# FI MU

# Some Remarks on Weak Bisimilarity of BPA-Processes

by

Ivana Černá
Jitka Stříbrná

   

# Some Remarks on Weak Bisimilarity of BPA-Processes

Ivana Černá [*]        Jitka Stříbrná[†]

Faculty of Informatics, Masaryk University
Botanická 68a, 60200 Brno
Czech Republic

## 1   Introduction

The purpose of this work is to examine the decidability problem of weak bisimilarity for BPA-processes. It has been known that strong bisimilarity, which may be considered a special case of weak bisimilarity, where the internal (silent) action $\tau$ is treated equally to observable actions, is decidable for BPA-processes. For strong bisimilarity, these processes are finitely branching and so for two non-bisimilar processes there exists a level $n$ that distinguishes the two processes. Additionally, from the decidability of whether two processes are equivalent at a given level $n$, semidecidability of strong non-bisimilarity directly follows. We examine the following two closely related approaches to semidecidability of strong equivalence:

1. construction of a (finite) bisimulation or expansion tree,

2. construction of a finite Caucal base.

We have attempted to find out if any of the above mentioned approaches could be generalized to (semi)decide weak bisimilarity. Our findings indicate that a direct generalization is not sufficient and an efficient (semi)decision procedure cannot be obtained in this way.

The technique of bisimulation trees was proposed by Hirshfeld in [6]. In the most general concept, bisimulation trees contain all possible derivative pairs of some initial pair. Hence the trees are complete and correctness is obviously maintained, however it may not be feasible to search such trees. Also when we discuss feasibility, for weak bisimulation we need to consider separately semidecidability of bisimilarity and semidecidability of

non-bisimilarity. To be more precise, in case of strong bisimulation the latter is guaranteed by the finite-image property which for weak equivalence fails to hold. Therefore, in the following we will only consider semidecidability of weak bisimulation equivalence.

In order to reach algorithmic feasibility it appears necessary to introduce some modification into the construction of bisimulation trees. There are two kinds of rules summarized by Jančar and Moller in [8]: omission and replacement. We can omit a pair from a reached node if it is in some sense implied by already visited pairs. We can replace one pair by a set of pairs in a newly constructed sibling node if we do not introduce "false" bisimulation witnesses in this process. It now has to be proved that completeness and correctness are maintained which is done by introducing an inductive invariant. The method has to be further modified for weak bisimulation and (general) BPA. More precisely, the criteria for omission and replacement need to be modified to comply with properties of weak bisimulation.

**Remark:** Here we shall not employ the notion of Caucal base (i.e. a set of pairs that would generate the maximal bisimulation by congruence closure - for more detail consult [3, 4]), which is used to semidecide strong bisimulation by enumeration of finite sets for which the Caucal condition is tested. In this way, in the positive case a finite bisimulation (Caucal) base is eventually constructed. The notion of Caucal base can be modified into weak Caucal base which serves as generation base for the maximal weak bisimulation equivalence. However, we can construct a pair of two weakly bisimilar processes for which there does not exist a finite weak (Caucal) bisimulation base, which indicates that it cannot always be used efficiently for weak bisimilarity.

# 2   Background

## 2.1   Basic Process Algebras

In order to define Basic Process Algebras we presuppose a finite set of *actions $Act$* that contains a special action $\tau$, and a finite set of process variables or atoms $\Sigma$. A *Basic Process Algebra (BPA)* is then a pair $(\Sigma^*, \Delta)$, where $\Sigma^*$ is the free monoid generated by $\Sigma$, and $\Delta = \{X \xrightarrow{a} \alpha \mid X \in \Sigma, \alpha \in \Sigma^*, a \in Act\}$ is a finite set of transitions. BPA-processes are identified with words from $\Sigma^*$. The empty word $\epsilon$ denotes the *empty* process that cannot perform any action. The transition rules of $\Delta$ determine a transition relation on general BPA-processes in this way:

$$X\beta \xrightarrow{a} \alpha\beta \text{ if there is a rule } X \xrightarrow{a} \alpha \text{ in } \Delta.$$

We will use capital letters $X, Y$ to range over process variables, $\alpha, \beta, \gamma, \delta$ to range over BPA-processes and $a, b, c$ to range over process actions.

If $\alpha$ is a process then the *norm* of $\alpha$, denoted by $|\alpha|$, is the minimum of lengths of derivation sequences leading from $\alpha$ to the empty process $\epsilon$. We say that a process is *normed* if it has a finite norm, otherwise it is *unnormed*. We also call this notion *strong norm* to distinguish it from *weak norm* which does not count the $\tau$-moves on the way to $\epsilon$, and is denoted by $\|\alpha\|$. When weak norm is considered, a process is called *normed* if it has a finite norm, *totally normed* if the norm is finite and positive, and *unnormed* otherwise.

## 2.2   Bisimulation equivalences

Each process $\alpha$ of a BPA $(\Sigma^*, \Delta)$ generates a labeled transition system (LTS) with $\alpha$ labeling the root, processes derivable from $\alpha$ labeling the nodes and the action leading from $\alpha$ to $\alpha'$ labeling the arc that leads from $\alpha$ to $\alpha'$.

**Example 1** *Here we present a simple BPA. The set of variables $\Sigma$ is $\{X, Y\}$ and the transition rules of $\Delta$ are given below:*

$$Y \xrightarrow{\ a\ } \epsilon \qquad X \xrightarrow{\ \tau\ } XY \qquad X \xrightarrow{\ b\ } \epsilon.$$

*The labeled transition system determined by the variable $X$ is sketched in figure 1.*
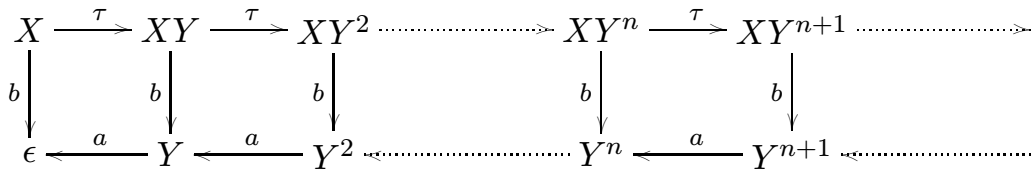


Figure 1: LTS determined by the process $X$

*The process $X$ can with a sequence of $n$ transitions $\xrightarrow{\tau}$ generate $n$ copies of $Y$ thus becoming $XY^n$. For $XY^n$ to perform any move of $Y$ the process has to dispose of the $X$ in front with an $X \xrightarrow{\ b\ } \epsilon$ move. Only then can an $a$ action of $Y$ be done as it is always the leftmost variable in a BPA-process that is allowed to carry out a transition.*   □

If two processes give rise to labeled transition systems that are isomorphic up to different names at the nodes then the processes are considered identical. Usually we want to identify a broader class of processes, namely processes which exhibit the same observable behavior. We will investigate two of the major equivalences: strong and weak bisimulations. Following Milner ([10]), we define the notion of a bisimulation as a binary relation on processes.

**Strong bisimulation**

**Definition 2** *Let* $(\Sigma^*, \Delta)$ *be a BPA. A binary relation* $\mathcal{R}$ *over* $\Sigma^*$ *is a* strong bisimulation *if for every pair* $(\alpha, \beta)$ *from* $\mathcal{R}$ *and and every action* $a$ *from* $Act$ *the following holds:*

- *for every* $\alpha \xrightarrow{a} \alpha'$ *there exists* $\beta \xrightarrow{a} \beta'$ *so that* $(\alpha', \beta') \in \mathcal{R}$;

- *for every* $\beta \xrightarrow{a} \beta'$ *there exists* $\alpha \xrightarrow{a} \alpha'$ *so that* $(\alpha', \beta') \in \mathcal{R}$.

Processes $\alpha$ and $\beta$ are strongly bisimilar, written $\alpha \sim \beta$, if they are related by some strong bisimulation.

It was shown in [10] that the union of all strong bisimulations is also a strong bisimulation. It is the largest strong bisimulation, denoted by $\sim$, and it is an equivalence relation. We will also call it *strong bisimulation equivalence*. Moreover, strong bisimulation is a congruence on every BPA, i.e. if $\alpha \sim \beta$ and $\gamma \sim \delta$ then $\alpha\gamma \sim \beta\delta$.

**Weak bisimulation**

The notion of strong bisimilarity requires a process to be capable of matching each transition that an equivalent process may perform. However, sometimes we want to distinguish between *observable (external)* and *internal* behavior of processes and we wish to regard two processes equivalent if they exhibit the same observable behavior, irrespective of any intermediate internal behavior that may occur. To this end we employ a special *silent* action $\tau$ which represents internal behavior.

In order to incorporate the notion of internal behavior we consider composite actions $\xrightarrow{a}$, where $\xrightarrow{a}$ is an abbreviation of $(\xrightarrow{\tau})^* \xrightarrow{a} (\xrightarrow{\tau})^*$ in case $a \neq \tau$, and $(\xrightarrow{\tau})^*$ in case $a = \tau$. The process $X$ from Example 1 gives rise to an infinitely branching tree that is shown in Figure 2.
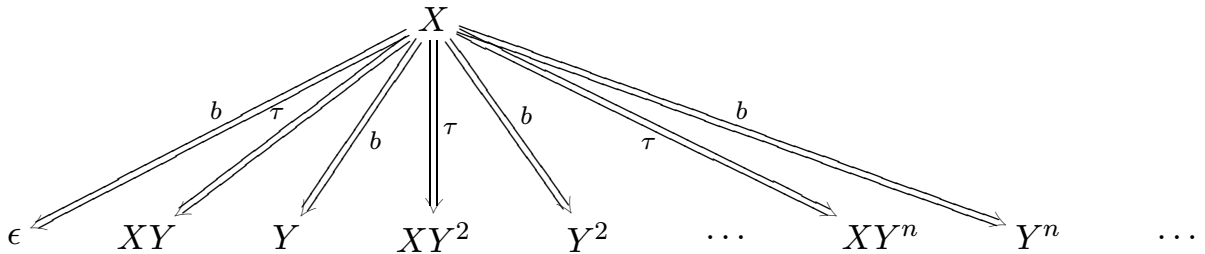


Figure 2: X as an infinitely branching tree

**Definition 3** *Let* $(\Sigma^*, \Delta)$ *be a BPA. A binary relation* $\mathcal{R}$ *over* $\Sigma^*$ *is a* weak bisimulation *if for every pair* $(\alpha, \beta)$ *from* $\mathcal{R}$ *and and every action* $a$ *from* $Act$ *the following holds:*

- *for every* $\alpha \xrightarrow{a} \alpha'$ *there exists* $\beta \xRightarrow{a} \beta'$ *so that* $(\alpha', \beta') \in \mathcal{R}$;

- *for every* $\beta \xrightarrow{a} \beta'$ *there exists* $\alpha \xRightarrow{a} \alpha'$ *so that* $(\alpha', \beta') \in \mathcal{R}$.

Processes $\alpha$ and $\beta$ are weakly bisimilar, written $\alpha \approx \beta$, if they are related by some weak bisimulation. The union of all weak bisimulations gives rise to the maximal weak bisimulation which is denoted by $\approx$. An equivalent definition of weak bisimulation is phrased in terms of composed transition in the premise followed by composite transition. Both definitions yield identical maximal weak bisimulations (see [10]).

The requirement on matching transitions is relaxed in comparison with strong bisimulation and so this definition yields a weaker notion of bisimulation. It is easy to convince oneself that for all pairs of processes $\alpha$ and $\beta$, if $\alpha \sim \beta$ then also $\alpha \approx \beta$. However, the converse does not always hold.

As opposed to strong bisimulation, a maximal weak bisimulation relation is not always necessarily a congruence — see [13] for counterexample. In order to ensure that this desirable property holds it is enough to require for a BPA $(\Sigma^*, \Delta)$ that for all variables $X \in \Sigma$, if $X \approx \epsilon$ then $X \xRightarrow{\tau} \epsilon$ (see [13]). Another trivial assumption is formulated in [7]. Here, in order to obtain congruence and simplify proofs, we will make a slightly stronger assumption throughout the rest of the paper that $P \approx \epsilon$ implies $P \equiv \epsilon$, i.e. the *only* process with no observable behavior is the empty process.

### Bisimulation game

A useful way to understand both strong and weak bisimulation relation is to consider it as a *bisimulation game* between two players Alice and Bob (for detailed description see i.e. [8]). For a given LTS and its two vertices $\alpha_0$ and $\beta_0$, the two players try to achieve opposite goals: Alice wants to show that $\alpha_0$ and $\beta_0$ are different while Bob tries to show their sameness. A *play* of the game is a sequence of pairs $(\alpha_0, \beta_0), (\alpha_1, \beta_1), \ldots$, where each consecutive pair arises in this way: Alice chooses an action $a$ and a transition $\alpha_i \xrightarrow{a} \alpha_{i+1}$, resp. $\beta_i \xrightarrow{a} \beta_{i+1}$. Bob then needs to produce a matching reply $\beta_i \xrightarrow{a} \beta_{i+1}$, resp. $\alpha_i \xrightarrow{a} \alpha_{i+1}$ (in the case of weak bisimulation composite actions need to be considered). Alice wins the play if Bob cannot respond to a move by Alice, otherwise the winner is Bob. Processes $\alpha_0$ and $\beta_0$ are bisimilar iff Bob is able to win every play of the game regardless of the moves made by Alice.

## 2.3 Approximation of bisimulations

In the previous sections, the strong, resp. weak, maximal bisimulations were obtained as the union of smaller strong, resp. weak, bisimulation relations. There exists an alternative approach (see [10]) where the maximal

equivalences are obtained as the limits of respective non-increasing chains of *bisimulation approximants.* These are binary relations on processes defined inductively in terms of *ordinal numbers.*

For the sake of simplicity, in this paper we shall view ordinals as generalization of (the well-ordered nature of) natural numbers. Ordinal numbers form a class, denoted by $On$, and are well-ordered by the element-of relation $<$. The initial segment of $On$ containing natural numbers and $\omega$ is $0, 1, 2, \ldots, n, \ldots, \omega$, after which follow $\omega + 1, \omega + 2, \ldots, \omega + n, \ldots, \omega + \omega$, etc. The ordinals $0, \omega, \omega + \omega$, are *limit* ordinals which means they have no predecessor, whereas ordinals such as $1, 2$, and $\omega + 1, \omega + 2$, are *successor* ordinals. We will be using the principle of *ordinal* or *transfinite induction*, the induction principle generalized to the class of all ordinal numbers. For a detailed instruction on ordinal numbers the reader should consult standard textbooks on set theory, such as [9].

Now we can define *weak bisimulation approximants* $\approx_\kappa$ for a fixed BPA $(\Sigma^*, \Delta)$ inductively on the class $On$:

- $\alpha \approx_0 \beta$ for all $\alpha$ and $\beta$ from $\Sigma^*$;

- $\alpha \approx_{\kappa+1} \beta$ if for all actions $a$,

    - whenever $\alpha \overset{a}{\Longrightarrow} \alpha'$ then there exists $\beta \overset{a}{\Longrightarrow} \beta'$ so that $\alpha' \approx_\kappa \beta'$;
    - whenever $\beta \overset{a}{\Longrightarrow} \beta'$ then there exists $\alpha \overset{a}{\Longrightarrow} \alpha'$ so that $\alpha' \approx_\kappa \beta'$;

- $\alpha \approx_\lambda \beta$ if $\alpha \approx_\kappa \beta$ for every $\kappa < \lambda$, for a limit ordinal $\lambda$.

*Strong bisimulation approximants* $\sim_\kappa$ are defined analogously, with composite transition $\overset{a}{\Longrightarrow}$ being replaced by single transition $\overset{a}{\longrightarrow}$, in both premise and conclusion.

It can be easily verified that binary relations $\approx_\alpha$ are equivalences for every ordinal $\alpha$. The following lemma sums up the structure of the chain of approximants and the relationship between individual approximants and the maximal bisimulation. A proof can be found in [10], [13].

**Lemma 4**

1. *for every $\kappa, \mu \in On$, $\kappa < \mu \implies \approx_\mu \subseteq \approx_\kappa$;*

2. *for every $\kappa \in On$, $\approx \subseteq \approx_\kappa$;*

3. *if there is an $\kappa$ such that $\approx_\kappa = \approx_{\kappa+1}$ then for all $\mu \geq \kappa$, $\approx_\kappa = \approx_\mu = \approx$;*

4. *$\approx = \bigcap_{\kappa \in On} \approx_\kappa$.*

1 says that approximants form a non-increasing sequence. 2 says that the maximal equivalence is contained in every approximant. 3 and 4 state that the sequence converges and the limit is $\approx$.

An analogous lemma holds also for strong bisimulation approximants, i.e. the sequence of strong bisimulation approximants converges with the limit being $\sim$. For BPA-processes, owing to their finite-branching structure, the convergence occurs at level $\omega$, that is $\sim \; = \; \sim_\omega \; = \; \bigcap_{n \in \omega} \; \sim_n$. Proof of this claim can be found in [5]. Additionally, this finite-branching property guarantees that each approximant $\sim_n$ is decidable. Therefore we obtain a straightforward semidecision procedure for non-bisimilarity by successive enumeration of all natural numbers $n$ and testing equivalence at $\sim_n$. However, this approach cannot be used for weak bisimulation approximants because infinite branching of BPA w.r.t. weak bisimilarity produces algebras where $\approx \; \subsetneq \; \approx_\omega$ (consult e.g. [13], [14] for more details).

When we deal with the whole class of ordinals the common induction principle for natural numbers becomes too weak for proving theorems. We need a more powerful proof method than that and, fortunately, the well-ordered structure of ordinal numbers enables us to formulate a statement which is a generalization of the induction principle. It is called *transfinite* or *ordinal induction*.

**The Principle of Transfinite Induction:** *Let $P(\kappa)$ be a statement for each ordinal $\kappa$. Assume that*

    *1. $P(0)$;*

    *2. $P(\kappa) \Rightarrow P(\kappa + 1)$ for every $\kappa$;*

    *3. if $\lambda$ is a limit ordinal then $(\forall \kappa < \lambda.\ P(\kappa)) \Rightarrow P(\lambda)$.*

*Then for every $\kappa \in On$, $P(\kappa)$.*

Now if we want to verify that some property $P$ holds for the class *On* we only have to test three cases: the base case $P(0)$, the successor case $P(\kappa) \Rightarrow P(\kappa + 1)$ and the limit case $(\forall \kappa < \lambda.\ P(\kappa)) \Rightarrow P(\lambda)$. If we manage to prove all three cases we can be confident that all ordinals possess the desired property $P$.

## 3   Decompositions

All known algorithms for deciding bisimilarity between two BPA-processes ([4], [7]) are strongly dependent on the notion of decomposability. Decomposition allows to transform the task of deciding bisimilarity between

given pairs of processes to a (finite) number of tasks of deciding bisimilarity between smaller pairs (for some suitably formulated criterion of process size).

Let $\alpha_1\alpha_2$ and $\beta_1\beta_2$ be two BPA-processes. Let us consider one particular bisimulation play, i.e. a sequence of pairs starting with $(\alpha_1\alpha_2, \beta_1\beta_2)$. This sequence can be divided into two subsequences: the first one beginning with $(\alpha_1\alpha_2, \beta_1\beta_2)$ and the second one with the (uniquely determined) pair $(\gamma\alpha_2, \delta\beta_2)$, such that in the next step of the play an action is emitted for the first time from $\alpha_2$ or from $\beta_2$. Both subsequences may be empty, finite or infinite.

The strategy for deciding bisimilarity between $\alpha_1\alpha_2$ and $\beta_1\beta_2$ based on this concept is the following. Let $A$ be a suitably chosen set of pairs of processes. Then $\alpha_1\alpha_2 \sim \beta_1\beta_2$ if

1. for every bisimulation play starting from $\alpha_1\alpha_2$ and $\beta_1\beta_2$,
   *either* Bob has a winning strategy leading to his victory without emitting any action neither from $\alpha_2$ nor from $\beta_2$,
   *or* the first pair such that in the next step an action is emitted from $\alpha_2$ or $\beta_2$ has the form $(\gamma\alpha_2, \delta\beta_2)$, where $(\gamma, \delta) \in A$, and

2. $\gamma\alpha_2 \sim \delta\beta_2$ for every $(\gamma, \delta) \in A$.

The procedure sketched above can be recursively applied to newly created pairs and is efficient under the assumption that the new pairs are "simpler". In order to implement this procedure we need a generalized notion of bisimulation relation that takes into account the sets of termination pairs that occur within a bisimulation play when the first halves of the original pair are removed. That gives rise to the notion of *bisimulation up to*, originally proposed by Hirshfeld in [7].

## 3.1   Bisimulation up to

**Definition 5** *Given an arbitrary set of pairs $A$, we say that a binary relation $R$ is a* weak bisimulation up to $A$ *if every pair from $R$ satisfies the following conditions:*

- *either $(\alpha, \beta) \in A$,*

- *or for every action $a$, if $\alpha \xLongrightarrow{a} \alpha'$ then there exists $\beta \xLongrightarrow{a} \beta'$ with $(\alpha', \beta') \in R$, and symmetrically.*

  *Furthermore we require that if $\alpha \equiv \epsilon$ or $\beta \equiv \epsilon$, then $(\alpha, \beta) \in A$.*

The processes $\alpha$ and $\beta$ are *weakly bisimilar up to $A$*, denoted by $\alpha \approx_A \beta$, if there exists a weak bisimulation up to $A$ that contains them. The union of all weak bisimulations up to $A$ is a maximal weak bisimulation up to $A$,

denoted also $\approx_A$. The last condition that we place on a pair $(\alpha, \beta)$ from a bisimulation up to $A$, is there to guarantee correctness of the decomposition approach described in the previous section.

The relationship between "classical" bisimulation and bisimulation up to can be characterized in this way: $\approx\ =\ \approx_A$ if and only if, for every pair $(\gamma, \delta) \in A$, $\gamma \approx \epsilon$ and $\delta \approx \epsilon$.

We can follow the alternative approach towards obtaining the maximal weak bisimulation that was sketched in Section 2.3 and define weak bisimulation approximants up to in an analogous way.

**Definition 6** *For a BPA $(\Sigma^*, \Delta)$, and a set up to $A$, weak bisimulation approximants up to $A$ are binary relations denoted by $\approx_{\kappa,A}$, defined by*

- *$\alpha \approx_{0,A} \beta$ for all $\alpha$ and $\beta \in \Sigma^*$,*

- *$\alpha \approx_{\kappa+1,A} \beta$ if $(\alpha, \beta) \in A$ or, for all actions $a$,*

  - *whenever $\alpha \stackrel{a}{\Longrightarrow} \alpha'$ then there exists $\beta \stackrel{a}{\Longrightarrow} \beta'$ so that $\alpha' \approx_{\kappa,A} \beta'$, and*
  - *whenever $\beta \stackrel{a}{\Longrightarrow} \beta'$ then there exists $\alpha \stackrel{a}{\Longrightarrow} \alpha'$ so that $\alpha' \approx_{\kappa,A} \beta'$;*

- *$\alpha \approx_{\lambda,A} \beta$ if $\alpha \approx_{\kappa,A} \beta$ for every $\kappa < \lambda$, for a limit ordinal $\lambda$.*

*Furthermore we require that if $\alpha \equiv \epsilon$ or $\beta \equiv \epsilon$ then $(\alpha, \beta) \in A$.*

For any set up to $A$, the respective approximants form a non-increasing chain that approximates the maximal bisimulation up to $A$ from above. The correctness of the two statements can be easily verified. Regarding the former, for every pair $(\alpha, \beta)$ from $\approx_{\kappa+1,A}$ there are two possibilities: either $(\alpha, \beta)$ belongs to the set $A$ in which case it is also included in $\approx_{\kappa,A}$ by definition, or there must exist pairs of matching derivatives $(\alpha', \beta')$ that appear in $\approx_{\kappa,A}$ and, by inductive reasoning, in all the approximants below. From this we can conclude that $(\alpha, \beta)$ must belong to $\approx_{\kappa,A}$ as well. The approximant labeled by 0 contains all pairs therefore this sequence indeed forms a non-increasing chain. The correctness of the latter claim is expressed by the proposition below:

**Proposition 7** $\approx_A\ =\ \bigcap_{\kappa \in \omega_1} \approx_{\kappa,A}$.

*Proof:* The first direction consists in proving that for any two BPA $\alpha$ and $\beta$, if $\alpha \approx_A \beta$ then for every ordinal $\kappa$, $\alpha \approx_{\kappa,A} \beta$. This needs to be done by transfinite induction on $\kappa$.

1. $\alpha \approx_{0,A} \beta$ is trivially true from the definition of approximants.

2. $\alpha \approx_{\kappa+1,A} \beta$ has to be proved from the premises that $\alpha \approx_A \beta$, and for any pair $(\alpha', \beta')$, $\alpha' \approx_A \beta'$ implies that $\alpha' \approx_{\kappa,A} \beta'$. In the case that $(\alpha, \beta) \in A$ we are done as then, by definition, $\alpha \approx_{\kappa+1,A} \beta$. We assume the contrary and consider any transition $\alpha \overset{a}{\Longrightarrow} \alpha'$. As $\alpha \approx_A \beta$, and $(\alpha, \beta) \notin A$, there exists a matching move $\beta \overset{a}{\Longrightarrow} \beta'$ such that $\alpha' \approx_A \beta'$. By the other assumption, $\alpha' \approx_{\kappa,A} \beta'$ and therefore we may conclude that $\alpha \approx_{\kappa+1,A} \beta$.

3. $\alpha \approx_{\lambda,A} \beta$, for a limit ordinal $\lambda$, is a straightforward consequence of the induction hypothesis that $\alpha \approx_{\kappa,A} \beta$ for every $\kappa < \lambda$.

The other direction falls into two cases. Firstly we need to realize that the chain will converge before reaching $\approx_{\omega_1,A}$, for the simple reason that we deal with countable algebras. Convergence will occur when we reach a level such that $\approx_{\kappa,A} = \approx_{\kappa+1,A}$, in which case $\approx_A = \approx_{\kappa,A} = \approx_{\mu,A}$, for every $\kappa < \mu$. Hence we assume that for some pair $(\alpha, \beta)$, $\alpha \approx_{\kappa,A} \beta$ for every $\kappa \in \omega_1$, and we will show that $\alpha \approx_A \beta$. In case $(\alpha, \beta) \in A$ we are done as then, by definition, $\alpha \approx_A \beta$. We assume that $(\alpha, \beta) \notin A$, and consider any move $\alpha \overset{a}{\Longrightarrow} \alpha'$. For every $\kappa \in \omega_1$ there exists a matching transition $\beta \overset{a}{\Longrightarrow} \beta'_\kappa$ with $\alpha' \approx_{\kappa,A} \beta'_\kappa$. However, as $\beta$ is a process in a countable algebra, there may be only countably many distinct derivatives $\beta'_\kappa$, and hence one $\beta'$ must occur among these derivatives uncountably often. Since approximants form a non-increasing chain, this $\beta'$ then satisfies the condition that $\alpha' \approx_{\kappa,A} \beta'$ for every $\kappa \in \omega_1$. So we can conclude that $\approx_{\omega_1,A}$ is in fact closed under expansion and hence included in $\approx_A$. $\square$

## 3.2 Properties of bisimulation up to

The largest strong (weak) bisimulation is (under the described assumptions) an equivalence relation and both $\sim$ and $\approx$ are congruences. This is the key property which allowed to build known algorithms for deciding bisimilarity as recursive algorithms. Unfortunately bisimulation up to is no longer an equivalence relation. Namely it is the property of transitivity that fails. Nevertheless some special form of transitivity and composition holds even in this case.

**Lemma 8 (Transitivity)** *If $\alpha \approx_A \beta$ and $\beta \approx \beta'$, then there exists a set $A'$ such that $\alpha \approx_{A'} \beta'$, and all pairs in $A$ and $A'$ are mutually bisimilar, i.e. for every $(\gamma, \delta) \in A$ there exists $(\gamma', \delta') \in A'$ with $\gamma \approx \gamma'$ and $\delta \approx \delta'$, and symmetrically for $A'$.*

**Proof:** From any $R$, weak bisimulation up to $A$, relating $\alpha$ and $\beta$, and any weak bisimulation $S$ relating $\beta$ and $\beta'$, we will construct a set up to $A'$ and
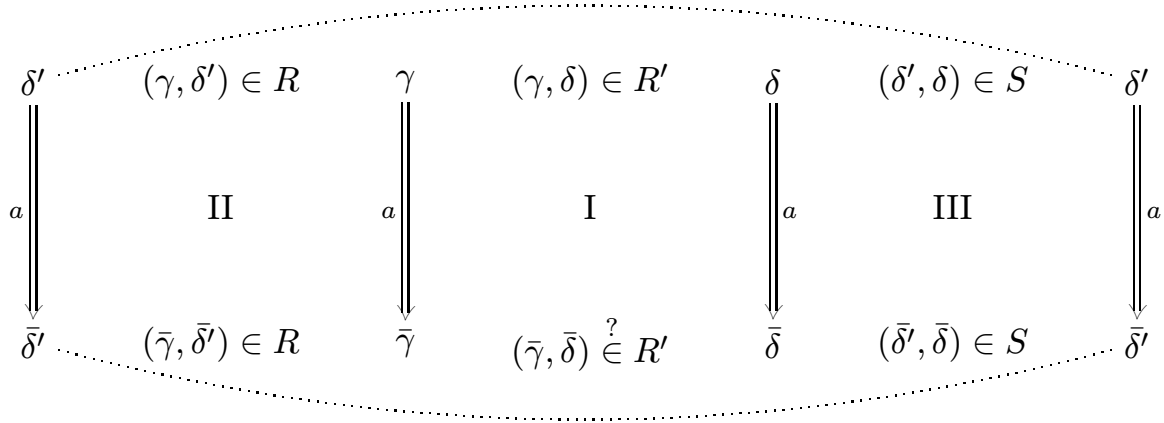
$R'$, a weak bisimulation up to $A'$, that will contain the pair $(\alpha, \beta')$. Additionally, the two sets $A$ and $A'$ will consist of mutually bisimilar couples, as described in the statement of the lemma. The two relations $R'$ and $A'$ are defined as follows:

$$A' = \{(\gamma, \delta) \mid \exists (\gamma, \delta') \in A \ \wedge \ (\delta', \delta) \in S\}$$

$$R' = \{(\gamma, \delta) \mid \exists (\gamma, \delta') \in R \ \wedge \ (\delta', \delta) \in S\}$$

First we shall verify that $A'$ satisfies the required conditions. Any pair $(\gamma, \delta)$ from $A'$ has its pre-image in some pair $(\gamma, \delta')$ from $A$, where $(\delta', \delta) \in S$. Since $S$ is a bisimulation relation, every pair contained within it must be weakly bisimilar, therefore $\delta' \approx \delta$. Obviously, $\gamma \approx \gamma$ and so we can conclude that every pair from $A'$ has a bisimilar pre-image in $A$. Obviously, the other implication is also true.

It remains to check that $R'$ is indeed a weak bisimulation up to $A'$. We shall express the expansion condition by means of the diagram below:



The starting point is the pair $(\gamma, \delta)$ from $R'$ at the top of square I, for which there must exist some $(\gamma, \delta')$ in $R$ (top of square II) satisfying $(\delta', \delta) \in S$ (top of square III). Either $(\gamma, \delta)$ is contained in $A'$ or we need to verify the expansion condition for the pair. We will assume the latter, i.e. $(\gamma, \delta) \notin A'$, from which also follows that $(\gamma, \delta')$ does not belong to $A$, and we will check the transitions.

If $\gamma$ does an $\overset{a}{\Longrightarrow}$ and evolves into $\bar{\gamma}$, then in diagram II we have a matching move from $\delta'$ into $\bar{\delta}'$ where $(\bar{\gamma}, \bar{\delta}') \in R$. The $\overset{a}{\Longrightarrow}$ transition of $\delta'$ (in diagram III) evokes a matching transition of $\delta$ owing to $\delta'$ and $\delta$ being in

$S$, with the resulting pair $(\bar{\delta}', \bar{\delta})$ also in $S$. Therefore, the pair of matching derivatives $(\bar{\gamma}, \bar{\delta})$ belongs to $R'$.

If $\delta$ does $\stackrel{a}{\Longrightarrow}$ into some $\bar{\delta}$ then, in diagram III, there must be a matching transition of $\delta'$ resulting in some $\bar{\delta}'$, where $(\bar{\delta}, \bar{\delta}') \in S$. The transition $\delta' \stackrel{a}{\Longrightarrow} \bar{\delta}'$ also appears as the left-most transition in diagram II, where from the assumption that $(\gamma, \delta') \notin A$ follows that $\gamma$ has a matching transition into some $\bar{\gamma}$. The pair $(\bar{\gamma}, \bar{\delta}')$ is in $R$ and $(\bar{\delta}', \bar{\delta})$ belongs to $S$ hence, from the definition of $R'$, we can conclude that $(\bar{\gamma}, \bar{\delta})$ is included in $R'$. We have verified that the expansion condition holds and therefore $R'$ is a bisimulation up to $A'$.

Lastly, we need to verify that if $\gamma \equiv \epsilon$ or $\delta \equiv \epsilon$ then $(\gamma, \delta) \in A'$, which readily follows from the assumptions that we have made.     □

Unfortunately, it seems impossible to say anything more precise about the exact correspondence of cardinalities of $A$ and $A'$, because the size of the (minimal w.r.t. inclusion) set up to depends on the size of branching of the two processes that we want to relate.

We need to be able to compose pairs of processes in a more general way:

**Lemma 9 (Composition)** *Whenever $\alpha_1 \approx_B \beta_1$ and $\gamma\alpha_2 \approx_A \delta\beta_2$, for every $(\gamma, \delta) \in B$, then $\alpha_1\alpha_2 \approx_A \beta_1\beta_2$.*

**Proof:** From the assumption that $\alpha_1 \approx_B \beta_1$ we can assume the existence of a bisimulation relation $R$ up to $B$, and a set of bisimulation relations up to $A$ for every pair $(\gamma, \delta)$ from $B$, that we denote $R_{(\gamma,\delta)}$. We define a relation $S = \{(\gamma\alpha_2, \delta\beta_2) \mid (\gamma, \delta) \in R\} \cup \bigcup_{(\gamma,\delta) \in B} R_{(\gamma,\delta)}$, and verify that this relation is a bisimulation up to $A$. We need to check only those pairs $(\gamma\alpha_2, \delta\beta_2)$, where $(\gamma, \delta) \in R$.

If $(\gamma, \delta)$ belongs directly to $B$ then from our assumptions, $(\gamma\alpha_2, \delta\beta_2)$ belongs to $R_{(\gamma,\delta)}$ and so we are done. In the other case we need to verify the expansion condition for $(\gamma\alpha_2, \delta\beta_2)$ w.r.t. $A$. A schema of the proof is drawn in Figure 3. An initial move $\gamma\alpha_2 \stackrel{a}{\Longrightarrow}$ may lead to some $\gamma'\alpha_2$ (diagram I) or it may dispose of $\gamma$ and end up in some $\alpha_2'$ (diagram II). In the first case we have a matching move $\delta\beta_2 \stackrel{a}{\Longrightarrow} \delta'\beta_2$ which belongs to $S$ by definition.

In the latter case, if $\gamma$ reduces to $\epsilon$, the process $\delta$ will evolve into $\delta'$ such that $(\epsilon, \delta') \in B$. Then we can use the assumption that $\epsilon\alpha_2 \approx_A \delta'\beta_2$ and hence, to the transition $\alpha_2 \stackrel{\tau}{\Longrightarrow} \alpha_2'$ there must be an equivalent move $\beta_2 \stackrel{\tau}{\Longrightarrow} \beta_2'$ leading to $(\alpha_2', \beta_2') \in R_{(\epsilon,\delta')}$.

Initial moves of $\delta$ and the combination $\gamma\alpha_2 \stackrel{\tau}{\Longrightarrow} \epsilon\alpha_2 \stackrel{a}{\Longrightarrow} \alpha_2'$ would be solved analogously. The $\epsilon$-condition on $A$ is also easy to verify.     □

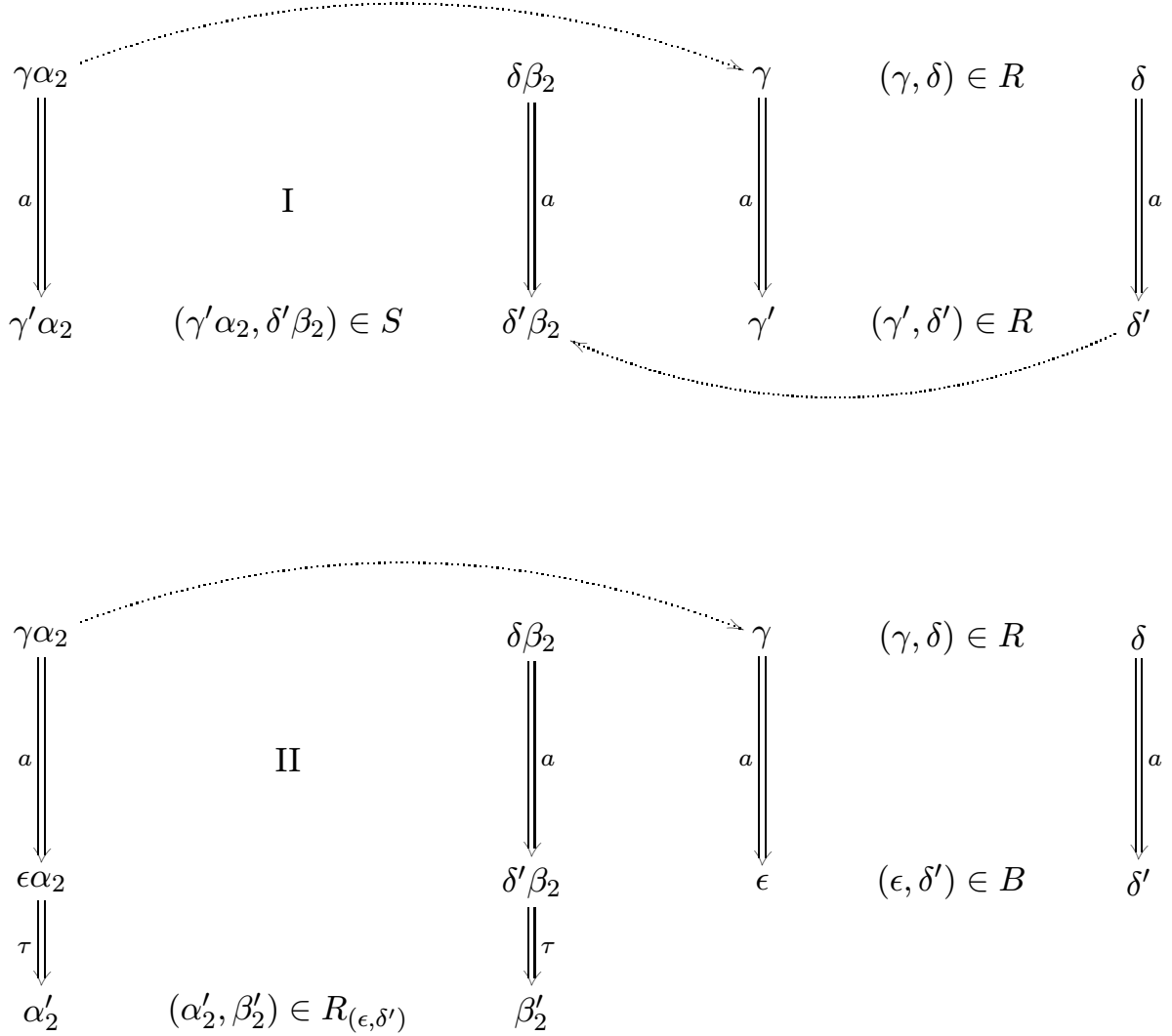Now we are ready to define the notion of decomposability we were seeking.

Figure 3: Case analysis

**Definition 10** *Let $\alpha, \beta$ be two processes bisimilar up to $A$. We say that processes $\alpha_1, \alpha_2, \beta_1, \beta_2$ and a set $B$ form a decomposition of $(\alpha, \beta)A$ up to $B$ if*

1. *$\alpha \equiv \alpha_1\alpha_2$ and $\beta \equiv \beta_1\beta_2$,*

2. *$\alpha_1 \approx_B \beta_1$,*

3. *$\gamma\alpha_2 \approx_A \delta\beta_2$, for every $(\gamma, \delta) \in B$.*

Intuitively, if we play a bisimulation game for any pair of bisimilar processes, we can always split the original processes into two pairs that will be "almost" bisimilar, up to some termination conditions. That is expressed in the following:

**Fact 11** *Every pair $(\alpha, \beta)$ bisimilar up to $A$ has some decomposition.*

# 4 Expansion trees

The notion of expansion tree is due to Hirshfeld [6] who put forward this idea in order to construct (semi)decision procedure for strong bisimulation on BPP and BPA-processes. The idea was then developed further, namely by Jančar and Moller in [8], and Hirshfeld in [7] . We first summarize this method for strong bisimulation on BPA processes.

**Definition 12** *Let $V \neq \emptyset$ and $U$ be two sets of pairs $(\alpha, \beta)$. $U$ is called a* strong expansion *of $V$ if it is a minimal set (w.r.t. inclusion) satisfying the following property: for every pair $(\alpha, \beta) \in V$, for every action $a$,*

- *if $\alpha \xrightarrow{a} \alpha'$ then $\beta \xrightarrow{a} \beta'$ with $(\alpha', \beta') \in U$;*

- *if $\beta \xrightarrow{a} \beta'$ then $\alpha \xrightarrow{a} \alpha'$ with $(\alpha', \beta') \in U$.*

A binary relation $R$ is a strong bisimulation iff it is a strong expansion of itself. A nonempty set $V$ does not have any expansion if it contains a pair $(\alpha, \beta)$ such that $\alpha \not\sim_1 \beta$, that is either $\alpha$ is able to emit an action $\beta$ is not able to emit or vice versa. The set $V = \{(\epsilon, \epsilon)\}$ has an empty expansion $\emptyset$ as neither of processes is able to emit any action. Moreover, for finitely branching processes, if a set $V$ is finite then every strong expansion of $V$ is finite and the number of different expansions of $V$ is finite, too.

The above mentioned properties give a hint how to decide bisimilarity: starting with a singleton containing the given pair expand it until a set which is an expansion of itself is achieved. This process is embodied into an expansion tree.

An *expansion tree* is a (generally infinite) tree whose nodes are labeled by sets of pairs of vertices, in which the children of a node are precisely the expansions of that node. A leaf of a tree is *successful* if it is empty; other leaves are *unsuccessful*. A branch is *successful* if it is infinite or finishes with a successful node. We may observe that the union of all nodes along a successful branch forms a bisimulation. The correctness of the expansion tree construction is spelled out in the following lemma:

**Lemma 13** $\alpha \sim \beta$ *iff the expansion tree rooted at $\{(\alpha, \beta)\}$ has a successful branch.*

As we are dealing with strong bisimulation on BPA the finiteness of an expansion as well as finite branching of an expansion tree are guaranteed. However, there may still occur branches of infinite length. To overcome this obstacle one has to introduce modification rules into the construction of expansion trees. In their paper [8], Jančar and Moller introduce the following rules:

**Rule 1 (Congruence rule)** Omit from any node $U$ the pair $(\alpha, \beta)$ if it belongs to the least congruence containing $U^{\Uparrow}$, where $U^{\Uparrow}$ denotes the union of all predecessors on the way from the root to $U$.

**Rule 2 (Decomposition rule)** If $(X\alpha, Y\beta)$ is in $U$ where $X$ and $Y$ are normed, then create a new sibling node $U' = U \setminus \{(X\alpha, Y\beta)\} \cup \{(X, Y\gamma), (\gamma\alpha, \beta)\}$, where $|X| = |Y\gamma|$ (and symmetrically).

**Rule 3 (Replacement rule)** If $(X\alpha, Y\beta)$ is in $U$ and some $(X\alpha', Y\beta')$ is in $U^{\Uparrow}$, then create a new sibling node $U' = U \setminus \{(X\alpha, Y\beta)\} \cup \{(\alpha, \alpha'), (\beta, \beta')\}$.

Obviously, when these rules are applied to the construction we need to verify that correctness is preserved. That boils down to checking that no *false* bisimulation witness is created, i.e. no pairs that would imply bisimilarity of two originally non-bisimilar processes are added. This is guaranteed by the following correctness criterion for (modified) expansion trees.

**Fact 14** *For any node $V \neq \emptyset$ and for any $n \in \mathbb{N}$, $V \subseteq \sim_{n+1}$ iff $V$ has a child $U \subseteq \sim_n$. As a consequence, $V \subseteq \sim$ iff $V$ has a child $U \subseteq \sim$.*

Rule 1 ensures that a pair is not considered if it can be composed from pairs that occurred previously (we use the fact that bisimilarity is a congruence). Rule 2 allows us to replace pairs by their decompositions which are strictly smaller in size (here the strong norm is taken as size criterion). However, one can easily find a pair which is not decomposable in the sense of Rule 2. Then Rule 3 will eventually be applied. Efficiency of the modification rules is asserted by a theorem from [4] that states that the number of undecomposable pairs is in some sense finite and therefore the modified expansion tree with bisimilar pair in its root always contains a finite successful branch. Hence the strong bisimilarity on BPA is semidecidable.

When dealing with weak bisimulation, we have to consider *weak expansions* and *weak expansion trees* that are obtained by replacing single transitions by composite ones. As in the case of the strong bisimulation, in order to cope with infiniteness we will introduce some modification rules that will employ decomposition and bisimulation up to. To this end, we need to define a generalized notion of expansion tree, *expansion tree up to*.

**Definition 15** *Let $V \neq \emptyset$ and $U$ be two sets of elements $(\alpha, \beta)A$. $U$ is called a weak expansion up to of $V$ if it is a minimal set (w.r.t. inclusion) satisfying the following property: for every pair $(\alpha, \beta)A \in V$,*

- *either $(\alpha, \beta) \in A$,*

- *or, for every action $a$,*

  - *if $\alpha \stackrel{a}{\Longrightarrow} \alpha'$ then $\beta \stackrel{a}{\Longrightarrow} \beta'$ with $(\alpha', \beta')A \in U$;*

– *if* $\beta \overset{a}{\Longrightarrow} \beta'$ *then* $\alpha \overset{a}{\Longrightarrow} \alpha'$ *with* $(\alpha', \beta')A \in U$.

The notion of *successful leaf*, resp. *successful branch*, generalizes to expansion trees up to in the obvious sense (in particular, an unsuccessful leaf contains an element $(\alpha, \beta)A$ with $\alpha \not\approx_{1,A} \beta$). Hirshfeld in [7] introduces the modification rules, for totally normed BPA-processes, which can be summarized as follows:

**Rule 4 (Omitting rule)** Omit $(\alpha, \beta)A$ from a node $U$ if it belongs to $U^{\Uparrow}$, or if $(\alpha, \beta) \in A$.

**Rule 5 (Decomposition rule)** If $(\alpha\alpha', \beta\beta')A$ belongs to $U$, then construct a sibling node $U'$ by replacing $(\alpha\alpha', \beta\beta')A$ by the set $\{(\alpha, \beta)B\} \cup \{(\gamma\alpha', \delta\beta')A \mid (\gamma, \delta) \in B\}$, where $B$ is a new set up to (plus some conditions on norms of $\alpha$ and $\beta$ and the norms of processes in $B$).

We are proposing the following generalization of the modification rules of [8], in the spirit of [7], for weak bisimulation and general BPA. The new modification rules are formulated below:

**Rule 6 (Omitting rule)** Omit $(\alpha, \beta)A$ from a node $U$ if any of the following occurs:

1. $(\alpha, \beta)A$ appears in $U^{\Uparrow}$;
2. $(\alpha, \beta)$ belongs to $A$;
3. $\alpha \equiv \beta$ and $(\epsilon, \epsilon) \in A$;
4. $\alpha \equiv \beta$, and they are unnormed processes.

**Rule 7 (Decomposition rule)** If $(X\alpha, Y\beta)A$ belongs to $U$, then construct a sibling node $U'$ by replacing $(X\alpha, Y\beta)A$ by the set $\{(X, Y)B\} \cup \{(\gamma\alpha, \delta\beta)A \mid (\gamma, \delta) \in B\}$, where $B$ is a new set up to.

**Rule 8 (Replacement rule)** If $(X\alpha, Y\beta)A$ is in $U$ and some $(X\alpha', Y\beta')A'$ is in $U^{\Uparrow}$, then create a sibling node $U'$ by replacing $(X\alpha, Y\beta)A$ with the set $\{(\alpha, \alpha')(\epsilon, \epsilon), (\beta, \beta')(\epsilon, \epsilon)\} \cup \{(\gamma, f(\gamma))(\epsilon, \epsilon), (\delta, f(\delta)(\epsilon, \epsilon) \mid (\gamma, \delta) \in A\} \cup \{(g(\gamma), \gamma)(\epsilon, \epsilon), (g(\delta), \delta)(\epsilon, \epsilon) \mid (\gamma, \delta) \in A'\}$, where $f : A \longrightarrow A'$, and $g : A' \longrightarrow A$ are arbitrary functions.

Rule 6 describes pairs whose presence in the tree is superfluous. Rule 7 is an analog of Decomposition Rules 2 and 5, and Rule 8 is a weak bisimulation analog of Replacement Rule 3. Obviously, as well as with strong bisimulation expansion trees, we need to check that the correctness of the construction has not been affected, in particular that no false witness can be added in this way. The correctness criterion needs to reflect the fact

that for weak bisimulation approximants, convergence (to the maximal relation) may occur at any ordinal less than $\omega_1$. Furthermore, we are dealing with pairs bisimilar up to. Both facts are taken into account in the criterion below.

**Fact 16** *For any node $V \neq \emptyset$ and for any $\mu < \omega_1$, there exists $(\alpha, \beta)A$ in $V$ such that $(\alpha, \beta) \notin \approx_{\mu,A}$ iff for every child $U$, there exist $\kappa < \mu$ and $(\alpha', \beta')A'$ in $U$ such that $(\alpha', \beta') \notin \approx_{\kappa,A'}$.*

As a consequence of Fact 16 together with the convergence criterion ($\approx_A = \bigcap_{\kappa \in \omega_1} \approx_{\kappa,A}$) we arrive at Fact 17:

**Fact 17** *For any node $V \neq \emptyset$, $\{(\alpha, \beta) \mid (\alpha, \beta)A \in V\} \subseteq \approx_A$, for every $A$, iff there exists a child $U$ with $\{(\alpha, \beta) \mid (\alpha, \beta)A \in U\} \subseteq \approx_A$, for every $A$.*

Clearly, if we start with a tree rooted at $(\alpha, \beta)(\epsilon, \epsilon)$ for a bisimilar pair then the root satisfies condition of Fact 17 and so it has a child also satisfying the condition, and so on. The sequence of such nodes forms a successful branch, finite or infinite. On the other hand, if the initial pair is not equivalent at some $\approx_\mu$, then every branch determines a sequence of inequivalent elements $(\alpha, \beta) \notin \approx_{\kappa,A_\kappa}$ where $\kappa$ is decreasing. Since every decreasing sequence of ordinals is finite every branch will eventually reach a node containing some2 $(\alpha, \beta) \notin \approx_{1,A_1}$ which denotes failure. This argument is reflected in the theorem that follows.

**Theorem 18** *If a (modified) expansion tree $T$ rooted at $(\alpha, \beta)(\epsilon, \epsilon)$ satisfies Fact 16, then $\alpha \approx \beta$ iff there exists a successful branch in $T$.*

This theorem states that every rule respecting Fact 16 maintains safeness. The next step is therefore to prove that Rules 6,7, and 8 satisfy Fact 16. The way of doing so is to assume that, given some node and its successors satisfying the condition of Fact 16, any new child arising by application of the rules will also respect it.

Rule 6 specifies when checking a pair $(\alpha, \beta)A$ would be superfluous, either because it has been considered previously (case 1), or its bisimilarity up to $A$ can be proved by some simple argument (cases 2,3,4). The correctness of Rule 7 comes as a consequence of the following lemma:

**Lemma 19** *If $(\alpha_1, \beta_1) \in \approx_{\kappa,B}$ and $(\gamma\alpha_2, \delta\beta_2) \in \approx_{\kappa,A}$ for every $(\gamma, \delta) \in B$, then $(\alpha_1\alpha_2, \beta_1\beta_2) \in \approx_{\kappa,A}$.*

**Proof:** Would be formally done by transfinite induction. First we need to consider the case when $\kappa = 0$; that holds trivially as by definition, all pairs are equivalent at $\approx_0$.

The successor case ($P(\kappa) \Rightarrow P(\kappa + 1)$) is spelt out as follows:

$$\underbrace{[(\alpha_1,\beta_1) \in \approx_{\kappa,B} \wedge \forall(\gamma,\delta) \in B.(\gamma\alpha_2,\delta\beta_2) \in \approx_{\kappa,A} \Rightarrow (\alpha_1\alpha_2,\beta_1\beta_2) \in \approx_{\kappa,A}]}_{P(\kappa)} \Longrightarrow$$

$$\underbrace{[(\alpha_1,\beta_1) \in \approx_{\kappa+1,B} \wedge \forall(\gamma,\delta) \in B.(\gamma\alpha_2,\delta\beta_2) \in \approx_{\kappa+1,A} \Rightarrow (\alpha_1\alpha_2,\beta_1\beta_2) \in \approx_{\kappa+1,A}]}_{P(\kappa+1)}$$

The goal therefore is to prove that, from the induction hypothesis $P(\kappa)$ and assumptions $(\alpha_1,\beta_1) \in \approx_{\kappa+1,B}$ and $(\gamma\alpha_2,\delta\beta_2) \in \approx_{\kappa+1,A}$, for every $(\gamma,\delta) \in B$, we can conclude that $(\alpha_1\alpha_2,\beta_1\beta_2) \in \approx_{\kappa+1,A}$. We will again make use of graphical description of the situation (Fig. 4). We assume an initial move of $\alpha_1\alpha_2$ which may either end up in some $\gamma\alpha_2$ (see diagram I), or lead to some $\alpha_2'$ with $\alpha_1$ removed along the way (diagram II). In the first case we have a matching equivalent move of $\beta_1 \overset{a}{\Longrightarrow} \beta_1'$ with $\alpha_1' \approx_{\kappa,B} \beta_1'$. By applying induction hypothesis to the pair $(\alpha_1',\beta_1')$ we obtain that $(\alpha_1'\alpha_2,\beta_1',\beta_2) \in \approx_{\kappa,A}$, which then validates the desired claim $\alpha_1\alpha_2 \approx_{\kappa+1,A} \beta_1\beta_2$.

In the second case we need to use the fact that if we reach $\epsilon$ from $\alpha_1$, a matching equivalent move of $\beta_1$ leads to some $\delta$ where $(\epsilon,\delta) \in B$. Then we can use the induction hypothesis to conclude that $(\alpha_2,\delta\beta_2) \in \approx_{\kappa+1,A}$ from which the equivalence of $\alpha_1\alpha_2$ and $\beta_1\beta_2$ at $\approx_{\kappa+1,A}$ follows. Moves initiating in $\beta_1$ and the combination $\alpha_1\alpha_2 \overset{\tau}{\Longrightarrow} \epsilon\alpha_2 \overset{a}{\Longrightarrow} \alpha_2'$ would be solved analogously.

The limit case would consist in proving that $\forall\kappa < \lambda.P(\kappa) \Rightarrow P(\lambda)$, and it would proceed analogously to the successor case. The $\epsilon$-condition on $\approx_{\kappa,A}$ is also straightforward to verify.
$\square$

As a consequence of the previous lemma we obtain that if there is a node $V$ containing some $(\alpha,\beta) \notin \approx_{\mu,A}$, then there must be some $(\alpha',\beta') \notin \approx_{\kappa,A'}$ in a new successor node $U'$, for some $\kappa < \mu$.

In order to prove safeness of Rule 8 we need to build a sequence of auxiliary results concerning restricted transitivity for approximants up to. In order to make our notation more concise we shall write $A \approx A'$ whenever for every $(\gamma,\delta) \in A$ there exists $(\gamma',\delta') \in A'$ with $\gamma \approx \gamma'$ and $\delta \approx \delta'$, and symmetrically for $A'$.

**Lemma 20** *If $\alpha \approx_{\kappa,A} \beta$ and $\beta \approx \beta'$, then there exists a set $A'$ such that $\alpha \approx_{\kappa,A'} \beta'$ and $A \approx A'$.*

**Proof:** The flavor of the proof is similar to the analogous lemma for bisimulation, however here we need to employ the principle of transfinite
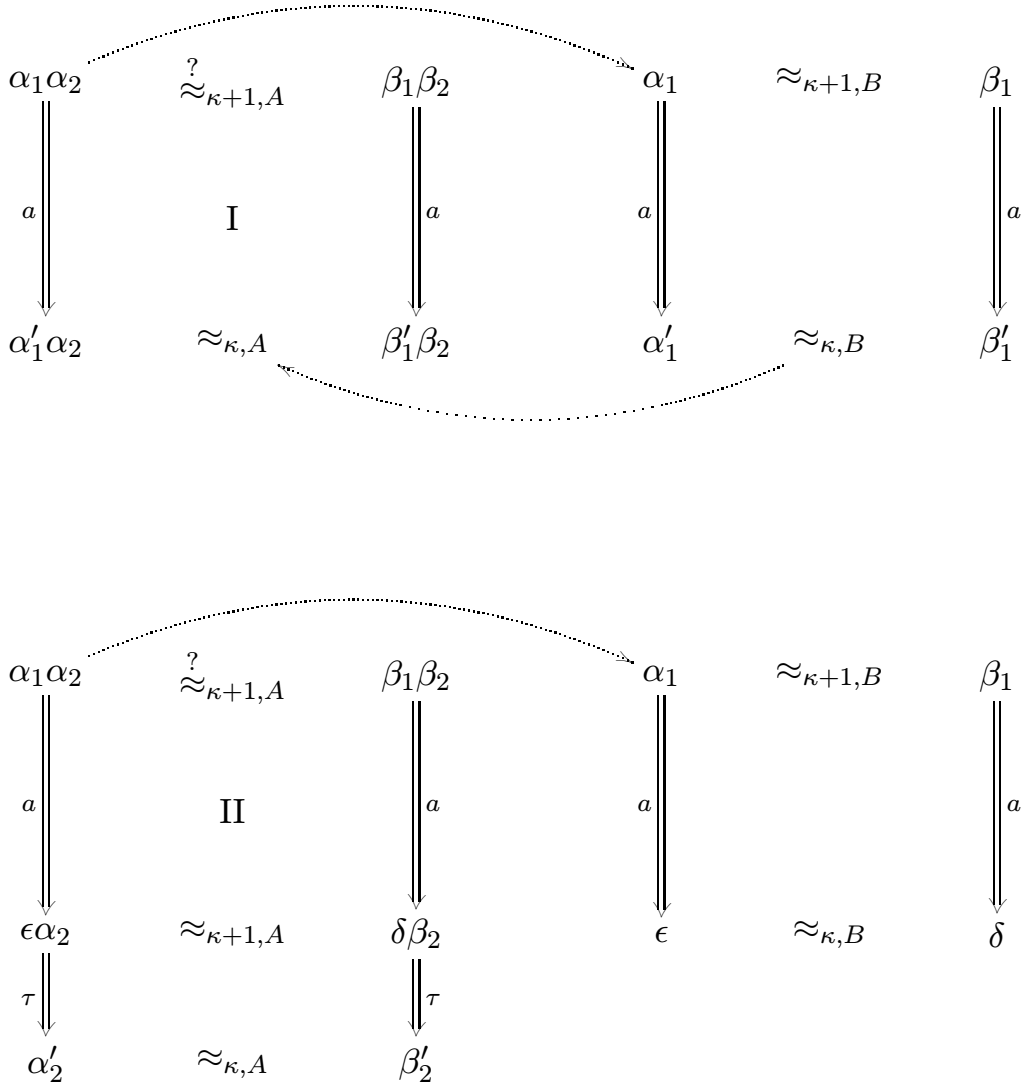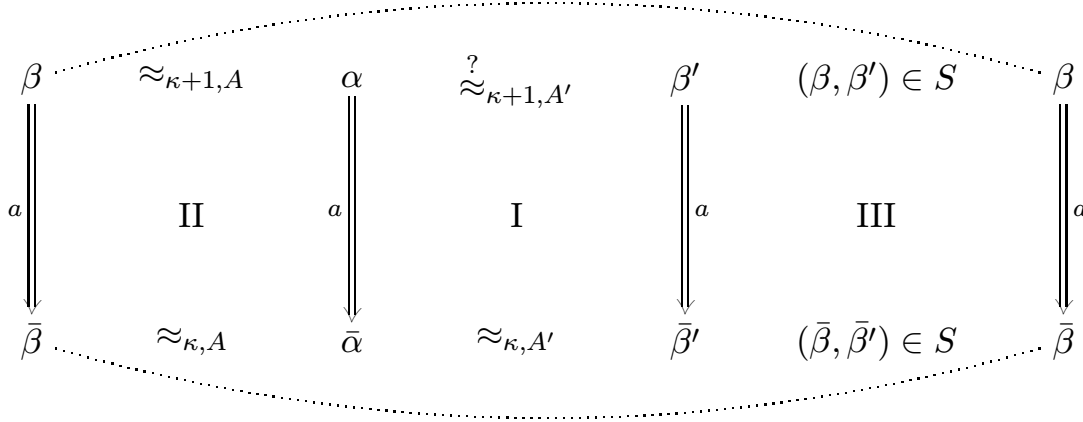
Figure 4: Case analysis

induction. For a fixed $\alpha$, $\beta$, $\beta'$, and $A$, the set $A'$ is defined using $A$ and $S$, some fixed weak bisimulation relating $\beta$ and $\beta'$:

$$A' = \{(\gamma, \delta) \mid \exists (\gamma, \delta') \in A \ \wedge \ (\delta', \delta) \in S\}$$

As in the proof of Lemma 9, it is not difficult to verify that indeed, $A \approx A'$, hence it remains to test the expansion condition. The case for $\kappa = 0$ is clear, and so we continue with the successor step. The induction hypothesis $P(\kappa)$ is the statement $\alpha \approx_{\kappa,A} \beta \wedge \beta \approx \beta' \Rightarrow \alpha \approx_{\kappa,A'} \beta'$. We are going to assume that $\alpha \approx_{\kappa+1,A} \beta$ and $\beta \approx \beta'$ and prove that $\alpha \approx_{\kappa+1,A'} \beta'$.

From the definition of $A'$ we can conclude that $(\alpha, \beta) \in A$ if and only if, $(\alpha, \beta') \in A'$. Therefore we can assume that if there is a move $\alpha \xoverset{a}{\Longrightarrow} \bar{\alpha}$, then

there exists $\beta \overset{a}{\Longrightarrow} \bar{\beta}$, where $\bar{\alpha} \approx_{\kappa,A} \bar{\beta}$ (square II). Then we have a matching bisimilar transition $\beta' \overset{a}{\Longrightarrow} \bar{\beta}'$ (square III). Now we can apply the induction hypothesis on the pairs $(\bar{\alpha}, \bar{\beta}) \in \approx_{\kappa,A}$ and $\bar{\beta} \approx \bar{\beta}'$. Here we need to realize that $(\bar{\beta}, \bar{\beta}')$ is a different pair than the original $(\beta, \beta')$, however as the former is a derivative of the latter, we may use the bisimulation $S$ to define the new set up to and thus we obtain the same set $A'$ with $\bar{\alpha} \approx_{\kappa,A'} \bar{\beta}'$. Therefore we may conclude that indeed, $\alpha \approx_{\kappa+1,A'} \beta'$.

If we start from a transition $\beta' \overset{a}{\Longrightarrow} \bar{\beta}'$, we make use of a matching bisimilar move $\beta \overset{a}{\Longrightarrow} \bar{\beta}$ (diagram III). Then in square II we have a move $\alpha \overset{a}{\Longrightarrow} \bar{\alpha}$ with $\bar{\alpha} \approx_{\kappa,A} \bar{\beta}$, and using an analogous argument, we can conclude that $\alpha \approx_{\kappa+1,A'} \beta'$.

For a limit ordinal $\lambda$, the proof relies on the fact that, $\alpha \approx_{\lambda,A} \beta$ if and only if, $\alpha \approx_{\kappa,A} \beta$, for every ordinal $\kappa < \lambda$. The argument is analogous to the successor case.   $\square$

As a corollary of this lemma we obtain the following:

**Lemma 21** *If $X\alpha \approx_{\kappa,A} Y\beta$, $\alpha \approx \alpha'$ and $\beta \approx \beta'$, then there exists a set $A'$ such that $X\alpha' \approx_{\kappa,A'} Y\beta'$ and $A \approx A'$.*

However, note that in order to obtain the lemma we need to employ a symmetric variant of Lemma 20 where we substitute a bisimilar pair on the left hand side. The reason for that is that in general, approximants up to (and also bisimulation up to) are not symmetric relations.
As a consequence of the previous lemma we obtain that if there is a node $V$ containing some $(\alpha, \beta) \notin \approx_{\mu,A}$, then there must be some $(\alpha', \beta') \notin \approx_{\kappa,A'}$ in a new successor node $U'$, for some $\kappa < \mu$.

# 5   Applications

In the previous section we have sketched the way of building up the (modified) expansion tree for a given pair of processes. Now we shall discuss applicability of this approach to deciding weak bisimilarity.

Necessary conditions for a (modified) expansion tree to be an algorithm are following

1. the tree is *finitely* branching

2. every vertex is labeled by a *finite* set

3. if the root is labeled by a (weakly) bisimilar pair then the tree has a *finite* successful branch.

The first condition is not valid as a finite set can have infinite number of different weak expansions due to the composite transitions. Nevertheless its invalidity is not critical. Searching the tree by dove-tailing technique results in semidecision procedure (if there is a finite successful branch than it is found otherwise the search never halts).

The second condition also need not be true. There are two sources of infinity. Firstly, while expanding a finite set we can come to an infinite one owing to composite transition on the attacker side. An example of such a process is given in Example 1. Secondly, infiniteness can arise while decomposing a pair of processes, namely in the set up to. A simple example is given below:

$$X \xrightarrow{a} \epsilon \qquad Y \xrightarrow{a} \epsilon \qquad B \xrightarrow{b} \epsilon \qquad U \xrightarrow{b} U$$
$$X \xrightarrow{a} XB \qquad Y \xrightarrow{a} Y$$

Although $XU \sim YU$ we cannot decompose the pair as $X \not\sim Y$, moreover, at a closer look we find out that any set $A$ with the property that $X$ is bisimilar to $Y$ up to $A$ is infinite and must contain $\{(B^i, \epsilon) \mid i \in \mathbb{N}\}$. However, (finiteness of) the decision procedure is based on the fact that any two non-bisimilar variables have only finitely many nonbisimilar completions.

In case of weak bisimulation, one can avoid problems 1. and 2. by considering the variant of weak bisimulation in which attacker is allowed to do only simple transitions (for the first type of infinity) or by allowing compact finite representation of the sets up to (i.e. via a finite or pushdown automaton). But there is still the third condition we have to cope with. As the following example shows this obstacle is the most serious one.

**Example 22** This simple BPA represents an algebra where violation of condition 3 appears, i.e. there exists a bisimilar pair for which the modified expansion tree has no finite successful branch.

$$X \xrightarrow{\tau} ZY \qquad X \xrightarrow{a} XW \qquad Z \xrightarrow{\tau} ZW \qquad Y \xrightarrow{c} Y$$
$$X \xrightarrow{\tau} \epsilon \qquad\qquad\qquad\quad Z \xrightarrow{\tau} \epsilon \qquad W \xrightarrow{b} \epsilon$$

All relevant (in)equivalence relationships are summarized below:

1. $Y \approx Y\alpha$, for any process $\alpha$;

2. $XW^iY \approx XW^jY$, for every $i, j$;

3. $XW^i \not\approx XW^j$, for every $i \neq j$;

4. $W^iY \not\approx W^jY$, for every $i \neq j$.

As $Y$ is an unnormed variable, the first equivalence is easy to observe. To verify item 3., we assume that $i < j$, and observe that after $XW^i$ disposes of $X$ it will do exactly $b^i$ to reach $\epsilon$, however $XW^j$ can in any case do at least $b^{i+1}$ as $i < j$. Similarly in case 4., if $i < j$ then $W^iY$ can do $c$ after $b^i$ which cannot be matched by $W^jY$. In order to test equivalence 2. we first analyze all possible (composite) moves of $X$. They are

$$X \xRightarrow{\tau} ZW^kY \qquad X \xRightarrow{a} ZW^kY \qquad X \xRightarrow{b} W^kY$$
$$X \xRightarrow{\tau} W^kY \qquad\quad X \xRightarrow{a} W^kY \qquad\quad X \xRightarrow{c} Y$$
$$X \xRightarrow{\tau} \epsilon$$

We will check weak bisimilarity of $XW^iY$ and $XW^jY$, for arbitrary $i$ and $j$, with the help of Figure 5. We will only analyze the moves of $XW^iY$ as the other process behaves analogously. Firstly, $XW^iY$ may dispose of the $X$ in front, then the other process $XW^jY$ evolves into $W^iYW^jY$ by means of the sequence $XW^jY \xrightarrow{\tau} ZYW^jY \xrightarrow{\tau^i} ZW^iYW^jY \xrightarrow{\tau} W^iYW^jY$, which is equivalent to $W^iY$ by equivalence 1. The other interesting move is $XW^iY \xrightarrow{a} XW^{i+1}Y$ that is matched by $XW^jY \xrightarrow{a} XW^{j+1}Y$. The remaining possibilities consist in $X$ generating $ZW^kY$ or $W^kY$ to which the other side responds by creating an exact copy (hence we obtain two bisimilar processes $Z^eW^kYW^iY$ and $Z^eW^kYW^jY$, where $e \in \{0, 1\}$).

Before we present the construction of a modified expansion tree we will make some observations about decomposability of bisimilar pairs in this algebra. From 3. and 4. above follows that for distinct $i$ and $j$ the pair $(XW^iY, XW^jY)$ has no classical decomposition, i.e. there is *no* way of
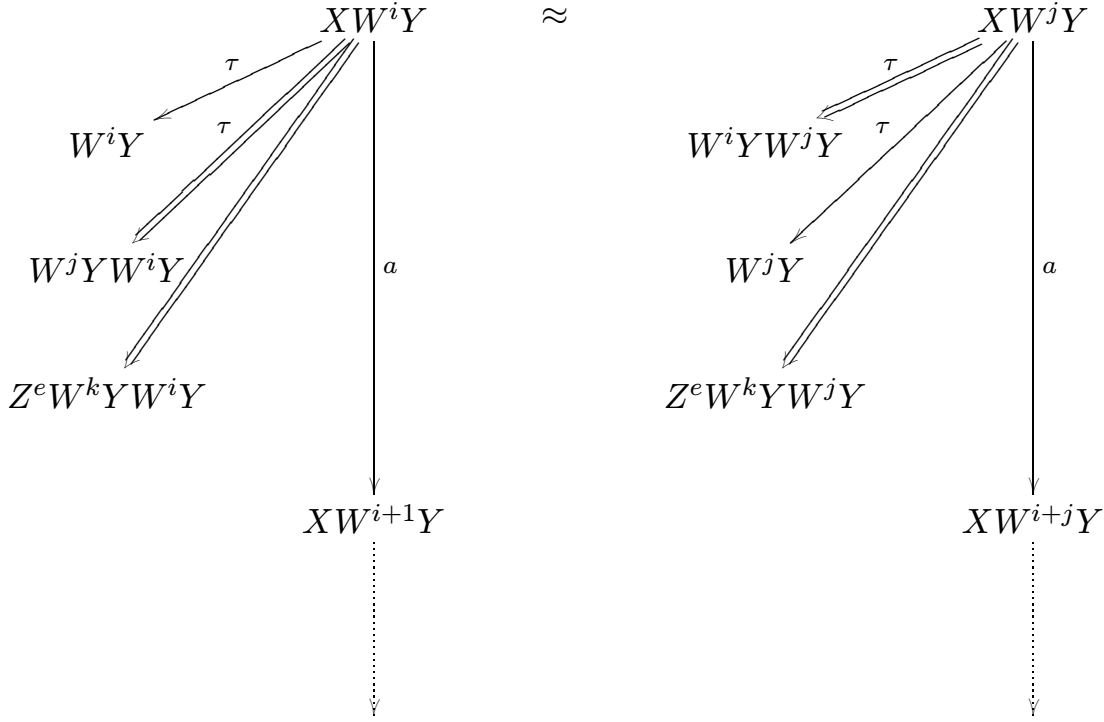
Figure 5: Bisimulation diagram

splitting $XW^iY$ and $XW^jY$ into two pairs of bisimilar processes. Furthermore, every bisimulation relating the pair is infinite and has no finite base as it must contain the set $\{(XW^{i+k}Y, XW^{j+k}Y \mid k \in \mathbb{N}\}$, which is not finitely generated.

At a closer look we may note that in any bisimulation play leading from the pair $(XW^iY, XW^jY)$, $X$ on either side may evolve into an unnormed process by choosing to perform $X \xrightarrow{\tau} ZY$, or it may disappear by doing $X \xrightarrow{\tau} \epsilon$. To the latter move the only correct response (of the other $X$) is $X \xLongrightarrow{\tau} W^kY$, where $k$ depends on $i$ or $j$ and the current depth of the play. Hence we may conclude that in general, $X \approx_A X$ for any set $A$ containing $(\epsilon, W^lY), (W^mY, \epsilon)$, where $l, m \in \mathbb{N}$.

Figure 6 represents a sketch of a construction of modified expansion tree for the pair $(XWY, XW^2Y)$, that only contains correct expansions and correct applications of modification rules. We will make use of equivalence 1. above and only consider those processes that contain at most one $Y$, as the final variable. We make the following conventions: in order to save space the set up to $\{(\epsilon, \epsilon)\}$ is denoted by $\varepsilon$; pairs that are not underlined are those omitted in further construction by application of Rule 6. We either omit identical pairs if the set up to contains $(\epsilon, \epsilon)$ (such as $(W^iY, W^iY)\varepsilon$), or identical pairs of unnormed processes. The other application of omitting rule is whenever a pair belongs to the respective set up

to (e.g. $(\epsilon, W^2Y)A$, where $A = \{(\epsilon, W^2Y), (W^3Y, \epsilon)\}$). The original root is labeled by $\{(XWY, XW^2Y)\varepsilon\}$, however a new root labeled by $\emptyset$ is added as a result of application of Rule 7 to the original one. The rightmost branch actually after a few steps becomes identical to the branch on the left which is denoted by an arrow in the picture.

The correct choices of sets up to when applying Rule 7 are influenced by the only correct response to the transition $X \xrightarrow{\tau} \epsilon$. When we decompose the original pair $(XWY, XW^2Y)$, the only correct set up to is $B = \{(\epsilon, WY), (W^2Y, \epsilon)\}$ as $X \approx_B X$ and also $WY \approx WY$, and $W^2Y \approx W^2Y$. When we move to $(XW^2Y, XW^3Y)$ we need to consider $B' = A = \{(\epsilon, W^2Y), (W^3Y, \epsilon)\}$. Then, as $X$ keeps generating further copies of $W$, also the exponents of $W$ in the consecutive sets up to grow. The sets are finite but unbounded in size of its elements. As the sets are all distinct (w.r.t. weak bisimilarity), any infinite branch cannot be terminated as a successful finite branch by the presented rules.
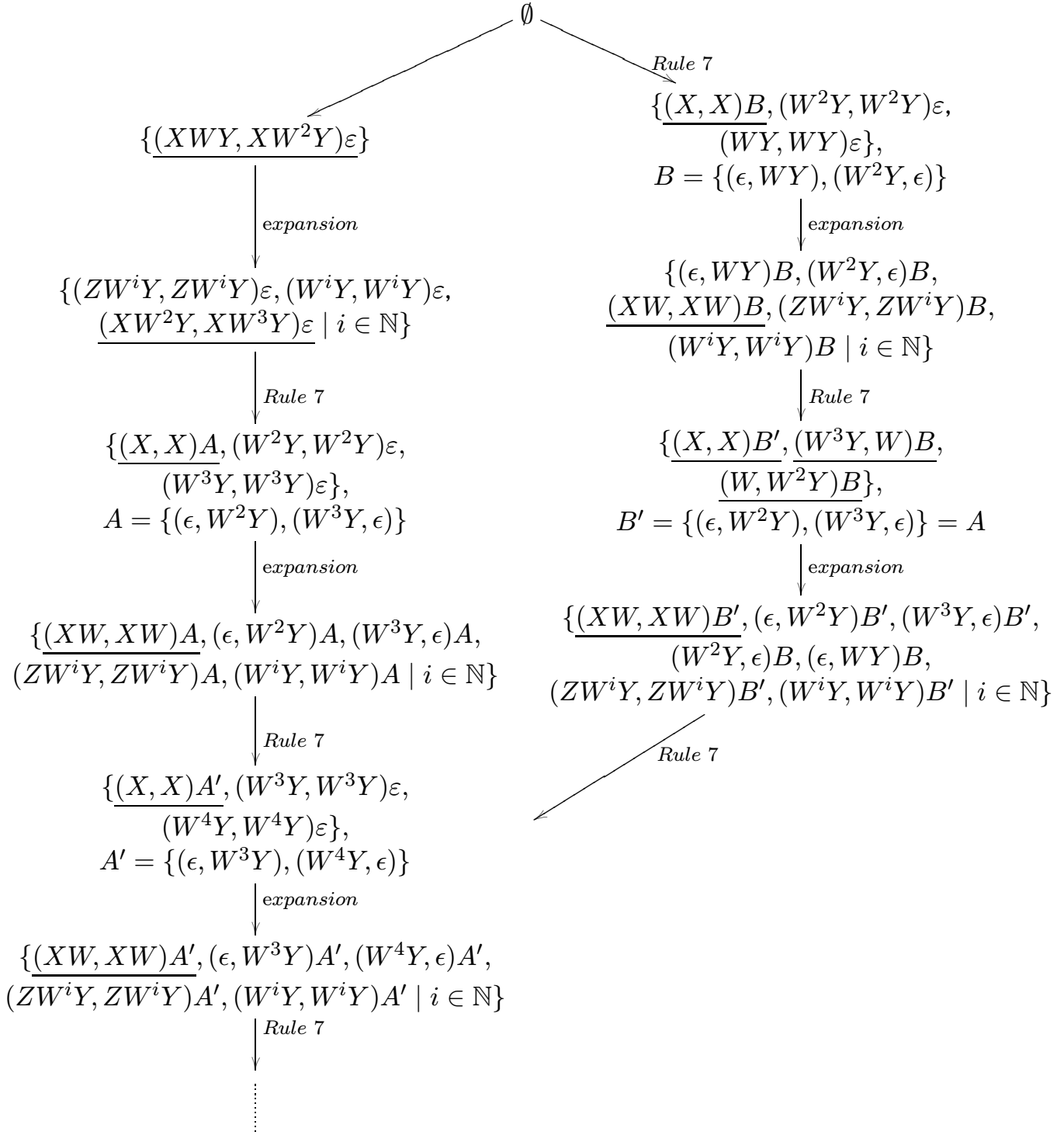
# 6   Conclusions

In this paper we have attempted to generalize the method of expansion trees for semideciding weak bisimilarity of BPA-processes. The main idea was to split a given problem (of deciding whether a given pair is weakly bisimilar) to a number a smaller tasks of the same type which would lead to a recursive procedure. In the Application section we have demonstrated an example of BPA-processes where even after application of the modification rules suggested in this paper we obtain larger and larger processes which results in non-termination of the proposed procedure.

The example presented in the previous section is an example of a process algebra where the maximal weak bisimulation does not have a finite Caucal base, moreover every weak bisimulation relating e.g. the pair $(XW^2Y, XW^3Y)$ also fails to have a finite base. However, we are able to provide a finite description of a Caucal base of any such bisimulation (for instance by means of a pushdown automaton). In general, any recursive description of a Caucal base suffices to semidecide weak bisimilarity. The existence of a recursive Caucal base of the maximal weak bisimulation and its efficient construction remain open questions. This would be one possible way of attacking the (semi)decidability problem for weak bisimilarity on BPA.

# References

[1] Baeten J.C.M., Bergstra J.A. and Klop J.W.: Decidability of bisimulation equivalence for processes generating context-free languages. In Proceedings of PARLE'87, LNCS 259, 1987.

Figure 6: Modified expansion tree for $(XWY, XW^2Y)$

[2] Burkart O., Caucal D. and Steffen B.: An elementary bisimulation decision procedure for arbitrary context-free processes. In Aachener Informatik - Berichte 94–28, 1994.

[3] Caucal D.: Graphes canoniques de graphes algébriques. In Informatique théorique et Applications (RAIRO), 24(4), 339–352,1990.

[4] Christensen S., Hüttel H. and Stirling C.: Bisimulation equivalence is decidable for all context-free processes. In Proceedings of CONCUR 92, LNCS 630, 138–147, 1992.

[5] Hennessy M. and Milner R.: Algebraic laws for nondeterminism and concurrency. In Journal of the ACM 32, 137–161, 1985.

[6] Hirshfeld Y.: Deciding equivalences in simple Process Algebras. In LFCS Report Series, University of Edinburgh, 1994.

[7] Hirshfeld Y.: Bisimulation Trees and the Decidability of Weak Bisimulation. In Proceedings of INFINITY'96, Steffen B., Caucal D. (Eds.), ENTCS 5, 1996.

[8] Jančar P. and Moller F.: Techniques for Decidability and Undecidability of Bisimilarity. In Proceedings of CONCUR'99, LNCS 1664, 1999.

[9] Levy A.: *Basic Set Theory*. Springer-Verlag, Berlin, 1979.

[10] Milner R.: *Communication and Concurrency*. Prentice-Hall, 1989.

[11] Hinman P.G.: *Recursion-Theoretic Hierarchies*. Springer-Verlag, Berlin Heidelberg New York, 1978.

[12] Sénizergues G.: Decidability of bisimulation equivalence for equational graphs of finite out-degree. In Proceedings of FOCS'98, IEEE, 120–129, 1998.

[13] Stříbrná J.: *Decidability and complexity of equivalences for simple process algebras.* PhD thesis, LFCS Report Series, ECS-LFCS-99-408, University of Edinburgh, 1999.

[14] Stříbrná J.: Approximanting Weak Bisimulation on Basic Process Algebra. In FI MU Report Series, Masaryk University Brno, FIMU-RS-99-05, 1999.

Publications in the FI MU Report Series are in general accessible
via WWW and anonymous FTP:

```
http://www.fi.muni.cz/informatics/reports/
ftp  ftp.fi.muni.cz (cd pub/reports)
```

Copies may be also obtained by contacting:

Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic