_1^{x_1} \dots P_r^{x_r}$ that satisfies the right-hand side of Claim 2. If there is such α it can be found in $\mathcal{T}$.

Claim 1 specialized to $i = k$ ensures $\equiv_{D_k} =$ gnrb($\equiv$) as required. Claim 2 indicates how to extend D_i to a unique decomposition base D_{i+1} that also satisfies Properties (1) and (2). By Claim 2, Property (1), and since gnrb($\equiv$) has unique decomposition there is at most one $\alpha = P_1^{x_1} \dots P_r^{x_r}$ such that $X_{i+1} \equiv \alpha$ and $(X_{i+1}, \alpha) \in$ n-r-exp($\equiv_{D_i} \cup \{(X_{i+1}, \alpha)\}$). If there is such α, declare it to be the decomposition of X_{i+1} in D_{i+1}. Otherwise declare X_{i+1} to be prime in D_{i+1}. It is clear that Properties (1) and (2) are satisfied. By Claim 3 the extension of D_i to D_{i+1} can be computed in $\mathcal{T}$. Then overall, D_k can also be computed in $\mathcal{T}$. $\square$

Apart from our core theorem we require two lemmas, which are analogous to insights of [6]. The first lemma gives us the means to efficiently check whether we have reached bisimilarity.

Lemma 1. *Let $D = (\Pi, \Gamma)$ be a unique decomposition base.*

1. *$\equiv_D$ is a bisimulation iff Γ satisfies expansion in $\equiv_D$, that is $\Gamma \subseteq$ exp($\equiv_D$).*
2. *It can be checked in $\mathcal{T}$ whether $\equiv_D$ is a bisimulation.*

The second lemma ensures that $\equiv_D \cap$ exp($\equiv_D$) is a congruence as required, and strictly refines $\equiv_D$ whenever we have not reached bisimilarity yet.

Lemma 2. *Let D be a unique decomposition base.*

1. *$\equiv_D \cap$ exp($\equiv_D$) is a norm-preserving congruence that can be decided for small processes in $\mathcal{T}$.*
2. *If $\equiv_D$ is not a bisimulation then some variable decomposable wrt. $\equiv_D$ is prime wrt. $\equiv_D \cap$ exp($\equiv_D$).*

The algorithm consists of the loop outlined below. In each iteration a given congruence $\equiv$ is refined to a congruence $\equiv_D$ that is represented by a unique decomposition base D. Then $\equiv_D$ is in turn reduced to a new congruence, which provides the value of $\equiv$ for the next iteration. Note that $\equiv$ will not be kept explicitly but be available in terms of its decision procedure. The following two invariants hold:

Invariant 1: $\equiv$ is a norm-preserving congruence decidable for small processes in $\mathcal{T}$.

Invariant 2: $\equiv$ and $\equiv_D$ both subsume bisimilarity: $\approx \subseteq \equiv$ and $\approx \subseteq \equiv_D$.

(1) Let $\equiv$ be the congruence defined by $\alpha \equiv \beta$ iff $|\alpha| = |\beta|$.
(2) Compute a unique decomposition base D such that $\equiv_D = \text{gnrb}(\equiv)$.
This can be done in $\mathcal{T}$ by using Theorem 2.
(3) If $\equiv_D$ is a bisimulation then halt and return D.
The condition can be checked in $\mathcal{T}$ by Lemma 1(2).
(4) Otherwise, redefine $\equiv$ to be the congruence $\equiv_D \cap \exp(\equiv_D)$. Go to step (2).

Claim 4 (Invariants). Invariants 1 and 2 are indeed satisfied.

After step (1), Invariant 1 holds by the definition of $\equiv$ and because by Prop. 4 $\equiv$ is computable in linear time. Invariant 2 follows since $\approx$ is clearly norm-preserving. Step (2) clearly preserves Invariant 2: $\approx$ is a norm-reducing bisimulation contained in $\equiv$ while $\equiv_D$ is the greatest such relation. After Step (4), Invariant 1 holds by Lemma 2(1). Invariant 2 follows since $\approx \subseteq \equiv_D$ and $\approx = \exp(\approx) \subseteq \exp(\equiv_D)$.

Claim 5 (Termination). The algorithm halts after at most n iterations.

By definition $\equiv_D \cap \exp(\equiv_D)$ is finer than $\equiv_D$. This implies that if a variable X_i is prime in D it will also be prime in the decomposition base of the next iteration. Lemma 2(2) ensures that on each iteration at least one variable becomes prime that was not prime before. Thus, the number of iterations is indeed bounded by n.

Claim 6 (Complexity). The algorithm works in $\mathcal{T}$.

This follows from Claim 5 and because each iteration can be completed in $\mathcal{T}$.

Claim 7 (Correctness). The unique decomposition base D output by the algorithm represents bisimilarity: $\equiv_D = \approx$.

The inclusion $\approx \subseteq \equiv_D$ follows by Invariant 2 while the opposite inclusion is immediate by $\equiv_D$ being a bisimulation. Altogether we have proved:

Theorem 3. *For $\mathcal{T}$-effective normed CRS' P the bisimilarity problem is in $\mathcal{T}$.*

6 Results for Concrete Cases

Finally we apply Theorem 3 to the concrete normed CRS' considered in Section 3.

Theorem 4.
(i) Strong bisimilarity on normed BPP_τ is decidable in polynomial time.
(ii) Distributed bisimilarity on BPP_τ is decidable in polynomial time.
(iii) Weak bisimilarity on totally normed BPP is decidable in elementary time.

(iv) Branching bisimilarity on totally normed BPP is decidable in elementary time.

The most direct way to obtain the elementary bounds for clause (c) in Def. 4 in the last two cases is by a Presburger encoding combined with an insight inherent in Theorem 3.3 of [4]. We conjecture that the complexity can be improved as much as to PSPACE, by directly maintaining semi-linear sets and by exploiting relatiohships between (totally normed) BPP and commutative context-free grammars [7]. We plan to investigate the complexity issues in detail in the full version of the paper.

Finally, we would like to emphesise generality of our approach. The above four bisimilarities are in fact only few relavant examples. We strongly believe than our approach is applicable to most of bisimilarities over BPP-like process algebras sufficiently "normed" to exhibit the unique decomposition property.

References

1. T. Basten. Branching bisimilarity is an equivalence indeed! *Information Processing Letters*, 58(3):141–147, 1996.
2. O. Burkart, D. Caucal, F Moller, and B. Steffen. Verification of infinite structures. In *Handbook of Process Algebra*, pages 545–623. Elevier, 2001.
3. S. Christensen. *Decidability and Decomposition in process algebras.* PhD thesis, Dept. of Computer Science, University of Edinburgh, UK, 1993.
4. J. Esparza. Petri nets, commutative context-free grammars, and basic parallel processes. *Fundamenta Informaticae*, 31:13–26, 1997.
5. Y. Hirshfeld. Bisimulation trees and the decidability of weak bisimulations. *Electr. Notes Theor. Comput. Sci.*, 5, 1996.
6. Y. Hirshfeld, M. Jerrum, and F. Moller. A polynomial time algorithm for deciding bisimulation equivalence of normed Basic Parallel Processes. *Mathematical Structures in Computer Science*, 6:251–259, 1996.
7. Dung T. Huynh. Commutative grammars: The complexity of uniform word problems. *Information and Control*, 57:21–39, 1983.
8. P. Jančar. Bisimilarity of Basic Parallel Processes is PSPACE-complete. In *Proc. LICS'03*, pages 218–227, 2003.
9. P. Jančar and M. Kot. Bisimilarity on normed Basic Parallel Processes can be decided in time $O(n^3)$. In *Proc. AVIS'04, ENTCS.* Elsevier, to appear.
10. P. Jančar and Z. Sawa. On distributed bisimilarity over Basic Parallel Processes. In *Proc. AVIS'05, ENTCS.* Elsevier, to appear.
11. S. Lasota. A polynomial-time algorithm for deciding true concurrency equivalences of Basic Parallel Processes. In *MFCS'03*, LNCS 2747, pages 521–530, 2003.
12. B. Luttik and V. van Oostrom. Decomposition orders—another generalisation of the fundamental theorem of arithmetic. *Theor. Comp. Sci.*, 335(2-3):147–186, 2005.
13. J. Srba. Complexity of weak bisimilarity and regularity for BPA and BPP. *Mathematical Structures in Computer Science*, 13:567–587, 2003.
14. J. Srba. Strong bisimilarity of simple process algebras: Complexity lower bounds. *Acta Informatica*, 39:469–499, 2003.
15. C. Stirling. Decidability of weak bisimilarity for a subset of basic parallel processes. In *FOSSACS'01*, LNCS 2030, pages 379–393, 2001.
16. G. Winskel. Synchronization trees. *Theor. Comput. Sci.*, 34:33–82, 1984.