# Semantic reachability for simple process algebras

Richard Mayr*

**Abstract**

This paper is an approach to combine the reachability problem with semantic notions like bisimulation equivalence. It deals with questions of the following form: Is a state reachable that is bisimulation equivalent to a given state ? Here we show some decidability results for process algebras and Petri nets.

**Keywords:** bisimulation, reachability problem, process algebras, Petri nets

## 1 Introduction

The reachability problem plays an important role in the theory of concurrent systems. The question is if a given state is reachable from the initial state by a sequence of actions. The complexity of this problem has been extensively studied (for example it is decidable and EXPSPACE-hard for general Petri nets and NP-complete for Basic Parallel Processes (BPP) [Esp95a]).

Here we generalize the reachability problem by regarding classes of semantically equivalent states instead of single states. The question is now if a state is reachable (from the initial state) that is a member of a given class. In other words: Is it possible to reach a state that is at least semantically equivalent to a given state ? It is natural to choose strong bisimulation equivalence as semantic equivalence, as it has become one of the most successful equivalence notions in concurrency theory. We will call this new problem the *bisimulation-reachability problem*.

The question is now for which classes of concurrent systems this problem is decidable, and if efficient algorithms can be found.

In section 2 we define strong bisimulation, the bisimulation-reachability problem and several process algebras. Section 3 contains some hardness results for the bisimulation-reachability problem. In sections 4, 5 and 6 we study the bisimulation-reachability problem for Basic Parallel Processes (BPP), context-free processes and normed PA-processes. Section 7 is about a related problem called the non-bisimulation-reachability problem and section 8 describes some open problems.

*Address: Institut für Informatik, Technische Universität München, Arcisstr. 21, D-80290 München, Germany; e-mail: `mayrri@informatik.tu-muenchen.de`

## 2 Preliminaries

**Definition 2.1** A binary relation $R$ over the states of a labeled transition system (LTS) is a *strong bisimulation* (often simply called bisimulation) iff

$$\begin{aligned}\forall (s_1, s_2) \in R\ \forall a \in Act.\ \ & (s_1 \xrightarrow{a} s_1' \ \Rightarrow \exists s_2 \xrightarrow{a} s_2'.\ s_1' R s_2') \ \wedge \\ & (s_2 \xrightarrow{a} s_2' \ \Rightarrow \exists s_1 \xrightarrow{a} s_1'.\ s_1' R s_2')\end{aligned}$$

There is always a largest bisimulation, which is an equivalence relation, denoted by $\sim$. Two states $s_1$ and $s_2$ are called *strongly bisimulation equivalent* (or strongly bisimilar) iff $s_1 \sim s_2$.

The most general form of the *bisimulation-reachability problem* can be formulated for labeled transition systems (LTS):
**Instance:** An LTS $R$ with initial state $s_0$ and a state $s$ in $R$.
**Question:** Is there a state $s'$ and a sequence of transitions $\sigma$ s.t. $s_0 \xrightarrow{\sigma} s'$ and $s \sim s'$ ?

In the sequel we consider transition systems described by process algebras or Petri nets. In process algebras processes are described by process terms and a set of dynamic rules of the form $t \xrightarrow{a} t'$, meaning that process $t$ can perform action $a$ and become process $t'$. Processes define an LTS, whose nodes are marked with process terms. The semantics of the processes is given by defining an equivalence relation over the term algebra. The equivalence classes then represent the intended processes. It follows that the dynamic rules describe unambiguously the dynamics of the quotient algebra, only if the chosen equivalence on the term algebra is a bisimulation. This is a main reason why bisimulation equivalence is the preferred choice for process equivalence.

The Basic Process Algebra PA is a simple model of infinite state concurrent systems. It has operators for nondeterministic choice, parallel composition and sequential composition. PA-processes and Petri nets are incomparable, meaning that neither model is more expressive that the other one. PA is not a syntactical subset of CCS [Mil89], because CCS does not have an explicit operator for sequential composition. However, as CCS can simulate sequential composition by parallel composition and synchronization, PA is still a weaker model for concurrent systems than CCS.

The definition of PA is as follows: Assume a countably infinite set of atomic actions $Act = \{a, b, c, \ldots\}$ and a countably infinite set of process variables $Var = \{X, Y, Z, \ldots\}$. The class of PA expressions is defined by the following abstract syntax

$$E ::= \epsilon \mid X \mid aE \mid E + E \mid E \| E \mid E.E$$

A PA is defined by a family of recursive equations $\{X_i := E_i \mid 1 \leq i \leq n\}$, where the $X_i$ are distinct and the $E_i$ are PA expressions at most containing the variables $\{X_1, \ldots, X_n\}$. We assume that every variable occurrence in the $E_i$ is *guarded*, i.e. appears within the scope of an action prefix, which ensures that PA-processes generate finitely branching transition graphs. This would not be true if unguarded expressions were allowed. For example, the process $X := a + a \| X$ generates an infinitely branching transition graph.

For every $a \in Act$ the transition relation $\xrightarrow{a}$ is the least relation satisfying the following inference rules:

$$aE \xrightarrow{a} E \qquad \frac{E \xrightarrow{a} E'}{E+F \xrightarrow{a} E'} \qquad \frac{F \xrightarrow{a} F'}{E+F \xrightarrow{a} F'} \qquad \frac{E \xrightarrow{a} E'}{X \xrightarrow{a} E'}(X := E)$$

$$\frac{E \xrightarrow{a} E'}{E\|F \xrightarrow{a} E'\|F} \qquad \frac{F \xrightarrow{a} F'}{E\|F \xrightarrow{a} E\|F'} \qquad \frac{E \xrightarrow{a} E'}{E.F \xrightarrow{a} E'.F}$$

Alternatively, PA-processes can be described by a state represented by a term of the form

$$G ::= \epsilon \mid X \mid G_1.G_2 \mid G_1\|G_2$$

and set of dynamic rules $\Delta$ of the form $X \xrightarrow{a} G$ whose application to states must respect sequential composition. This is described by the following inference rules:

$$X \xrightarrow{a} G \quad \text{if } (X \xrightarrow{a} G) \in \Delta$$

$$\frac{E \xrightarrow{a} E'}{E\|F \xrightarrow{a} E'\|F} \qquad \frac{F \xrightarrow{a} F'}{E\|F \xrightarrow{a} E\|F'} \qquad \frac{E \xrightarrow{a} E'}{E.F \xrightarrow{a} E'.F}$$

Basic Parallel Processes (BPP) are the subset of PA-processes without sequential composition, while context-free processes are the subset of PA-processes without parallel composition.

Unlike for PA-processes there is a one-to-one correspondence between BPPs and a class of labeled Petri nets, the *communication-free nets* [Esp95a]. In these nets every transition has exactly one input place with an arc labeled by 1. The translation of a BPP algebra into a communication-free net goes as follows: Introduce a place for each process variable and a transition for each transition rule. For a rule $X \xrightarrow{a} Y_1^{m_1}\| \cdots \| Y_n^{m_n}$ introduce a transition $t$ labeled by $a$, an arc labeled by 1 leading from place $X$ to $t$ and arcs labeled by $m_i$ leading from $t$ to places $Y_i$. The other direction is analogous.

# 3 General hardness results

How does the computational complexity of the bisimulation-reachability problem compare to the complexities of the reachability problem and the problem of deciding strong bisimilarity ? For most models of systems the bisimulation-reachability problem is at least as hard as the other two problems.

**Lemma 3.1** *For all classes of Petri nets that are at least as powerful as communication-free nets the bisimulation-reachability problem is at least as hard as the problem of deciding strong bisimilarity.*

**Proof** The problem of deciding strong bisimilarity can be reduced to the bisimulation reachability problem by constructing a slightly modified system s.t. the only reachable state that can possibly be bisimilar to the given state is the initial state itself. This construction is possible for all models that allow the creation of new parallel processes.

Without restriction we can regard the problems for different markings $\Sigma_1, \Sigma_2$ in the same Petri net $N$. Let $A = \{a_1, \ldots, a_n\}$ be the set of actions occurring in $N$ and $A' = \{a'_1, \ldots, a'_n\}$ a new set of actions s.t. $A \cap A' = \emptyset$. Now construct a new net $N'$ in the following way. For each transition $t$ in $N$ labeled by $a_i$ introduce a new place $s$ and a new transition $t'$ labeled by $a'_i$ with an arc from $t$ to $s$ and from $s$ to $t'$. Any marking $\Sigma$ of $N$ can be extended to a marking of $N'$ in a natural way by defining $\Sigma(s) = 0$ for all places $s$ that are not in $N$.

Let $L$ be the labeling function that assigns actions to the transitions in $N'$.

We show that $(N, \Sigma_1) \sim (N, \Sigma_2) \iff \exists (N', \Sigma_1) \xrightarrow{\sigma} (N', \Sigma'_1) \sim (N', \Sigma_2)$.

$\Rightarrow$ If $(N, \Sigma_1) \sim (N, \Sigma_2)$ then

$$\{((N', \Sigma), (N', \Sigma')) \mid (N, \Sigma_{|N}) \sim (N, \Sigma'_{|N}) \wedge \forall a' \in A'. \sum_{L(s\cdot)=a'} \Sigma(s) = \sum_{L(s\cdot)=a'} \Sigma'(s)\}$$

is a strong bisimulation. It follows that $(N', \Sigma_1) \sim (N', \Sigma_2)$, because $(N, \Sigma_1) \sim (N, \Sigma_2)$ and $\sum_{L(s\cdot)=a'} \Sigma(s) = 0 = \sum_{L(s\cdot)=a'} \Sigma'(s)$ for every $a' \in A'$. So we can simply choose $\sigma = \epsilon$ and $\Sigma'_1 = \Sigma_1$ and the condition is satisfied.

$\Leftarrow$ Assume that $\exists (N', \Sigma_1) \xrightarrow{\sigma} (N', \Sigma'_1).(N', \Sigma'_1) \sim (N', \Sigma_2)$. It follows that $length(\sigma) = 0$ and $\Sigma'_1 = \Sigma_1$, because otherwise $(N', \Sigma'_1)$ could do an action from $A'$ while $(N', \Sigma_2)$ can't. As $A$ and $A'$ are disjoint $\{((N, \Sigma_{|N}), (N, \Sigma'_{|N})) \mid (N', \Sigma) \sim (N', \Sigma')\}$ is a bisimulation and therefore $(N, \Sigma_1) \sim (N, \Sigma_2)$.

This construction is possible for communication-free nets and T-systems, as well as for all models more general than them. This is because the construction does not exceed the bounds of the model (i.e. if $N$ is a communication-free net then $N'$ is a communication-free net as well). Figure 1 illustrates the construction for a Petri net. □

**Theorem 3.2** *The bisimulation-reachability problem is undecidable for Petri nets.*

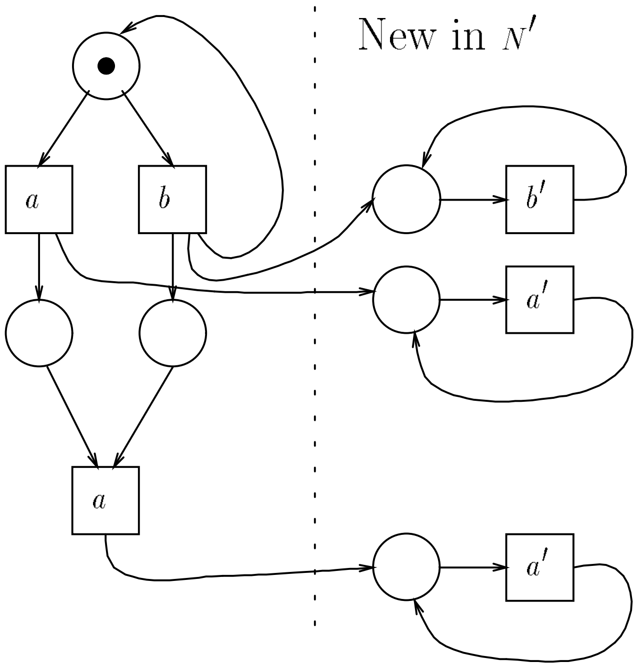
**Proof** Directly from Lemma 3.1 and the result from Jančar [Jan95] that strong bisimilarity is undecidable for general Petri nets. □

Unfortunately Lemma 3.1 yields no complexity bounds for BPPs as, to our knowledge, there is no hardness result for the problem of deciding strong bisimilarity of BPPs yet. We can give complexity bounds by showing that for many models of systems the bisimulation reachability problem is at least as hard as the reachability problem.

**Lemma 3.3** *For BPPs the bisimulation-reachability problem is at least as hard as the reachability problem.*

**Proof** by reduction of the reachability problem to the bisimulation-reachability problem. Let $N$ be a communication-free net, $A$ the set of atomic actions occurring in $A$, $\Sigma_0$ the

Figure 1: How to reduce the problem of deciding strong bisimilarity to the bisimulation-reachability problem.




initial marking and $\Sigma$ a marking of $N$. Now construct a new larger net $N'$: for each place $s$ in $N$ add a new transition $t_s$ in the postset of $s$ that is labeled by a unique new action $a_s$. Then add one new place $\hat{s}$ and one new transition $\hat{t}$ to the net and an arc from every $t_s$ to $\hat{s}$ and an arc from $\hat{s}$ to $\hat{t}$ and from $\hat{t}$ to $\hat{s}$. Let $\hat{t}$ be labeled by a unique new action $\hat{a}$. Figure 2 illustrates the construction.

It follows that if $\Sigma_1$ and $\Sigma_2$ are markings of $N'$ and $\Sigma_1 \sim \Sigma_2$ and $\Sigma_1(\hat{s}) = 0$ then $\Sigma_1 = \Sigma_2$. Now we show that

$$\exists \sigma.(N, \Sigma) \xrightarrow{\sigma} (N, \Sigma') \iff \exists \sigma'.(N', \Sigma) \xrightarrow{\sigma'} (N', \Sigma'') \sim (N', \Sigma')$$

$\Rightarrow$ If $(N, \Sigma) \xrightarrow{\sigma} (N, \Sigma')$ then $(N', \Sigma) \xrightarrow{\sigma} (N', \Sigma')$. Choose $\sigma' = \sigma$ and $\Sigma'' = \Sigma'$ and the condition is satisfied.

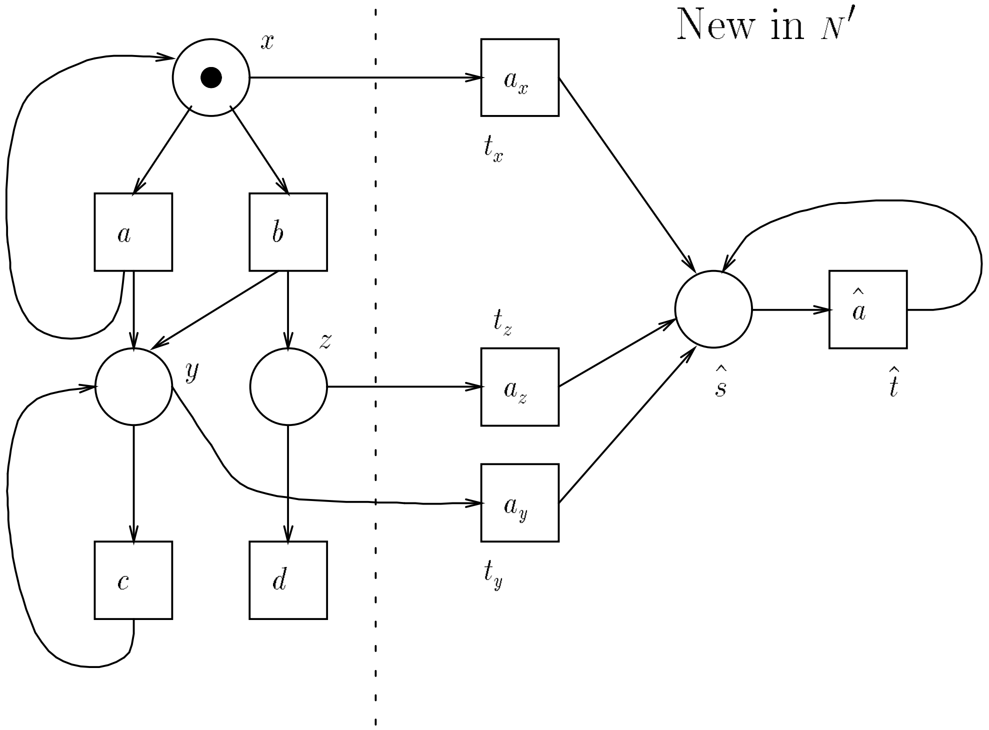
$\Leftarrow$ As $\Sigma'(\hat{s}) = 0$ it follows that $\Sigma'' = \Sigma'$ and thus $\Sigma''(\hat{s}) = 0$. Therefore no transition $t_s$ occurs in $\sigma'$. So $\sigma'$ is a sequence enabled by $(N, \Sigma)$ and the condition is satisfied with $\sigma = \sigma'$.

□

**Corollary 3.4** *The bisimulation-reachability problem for BPPs is NP-hard.*

**Proof** Directly from Lemma 3.3 and the fact that the reachability problem for BPPs is NP-compete [Esp95a]. □

Figure 2: How to reduce the reachability problem to the bisimulation-reachability problem.




**Remark 3.5** *The construction used in Lemma 3.3 is possible for many classes of Petri nets, but not for T-systems and free-choice nets. Also the newly constructed net is never normed, so Lemma 3.3 yields no hardness result for normed BPP.*

A process $t$ is *normed* if every process $t'$ reachable from $t$ has a terminating computation. The length of the shortest terminating computation is called the norm of $t$. It is denoted by $[t]$.

A BPP is normed iff in the corresponding communication-free net $N$ with initial marking $\Sigma_0$ it is impossible to reach a marking $\Sigma$ s.t. $\Sigma$ marks a trap of $N$. This property can be decided in polynomial time, because the maximal trap can be computed in polynomial time and because in these nets tokens can move independently.

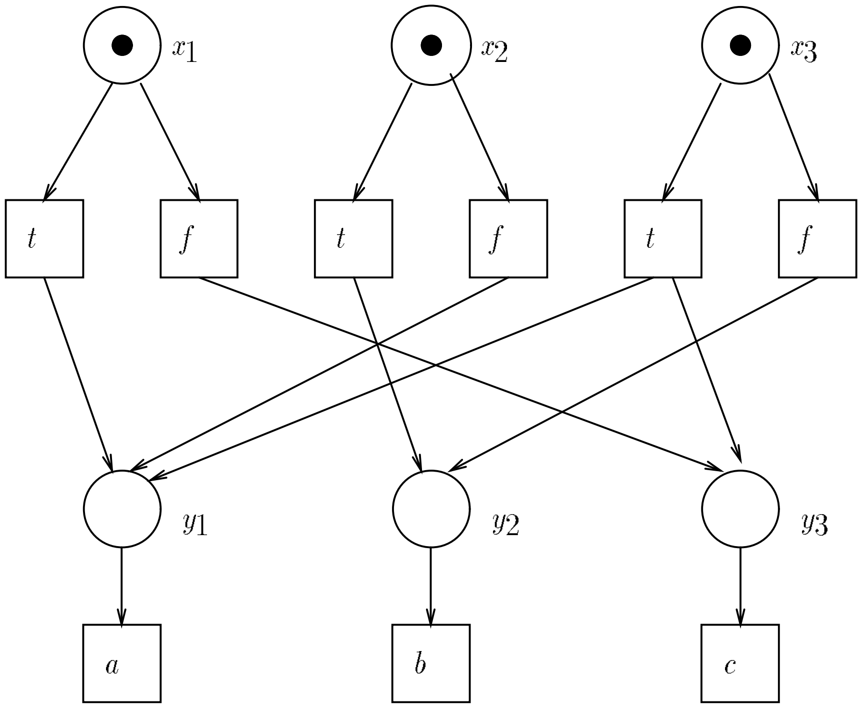
**Lemma 3.6** *The bisimulation-reachability problem is NP-hard even for normed BPP.*

**Proof** The proof is done by reduction of SAT to the bisimulation-reachability problem. We illustrate the construction by an example (see Figure 3): The formula

$$(x_1 \vee \neg x_2 \vee x_3) \wedge (x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_3)$$

is satisfiable iff a state is reachable from $\{x_1, x_2, x_3\}$ that is bisimilar to $\{y_1, y_2, y_3\}$. (The only such state is $\{y_1, y_2, y_3\}$ itself). Note that the constructed communication-free net is finite state and normed. $\square$

Figure 3: NP-hardness of the bisimulation-reachability problem for normed BPP.




In the next sections we show that the bisimulation-reachability problem is decidable for several special classes of process algebras.

# 4 Basic Parallel Processes

**Lemma 4.1** *Let $t_0$ be a BPP and $t$ a BPP that has a terminating computation. It is decidable if there is a sequence $t_0 \xrightarrow{\sigma} t'$ s.t. $t \sim t'$.*

**Proof** We know that $[t] \in \mathbb{N}$. If such a $t'$ exists that is reachable from $t_0$ and bisimilar to $t$, then $[t'] = [t]$ and therefore $size(t') \leq [t]$. There are only finitely many candidates for such a $t'$. It is decidable if a candidate $t'$ is reachable from $t_0$ and it is decidable if $t' \sim t$ [CHM93b]. Check all candidates until a correct one is found (answer "yes") or none is left (answer "no"). □

Note that this lemma is especially true if $t$ is normed.

**Lemma 4.2** *Let $t_0$ be a normed BPP and $t$ a BPP. It is decidable if there is a sequence $t_0 \xrightarrow{\sigma} t'$ s.t. $t \sim t'$.*

**Proof** As $t_0$ is normed, every $t'$ that is reachable from $t_0$ has a terminating computation. It can be decided in polynomial time if $t$ has a terminating computation. First apply a marking algorithm that marks all variables $X$ s.t. $\exists \sigma.\ X \xrightarrow{\sigma} \epsilon$. Then check if all variables in $t$ are marked. There are two cases:

1. If $t$ has no terminating computation, then $t \not\sim t'$ for every $t'$ that is reachable from $t_0$, because $t'$ is normed. Thus the answer to the question is "no".

2. If $t$ has a terminating computation, then we have the same case as in Lemma 4.1.

□

The algorithms used in Lemma 4.1 and Lemma 4.2 have non-elementary complexity, as they use the algorithm from [CHM93a] for deciding strong bisimilarity of BPPs. The only known lower bound is NP-hardness.

However, for the special case of two normed BPPs we can give an accurate complexity measure.

**Lemma 4.3** *Let $t_0$ and $t$ be normed BPPs. It is decidable in NP if there is a sequence $t_0 \xrightarrow{\sigma} t'$ s.t. $t \sim t'$.*

**Proof** It suffices to prove the property for normed markings $\Sigma_0$ (corresponding to $t_0$) and $\Sigma$ (corresponding to $t$) of a communication-free net. As $\Sigma_0$ is normed, every $\Sigma'$ reachable from $\Sigma_0$ is normed. The norm $[\Sigma]$ of $\Sigma$ is at most exponential. If a correct $\Sigma'$ exists, then it must have the same norm as $\Sigma$. So it can contain at most exponentially many tokens and can therefore be described in polynomial space. Thus it can be reached by a sequence of at most exponential length [Esp95a]. So the Parikh-vector of this sequence of transitions can be described in polynomial space. As $\Sigma$ and $\Sigma'$ are normed, it is decidable in polynomial time if $\Sigma \sim \Sigma'$ [CHM93b].

The algorithm goes like this: Nondeterministically guess a Parikh-vector of transitions of polynomial size. Then check in polynomial time if there is a fireable sequence of transitions starting at $\Sigma_0$ with this Parikh-vector [Esp95b], calculate the result $\Sigma'$ (also in polynomial time) and check in polynomial time if $\Sigma' \sim \Sigma$ [CHM93b]. □

**Theorem 4.4** *The bisimulation-reachability problem for normed BPP is NP-complete.*

**Proof** Directly from Lemma 4.3 and Lemma 3.6. □

# 5 Context-free processes

**Lemma 5.1** *The bisimulation-reachability problem is decidable for context-free processes if the initial state $t_0$ is normed or the given state $t$ has a terminating computation.*

**Proof** The proofs are analogous to the proofs of Lemma 4.1 and Lemma 4.2. The algorithm used for deciding bisimilarity of context-free processes is the one described in [CHS92]. □

The complexity of the algorithms used in Lemma 5.1 is non-elementary, because the algorithm for deciding bisimilarity of context-free processes has non-elementary complexity. Unfortunately, there is no hardness result yet. Lemma 3.1 does not carry over to context-free processes.

However, just like for BPPs, the problem for two normed processes is easier to solve.

**Theorem 5.2** *The bisimulation-reachability problem for normed context-free processes is decidable in exponential space.*

**Proof** Let $t_0$ be the initial state and $t$ the given state. As $t_0$ is normed, every $t'$ that is reachable from $t_0$ is normed. The norm $[t]$ of $t$ is at most exponential in $size(t)$. If a correct $t'$ exists, then it must have the same norm as $t$ and can therefore be described in exponential space. Now for each of these candidates for $t'$ with $size(t') \leq O(2^{size(t)})$ first check if it is reachable from $t_0$. This requires $O(2^{size(t)})$ time as the reachability problem for context-free processes is polynomial. Then check if it is bisimilar to $t$. As both $t'$ and $t$ are normed the time needed for this is polynomial in $size(t_0) + size(t') \leq size(t_0) + O(2^{size(t)})$. This is because deciding bisimilarity for normed context-free processes is polynomial [HJM94].

So overall the algorithm requires at most exponential space. □

# 6 Normed PA-processes vs. finite state systems

As a first step we prove decidability of the reachability problem for PA-processes.

**Lemma 6.1** *The reachability problem for PA is decidable in polynomial space.*

**Proof** Let $n$ be the size of the instance of the problem. We show that if a $t$ can be reached from $t_0$, then it can be reached via a path s.t. the size of every intermediate state $t'$ is bounded by a constant $c \leq O(n^2)$. Every intermediate state $t'$ consists of three parts:

**A** The **stable** part. This part will not change in the rest of the sequence and will be part of $t$.

**B** The **active** part. This part will change in the rest of the sequence and at least a part of the result will be part of $t$.

**C** The **waste** part. This part will be reduced to $\epsilon$ in the rest of the sequence.

It is a valid strategy to reduce part $C$ to $\epsilon$ first whenever $C$ isn't empty, before doing anything else. It is clear that the sum of the sizes of part $A$ and $B$ must never exceed $size(t)$. To keep part $C$ small we will first reduce the accessible variables in $C$ that have the lowest norm. How big can part $C$ ever become if we follow this strategy ? Let $m$ be the number of variables in the PA-algebra and $l$ the maximal size of the right hand side of a rule $X \xrightarrow{a} G$. So the size of the waste descending from a variable $X$ will never exceed $(l-1)(m-2)+l$. The size of the $C$-part of $t_0$ is at most $size(t_0)-1$ and the waste generated by the application of a reduction rule is at most $l-1$. Therefore the size of the $C$-part of $t'$ never exceeds $max(size(t_0)-1, l-1)+(l-1)(m-2)+l$. Thus $size(t') \leq size(t) + max(size(t_0)-1, l-1)+(l-1)(m-2)+l \leq O(n^2)$. There are only exponentially many such terms $t'$. So if $t$ can be reached at all, then it can be reached by a sequence of at most exponential length, and thus the problem can be decided in polynomial space. □

**Remark 6.2** *The argument in Lemma 6.1 about the maximal length of the sequence needed to reach a given term is somewhat crude. A longer and more careful analysis of the structure of PA terms shows that the problem is in fact NP-complete. However, in the sequel we only need containment in PSPACE.*

**Theorem 6.3** *Let $t_0$ be a normed PA-process and $R$ a finite state LTS with initial state $r_0$. It is decidable in PSPACE if there is a sequence $t_0 \xrightarrow{\sigma} t$ s.t. $t \sim r_0$.*

**Proof** It is decidable in polynomial time if $r_0$ is normed.

1. If $r_0$ is not normed then the answer is "no". This is because $t$ is always normed and a normed process is never bisimilar to an unnormed one.

2. If $r_0$ is normed, then $[r_0] \leq k-1$, where $k$ is the number of states in $R$. So if any correct $t$ exists, then $[t] = [r_0] \leq k-1$ and thus $size(t) \leq k-1$. So there are only finitely many candidates for $t$, each of which has only polynomial size.

   It remains to check for each candidate $t$ if it is reachable from $t_0$ and if $t \sim r_0$. By Lemma 6.1 the first condition can be checked in polynomial space. The second condition can be checked in polynomial space as well, as in a bisimulation game the size of a child $\hat{t}$ of $t$ must never exceed $k-1$, because $\hat{t}$ and $r_0$ are normed and $R$ has $k$ states.

□

**Remark 6.4** *As the BPPs are a subset of PA, the problem of Theorem 6.3 is at least NP-hard, because of Lemma 3.6.*

# 7 The non-bisimulation-reachability problem

So far we have studied the bisimulation-reachability problem: "Is there a reachable state that is bisimilar to a given state ?". The opposite question is if there is a reachable state that is **not** bisimilar to a given state. This can be generalized to finite sets of states.

The non-bisimulation-reachability problem.
**Instance:** An LTS with initial state $s$ and a finite set of given states $\{s_1, \ldots, s_n\}$.
**Question:** Is there a sequence of transitions $\sigma$ s.t. $s \xrightarrow{\sigma} s'$ and $\forall i \in \{1, \ldots, n\}.\ s' \not\sim s_i$.

In section 3 we have shown that the bisimulation-reachability problem is undecidable for Petri nets (Theorem 3.2). On the other hand the non-bisimulation-reachability problem is decidable.

**Theorem 7.1** *The non-bisimulation-reachability problem is decidable for Petri nets.*

**Proof** Let $N$ be a Petri net with initial marking $\Sigma$ and $\Sigma_1, \ldots, \Sigma_n$ markings of $N$. Let $R(\Sigma)$ be the set of reachable markings. There are two cases:

1. If $|[R(\Sigma)]_\sim| = k \leq n$ then there is a finite state LTS $U$ with $k$ states and an initial state $u_0$ s.t. $u_0 \sim \Sigma$. If $\exists u \in U. \forall i \in \{1, \ldots, n\}. u \not\sim \Sigma_i$ then the answer is "yes" else the answer is "no".

2. In this case the system $(N, \Sigma)$ has more than $n$ different states w.r.t. strong bisimilarity (possibly even infinitely many). If $|[R(\Sigma)]_\sim| > n$ then there is at least one $\Sigma' \in R(\Sigma)$ s.t. $\forall i \in \{1, \ldots, n\}$. $\Sigma' \not\sim \Sigma_i$ and the answer to the question is "yes".

This yields a decision procedure, because there are only finitely many finite state LTS with $\leq n$ states and it is decidable if a Petri net and a finite state LTS are strongly bisimilar [JM95]. □

# 8 Conclusion

The bisimulation-reachability problem is decidable for several simple classes of process algebras, but many cases are still open. We conjecture that the problem is decidable even for unnormed BPP and context-free processes. Decidability for (normed) PA-processes is also open. To our knowledge, it isn't even known yet if strong bisimulation equivalence is decidable for (normed) PA-processes. Another interesting field would be studying the same problems for weak bisimulation equivalence.

# References

[CHM93a] S. Christensen, Y. Hirshfeld, and F. Moller. Bisimulation equivalence is decidable for basic parallel processes. In E. Best, editor, *Proceedings of CONCUR 93*, number 715 in LNCS. Springer Verlag, 1993.

[CHM93b] S. Christensen, Y. Hirshfeld, and F. Moller. Decomposability, decidability and axiomatisability for bisimulation equivalence on basic parallel processes. In *Proceedings of LICS93*. IEEE Computer Society Press, 1993.

[CHS92] S. Christensen, H. Hüttel, and C. Stirling. Bisimulation equivelence is decidable for all context-free processes. In W.R. Cleaveland, editor, *Proceedings of CONCUR 92*, number 630 in LNCS. Springer Verlag, 1992.

[Esp95a] Javier Esparza. Decidability of model checking for infinite-state concurrent systems. *Acta Informatica*, 1995.

[Esp95b] Javier Esparza. Petri nets, commutative context-free grammars and basic parallel processes. In Horst Reichel, editor, *Fundamentals of Computation Theory*, number 965 in LNCS. Springer Verlag, 1995.

[HJM94] Y. Hirshfeld, M. Jerrum, and F. Moller. A polynomial algorithm for deciding bisimulation of normed context free processes. Technical report, LFCS report series 94-286, Edinburgh University, 1994.

[Jan95] P. Jančar. Undecidability of bisimilarity for petri nets and related problems. *Theoretical Computer Science*, 1995.

[JM95] P. Jančar and F. Moller. Checking regular properties of petri nets. In Insup Lee and Scott A. Smolka, editors, *Proceedings of CONCUR'95*, number 962 in LNCS. Springer Verlag, 1995.

[Mil89] R. Milner. *Communication and Concurrency.* Prentice Hall, 1989.