# An Elementary Bisimulation Decision Procedure for Arbitrary Context-Free Processes

Olaf Burkart*
Lehrstuhl für Informatik II
RWTH Aachen
Ahornstraße 55
52074 Aachen
Germany

Didier Caucal†
IRISA
Campus de Beaulieu
35042 Rennes
France

Bernhard Steffen‡
Fakultät für Mathematik und Informatik
Universität Passau
Innstraße 33
94032 Passau
Germany

*Aachener Informatikbericht*
*Nr. 94 – 28*

**Abstract**

We present an elementary algorithm for deciding bisimulation equivalence between arbitrary context-free processes. This improves on the state of the art algorithm of Christensen, Hüttel and Stirling consisting of two semi-decision procedures running in parallel, which prohibits any complexity estimation. The point of our algorithm is the effective construction of a finite relation characterizing all bisimulation equivalence classes, whose mere existence was exploited for the above mentioned decidability result.

*e-mail: burkart@informatik.rwth-aachen.de
†e-mail: caucal@irisa.fr
‡e-mail: steffen@fmi.uni-passau.de

# 1 Introduction

Algebraic descriptions of concurrent systems are usually formulated using a process calculus such as CCS [Mil89], ACP [BW90], or CSP [Hoa85], and their semantics in terms of labelled transition graphs together with a notion of behavioural equivalence which reflects some notion of observation. A number of such behavioural equivalences have been considered depending on the suggested observational power (cf. [Gla90]): their main rationale has been to capture aspects like nondeterministic, parallel or nonterminating behaviours, which are not taken into account in classical language theory.

Besides their observational power, decidability is of practical interest. For example, from classical language theory it is known that language equivalence is decidable for regular systems. However, if one moves further to context-free languages the problem becomes undecidable. In the 'finer' process algebraic setting the situation is different, as *bisimilarity* is decidable for *normed* context-free processes (BPA processes)[BBK87], which are defined by reduced context-free grammars, and can terminate in finitely many steps at any point of the execution. This exceptional property of the bisimulation equivalence led to an intense investigation (cf. [Cau90, Gro91, HS91, HT94, HM94]), which resulted in a polynomial time decision procedure for normed processes [HJM94].

The treatment of general context-free processes requires completely new techniques, as the decomposition properties for the normed case fail to hold. Nevertheless, considering *bisimulation bases* $B$ characterizing the bisimulation equivalence as the least congruence w.r.t. sequential composition containing $B$, it is possible to prove that bisimulation is decidable also in this case [CHS92]. The existence of such a finite relation $B$ can be exploited for a decision algorithm based on two semi-decision procedures: one for enumerating all possible relations $B$ which may contain the pair $(\alpha, \beta)$, and one for the enumeration of all non-bisimilar pairs of processes.

In this paper we show how to effectively compute a bisimulation base $B$, and exploit it for the construction of an elementary bisimulation decision procedure for arbitrary context-free processes. Key idea behind the construction of $B$ is the determination of a new bound for the number of transitions needed to separate two normed non-bisimilar processes along the lines of [Cau89]. This bound allows the construction of an 'initial' base, which subsequently must be refined by means of a fixpoint iteration similar to the one used in [HJM94] to obtain $B$. The decision algorithm is then completed by a straightforward branching algorithm.

Section 2 introduces the kind of processes and the behavioural equivalence we are interested in, while Section 3 describes a branching algorithm for deciding bisimilarity of normed BPA, from which we derive a bound for the separability of two normed non-bisimilar processes. Finally, in Section 4 we develop our new bisimulation base construction algorithm to obtain the proposed decision procedure.

## 2 Context-Free Processes and Bisimulation

We first review some basic definitions and facts concerning the class of processes and the behavioural equivalence we want to consider.

**Definition 2.1** *A* labelled transition system *is a triple* $(\mathcal{S}, Act, \rightarrow)$ *consisting of a set of* states *or* processes $\mathcal{S}$*, a set of* actions $Act$ *and a* transition relation $\rightarrow \subseteq \mathcal{S} \times Act \times \mathcal{S}$*, written* $s \stackrel{a}{\rightarrow} s'$ *for* $(s, a, s') \in \rightarrow$*.*

As usual, we extend the transition relation by reflexivity and transitivity to allow $s \stackrel{w}{\rightarrow} s'$ for $w \in Act^*$ and we call a process $s$ *terminating* if no transition evolves from $s$, also written as $s \not\rightarrow$. A labelled transition system is called *finitely branching* if for each state $s$ the set $\{ s \stackrel{a}{\rightarrow} s' \mid a \in Act, s' \in \mathcal{S} \}$ is finite.

We will concentrate on the class of processes defined by guarded recursive BPA (Basic Process Algebra) systems [BBK87].

**Definition 2.2** *A* BPA system $\mathcal{C}$ *is a quadruple* $(V, Act, \Delta, X_1)$ *consisting of*

- *a finite set of* variables *or* non-terminals $V = \{ X_1, \ldots, X_n \}$*,*
- *a finite set of* actions *or* terminals $Act$*,*
- *a finite set of recursive process equations* $\Delta = \{ X_i =_{\mathrm{df}} E_i \mid 1 \leq i \leq n \}$*, where each* $E_i$ *is a BPA expression obeying the following abstract syntax*

  $$E ::= a \mid X \mid E + E \mid E \cdot E$$

  *such that* $a$ *ranges over* $Act$ *and* $X$ *over* $V$*, and*
- *a variable* $X_1 \in V$*, called the* root*.*

*The operator* $+$ *is interpreted as nondeterministic choice while* $\cdot$ *is sequential composition – henceforth we usually omit the* $\cdot$*.*

**Definition 2.3** *We call an occurrence of a variable* $X$ *in a BPA term* $t$ guarded *if* $t$ *has a subterm* $a.t'$ *such that* $a$ *is an atomic action and* $t'$ *contains the occurrence of* $X$*. A BPA term* $t$ *is guarded if every variable occurrence is guarded, and a BPA system is guarded if each* $E_i$ *is guarded for* $1 \leq i \leq n$*.*

We use $X, Y, \ldots$ to range over variables in $V$ and Greek letters $\alpha, \beta, \ldots$ to range over elements in $V^*$. For technical convenience, we use $\epsilon$ to denote the empty variable sequence with the convention $\epsilon\alpha = \alpha\epsilon = \alpha$. The function $|\,.\,|$ gives the length of a sequence.

**Definition 2.4** *The* operational semantics *of a guarded BPA system is given by the labelled transition system* $(V^*, Act, \rightarrow)$ *where the transition relation* $\rightarrow$ *is the least relation satisfying the following rules:*

$$a \xrightarrow{a} \epsilon, \quad a \in Act \qquad \frac{E \xrightarrow{a} E'}{X \xrightarrow{a} E'} \; X =_{\mathrm{df}} E \in \Delta \qquad \frac{E \xrightarrow{a} E'}{EF \xrightarrow{a} E'F}$$

$$\frac{E \xrightarrow{a} E'}{E + F \xrightarrow{a} E'} \qquad \frac{F \xrightarrow{a} F'}{E + F \xrightarrow{a} F'}$$

Over the last years within process theory a lot of behavioural equivalences have been proposed. Bisimulation equivalence [Par81], which we are going to decide for arbitrary context-free processes in this paper, is the finest equivalence in the van Glabbek hierarchy [Gla90]. In particular, it is strictly finer than "classical" language equivalence.

**Definition 2.5** *A binary relation* $R$ *between processes is a* bisimulation *if whenever* $(P, Q) \in R$ *then for each* $a \in Act$

1. $P \xrightarrow{a} P'$ *implies* $\exists\, Q'.\ Q \xrightarrow{a} Q' \ \wedge\ (P', Q') \in R$
2. $Q \xrightarrow{a} Q'$ *implies* $\exists\, P'.\ P \xrightarrow{a} P' \ \wedge\ (P', Q') \in R$

Two processes $P$ and $Q$ are said to be *bisimulation equivalent* or *bisimilar*, written $P \sim Q$, if $(P, Q) \in R$ for some bisimulation $R$. In the remainder of the paper we write $[P]_\sim$ for the bisimulation equivalence class of $P$. Now a basic result of [BBK87] states that any guarded BPA system can effectively be transformed into $K$*-Greibach Normal Form* ($K$-GNF), while preserving bisimulation equivalence. A BPA system is said to be in $K$-GNF, when each equation is of the form $X_i =_{\mathrm{df}} \sum_{j=1}^{m_i} a_{ij}\alpha_{ij}$ with the additional restriction that each variable sequence $\alpha_{ij}$ has length of at most $K$.

**Definition 2.6** *Given a binary relation* $R$ *between processes we define* $P \cong_R Q$ *if for each* $a \in Act$

1. $P \xrightarrow{a} P'$ *implies* $\exists\, Q'.\ Q \xrightarrow{a} Q' \ \wedge\ P' \overset{*}{\leftrightarrow}_R Q'$
2. $Q \xrightarrow{a} Q'$ *implies* $\exists\, P'.\ P \xrightarrow{a} P' \ \wedge\ P' \overset{*}{\leftrightarrow}_R Q'$

*A binary relation* $R$ *between processes is then a* self-bisimulation *if* $R \subseteq \cong_R$.

The importance of self-bisimulations is revealed by the following lemma which was first proved in [Cau90].

**Lemma 2.7** *If* $R$ *is a self-bisimulation then* $\overset{*}{\leftrightarrow}_R \subseteq \sim$.

## 2.1 Normedness and Cancellations

The *norm* of a process $s$, written as $||s||$, is the length of the shortest transition sequence from $s$ to a terminating state. A process is said to be *normed* if its norm is finite, and a transition $s \xrightarrow{a} s'$ of a normed process $s$ is called a *norm-reducing* transition if $||s'|| = ||s|| - 1$. Henceforth, we will abbreviate $\min\{||\alpha||, ||\beta||\}$ by $||\alpha, \beta||$. In the remainder of this section we state some well-known basic properties of normed processes. In particular, normedness allows to take advantage of some important cancellation rules.

**Lemma 2.8** *Let $\alpha \xrightarrow{w} \alpha'$ with $\alpha, \alpha'$ normed and $|w| = l$. Then*

$$||\alpha'|| \leq (l(K-1) + |\alpha|)N.$$

**Proof:** The proof is accomplished by induction on $l$. For $l = 0$ we obtain $||\alpha'|| = ||\alpha|| \leq |X\alpha|N$. Now assume $\alpha = X\beta \xrightarrow{a} \gamma\beta \xrightarrow{w} \alpha'$ with $|w| = l$. Then we conclude by induction hypothesis $||\alpha'|| \leq (l(K-1) + |\gamma\beta|)N \leq (l(K-1) + (K + |\beta|))N = (l(K-1) + ((K-1) + |X\beta|))N \leq ((l+1)(K-1) + |\alpha|)N$. $\square$

**Lemma 2.9**

- $||\alpha\beta|| = ||\alpha|| + ||\beta||$
- *$\alpha \sim \beta$ implies $||\alpha|| = ||\beta||$*
- *If $\alpha \sim \alpha'$ and $\beta \sim \beta'$ then $\alpha\beta \sim \alpha'\beta'$.*

**Lemma 2.10 (Cancellation rules for normed BPA processes)**
*Let $\alpha, \beta$ and $\gamma$ be normed. Then*

1. *$\gamma\alpha \sim \gamma\beta$ implies $\alpha \sim \beta$ and*
2. *$\alpha\gamma \sim \beta\gamma$ implies $\alpha \sim \beta$.*

Note however that both implications of the cancellation lemma are invalid for unnormed processes:

1. Let $X = a + aX + aY$ and $Y = bY$ then we have $XY \sim XXY$ but $Y \not\sim XY$.
2. Let $X = a$ and $Y = aY$ then we have $XY \sim XXY$ but $X \not\sim XX$.

A crucial property of normed processes is stated in the following splitting lemma [Cau90].

**Lemma 2.11 (Splitting rule for normed BPA processes)**
*Let $X\alpha, Y\beta \in V^+$ be normed such that $||X|| \leq ||Y||$. Then*

$$X\alpha \sim Y\beta \quad \textit{iff} \quad X\gamma \sim Y \textit{ and } \alpha \sim \gamma\beta \textit{ for some } \gamma.$$

In the presence of unnormed processes we only have the trivial right-cancellation rule.

**Lemma 2.12 (Right-cancellation rule for unnormed BPA processes)**
*If $X$ is unnormed, then $\alpha X \beta \sim \alpha X$.*

Henceforth assume that we have a guarded BPA system $\mathcal{C} = (V, Act, \Delta, X_1)$ in $K$-GNF, where the variables $V = \{ X_1, \ldots, X_n \}$ are ordered by nondecreasing norm. Moreover, we divide the set of variables into disjoint subsets $V_N = \{ X \in V \mid X \text{ is normed} \}$ and $V_U = V \setminus V_N$. Due to the above stated right-cancellation rule for unnormed processes we restrict our attention to variable sequences $\alpha \in \mathcal{S}(V^*) =_{\mathrm{df}} V_N^* \cup V_N^* V_U$. Additionally, we write $\mathcal{S}(V^+)$ for $\mathcal{S}(V^*) \setminus \{ \epsilon \}$, and we denote the maximal finite norm of all given variables by $\mathcal{N}$.

Finally, we assume that the given BPA system is *normalized*, i.e. if $\alpha$ is an unnormed right-hand side summand of some normed variable $X$, then $\alpha$ must be of the form $a.Y$. This additional assumption does not impose any restriction, since every guarded BPA system in $K$-GNF can be normalized by means of the following transformation:

> If $a.\beta$ with $\beta \notin V$ is an unnormed right-hand side summand of some normed variable $X$, replace $\beta$ by some fresh variable $Y$ and add the equation $Y =_{\mathrm{df}} \beta$ to $\Delta$. Then transform the resulting BPA system again into $K$-GNF.

The transformation will eventually terminate since every newly created variable is unnormed.

## 2.2 Separability

The development of this section is based on a fixed-point construction, which Milner [Mil89] proposed to characterize bisimulation of finitely branching transition systems in terms of sequences of approximations. *Separability* is defined in terms of the complements of these approximations. It characterizes the number of transitions needed to separate two non-bisimilar processes.

**Definition 2.13** *Let $R$ be a binary relation between processes. Then $(P, Q) \in \mathcal{F}(R)$ iff for each $a \in Act$:*

1. *$P \xrightarrow{a} P'$ implies $\exists\, Q'.\ Q \xrightarrow{a} Q' \ \wedge\ (P', Q') \in R$*
2. *$Q \xrightarrow{a} Q'$ implies $\exists\, P'.\ P \xrightarrow{a} P' \ \wedge\ (P', Q') \in R$*

**Lemma 2.14**

1. *$\mathcal{F}$ is monotonic, i.e. if $R_1 \subseteq R_2$ then $\mathcal{F}(R_1) \subseteq \mathcal{F}(R_2)$.*
2. *$R$ is a bisimulation iff $R \subseteq \mathcal{F}(R)$.*

3. $\sim \; = \; \bigcup \{ R \mid R \subseteq \mathcal{F}(R) \}$.

4. *If the labelled transition graph* $(\mathcal{S}, Act, \rightarrow)$ *is finitely branching, then* $\mathcal{F}$ *is continuous on* $\mathcal{S} \times \mathcal{S}$. *In this case we also have* $\sim \; = \; \bigcap \{ \mathcal{F}^i(\mathcal{S} \times \mathcal{S}) \mid i \geq 0 \}$.

Based on this iterative characterization of bisimulation we define when two processes are said to be $m$-bisimilar, which leads to the notion of separability.

**Definition 2.15** *Let* $P, Q \in \mathcal{S}$ *be two processes.* $P$ *and* $Q$ *are said to be* $m$-bisimilar, *written* $P \sim_m Q$, *if* $(P, Q) \in \mathcal{F}^{\,m}(\mathcal{S} \times \mathcal{S})$. *Moreover, we define the* separability *of* $P$ *and* $Q$ *as*

$$Sep(P, Q) =_{\mathrm{df}} \max\{ m \mid P \sim_m Q \} + 1$$

*If the separability of* $P$ *and* $Q$ *is finite*[1], *let us say* $m$, *then* $P$ *and* $Q$ *are also called* $m$-separable.

In contrast to bisimilarity, separability seems not to be very well elaborated. As a first step to overcome this problem, we give some new results concerning the separability of context-free processes. Most of the results were first developed for *simple grammars*, a proper subclass of context-free processes [Cau89]. Simple grammars can be interpreted as normed *deterministic* BPA systems, so it is surprising that these results also hold in the nondeterministic case.

**Lemma 2.16** *If two processes* $P, Q \in \mathcal{S}$ *are* $m$*-separable, then one of the following conditions hold.*

1. $P \stackrel{a}{\rightarrow} P'$ *for some* $a, P'$ *and* $\forall\, Q'.\ Q \stackrel{a}{\rightarrow} Q' \Rightarrow P' \not\sim_{m-1} Q'$; *or*

2. $Q \stackrel{a}{\rightarrow} Q'$ *for some* $a, Q'$ *and* $\forall\, P'.\ P \stackrel{a}{\rightarrow} P' \Rightarrow P' \not\sim_{m-1} Q'$.

*In the first case we call* $P \stackrel{a}{\rightarrow} P'$ *a* separating transition *for* $P$ *and* $Q$, *and in the later case* $Q \stackrel{a}{\rightarrow} Q'$, *respectively.*

**Proof:** This lemma is a simple consequence of the definition of $m$-bisimilarity. $\Box$

Iterative application of the above lemma to a non-bisimilar pair of processes yields a separability-reducing *transition track*.

**Definition 2.17** *Let* $P \not\sim_m Q$. *A* transition track *for* $(P, Q)$ *is a pair of transition sequences* $(P = P_m \stackrel{a_{m-1}}{\rightarrow} P_{m-1} \ldots \stackrel{a_r}{\rightarrow} P_r, Q = Q_m \stackrel{a_{m-1}}{\rightarrow} Q_{m-1} \ldots \stackrel{a_r}{\rightarrow} Q_r)$, $1 \leq r < m$ *with the property* $P_i \not\sim_i Q_i$ *for each* $r \leq i \leq m$.

Figure 1 gives an illustration of a transition track for two non-bisimilar processes $P, Q$.

We proceed by presenting some basic bounds for separability.

[1]Note that $P \sim Q$ implies $\mathrm{Sep}(P, Q) = \infty$.

$$\begin{tikzcd}[row sep=small]
P = P_m \arrow[d, "a_{m-1}"] & \not\sim_m & Q_m = Q \arrow[d, "a_{m-1}"] \\
P_{m-1} \arrow[d, "a_{m-2}"] & \not\sim_{m-1} & Q_{m-1} \arrow[d, "a_{m-2}"] \\
\cdots \arrow[d, "a_r"] & & \cdots \arrow[d, "a_r"] \\
P_r & \not\sim_r & Q_r
\end{tikzcd}$$

Figure 1: A transition track for $P, Q$.

**Lemma 2.18** *If* $||\alpha|| < ||\beta||$ *then* $\mathit{Sep}(\alpha, \beta) \leq ||\alpha|| + 1$.

**Proof:** Let $\alpha \stackrel{w}{\rightarrow} \epsilon$ be a norm-reducing transition sequence. Then either $\beta \stackrel{w}{\not\rightarrow}$ and therefore $\mathrm{Sep}(\alpha, \beta) \leq |w| \leq ||\alpha||$ or all $\beta'$ with $\beta \stackrel{w}{\rightarrow} \beta'$ satisfy $\epsilon \not\sim_1 \beta' \neq \epsilon$ and therefore $\mathrm{Sep}(\alpha, \beta) \leq ||\alpha|| + 1$. □

**Lemma 2.19** *Let* $\mathit{Sep}(\alpha, \beta) = m + 1$ *and* $\alpha \stackrel{w}{\rightarrow} \alpha'$ *with* $|w| \leq m$. *Then there exists some* $\beta'$ *such that* $\beta \stackrel{w}{\rightarrow} \beta'$ *and* $\mathit{Sep}(\alpha, \beta) \leq |w| + \mathit{Sep}(\alpha', \beta')$.

**Proof:** The lemma is proved by induction on $|w|$. For $|w| = 0$ there is nothing to show. Now let $\mathrm{Sep}(\alpha, \beta) = m + 1$ and $\alpha \stackrel{a}{\rightarrow} \alpha' \stackrel{w}{\rightarrow} \alpha''$, and therefore in particular $\alpha \sim_m \beta$. Then according to the definition of $m$-bisimilarity there exists some $\beta'$ satisfying $\beta \stackrel{a}{\rightarrow} \beta'$ and $\alpha' \sim_{m-1} \beta'$. Now the induction hypothesis yields the existence of some $\beta''$ such that $\beta' \stackrel{w}{\rightarrow} \beta''$ and $\mathrm{Sep}(\alpha', \beta') \leq |w| + \mathrm{Sep}(\alpha'', \beta'')$. Thus together we obtain as desired $\mathrm{Sep}(\alpha, \beta) \leq \mathrm{Sep}(\alpha', \beta') + 1 \leq 1 + |w| + \mathrm{Sep}(\alpha'', \beta'')$. □

The connection between separability and sequential composition is revealed by the following lemma.

**Lemma 2.20** *For all* $\alpha, \beta, \zeta, \eta \in V^*$ *such that* $\zeta \sim \eta$, *we have*

1. $1 \leq \mathit{Sep}(\alpha, \beta) \leq \mathit{Sep}(\alpha\zeta, \beta\eta) \leq \mathit{Sep}(\alpha, \beta) + ||\zeta|| \leq \mathit{Sep}(\zeta\alpha, \eta\beta)$
2. $\min\{\, \mathit{Sep}(\alpha, \beta), \mathit{Sep}(\beta, \zeta)\,\} \leq \mathit{Sep}(\alpha, \zeta)$

**Proof:** First note that $1 \leq \mathrm{Sep}(\alpha, \beta)$ is a direct consequence of the definition. If not all separabilities occurring in the lemma are finite, only the inequality $\mathrm{Sep}(\alpha\zeta, \beta\eta) \leq \mathrm{Sep}(\alpha, \beta) + ||\zeta||$, where we assume $\alpha\zeta \sim \beta\eta$, is interesting. Moreover, we can obviously assume that $||\zeta||$ is finite. Now either $||\alpha||$ and $||\beta||$ are infinite, which implies by Lemma 2.12 $\alpha \sim \beta$, or both norms are finite from which we also deduce by means of Lemma 2.10(2.) that $\alpha \sim \beta$, and therefore $\mathrm{Sep}(\alpha, \beta) = \infty$.

To see $\mathrm{Sep}(\alpha,\beta) \leq \mathrm{Sep}(\alpha\zeta,\beta\eta)$, let $\mathrm{Sep}(\alpha,\beta) = m+1$, i.e. $\alpha \sim_m \beta$. Since we can easily prove that $\alpha \sim_m \beta$ implies $\alpha\zeta \sim_m \beta\eta$ by indcution on $m$, we obtain $\mathrm{Sep}(\alpha\zeta,\beta\eta) \geq m+1 = \mathrm{Sep}(\alpha,\beta)$.

The succeeding inequality $\mathrm{Sep}(\alpha\zeta,\beta\eta) \leq \mathrm{Sep}(\alpha,\beta) + ||\zeta||$ is shown by induction on $\mathrm{Sep}(\alpha,\beta)$. For $\mathrm{Sep}(\alpha,\beta) = 1$, we either have wlog. $\alpha = \epsilon$ and $\beta \neq \epsilon$, which implies by Lemma 2.18 $\mathrm{Sep}(\alpha\zeta,\beta\eta) \leq ||\zeta||+1 = \mathrm{Sep}(\alpha,\beta)+||\zeta||$, or we have $\alpha,\beta \neq \epsilon$, which implies $\mathrm{Sep}(\alpha\zeta,\beta\eta) = 1 = \mathrm{Sep}(\alpha,\beta) \leq \mathrm{Sep}(\alpha,\beta) + ||\zeta||$. Now let $\mathrm{Sep}(\alpha,\beta) = m+1$ and wlog. $\alpha \xrightarrow{a} \alpha'$ be a separating transition for $\alpha$ and $\beta$. Furthermore assume that $\mathrm{Sep}(\alpha\zeta,\beta\eta) = k+1$. Then we have $\alpha\zeta \sim_k \beta\eta$ and the existence of some $\beta'$ with $\beta \xrightarrow{a} \beta'$ and $\alpha'\zeta \sim_{k-1} \beta'\eta$. This yields $\mathrm{Sep}(\alpha'\zeta,\beta'\eta) \geq k = \mathrm{Sep}(\alpha\zeta,\beta\eta)-1$. Since $\alpha \xrightarrow{a} \alpha'$ is separating, we conclude that $\mathrm{Sep}(\alpha',\beta') \leq m$, which allows us to apply the induction hypothesis. Summarizing we obtain $\mathrm{Sep}(\alpha\zeta,\beta\eta) \leq \mathrm{Sep}(\alpha'\zeta,\beta'\eta)+1 \leq \mathrm{Sep}(\alpha',\beta')+1+||\zeta|| \leq \mathrm{Sep}(\alpha,\beta)+||\zeta||$.

In order to show $\mathrm{Sep}(\alpha,\beta) + ||\zeta|| \leq \mathrm{Sep}(\zeta\alpha,\eta\beta)$, we construct a separability-reducing transition track for $(\zeta\alpha,\eta\beta)$ with the following strategy: if the actual pair is of the form $\zeta'\alpha \not\sim_i \eta'\beta$ let without loss of generality $\zeta'\alpha \xrightarrow{a} \zeta''\alpha$ be the separating transition[2], and choose some $\eta''$ such that $\eta' \xrightarrow{a} \eta''$ and $\eta'' \sim \zeta''$ which yields $\eta'\beta \xrightarrow{a} \eta''\beta$ as the matching transition together with $\zeta''\alpha \not\sim_{i-1} \eta''\beta$. Eventually this will lead us $\alpha \not\sim_m \beta$ where $m = \mathrm{Sep}(\alpha,\beta)$. Thus together we obtain $||\zeta|| + \mathrm{Sep}(\alpha,\beta) \leq \mathrm{Sep}(\zeta\alpha,\eta\beta)$.

The second part can be proved as follows. Assume without loss of generality that $m+1 = \mathrm{Sep}(\alpha,\beta) \leq \mathrm{Sep}(\beta,\zeta)$. Then $\alpha \sim_m \beta \sim_m \zeta$, and by transitivity also $\alpha \sim_m \zeta$. Thus we obtain $\mathrm{Sep}(\alpha,\beta) \leq \mathrm{Sep}(\alpha,\zeta)$. $\square$

An example that the inequality $\mathrm{Sep}(\alpha,\beta) + ||\zeta|| \leq \mathrm{Sep}(\zeta\alpha,\eta\beta)$ may be proper is given by $\alpha = a, \beta = b$ and $\zeta = a + a\alpha + a\beta$. In this case $\mathrm{Sep}(\alpha,\beta) = 1, ||\zeta|| = 1$, but $\mathrm{Sep}(\zeta\alpha,\zeta\beta) = 3$. Note that we always have equality when considering simple grammars.

The following extension of the notion of separability to relations is important.

**Definition 2.21** *Let* $Sep(R) =_{\mathrm{df}} \min\{\, Sep(\alpha,\beta) \mid (\alpha,\beta) \in R \,\}$.

**Lemma 2.22** *Let $R$ be a binary relation on $V^*$. Then for all $\alpha \overset{*}{\leftrightarrow}_R \alpha', \beta \overset{*}{\leftrightarrow}_R \beta'$ we have*

$$Sep(R \cup \{\,(\alpha,\beta)\,\}) \leq Sep(\alpha',\beta')$$

**Proof:** The one-step rewriting $\rightarrow_R$ can be extended to $(V^*)^2 \times (V^*)^2$ by

$$(\alpha,\beta) \rightarrow_R (\alpha',\beta') \quad \text{if} \quad (\alpha \rightarrow_R \alpha' \wedge \beta = \beta') \text{ or } (\alpha = \alpha' \wedge \beta \rightarrow_R \beta')$$

which allows us to define $(\alpha,\beta) \overset{i}{\leftrightarrow}_R (\alpha',\beta')$ in the usual fashion. The lemma is then shown by induction on $i$. For $i = 0$ simply observe that

$$\mathrm{Sep}(R \cup \{\,(\alpha,\beta)\,\}) = \min\{\,\mathrm{Sep}(R), \mathrm{Sep}(\alpha,\beta)\,\} \leq \mathrm{Sep}(\alpha,\beta).$$

[2]This transition needs not to be norm-reducing as in the case of simple grammars.

Now assume $(\alpha, \beta) \overset{i}{\leftrightarrow}_R (\alpha'', \beta'') \leftrightarrow_R (\alpha', \beta')$ and wlog. $\alpha'' \leftrightarrow_R \alpha'$. Then by induction hypothesis we have $\mathrm{Sep}(R \cup \{ (\alpha, \beta) \}) \leq \mathrm{Sep}(\alpha'', \beta'') = \mathrm{Sep}(\alpha'', \beta')$ and by definition $\mathrm{Sep}(R) \leq \mathrm{Sep}(\alpha'', \alpha')$. Summarizing we obtain

$$\mathrm{Sep}(R \cup \{ (\alpha, \beta) \}) \leq \min\{ \mathrm{Sep}(\alpha', \alpha''), \mathrm{Sep}(\alpha'', \beta') \} \leq \mathrm{Sep}(\alpha', \beta').$$

□

# 3 A branching algorithm for deciding bisimilarity of normed BPA

In this section we present a branching algorithm for deciding bisimilarity of normed context-free processes. Thereby we improve on a similar tableau system presented in [HS91] in two respects: first we introduce an explicit `SPLIT` rule which replaces the "eliminating subtableau" construction and secondly, we use a depth-first, left-to-right derivation strategy together with variable reduction. Both techniques reduce the exponentiality of the size of a tableau. Subsequently, we use the branching algorithm to give a bound on the number of transitions needed to separate two non-bisimilar normed BPA processes, formerly shown for simple grammars in [Cau89].

Since our branching algorithm is tree-oriented, we start by presenting the neccessary notations. Given a set $M$, an *$M$-valued tree $T$* is a map from the set of nodes $N(T) \subseteq \mathbb{N}_{>0}^*$ to $M$ such that $N(T)$ is prefix closed and satisfies $uj \in N(T) \Rightarrow \forall\, i < j.\ ui \in N(T)$.

The branching algorithm will construct a tree where each node is labelled with an equation $\alpha = \beta$ for some $\alpha, \beta \in V^*$. Starting with the root each node is subsequently expanded in lexical order by using a *rule* which has the form

$$\frac{\alpha = \beta}{\alpha_1 = \beta_1 \qquad \ldots \qquad \alpha_m = \beta_m}$$

with side conditions restricting the applicability. Intuitively, this rule expresses that the goal $\alpha \sim \beta$, the node under consideration, can be proved by verifying the subgoals $\alpha_1 = \beta_1 \quad \ldots \quad \alpha_m = \beta_m$ constituting sons in the tree. During the construction of the tree we will extract a *fundamental* relation which is defined as follows.

**Definition 3.1** *Let $R$ be a binary relation over $V^*$.*

- *The* domain *of $R$ is the set $dom(R) =_{\mathrm{df}} \{ \alpha \mid \alpha R \beta \}$ while the* image *of $R$ is the set $im(R) =_{\mathrm{df}} \{ \beta \mid \alpha R \beta \}$.*
- *$R$ is called* fundamental*, if it satisfies the following three conditions:*
  1. *$dom(R) \subseteq V$ and $im(R) \subseteq (V \setminus dom(R))^*$,*
  2. *$R$ is* functional*: if $XR\alpha$ and $XR\beta$ then $\alpha = \beta$, and*
  3. *$R$ is* norm-preserving*: if $XR\alpha$ then $||X|| = ||\alpha||$.*

| | | |
|---|---|---|
| `REDUCE` | $\dfrac{\alpha = \beta}{\alpha\downarrow R = \beta\downarrow R}$ | where $\alpha \neq \alpha\downarrow R$ or $\beta \neq \beta\downarrow R$. |
| `SWAP` | $\dfrac{X_j\beta = X_i\alpha}{X_i\alpha = X_j\beta}$ | where $X_j\beta, X_i\alpha$ are in normal form, $j > i$. |
| `LCANCEL` | $\dfrac{X_i\alpha = X_i\beta}{\alpha = \gamma\beta}$ | where $X_i\alpha, X_i\beta$ are in normal form, $\|\|X_i\alpha\|\| = \|\|X_i\beta\|\|, \exists\, w.\ \|w\| = \|\|X_i\|\|$, $X_i \stackrel{w}{\to} \epsilon$ and $X_i \stackrel{w}{\to} \gamma$. |
| `SPLIT` | $\dfrac{X_i\alpha = X_j\beta}{X_i\gamma = X_j \qquad \alpha = \gamma\beta}$ | where $X_i\alpha, X_j\beta$ are in normal form, $i < j, \|\|X_i\alpha\|\| = \|\|X_j\beta\|\|, \alpha, \beta \neq \epsilon$, $\exists\, w.\ \|w\| = \|\|X_i\|\|, X_i \stackrel{w}{\to} \epsilon$ and $X_j \stackrel{w}{\to} \gamma$. |
| `UNFOLD` | $\dfrac{X_i\alpha = X_j}{\{\,\alpha_i\alpha = \beta_{f(i)}\,\}_{i=1}^k \qquad \{\,\alpha_{g(j)}\alpha = \beta_j\,\}_{j=1}^l}$ | where $X_i\alpha, X_j$ are in normal form, $i < j, \|\|X_i\alpha\|\| = \|\|X_j\|\|, X_i \sim_1 X_j$, $X_i =_{\rm df} \sum_{i=1}^k a_i\alpha_i, X_j =_{\rm df} \sum_{j=1}^l b_j\beta_j$, $f : [k] \to [l],\ \forall\, i \in [k].\ a_i = b_{f(i)}$, $g : [l] \to [k],\ \forall\, j \in [l].\ a_{g(j)} = b_j$. |

Table 1: The expansion rules of the branching algorithm.

It is easy to see that any fundamental relation $R$ is confluent and noetherian. Thus any $\alpha$ possesses a unique normal form $\alpha\downarrow R$. Moreover, if *card* denotes the cardinality of either sets or relations, we have $card(R) < card(V)$ due to property (1). This puts us in a position to give the details of our branching algorithm.

Let `BA` be the recursive procedure which takes as input parameter a tuple $(T, u, R)$ where $T$ is a tree, $u$ is a node of the tree and $R$ a fundamental relation, and returns either "failure" or a triple $(T', u', R')$ of the same type as $(T, u, R)$. Abbreviating $\{\,1, \ldots, n\,\}$ by $[n]$, the procedure `BA`$(T, u, R)$ is defined by means of the rules listed in Table 1 and the instructions given in Table 2.

Even though there is at most one rule applicable during the execution of the procedure the algorithm is still *nondeterministic*, since we have, for example, to guess the right functions $f$ and $g$ for every application of the `UNFOLD` rule. Thus there may be many execution sequence of `BA`, which we call *runs* of `BA` in the sequel.

To determine whether $\alpha \sim \beta$, we start the branching algorithm with `BA`$((\epsilon, \alpha = \beta), \epsilon, \emptyset)$. Before proving the soundness, completeness and termination of the algorithm, we summarize some basic facts about rule applications.

**Lemma 3.2** *Concerning the minimal norm of left and right-hand sides of equations occuring as premises in rule applications, the `REDUCE` and the `SWAP` rule are norm-preserving,* Let $T(u)$ be labelled by $\alpha = \beta$.

1. If no rule is applicable then
   - If $\alpha, \beta = \epsilon$
     then if $u$ has a successor $v$ in the lexical order
     - then continue with $\mathtt{BA}(T, v, R)$
     - else stop successfully with $(T, u, R)$.
   - Otherwise stop with "failure".
2. Else apply the unique applicable rule $r$ to $u$. Let $e_1, \ldots, e_m$ be the consequents of the rule application, $T' = T \cup \{\, (ui, e_i) \mid 1 \leq i \leq m \,\}$ and $u' = u1$. If $r$ is the `UNFOLD` rule
   - then continue with $\mathtt{BA}(T', u', R \cup \{\, (X_j, X_i\alpha) \,\})$
   - else continue with $\mathtt{BA}(T', u', R)$.

Table 2: The instructions of the branching algorithm.

*while the `LCANCEL` and the `SPLIT` rule are strictly norm-reducing, i.e.*

$$\begin{array}{llcl}
\mathit{REDUCE}: & ||\alpha, \beta|| & = & ||\alpha \downarrow R, \beta \downarrow R|| \\
\mathit{SWAP}: & ||X_j\beta, X_i\alpha|| & = & ||X_i\alpha, X_j\beta|| \\
\mathit{LCANCEL}: & ||X_i\alpha, X_i\beta|| & > & ||\alpha, \gamma\beta|| \\
\mathit{SPLIT}: & ||X_i\alpha, X_j\beta|| & > & \max\{\, ||X_i\gamma, X_j||, ||\alpha, \gamma\beta|| \,\}
\end{array}$$

**Proof:** The equality for the `REDUCE` rule follows from the norm-preserving property of $R$, while the other equations are a direct consequence of the definitions. □

**Proposition 3.3** *There are only finitely many runs for $\alpha = \beta$, each of which is finite.*

**Proof:** Consider the sequence $(T_i, u_i, R_i)_{i \geq 0}$ of successive call parameters of `BA` with $T_0 = \{\, (\epsilon, (\alpha, \beta)) \,\}$, $u_0 = \epsilon$ and $R = \emptyset$. By induction on $i$ it is easy to show that each $R_i$ is a fundamental relation. Now let $T$ be the tree constructed by a run of `BA`. Then the sequence of rules applied to the nodes occuring on a path in the tree are a word of the regular language represented by the BNF-expression:

$$\begin{array}{ll}
( & [\mathtt{REDUCE}]\ \mathtt{LCANCEL} \mid \\
 & [\mathtt{REDUCE}]\ [\mathtt{SWAP}]\ (\mathtt{SPLIT} \mid \mathtt{UNFOLD}) \quad )^*
\end{array}$$

Since each $R_i$ is fundamental, we have at most $card(V) - 1$ many `UNFOLD` applications. Considering $||\alpha, \beta||$ for each node label $\alpha = \beta$, we deduce from Lemma 3.2 that every path in $T$ has finite length. Since also every rule application has finitely many consequents only, we obtain that $T$ must be finite. Finally, the finite-branching property of BPA processes keeps the number of runs of `BA` finite. □

**Proposition 3.4** *The branching algorithm is sound and complete, i.e. we have $\alpha \sim \beta$ iff there exists a successful run for $\alpha = \beta$.*

**Proof:** Assume $\alpha \sim \beta$. Then we build a tree for $\alpha = \beta$ in such a way that for each node labelled with $\zeta = \eta$ we have $\zeta \sim \eta$. If we have to apply the `LCANCEL` rule to $X_i\alpha = X_i\beta$, let $\gamma = \epsilon$ and thus $X_i \stackrel{w}{\to} \beta$. From Lemma 2.10(1.) it easily follows that the consequents resulting from such an application are bisimilar if the antecedent is. Obviously the same holds for any application of the `SWAP` rule. If we have to apply the `SPLIT` rule choose $\gamma$ as given by the splitting Lemma 2.11, and in the case of an `UNFOLD` application simply choose the matching transitions which must exist due to the definition of bisimulation. This way only bisimilar pairs are added to $R$ during an `UNFOLD` application. Then the congruence Lemma 2.9(3.) yields the correctness of the `REDUCE` rule. By Proposition 3.3 this construction must terminate, and from instruction (1) of the branching algorithm we see that all resulting terminals are of the form $\epsilon = \epsilon$.

Let now $(T, u, R)$ be the value returned by `BA`. Then we prove soundness by showing that

$$S = \{ (\alpha, \beta) \mid \alpha = \beta \text{ is a node labelling in the tree } T \} \cup \{ (\alpha, \alpha) \mid \alpha \in V^* \}$$

is a self-bisimulation, by induction on the depth of subtrees.

For leaves ($\epsilon = \epsilon$) nothing needs to be shown. Thus assume $\alpha = \beta$ is an inner node. First, we deal with the easiest case, i.e. the `UNFOLD` rule is applied to $\alpha = \beta$. This guarantees that for each $\alpha \stackrel{a}{\to} \alpha'$ we have $\beta \stackrel{a}{\to} \beta'$ for some $\beta'$ such that $(\alpha', \beta') \in S$ and symmetrically, for each $\beta \stackrel{a}{\to} \beta'$ we have $\alpha \stackrel{a}{\to} \alpha'$ for some $\alpha'$ such that $(\alpha', \beta') \in S$, as desired.

Second, if $\alpha = \beta$ is the premise of a `SWAP` application the self-bisimulation condition follows by induction hypothesis from the self-bisimulation property of $(\beta, \alpha)$. Third, if the `SPLIT` rule is applied to $\alpha = \beta$, let $X_i\gamma$ and $\alpha = \gamma\beta$ be the consequents. Now consider $X_i\alpha \stackrel{a}{\to} \alpha'\alpha$ which implies $X_i\gamma \stackrel{a}{\to} \alpha'\gamma$. Then we know that $X_j \stackrel{a}{\to} \beta'$ for some $\beta'$ such that by induction hypothesis $\alpha'\gamma \stackrel{*}{\leftrightarrow}_S \beta'$. This yields $X_j\beta \stackrel{a}{\to} \beta'\beta$, and observing that $(\alpha, \gamma\beta) \in S$ we obtain $\alpha'\alpha \stackrel{*}{\leftrightarrow}_S \alpha'\gamma\beta \stackrel{*}{\leftrightarrow}_S \beta'\beta$. The symmetric case is shown in a similar way. Forth, since $T$ is the result of a successful run of `BA`, in case of an `LCANCEL` application to $X_i\alpha = X_i\beta$, it is easy to prove that the consequent must be $\alpha = \beta$ and that $\gamma\alpha \stackrel{*}{\leftrightarrow}_S \gamma\beta$ for all $\gamma$ with $X_i \stackrel{a}{\to} \gamma$.

Sixth, in the remaining case of a `REDUCE` application, the self-bisimulation condition is a consequence of the induction hypothesis and the following claim, as

$$X_i\alpha \cong_S X_i\alpha \downarrow R \cong_S X_j\beta \downarrow R \cong_S X_j\beta$$

**Claim :** For any process $X_i\alpha \in V^+$ we have $X_i\alpha \cong_S X_i\alpha \downarrow R$.

For the proof of the claim, which proceeds by induction on $n$, let $X_i\alpha \stackrel{n}{\to}_R X_i\alpha \downarrow R$. If $n = 0$ the claim trivially holds. Now let $X_i\alpha \to_R \zeta\alpha' \stackrel{n}{\to}_R X_i\alpha \downarrow R$.

**Case 1:** $X_i\alpha \to_R \zeta\alpha' \stackrel{n}{\to}_R$ is due to $X_i \to_R \zeta$ which implies $\alpha' = \alpha$. Since $X_i = \zeta$ is then the label of a node formerly expanded by the `UNFOLD` rule we have $X_i \cong_S \zeta$ and by induction hypothesis also $\zeta\alpha' \cong_S X_i\alpha \downarrow R$. Thus we obtain $X_i\alpha \cong_S \zeta\alpha \cong_S X_i\alpha \downarrow R$.

**Case 2:** $X_i\alpha \to_R \zeta\alpha' \stackrel{n}{\to}_R$ is due to $\alpha \to_R \alpha'$ which implies $\zeta = X_i$. Since $R \subseteq S$ we obtain for any $a \in Act$:

1. $X_i\alpha \stackrel{a}{\to} \gamma\alpha$ implies $X_i\alpha' \stackrel{a}{\to} \gamma\alpha'$ and $\gamma\alpha \stackrel{*}{\leftrightarrow}_S \gamma\alpha'$,
2. $X_i\alpha' \stackrel{a}{\to} \gamma\alpha'$ implies $X_i\alpha \stackrel{a}{\to} \gamma\alpha$ and $\gamma\alpha' \stackrel{*}{\leftrightarrow}_S \gamma\alpha$.

Thus also in this case we have $X_i\alpha \cong_S X_i\alpha' \cong_S X_i\alpha \downarrow R$.

□

## 3.1 A bound for separability

In this section we develop a bound for the separability of two non-bisimilar normed BPA processes from the branching algorithm presented above. First, we give a bound for the separability of two processes occuring in a node which cannot be further expanded. Then we relate the separabilities of the premise and the consequents of rule applications.

**Lemma 3.5** *If no rule is applicable to an equation $X_i\alpha = X_j\beta$, we have*

$$Sep(X_i\alpha, X_j\beta) \leq ||X_i\alpha, X_j\beta|| + 1.$$

**Proof:** Since none of the rules `REDUCE`, `LCANCEL` and `SWAP` is applicable, we conclude that $X_i\alpha$ as well as $X_j\beta$ are in normal form, and that $i < j$. Thus either the left and right-hand side of the equation have different norms and the inequality follows from Lemma 2.18, or we have $||X_i\alpha|| = ||X_j\beta||$ and must distinguish between $\min\{\,|X_i\alpha|, |X_j\beta|\,\} = 1$ and $\min\{\,|X_i\alpha|, |X_j\beta|\,\} > 1$. In the first case the `UNFOLD` rule is not applicable guaranteeing $\mathrm{Sep}(X_i\alpha, X_j\beta) = 1$, and in the latter case the `SPLIT` rule cannot be applied and hence $\mathrm{Sep}(X_i\alpha, X_j\beta) \leq ||X_i|| + 1 \leq ||X_i\alpha|| + 1$. □

**Remark 3.6** *For any `SWAP` application to $X_j\beta = X_i\alpha$ we obviously have:*

$$Sep(X_j\beta, X_i\alpha) \leq Sep(X_i\alpha, X_j\beta) + ||X_j\beta, X_i\alpha|| - ||X_i\alpha, X_j\beta||.$$

**Lemma 3.7** *There exists an application of the `LCANCEL` rule to $X_i\alpha = X_i\beta$ such that for the consequent $\alpha = \gamma\beta$ we have:*

$$Sep(X_i\alpha, X_i\beta) \leq Sep(\alpha, \gamma\beta) + ||X_i\alpha, X_i\beta|| - ||\alpha, \gamma\beta||$$

**Proof:** Let $X_i\alpha \stackrel{w}{\to} \alpha$ with $|w| = ||X_i||$. Due to Lemma 2.19 there exists some $\gamma$ with $X_i\beta \stackrel{w}{\to} \gamma\beta$ and $\mathrm{Sep}(X_i\alpha, X_i\beta) \leq |w| + \mathrm{Sep}(\alpha, \gamma\beta)$. Thus we obtain:

$$\begin{aligned}
\mathrm{Sep}(X_i\alpha, X_i\beta) &\leq |w| + \mathrm{Sep}(\alpha, \gamma\beta) \\
&= \mathrm{Sep}(\alpha, \gamma\beta) + ||X_i|| + ||\alpha, \beta|| - ||\alpha, \beta|| \\
&= \mathrm{Sep}(\alpha, \gamma\beta) + ||X_i\alpha, X_i\beta|| - ||\alpha, \gamma\beta||
\end{aligned}$$

□

**Lemma 3.8** *There exists an application of the* SPLIT *rule to* $X_i\alpha = X_j\beta$ *such that for every consequent* $\zeta = \eta$ *we have:*

$$Sep(X_i\alpha, X_j\beta) \leq Sep(\zeta, \eta) + ||X_i\alpha, X_j\beta|| - ||\zeta, \eta||.$$

**Proof:** First observe that for every application of the `SPLIT` rule the following two equalities hold:

$$\begin{array}{ll} (*) & ||X_i\alpha, X_j\beta|| = ||X_i\alpha|| = ||X_i|| + ||\alpha|| = ||X_i|| + ||\alpha, \gamma\beta||, \text{ and} \\ (**) & ||X_i\alpha, X_j\beta|| = ||X_j\beta|| = ||X_j|| + ||\beta|| = ||\beta|| + ||X_i\gamma, X_j|| \end{array}$$

Now let $X_i\alpha \stackrel{w}{\to} \alpha$ be a norm-reducing transition sequence. Due to Lemma 2.19 there exists some $\gamma$ such that $X_j\beta \stackrel{w}{\to} \gamma$ and $\mathrm{Sep}(X_i\alpha, X_j\beta) \leq |w| + \mathrm{Sep}(\alpha, \gamma\beta) = ||X_i|| + \mathrm{Sep}(\alpha, \gamma\beta)$. Fixing this $\gamma$, the lemma follows for the consequent $\alpha = \gamma\beta$ from $\mathrm{Sep}(X_i\alpha, X_j\beta) \leq ||X_i|| + \mathrm{Sep}(\alpha, \gamma\beta) = \mathrm{Sep}(\alpha, \gamma\beta) + ||X_i\alpha, X_j\beta|| - ||\alpha, \gamma\beta||$ which relying on (*), while for the consequent $X_i\gamma = X_j$ it follows from the inequality $\mathrm{Sep}(X_i\alpha, X_j\beta) \leq ||X_i|| + \mathrm{Sep}(\alpha, \gamma\beta) \leq \mathrm{Sep}(X_i\alpha, X_i\gamma\beta)$ and (**) by means of

$$\begin{array}{rcl} \mathrm{Sep}(X_i\alpha, X_j\beta) & = & \min\{\,\mathrm{Sep}(X_i\gamma\beta, X_i\alpha), \mathrm{Sep}(X_i\alpha, X_j\beta)\,\} \\ & \leq & \mathrm{Sep}(X_i\gamma\beta, X_j\beta) \\ & \leq & \mathrm{Sep}(X_i\gamma, X_j) + ||\beta|| \\ & = & \mathrm{Sep}(X_i\gamma, X_j) + ||X_i\alpha, X_j\beta|| - ||X_i\gamma, X_j|| \end{array}$$

□

Let $\alpha \not\sim \beta$ and $T$ be an unsuccessful tree with root $\alpha = \beta$. Moreover assume that every consequent $\zeta = \eta$ of an `UNFOLD` application to $X_i\alpha = X_j$ satisfies $\mathrm{Sep}(X_i\alpha, X_j) \leq 1 + \mathrm{Sep}(\zeta, \eta)$, which can be guaranteed by Lemma 2.19, as well as that every `LCANCEL` (`SPLIT`) application in $T$ satisfies the condition of Lemma 3.7 (3.8). Henceforth, we order the nodes of $T$ by the usual lexical ordering $\leq_{lex}$ (depth-first, left-to-right traversal of the tree), split $N(T)$ into the set of nodes $N_U(T)$ whose labels have been expanded by the `UNFOLD` rule, and abbreviate $N(T) \setminus N_U(T)$ by $N_O(T)$. To shorten notation we omit references to $T$ in the sequel. For any $u \in N$, let

$$\begin{array}{rcl} L_u & = & \{\, n \in N_U \mid n \leq_{lex} u\,\}, \text{ and} \\ D_u & = & \{\, uv \in N \mid \forall w.\ \epsilon <_{lex} w <_{lex} v, uw \notin N_U\,\} \end{array}$$

If $N_U = \{\, u_1, \ldots, u_p\,\}$ with $p < n = card(V)$ and $u_i \leq_{lex} u_{i+1}$ the set of tree nodes can be decomposed as $N = \bigcup\{\, D_{u_i} \mid i \in [p]\,\}$. Writing $\mathcal{E} =_{\mathrm{df}} \max\{\, ||E|| \mid X =_{\mathrm{df}} E \in \Delta\,\} - 1$ for the greatest norm of the right-hand sides minus 1, the next two lemmata state properties about the separability of processes occuring as labels of nodes contained in $D_u$.

**Lemma 3.9** $\forall u \in N_O,\ \forall v \in D_u$

- $||T(v)|| \leq ||T(u)||$
- $Sep(T(L_u) \cup \{\, T(u)\,\}) \leq Sep(T(v)) + ||T(u)|| - ||T(v)||$

**Proof:** Both parts will be shown by induction on $|v|-|u|$. If $|v| = |u|$ we simply conclude $u = v$ and the lemma is trivially true. Now let for the induction step $u \to v_1 \to \ldots \to v_n = v$ be a path in the tree. Then the first part follows from the induction hypothesis and Lemma 3.2: $||T(u)|| \geq ||T(v_{n-1})|| \geq ||T(v_n)||$. For the second part we distinguish whether $v_{n-1}$ was expanded by the `REDUCE` rule or one of the rules `SWAP`, `LCANCEL` or `SPLIT`. In the latter case we conclude as follows:

$$\begin{aligned}
\mathrm{Sep}(T(L_u) \cup \{\, T(u) \,\}) &\leq \mathrm{Sep}(T(v_{n-1})) + ||T(u)|| - ||T(v_{n-1})|| \\
&\phantom{\leq} \text{(by induction hypothesis)} \\
&\leq \mathrm{Sep}(T(v_n)) + ||T(v_{n-1})|| - ||T(v_n)|| + ||T(u)|| - ||T(v_{n-1})|| \\
&\phantom{\leq} \text{(by Remark 3.6, Lemma 3.7 or Lemma 3.8, respectively)} \\
&= \mathrm{Sep}(T(v)) + ||T(u)|| - ||T(v)||.
\end{aligned}$$

If, however, $v_{n-1}$ was expanded by the `REDUCE` rule, observe that $||T(v_{n-1})|| = ||T(v_n)||$. We distinguish two cases:

**Case 1:**: $\min\{\, \mathrm{Sep}(T(L_u)), \mathrm{Sep}(T(v_{n-1})) \,\} = \mathrm{Sep}(T(v_{n-1}))$
From Lemma 2.22, we obtain $\mathrm{Sep}(T(v_{n-1})) = \mathrm{Sep}(T(L_u) \cup \{\, T(v_{n-1}) \,\}) \leq \mathrm{Sep}(T(v_n))$. Thus we can conclude as above.

**Case 2:**: $\min\{\, \mathrm{Sep}(T(L_u)), \mathrm{Sep}(T(v_{n-1})) \,\} = \mathrm{Sep}(T(L_u))$
In this case, the application of Lemma 2.22 yields

$$\begin{aligned}
\mathrm{Sep}(T(L_u) \cup \{\, T(u) \,\}) &\leq \mathrm{Sep}(T(L_u)) = \mathrm{Sep}(T(L_u) \cup \{\, T(v_{n-1}) \,\}) \leq \mathrm{Sep}(T(v_n)) \\
&\leq \mathrm{Sep}(T(v_n)) + ||T(v_{n-1})|| - ||T(v_n)|| + ||T(u)|| - ||T(v_{n-1})|| \\
&= \mathrm{Sep}(T(v)) + ||T(u)|| - ||T(v)||.
\end{aligned}$$

□

**Lemma 3.10** $\forall u \in N_U,\ \forall v \in D_u$

- $||T(v)|| \leq \mathcal{E}$
- $\mathit{Sep}(T(L_u)) \leq \mathit{Sep}(T(v)) + \mathcal{E} + 1 - ||T(v)||$

**Proof:** Let $u$ labelled by $X_i\alpha = X_j$ and $u \overset{\texttt{UNFOLD}}{\to} v_1 \to \ldots \to v_n = v$ be a path in the tree such that $v_i \in D_u$ for all $1 \leq i \leq n$. Since $u$ was unfolded, for any $a \in \mathit{Act}$ we have $X_j \overset{a}{\to} \gamma$ implies $||\gamma|| \leq \mathcal{E}$. Hence, we have $||T(v_1)|| \leq \mathcal{E}$ and by the assumption about the `UNFOLD` rule $\mathrm{Sep}(T(L_u)) = \min\{\, \mathrm{Sep}(T(L_u \setminus \{\, u \,\})), \mathrm{Sep}(T(u)) \,\} \leq \mathrm{Sep}(T(u)) \leq 1 + \mathrm{Sep}(T(v_1))$. We proceed by case analysis.

**Case 1:**: $v_1 \in N_U$
The definition of $D_u$ implies that $v = v_1$. Thus $||T(v)|| \leq \mathcal{E}$ as well as $\mathrm{Sep}(T(L_u)) \leq \mathrm{Sep}(T(v_1)) + 1 \leq \mathrm{Sep}(T(v_1)) + 1 + \mathcal{E} - ||T(v_1)||$.

**Case 2:**: $v_1 \notin N_U$
In this case we have by definition of $D_u$ $L_{v_1} = L_u$. Thus Lemma 3.9 yields: $||T(v)|| \leq ||T(v_1)|| \leq \mathcal{E}$ and

$$\begin{aligned}
\mathrm{Sep}(T(L_u)) &\leq \min\{\, \mathrm{Sep}(T(L_{v_1})), 1 + \mathrm{Sep}(T(v_1)) \,\} \\
&\leq \mathrm{Sep}(T(L_{v_1}) \cup \{\, T(v_1) \,\}) + 1 \\
&\leq \mathrm{Sep}(T(v)) + 1 + ||T(v_1)|| - ||T(v)|| \\
&\leq \mathrm{Sep}(T(v)) + 1 + \mathcal{E} - ||T(v)||.
\end{aligned}$$

□

**Lemma 3.11** *Whenever the root $\alpha = \beta$ of the tree is expanded by the* `UNFOLD` *rule, i.e. $T(\epsilon) \in N_U$, we have for every $v \in N$:*

$$Sep(\alpha, \beta) \leq Sep(T(v)) + (n-1)\mathcal{E} + 1 - ||T(v)||$$

**Proof:** The lemma is proved by showing for any $v \in D_{u_i}$:

$$\mathrm{Sep}(\alpha, \beta) \leq \mathrm{Sep}(T(v)) + i\mathcal{E} + 1 - ||T(v)||$$

by induction on $i$. For $i = 1$ and $v \in D_{u_1}$ Lemma 3.10 yields $\mathrm{Sep}(\alpha, \beta) = \mathrm{Sep}(T(u_1)) = \mathrm{Sep}(T(L_{u_1})) \leq \mathrm{Sep}(T(v)) + \mathcal{E} + 1 - ||T(v)||$. In order to show the induction step let now $v \in D_{u_{i+1}}$. Then, again by Lemma 3.10, we know $\mathrm{Sep}(T(L_{u_{i+1}})) \leq \mathrm{Sep}(T(v)) + \mathcal{E} + 1 - ||T(v)||$. Now either $\mathrm{Sep}(T(u_1)) = \mathrm{Sep}(T(L_{u_{i+1}})) \leq \mathrm{Sep}(T(v)) + \mathcal{E} + 1 - ||T(v)|| \leq \mathrm{Sep}(T(v)) + (i+1)\mathcal{E} + 1 - ||T(v)||$ or $\mathrm{Sep}(T(L_{u_{i+1}})) = \mathrm{Sep}(T(u_j))$ for some $1 < j \leq i+1$ with $u_j \in D_{u_{j-1}}$ from which we obtain by induction hypothesis:

$$\begin{aligned}
\mathrm{Sep}(\alpha, \beta) &\leq \mathrm{Sep}(T(u_j)) + (j-1)\mathcal{E} + 1 - ||T(u_j)|| \\
&\leq \mathrm{Sep}(T(v)) + \mathcal{E} + 1 - ||T(v)|| + (j-1)\mathcal{E} + 1 - ||T(u_j)|| \\
&\leq \mathrm{Sep}(T(v)) + j\mathcal{E} + 1 - ||T(v)|| \\
&\leq \mathrm{Sep}(T(v)) + (i+1)\mathcal{E} + 1 - ||T(v)||
\end{aligned}$$

□

Finally, we obtain the main theorem of this section.

**Theorem 3.12** *If $\alpha$ is not bisimilar to $\beta$, then $Sep(\alpha, \beta) \leq \mathcal{B}_{\alpha,\beta}$, where*

$$\mathcal{B}_{\alpha,\beta} = \begin{cases} ||\alpha, \beta|| + 1 & \text{if } ||\alpha|| \neq ||\beta||, \text{ and} \\ (n-1)\mathcal{E} + 1 + ||\alpha, \beta|| & \text{if } ||\alpha|| = ||\beta||. \end{cases}$$

**Proof:** Let $\alpha \not\sim \beta$. If $||\alpha|| \neq ||\beta||$, we conclude by Lemma 2.18 that $\mathrm{Sep}(\alpha, \beta) \leq ||\alpha, \beta|| + 1 = \mathcal{B}_{\alpha,\beta}$. Now suppose $||\alpha|| = ||\beta||$. The theorem is then proved by induction on $||\alpha, \beta||$. For the base case assume $||\alpha, \beta|| = 1$ and let $z$ be the last node of the tree construction which could not be further expanded. By Lemma 3.5 we then know $\mathrm{Sep}(T(z)) \leq ||T(z)|| + 1$. If $z = \epsilon$, we immediately get $\mathrm{Sep}(\alpha, \beta) \leq 1 + ||\alpha, \beta|| \leq \mathcal{B}_{\alpha,\beta}$. Otherwise we have to distinguish which rule could be applied to $\alpha = \beta$. It is easy to see that the `REDUCE` rule is not applicable to $\alpha = \beta$ due to $R_0 = \emptyset$. The same applies for the `LCANCEL` rule, since $||\alpha, \beta|| = 1$ would imply that $\alpha, \beta$ are identical to some $X_i$ yielding a contradiction to the assumption $\alpha \not\sim \beta$. If the root was expanded by the `UNFOLD` rule, we apply Lemma 3.11 to obtain:

$$\begin{aligned}
\mathrm{Sep}(\alpha, \beta) &\leq (n-1)\mathcal{E} + 1 + \mathrm{Sep}(T(z)) - ||T(z)|| \\
&\leq (n-1)\mathcal{E} + 1 + 1 \\
&\leq (n-1)\mathcal{E} + 1 + ||\alpha, \beta|| = \mathcal{B}_{\alpha,\beta}
\end{aligned}$$

Moreover, in case of a `SWAP` rule application, we have $\mathrm{Sep}(\alpha, \beta) = \mathrm{Sep}(\beta, \alpha)$. Thus we can conclude for the consequent as in the former cases.

For the induction step assume $||\alpha, \beta|| = i + 1$. Then the `REDUCE` rule is still not applicable,. In case of an `UNFOLD` expansion we apply Lemma 3.11 again, and for a `SWAP` rule application the arguments for the base case apply. Let now the root be of the form $X_i\alpha = X_i\beta$ such that the `LCANCEL` rule is applied to it yielding the consequent $\alpha = \gamma\beta$. Then by induction hypothesis

$$\begin{aligned}
\mathrm{Sep}(X_i\alpha, X_i\beta) &\leq ||X_i|| + \mathrm{Sep}(\alpha, \gamma\beta) \\
&\leq ||X_i|| + \mathcal{B}_{\alpha,\gamma\beta} \\
&\leq ||X_i|| + (n-1)\mathcal{E} + 1 + ||\alpha, \gamma\beta|| \\
&= (n-1)\mathcal{E} + 1 + ||X_i\alpha, X_i\beta|| = \mathcal{B}_{X_i\alpha, X_i\beta}.
\end{aligned}$$

Finally, we consider a `SPLIT` rule application to a root of the form $X_i\alpha = X_j\beta$. By Lemma 3.8 we know that there exists a consequent $\zeta = \eta$ such that we have

$$\begin{aligned}
\mathrm{Sep}(\alpha, \beta) &\leq \mathrm{Sep}(\zeta, \eta) + ||\alpha, \beta|| - ||\zeta, \eta|| \\
&\leq \mathcal{B}_{\zeta,\eta} + ||\alpha, \beta|| - ||\zeta, \eta|| = \mathcal{B}_{\alpha,\beta}.
\end{aligned}$$

This completes the proof. □

In the next section we will apply this theorem to a slightly larger class of processes, namely BPA$_\delta$. The constant symbol $\delta$ represents *deadlock*, i.e. a process which cannot proceed. Its behaviour is captured by the axioms $E+\delta = E$ and $\delta E = \delta$. A BPA$_\delta$ system is *normed*, if each variable $X$ is either normed in the usual sense or *every* transition sequence starting at $X$ reaches $\delta$ in finitely many steps. It is easy to show that the splitting rule is still valid for normed BPA$_\delta$ systems, which allows us to apply the bound given above. This technical but straightforward variant is important for the proof of Proposition 4.8.

# 4 The Algorithm

In this section we present our three step algorithm for deciding bisimilarity of context-free processes. The first two steps are required for the construction of a *bisimulation base* $B$. Bisimulation bases characterize bisimulation equivalence as the least congruence w.r.t. sequential composition containing $B$. This can be exploited for a branching algorithm, which completes the decision procedure in a third step. We will concentrate on the first step here, as the second step can be obtained by combining results from (cf. [CHS92, HJM94]), and the third step is rather straightforward. Thus we start by sketching the second step, which also serves as a good motivation for the subsequently presented first step of our algorithm. Finally, we summarize the results to obtain the complete decision procedure.

## 4.1 Bisimulation Bases

An important difference between the theory of normed and unnormed BPA processes is the existence of two kinds of bisimilar pairs, *decomposable* and *elementary* pairs (cf. [CHS92]).

**Definition 4.1** *Let $X_i\alpha \sim X_j\beta$ with $i \leq j$ and $X_i, X_j \in V_N$. We say that the pair $(X_i\alpha, X_j\beta)$ is* decomposable *if there exists some $\gamma$ such that $\alpha \sim \gamma\beta$ and $X_i\gamma \sim X_j$. If the pair is not decomposable it is said to be* elementary.

An immediate consequence of this definition is that if $(X_i\alpha, X_j\beta)$ is elementary, then $\alpha$ and $\beta$ must be unnormed. Thus in the normed case only decomposable pairs can occur.

In order to prove termination, we extend the notion of norm to a *seminorm* on arbitrary context-free processes as follows: we define the *seminorm* of a variable sequence $\alpha \in \mathcal{S}(V^*)$ as $||\alpha X||_s = ||\alpha X||$ if $X \in V_N$ and $||\alpha X||_s = ||\alpha||$ otherwise. The seminorm is used to define a well-founded ordering on $\mathcal{S}(V^*) \times \mathcal{S}(V^*)$ by $(\alpha_1, \alpha_2) \sqsubseteq (\beta_1, \beta_2)$ iff $\max\{\, ||\alpha_1||_s, ||\alpha_2||_s \,\} \leq \max\{\, ||\beta_1||_s, ||\beta_2||_s \,\}$.

**Definition 4.2**

- *A* base *is a binary relation consisting of pairs $(X_i\alpha, X_j\beta) \in \mathcal{S}(V^+) \times \mathcal{S}(V^+)$ with $i \leq j$.*
- *A base $B$ is called* bisimulation-complete *iff whenever $X_i\alpha \sim X_j\beta$ with $i \leq j$ then one of the following conditions hold:*
  1. *$(X_i\alpha, X_j\beta)$ is decomposable, i.e. we have in particular $X_i\gamma \sim X_j$ for some $\gamma$, and $(X_i\gamma', X_j) \in B$ for some $\gamma' \sim \gamma$.*
  2. *$(X_i\alpha, X_j\beta)$ is elementary and $(X_i\alpha', X_j\beta') \in B$ for some $\alpha' \sim \alpha$ and $\beta' \sim \beta$ such that $(\alpha', \beta') \sqsubseteq (\alpha, \beta)$.*
- *The relation $\equiv_{\mathcal{B}}$ is defined recursively by:*
  1. *$\epsilon \equiv_{\mathcal{B}} \epsilon$ and*
  2. *$X_i\alpha \equiv_{\mathcal{B}} X_j\beta$ iff (a) $(X_i\gamma, X_j) \in B$ and $\alpha \equiv_{\mathcal{B}} \gamma\beta$ or*

     *(b) $(X_i\alpha', X_j\beta') \in B$, $\alpha \equiv_{\mathcal{B}} \alpha'$ and $\beta \equiv_{\mathcal{B}} \beta'$.*

The importance of the relation $\equiv_{\mathcal{B}}$ is revealed by the following lemma.

**Lemma 4.3**

1. *$\equiv_{\mathcal{B}} \subseteq \overset{*}{\leftrightarrow}_{\mathcal{B}}$*
2. *If $\mathcal{B}$ is bisimulation-complete then $\sim \subseteq \equiv_{\mathcal{B}}$.*

**Proof:** The first part is obvious. The second part is shown by induction with respect to $\sqsubseteq$. First observe that $\epsilon \equiv_{\mathcal{B}} \epsilon$. Let us now assume $X_i \sim X_j$ for some $X_i, X_j \in V_U$. The pair $(X_i, X_j)$ is elementary, so by bisimulation-completeness of $\mathcal{B}$ we know $(X_i, X_j) \in \mathcal{B}$ which implies $X_i \equiv_{\mathcal{B}} X_j$. Thus the induction base holds. For the induction step now assume $X_i\alpha \sim X_j\beta$. We have to distinguish two cases. First assume that $(X_i\alpha, X_j\beta)$ is decomposable into $X_i\gamma \sim X_j$ and $\alpha \sim \gamma\beta$ for some $\gamma$. By bisimulation-completeness of $\mathcal{B}$

we have $(X_i\gamma', X_j) \in \mathcal{B}$ for some $\gamma' \sim \gamma$ which implies $\alpha \sim \gamma'\beta$. Observing that $(\alpha, \gamma'\beta) \sqsubset (X_i\alpha, X_j\beta)$ the induction hypothesis yields $\alpha \equiv_{\mathcal{B}} \gamma'\beta$ and therefore $X_i\alpha \equiv_{\mathcal{B}} X_j\beta$. To show the remaining case assume that $(X_i\alpha, X_j\beta)$ is elementary. Then the bisimulation-completeness of $\mathcal{B}$ delivers $(X_i\alpha', X_j\beta') \in \mathcal{B}$ for some $\alpha' \sim \alpha$ and $\beta' \sim \beta$ such that $(\alpha', \beta') \sqsubseteq (\alpha, \beta)$. Using $(\alpha', \beta') \sqsubseteq (\alpha, \beta) \sqsubset (X_i\alpha, X_j\beta)$ we conclude by induction $\alpha \equiv_{\mathcal{B}} \alpha'$ and $\beta \equiv_{\mathcal{B}} \beta'$ and therefore $X_i\alpha \equiv_{\mathcal{B}} X_j\beta$. $\square$

The key structure for our decidability result are *bisimulation bases*, which are relations $B$ satisfying $\sim \; = \; \overset{*}{\leftrightarrow}_{\mathcal{B}}$. They are important as they characterize bisimulation equivalence as the least congruence w.r.t. sequential composition containing $B$. As a consequence of Lemma 4.3, one of the inclusions, $\sim \; \subseteq \; \equiv_{\mathcal{B}}$, is guaranteed for bisimulation-complete relations $B$. A sufficient condition for the inverse inclusion, which can be proved for our construction, is 'self-bisimulation'.

**Definition 4.4** *Given a base $B$, define the subbase $\mathcal{R}(B)$ by: $(\alpha, \beta) \in \mathcal{R}(B)$ iff $(\alpha, \beta) \in B$ and*

1. *$\alpha \xrightarrow{a} \alpha'$ implies $\exists\, \beta'.\ \beta \xrightarrow{a} \beta' \;\wedge\; \alpha' \equiv_{\mathcal{B}} \beta'$*
2. *$\beta \xrightarrow{a} \beta'$ implies $\exists\, \alpha'.\ \alpha \xrightarrow{a} \alpha' \;\wedge\; \alpha' \equiv_{\mathcal{B}} \beta'$*

That $\mathcal{R}$ is a good candidate for successively reducing a bisimulation-complete relation to a bisimulation base is a consequence of the following lemma.

**Lemma 4.5** *Let $\mathcal{B}$ be a bisimulation-complete base. Then the following holds:*

1. *If $(\alpha, \beta) \in \mathcal{B}$ and $\alpha \sim \beta$ then $(\alpha, \beta) \in \mathcal{R}(\mathcal{B})$.*
2. *$\mathcal{R}(\mathcal{B})$ is bisimulation-complete.*

**Proof:** For the first part assume $(\alpha, \beta) \in \mathcal{B}$ and $\alpha \sim \beta$. Whenever $\alpha \xrightarrow{a} \alpha'$ then $\beta \xrightarrow{a} \beta'$ and $\alpha' \sim \beta'$ for some $\beta'$. Thus by Lemma 4.3 we have $\alpha \equiv_{\mathcal{B}} \beta$. Similarly, if $\beta \xrightarrow{a} \beta'$ then $\alpha \xrightarrow{a} \alpha'$ and $\alpha' \sim \beta'$ for some $\alpha'$. Again by Lemma 4.3 we conclude $\alpha \equiv_{\mathcal{B}} \beta$. So $(\alpha, \beta) \in \mathcal{R}(\mathcal{B})$.

For the second part, assume $X_i\alpha \sim X_j\beta$. If $(X_i\alpha, X_j\beta)$ is decomposable, i.e. if $X_i\gamma \sim X_j$ and $\alpha \sim \gamma\beta$ for some $\gamma$, then by the bisimulation-completeness of $\mathcal{B}$ we know $(X_i\gamma', X_j) \in \mathcal{B}$ for some $\gamma' \sim \gamma$. Thus $X_i\gamma' \sim X_j$, and by part one the pair $(X_i\gamma', X_j)$ is also contained in $\mathcal{R}(\mathcal{B})$. On the other side, if $(X_i\alpha, X_j\beta)$ is elementary the we have by bisimulation-completeness of $\mathcal{B}$ that $(X_i\alpha', X_j\beta') \in \mathcal{B}$ for some $\alpha' \sim \alpha$ and $\beta' \sim \beta$. Using the first part again, we conclude from $X_i\alpha' \sim X_i\alpha \sim X_j\beta \sim X_j\beta'$ that $(X_i\alpha', X_j\beta') \in \mathcal{R}(\mathcal{B})$. $\square$

Along the lines of [HJM94], this can straightforwardly be exploited to verify that the successive $\mathcal{R}$-refinement of a bisimulation-complete relation yields indeed a bisimulation base: the (additional) fixpoint property of $B_\sim$ is sufficient to establish that $B_\sim$ is a self-bisinulation.

**Theorem 4.6** *If $B_0$ is a bisimulation-complete base then $B_\sim =_{\mathrm{df}} \bigcap \{\, \mathcal{R}^i(B_0) \mid i \geq 0 \,\}$ is a bisimulation base, i.e. we have $\sim \; = \; \overset{*}{\leftrightarrow}_{B_\sim}$.*

## 4.2 The Computation of an Initial Base

Recall that we have given a normalized unnormed BPA system $\mathcal{C} = (V, Act, \Delta, X_1)$. We say a variable $Y \in V_U$ is *crossing*, if $Y$ occurs on the right-hand side of some $X \in V_N$, and we denote the set of all crossing variables by $V_C$. Intuitively, starting with a normed process we can only reach an unnormed one by "crossing" a variable of $V_C$. If $\sim_{\mathcal{C}}$ denotes bisimilarity wrt. $\mathcal{C}$, we construct a normed BPA$_\delta$ system $\mathcal{C}' = (V', Act', \Delta', X_1)$ from $\mathcal{C}$ as follows.

- $V' = V_N \cup V_C$,
- $Act' = Act \cup \{\, a_{[Y]_{\sim_{\mathcal{C}}}} \mid Y \in V_C \,\}$, where each $a_{[Y]_{\sim_{\mathcal{C}}}}$ is an action not occuring in $Act$,
- $\Delta' = \{\, X =_{\mathrm{df}} E \in \Delta \mid X \in V_N \,\} \cup \{\, Y =_{\mathrm{df}} a_{[Y]_{\sim_{\mathcal{C}}}} \delta \mid Y \in V_C \,\}$.

The constructed BPA$_\delta$ $\mathcal{C}'$ system is obviously normed. It represents the behaviour of the normed part of $\mathcal{C}$ as stated in the following proposition.

**Proposition 4.7** *For all $\alpha, \beta \in V_N^* \cup V_C$ we have: $\alpha \sim_{\mathcal{C}} \beta$ iff $\alpha \sim_{\mathcal{C}'} \beta$.*

**Proof:** To prove the direction from left to right, we will show that $R =_{\mathrm{df}} \sim_{\mathcal{C}\,|\,V_N^* \cup V_C} \cup \{\, (\delta, \delta) \,\}$ is a bisimulation wrt. $\mathcal{C}'$. As $\delta$ cannot perform any action we distinguish two cases:

**Case 1:** $\alpha \sim_{\mathcal{C}} \beta$ where $\alpha, \beta \in V_C$.
Let $\alpha \overset{[\alpha]_{\sim_{\mathcal{C}}}}{\longrightarrow}_{\mathcal{C}'} \delta$ be the only possible transition of $\alpha$. Since $\alpha \sim_{\mathcal{C}} \beta$ and therefore $[\alpha]_{\sim_{\mathcal{C}}} = [\beta]_{\sim_{\mathcal{C}}}$ we obtain by construction of $\mathcal{C}'$ that $\beta \overset{[\beta]_{\sim_{\mathcal{C}}}}{\longrightarrow}_{\mathcal{C}'} \delta$ and $(\delta, \delta) \in R$. The other direction is shown by symmetric arguments.

**Case 2:** $\alpha \sim_{\mathcal{C}} \beta$ where $\alpha, \beta \in V_N^*$.
Let $\alpha \overset{a}{\rightarrow}_{\mathcal{C}'} \alpha'$. The we know by the construction of $\mathcal{C}'$ that also $\alpha \overset{a}{\rightarrow}_{\mathcal{C}} \alpha'$, and as $\alpha \sim_{\mathcal{C}} \beta$ this together implies $\beta \overset{a}{\rightarrow}_{\mathcal{C}} \beta'$ and $\alpha' \sim_{\mathcal{C}} \beta'$ for some $\beta'$. Again from the construction of $\mathcal{C}'$ and the fact that $\alpha', \beta' \in V_N^* \cup V_C$ we finally conclude that $\beta \overset{a}{\rightarrow}_{\mathcal{C}'} \beta'$ and $(\alpha', \beta') \in R$. The other direction follows in a similar fashion.

For the remaining direction from right to left., we first assume that $\alpha \sim_{\mathcal{C}'} \beta$ and $\alpha, \beta \in V_C$. Since $\alpha \overset{[\alpha]_{\sim_{\mathcal{C}}}}{\longrightarrow}_{\mathcal{C}'} \delta$ and $\beta \overset{[\beta]_{\sim_{\mathcal{C}}}}{\longrightarrow}_{\mathcal{C}'} \delta$ are the only possible transitions in $\mathcal{C}'$, we conclude from the assumption that $[\alpha]_{\sim_{\mathcal{C}}} = [\beta]_{\sim_{\mathcal{C}}}$ and therefore $\alpha \sim_{\mathcal{C}} \beta$. Finally, the proof is completed by showing that $R =_{\mathrm{df}} \sim_{\mathcal{C}'\,|\,V_N^*} \cup \sim_{\mathcal{C}\,|\,V_N^* V_U}$ is a bisimulation wrt. $\mathcal{C}$. As the case $\alpha \sim_{\mathcal{C}} \beta$, where $\alpha, \beta \in V_N^* V_U$, is trivial, we only consider the situation when $\alpha \sim_{\mathcal{C}'} \beta$ with $\alpha, \beta \in V_N^*$. Thus let $\alpha \overset{a}{\rightarrow}_{\mathcal{C}} \alpha'$. From the construction of $\mathcal{C}'$, we conclude that $\alpha \overset{a}{\rightarrow}_{\mathcal{C}'} \alpha'$, and as $\alpha \sim_{\mathcal{C}'} \beta$ we have $\beta \overset{a}{\rightarrow}_{\mathcal{C}'} \beta'$ and $\alpha' \sim_{\mathcal{C}'} \beta'$ for some $\beta'$. Now we have either $\alpha', \beta' \in V_C$, which according to the first case yields $\alpha' \sim_{\mathcal{C}} \beta'$ and therefore $(\alpha', \beta') \in R$, or $\alpha', \beta' \in V_N^*$, which also guarantees $(\alpha', \beta') \in R$. Since the other direction is symmetric this completes the proof. $\square$

Using this proposition, we are now able to apply the separability bound $\mathcal{B}_{\alpha,\beta}$ obtained in the previous section.

**Proposition 4.8** *Consider the BPA system $\mathcal{C}$ and let $\zeta$ be normed with $\zeta \not\sim \eta$, $||\zeta|| \leq ||\eta||$ and $\zeta\beta \sim \eta\beta$. Then $\beta \sim X$, where $X =_{\mathrm{df}} \gamma X$ for some $\gamma \neq \epsilon$ such that the following conditions hold.*

- *If $||\zeta|| < ||\eta||$ then $\eta \xrightarrow{w} \gamma$ for some fixed $w$ where $\zeta \xrightarrow{w} \epsilon$ is a norm-reducing transition sequence.*
- *If $||\zeta|| = ||\eta||$ then without loss of generality $\zeta \xrightarrow{w} \epsilon$ and $\eta \xrightarrow{w} \gamma$ in $|w| \leq \mathcal{B}(\zeta,\eta)$ steps where $\mathcal{B}(\zeta,\eta) =_{\mathrm{df}} \mathcal{B}_{\zeta,\eta} + (\mathcal{B}_{\zeta,\eta}(K-1) + |\zeta|)N$.*

**Proof:** First assume $||\zeta|| < ||\eta||$. Let $\zeta \xrightarrow{w} \epsilon$ be a norm-reducing transition sequence. Then $\zeta\beta \xrightarrow{w} \beta, \eta\beta \xrightarrow{w} \gamma\beta$ and $\beta \sim \gamma\beta$ for some $\gamma$ such that $\eta \xrightarrow{w} \gamma$ in $|w| = ||\zeta||$ steps. Since any system of guarded equations has a unique solution up to bisimulation, we conclude that $\beta \sim X$ where $X =_{\mathrm{df}} \gamma X$, which completes the first part.

Now suppose $||\zeta|| = ||\eta||$. By Proposition 4.7, $\mathrm{Sep}_{\mathcal{C}'}(\zeta,\eta) = m$ for some $m \geq 1$. Thus we have the following situation: $\zeta = \zeta_m \not\sim_{m,\mathcal{C}'} \eta_m = \eta$. Let $\zeta_m \xrightarrow{a} \zeta_{m-1}$ wlog. be a separating transition. Since $\zeta_m\beta \xrightarrow{a} \zeta_{m-1}\beta$ we also have $\eta_m\beta \xrightarrow{a} \eta_{m-1}\beta$ for some $\eta_{m-1}$ such that $\zeta_{m-1}\beta \sim_{\mathcal{C}} \eta_{m-1}\beta$ and $\zeta_{m-1} \not\sim_{m-1,\mathcal{C}'} \eta_{m-1}$ due to the separating property of the transition. This construction can repeatedly be applied to obtain sequences $\zeta_m, \ldots, \zeta_1$ and $\eta_m, \ldots, \eta_1$ such that $\zeta_i \not\sim_{i,\mathcal{C}'} \eta_i$ and $\zeta_i\beta \sim_{\mathcal{C}} \eta_i\beta$ for all $1 \leq i \leq m$. An illustration is given in Figure 2. As the situation is symmetric in this case we may assume $||\zeta_1|| \leq ||\eta_1||$. In order to complete the proof we consider two cases:

**Case 1:** $||\eta_1|| < \infty$. This implies $||\zeta_1|| < \infty$, and since $\mathcal{C}$ and $\mathcal{C}'$ coincide on $V_N^* \times V_N^*$, we have also $\zeta_1 \not\sim_{1,\mathcal{C}} \eta_1$. As $\zeta_1\beta \sim_{\mathcal{C}} \eta_1\beta$, we obtain $\zeta_1 = \epsilon$. Hence $\gamma = \eta_1$ suits and we have $|w| = m - 1 = \mathrm{Sep}_{\mathcal{C}'}(\zeta,\eta) - 1 \leq \mathcal{B}_{\zeta,\eta} - 1$.

**Case 2:** $||\eta_1|| = \infty$. From $\zeta_1 \not\sim_{\mathcal{C}'} \eta_1$ we conclude by Proposition 4.7 that $\zeta_1 \not\sim_{\mathcal{C}} \eta_1$. Since $\zeta_1\beta \sim_{\mathcal{C}} \eta_1\beta$, we have $||\zeta_1|| < \infty$. Hence with $|w| = m - 1 + ||\zeta_i|| \leq \mathcal{B}_{\zeta,\eta} + (\mathcal{B}_{\zeta,\eta}(K-1) + |\zeta|)N$ there exists again some $\gamma$ such that $\eta = \eta_m \xrightarrow{w} \eta_1 = \gamma$ in $m - 1 = |w| \leq \mathcal{B}_{\zeta,\eta} + (\mathcal{B}_{\zeta,\eta}(K-1) + |\zeta|)N$ steps and $\beta \sim X$ where $X =_{\mathrm{df}} \eta_1 X$.

□

The first step of the initial base construction consists of a *completion* procedure for the given BPA system.

**Algorithm 4.9** *The* elementary completion $\Delta^\omega$ *of a set of equations $\Delta$ is defined as follows:*

- *Every equation of $\Delta$ is contained in $\Delta^\omega$.*
- *For each $i$ and $j$ in the range $1 \leq i \leq j \leq n$ such that $X_i, X_j \in V_N$ fix some norm-reducing transition sequence $X_i \xrightarrow{u} \epsilon$. Then for each pair $(X_i\gamma, X_j)$ such that $X_j \xrightarrow{u} \gamma$ for some $\gamma$ we do the following:*

$$
\begin{tikzcd}
\zeta_m \arrow[d, "a_{m-1}"] & \not\sim_{m,\mathcal{C}'} & \eta_m \arrow[d, "a_{m-1}"] \\
\zeta_{m-1} \arrow[d, "a_{m-2}"] & \not\sim_{m-1,\mathcal{C}'} & \eta_{m-1} \arrow[d, "a_{m-2}"] \\
\cdots \arrow[d, "a_1"] & & \cdots \arrow[d, "a_1"] \\
\zeta_1 & \not\sim_{1,\mathcal{C}'} & \eta_1
\end{tikzcd}
\qquad
\begin{tikzcd}
\zeta_m\beta \arrow[d, "a_{m-1}"] & \sim_{\mathcal{C}} & \eta_m\beta \arrow[d, "a_{m-1}"] \\
\zeta_{m-1}\beta \arrow[d, "a_{m-2}"] & \sim_{\mathcal{C}} & \eta_{m-1}\beta \arrow[d, "a_{m-2}"] \\
\cdots \arrow[d, "a_1"] & & \cdots \arrow[d, "a_1"] \\
\zeta_1\beta & \sim_{\mathcal{C}} & \eta_1\beta
\end{tikzcd}
$$

Figure 2: The construction of a separating transition track.

1. *If $||X_i\gamma|| < ||X_j||$ then fix some norm-reducing transition sequence $X_i\gamma \xrightarrow{w} \epsilon$ and add for each $\zeta$ such that $X_j \xrightarrow{w} \zeta$ the equations $Y =_{\mathrm{df}} \zeta Y$ and $Y' =_{\mathrm{df}} \gamma Y$ to $\Delta^\omega$ where $Y, Y'$ are fresh variables.*
2. *If $||X_i\gamma|| > ||X_j||$ then fix some norm-reducing transition sequence $X_j \xrightarrow{w} \epsilon$ and add for each $\zeta$ such that $X_i\gamma \xrightarrow{w} \zeta$ the equations $Y =_{\mathrm{df}} \zeta Y$ and $Y' =_{\mathrm{df}} \gamma Y$ to $\Delta^\omega$ where $Y, Y'$ are fresh variables.*
3. *If $||X_i\gamma|| = ||X_j||$ then add for each $\zeta$ such that*
   - *$X_i\gamma \xrightarrow{w} \epsilon, X_j \xrightarrow{w} \zeta$ in $|w| \leq B(X_i\gamma)$ steps, or*
   - *$X_j \xrightarrow{w} \epsilon, X_i\gamma \xrightarrow{w} \zeta$ in $|w| \leq B(X_j)$ steps*

   *the equations $Y =_{\mathrm{df}} \zeta Y$ and $Y' =_{\mathrm{df}} \gamma Y$ to $\Delta^\omega$ where $Y, Y'$ are fresh variables.*

*The equations generated from a pair $(X_i\gamma, X_j)$ are also called the* elementary completion *of $(X_i\gamma, X_j)$.*

During this completion we will add new equations to $\Delta$ in order to be able to explicitely denote some needed bisimulation classes. Note, however, that each equation added to $\Delta$ defines an unnormed variable which is bisimilar to $\gamma\zeta^\omega$ for some $\gamma, \zeta$, and that no new variable is reachable from any of the original variables. Thus the bisimulation classes generated by finite sequences over $V$ are unchanged after completion.

**Algorithm 4.10 (Initial base $B_0$ construction)**
*Let $\Delta^\omega$ be the elementary completion of $\Delta$ with variables $V_N$ and $V_U^\omega = V_U \cup V_U^{new}$.*

**Step 1:** *For each $i$ and $j$ in the range $1 \leq i \leq j \leq n$ such that $X_i, X_j \in V_N$ fix some $[X_j]_{||X_i||}$ such that $X_j \xrightarrow{w} [X_j]_{||X_i||}$ in $|w| = ||X_i||$ norm-reducing steps. Then $(X_i[X_j]_{||X_i||}, X_j) \in B_0$.*

**Step 2:** *For each $i$ and $j$ in the range $1 \leq i \leq j \leq n$ such that $X_i, X_j \in V_N$ let $u$ be the path labelling choosen during the elementary completion. Then for each $\gamma$ such that $X_j \xrightarrow{u} \gamma$ let $(X_iY', X_jY) \in B_0$, if $Y =_{\mathrm{df}} \zeta Y$ and $Y' =_{\mathrm{df}} \gamma Y$ are contained in the elementary completion of $(X_i\gamma, X_j)$.*

**Step 3:** *For each $i$ and $j$ in the range $1 \leq i \leq j \leq n$ such that $X_i, X_j \in V_U^\omega$ let $(X_i, X_j) \in B_0$.*

**Step 4:** *For each $X_i \in V_N$ and $X_j \in V_U^\omega$ let*

$$\{ (X_i[X_j]_{||X_i||}, X_j) \mid X_j \xrightarrow{w} [X_j]_{||X_i||} \textit{ in } |w| = ||X_i|| \textit{ steps} \} \subseteq B_0.$$

*Moreover, let $\Omega$ be the maximal seminorm of all $[X_j]_{||X_i||}$ obtained. Then add also $\{ (X_i\alpha, X_j) \mid ||\alpha||_s < \Omega \}$ to $B_0$.*

Bisimulation completeness follows by construction:

**Theorem 4.11** *The initial base $B_0$ is bisimulation-complete.*

**Proof:** Assume $X_i\alpha \sim X_j\beta$, and let us first consider the case when $(X_i\alpha, X_j\beta)$ is decomposable, i.e. $X_i\gamma \sim X_j$ and $\alpha \sim \gamma\beta$ for some $\gamma$. Let $w$ be the norm-reducing transition sequence of length $||X_i||$ chosen in step (1) of the initial base construction. Then due to $X_j \xrightarrow{w} [X_j]_{||X_i||}$ we have $X_i\gamma \xrightarrow{w} \gamma$ and $\gamma \sim [X_j]_{||X_i||}$. Thus the pair $(X_i[X_j]_{||X_i||}, X_j) \in \mathcal{B}_0$ satisfies condition (1) for bisimulation-completeness.

Now assume $(X_i\alpha, X_j\beta)$ is elementary.

**Case 1:** $X_i, X_j \in V_N$

Let $X_i \xrightarrow{u} \epsilon$ be the norm-reducing transition sequence chosen during the elementary completion. Then $X_i\alpha \xrightarrow{u} \alpha, X_j\beta \xrightarrow{u} \gamma\beta$ and $\alpha \sim \gamma\beta$ for some $\gamma$ such that $X_j \xrightarrow{u} \gamma$. Thus we also have $X_i\gamma\beta \sim X_i\alpha \sim X_j\beta$. Moreover, we know $X_i\gamma \not\sim X_j$ due to the undecomposability of the pair $(X_i\alpha, X_j\beta)$. Observing that $\min\{ ||X_i\gamma||, ||X_j|| \}$ is finite allows us to apply Lemma 4.8. We distinguish the following cases.

1. $||X_i\gamma|| < ||X_j||$
   For the norm-reducing transition sequence $X_i\gamma \xrightarrow{w} \epsilon$ fixed during elementary completion we have $X_j \xrightarrow{w} \zeta$ and $\beta \sim \zeta\beta$ for some $\zeta$. So there exist equations $Y =_{\mathrm{df}} \zeta Y,\ Y' =_{\mathrm{df}} \gamma Y \in \Delta^\omega$ with $Y \sim \beta$ and $Y' \sim \gamma Y \sim \gamma\beta \sim \alpha$. Due to step (2.) we have $(X_iY', X_jY) \in \mathcal{B}_0$ and obviously this pair is minimal wrt. $\sqsubseteq$.
2. $||X_i\gamma|| > ||X_j||$
   This case follows a similar way as the previous one.
3. $||X_i\gamma|| = ||X_j||$
   Then without loss of generality for a transition sequence $X_i\gamma \xrightarrow{w} \epsilon$ with $|w| \leq B(X_i\gamma)$ we have $X_j \xrightarrow{w} \zeta$ and $\beta \sim \zeta\beta$. Again the elementary completion ensures the existence of equations $Y =_{\mathrm{df}} \zeta Y,\ Y' =_{\mathrm{df}} \gamma Y \in \Delta^\omega$ with $Y \sim \beta$ and $Y' \sim \gamma Y \sim \gamma\beta \sim \alpha$. Due to step (2.) we have $(X_iY', X_jY) \in \mathcal{B}_0$ and obviously this pair is also minimal wrt. $\sqsubseteq$.

**Case 2:** $X_i, X_j \in V_U^\omega, \alpha = \beta = \epsilon$

Then by step (3.) $(X_i, X_j) \in \mathcal{B}_0$. Note that this pair is minimal wrt. $\sqsubseteq$.

**Case 3:** $X_i \in V_N, X_j \in V_U^\omega, \beta = \epsilon$

Let $X_i \stackrel{w}{\to} \epsilon$ be a norm-reducing transition sequence. Then $X_i\alpha \stackrel{w}{\to} \alpha, X_j \stackrel{w}{\to} \alpha'$ and $\alpha \sim \alpha'$ for some $\alpha'$. Due to step (4.) we have $(X_i\alpha', X_j) \in \mathcal{B}_0$. If there exists $\alpha''$ with $\alpha'' \sim \alpha'$ and $||\alpha''||_s < ||\alpha'||_s$ then the seminorm down closure ensures that $(X_i\alpha'', X_j)$ is also contained in $\mathcal{B}_0$.

$\square$

## 4.3 Summary of the Decision Procedure

The overall algorithm proceeds in three steps:

1. Computation of an intial base $B_0$ by means of the algorithm presented in the previous subsection. This algorithm terminates with a base of elementary size.
2. Refinement of $B_0$ by iterative application of R until a bisimulation base is reached, which is the case after at most $|B_0|$ iterations. As $B_0$ is of elementary size and each iteration itself is elementary also this step is elementary.
3. Decision of $\alpha \sim \beta$ by means of a straightforward (obviously elementary) branching algorithm.

Summarizing we can conclude:

**Theorem 4.12 (Main Theorem)**
*Our three step bisimulation decision algorithm is elementary.*

**Remark:** Our complexity analysis is quite rough. A more careful analysis would reveal quite a small 'exponentiality'. Indeed we conjecture that some straightforward optimizations will yield a doubly exponential algorithm.

# References

[BBK87] J.C.M. Baeten, J.A. Bergstra, and J.W. Klop. Decidability of Bisimulation Equivalence for Processes Generating Context-Free Languages. In *PARLE '87, LNCS 259*, pages 94–113. Springer, 1987. Full version appeared as: Technical Report CS-R8632, CWI, Sep. 1987.

[BW90] J.C.M. Baeten and W.P. Weijland. *Process Algebra*, volume 18 of *Cambridge Tracts in TCS*. Cambridge University Press, 1990.

[Cau89] D. Caucal. A Fast Algorithm to Decide on Simple Grammars Equivalence. In *Int. Symposium on Optimal Algorithms, LNCS 401*, pages 66–85. Springer, 1989.

[Cau90] D. Caucal. Graphes Canoniques de Graphes Algébriques. *RAIRO*, 24(4):339–352, 1990. A preliminary version appeared as: Rapport de Recherche 872, INRIA, July 1988.

[CHS92] S. Christensen, H. Hüttel, and C. Stirling. Bisimulation Equivalence is Decidable for all Context-Free Processes. In *CONCUR '92, LNCS 630*, pages 138–147. Springer, 1992.

[GH91] J.F. Groote and H. Hüttel. Undecidable Equivalences for Basic Process Algebra. Technical Report CS-R9137, CWI, 1991.

[Gla90] R.J. van Glabbeek. The Linear Time - Branching Time Spectrum. In *CONCUR 90, LNCS 458*, pages 278–297. Springer, 1990.

[Gro91] J.F. Groote. A Short Proof of the Decidability of Bisimulation for Normed BPA-Processes. Technical Report CS-R9151, CWI, Dec 1991.

[HJM94] Y. Hirshfeld, M. Jerrum, and F. Moller. A Polynomial Algorithm for Deciding Bisimilarity of Normed Context-Free Processes. Technical Report ECS-LFCS-94-286, LFCS, Edinburgh, March 1994. To be presented at FOCS '94.

[HM94] Y. Hirshfeld and F. Moller. A Fast Algorithm for Deciding Bisimilarity of Normed Context-Free Processes. In *CONCUR '94, LNCS 836*, pages 48–63, 1994.

[Hoa85] C.A.R. Hoare. *Communicating Sequential Processes*. Prentice–Hall, 1985.

[HS91] H. Hüttel and C. Stirling. Actions Speak Louder than Words: Proving Bisimilarity for Context-Free Processes. In *LICS '91*, pages 376–386. IEEE Computer Society Press, 1991.

[HT91] D.T. Huynh and L. Tian. Complexity of Deciding Readiness and Failure Equivalences for Processes. In *Proc. 3rd IEEE Symp. on Parallel and Distributed Processing*, pages 738–745, 1991.

[HT94] D.T. Huynh and L. Tian. Deciding Bisimilarity of Normed Context-Free Processes is in $\Sigma_2^p$. *Theoretical Computer Science*, 123:183–197, 1994.

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

[Par81] D. Park. Concurrency and Automata on Infinite Sequences. In *5th GI Conference, LNCS 104*, pages 167–183. Springer, 1981.