

Decidability of Weak Bisimilarity for a Subset of BPA

Haiyan Chen^{1,2}

*State Key Laboratory Computer Science, Institute of Software,
Chinese Academy of Sciences, P.O.Box 8718, 100080 Beijing, China and
Graduate School of the Chinese Academy of Sciences,
YuQuan Streets, 100039 Beijing, China*

Abstract

In this paper we use the tableau as a means to prove decidability of weak bisimulation for normed BPA. Decidability was proved for a restricted subclass, the totally normed processes by Hirshfeld [9]. However in the case that weak bisimilarity to be decidable is inequivalence is finitely approximable. In this paper we relax restriction of totally normedness and prove decidability of weak bisimilarity for a subset of normed BPA which permit the norm of BPA processes is zero.

Keywords: BPA, Normed, Tableau, Decidability

1 Introduction

Decidability results for bisimulation equivalence between context-free processes have been flourishing since Baeten, Bergstra, and Klop [4] first proved that bisimulation equivalence is decidable for normed BPA processes, a class of context-free processes. The same fact has been proved by a series of simpler proofs later by Caucal [17], Hans Hüttel and Colin Stirling [11], and Groote [23]. Also algorithms and various complexity results for deciding bisimilarity of normed context-free processes have been obtained by Huynh and Tian [24]. Finally, the decidability result has later been extended to the class of all(not necessarily normed)BPA processes in [5,6] and to pushdown automata [5,19,20]. Above most of the results on infinite state system are concerning strong bisimilarity. For weak bisimilarity, much less is known. Hirshfeld proved a decomposition property for a generalized weak bisimilarity of

¹ Research was supported by the National Natural Science Foundation of China under Grant Nos. 60673045, 60496321

² Email: chy@ios.ac.cn

totally normed context-free processes, and with this directly obtained decidability of bisimilarity of totally normed BPA in [9]. This decidability result is also a consequence of a more elaborate theorem proved by Stirling in [10].

Now families of infinite state systems for which weak bisimilarity is known to be decidable are finitely branching. For each label and for each configuration the set of its successors is finite and easily computable. Therefore if two systems are not bisimulation equivalent then there is a least approximant $n > 0$ such that they are not equivalent at level n , and for each n the equivalence at level n is decidable. But if processes are infinite branching then inequivalence may be manifested at higher ordinals and therefore a new technique is required to establish semidecidability of inequivalence [20].

Hirshfeld proved the decidability of weak bisimilarity on BPA whose norm is greater than 0. And Kučera, Mayr show weak bisimilarity between finite-state systems and BPA or normed BPP is decidable [21]. However both these cases are finite branching. We will resolve broader problem than Hirshfeld. Here we permit that the norm of BPA processes is zero, with the consequence that BPA processes are infinitely branching. Weak bisimulation inequivalence is then generally not finitely approximable.

In this paper by refining Hirshfeld's notion of weak bisimulation up to, we obtain an equivalence relation which enables us to devise a tableau method for deciding weak bisimilarity of totally normed context-free processes. In [11], Hüttel and Stirling proposed a tableau decision procedure for deciding strong bisimilarity of normed context-free processes. Later in [12], Hüttel adapted the tableau method and proved the decidability of branching bisimulation of totally normed context-free processes. As Hüttel pointed out in [12], the key for the tableau method to work is a nice decomposition property which holds for strong bisimulation and branching bisimulation, but fails for weak bisimulation. Our work in some sense is to propose a version of weak bisimulation equivalence for which certain decomposition property (Proposition 3.2) makes the tableau method work correctly. Therefore we use a finite symbolic characterisation of the infinite branching of normed BPA to show weak bisimulation equivalence.

The paper is organized as follows. In section 2 we review some important concepts about BPA processes and weak bisimulation and describe the finite characterization of the infinite transition relations. In section 3 we present the tableau decision method and show the sound and completeness result. Section 4 contains conclusion and suggestions for further work.

2 Preliminaries

2.1 BPA Processes and Weak Bisimulation

We present BPA processes as states in a sequential labeled rewrite system.

Definition 2.1 A sequential labeled rewrite system is a tuple $\langle \mathcal{V}, Act_\tau, \Delta \rangle$ where

- (i) \mathcal{V} is a finite set of variables; the elements of \mathcal{V}^* , finite strings on \mathcal{V} , are referred

to as states.

- (ii) Act_τ is a finite set of labels, containing a special label τ .
- (iii) $\Delta \subseteq \mathcal{V} \times Act_\tau \times \mathcal{V}^*$ is a finite set of rewrite rules.

We use X, Y, Z to range over elements of \mathcal{V} ; a, b, c to range over elements of Act ; α, β, γ to range over elements of \mathcal{V}^* , and write $\alpha\beta$ for the concatenation of α and β . Also we shall write X^n to represent the term $X \cdots X$ consisting of n copies of X combined in sequence. The operational semantics of the processes (states) can be simply given by a labeled transition system $(\mathcal{V}^*, Act_\tau, \longrightarrow)$ where $\longrightarrow \subseteq \mathcal{V}^* \times Act_\tau \times \mathcal{V}^*$ is as follows:

$$\longrightarrow = \{(X\beta, a, \alpha\beta) \mid (X, a, \alpha) \in \Delta, \beta \in \mathcal{V}^*\}.$$

As usual we write $\alpha \xrightarrow{a} \beta$ for $(\alpha, a, \beta) \in \longrightarrow$, write $\alpha \xRightarrow{\epsilon} \beta$ or simply $\alpha \Longrightarrow \beta$ for $\alpha \xrightarrow{(\tau)^*} \beta$, and write $\alpha \xRightarrow{a} \beta$ for $\alpha \xRightarrow{\epsilon} \xrightarrow{a} \xRightarrow{\epsilon} \beta$. We say α is terminating, written $\alpha \Downarrow$, if $\alpha \xrightarrow{(\tau)^*} \epsilon$ where ϵ is the empty string.

Example 2.2 The variables set \mathcal{V} is $\{X, Y, A, B, E, F, D, G\}$, Act_τ is $\{\tau, a, b, c, d\}$ and the basic transitions are $X \xrightarrow{a} \epsilon$, $X \xrightarrow{\tau} XA$, $X \xrightarrow{c} EF$, $Y \xrightarrow{a} \epsilon$, $Y \xrightarrow{\tau} YB$, $Y \xrightarrow{c} GD$, $E \xrightarrow{a} \epsilon$, $F \xrightarrow{\tau} FD$, $F \xrightarrow{a} \epsilon$, $G \xrightarrow{a} FD$, $A \xrightarrow{a} \epsilon$, $A \xrightarrow{\tau} \epsilon$, $B \xrightarrow{a} \epsilon$, $B \xrightarrow{\tau} \epsilon$, $D \xrightarrow{d} \epsilon$, $D \xrightarrow{\tau} \epsilon$. For each $n \geq 0$ there is the extended transition $X \xRightarrow{\epsilon} XA^n$.

Let $\hat{\cdot} : Act_\tau \longrightarrow Act_\tau^*$ be the function such that $\hat{a} = a$ when $a \neq \tau$ and $\hat{\tau} = \epsilon$, then the following general definition of weak bisimulation on \mathcal{V}^* is standard.

Definition 2.3 A binary relation $R \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is a weak bisimulation if for all $(\alpha, \beta) \in R$ the following hold:

- (i) $\alpha \Downarrow$ if and only if $\beta \Downarrow$;
- (ii) whenever $\alpha \xrightarrow{a} \alpha'$, then $\beta \xRightarrow{\hat{a}} \beta'$ for some β' with $(\alpha', \beta') \in R$;
- (iii) whenever $\beta \xrightarrow{a} \beta'$, then $\alpha \xRightarrow{\hat{a}} \alpha'$ for some α' with $(\alpha', \beta') \in R$.

Two states α and β are said to be weak bisimulation equivalent, written $\alpha \approx \beta$, if there is a weak bisimulation R such that $(\alpha, \beta) \in R$.

It is standard to prove that \approx is an equivalence relation between processes. Moreover it is a congruence with respect to string composition on \mathcal{V}^* :

Proposition 2.4 If $\alpha \approx \beta$ and $\alpha' \approx \beta'$ then $\alpha\alpha' \approx \beta\beta'$.

Note that in general clause 1. of Definition 2.3 is necessary. Otherwise any $X \in \mathcal{V}$ which has no transitions will be equated with ϵ , and as a consequence Proposition 2.4 would fail.

2.2 Relative Weak Bisimulation Equivalence

In this section we will present a version of weak bisimulation, i.e. a notion of weak bisimulation relative to a binary relation on states. This notion is a refinement of

Hirshfeld's notion of “bisimulation up to” introduced in [9]. For the induced new equivalence, we then study its decomposition properties, its relationship to \approx . This provides a foundation for the tableau decision method discussed in the next section.

The following definition settles some notations and terminologies.

Definition 2.5 A state α is said to be *normed* if there exists a finite sequence of transitions from α to ϵ , and un-normed otherwise. The weak norm of a state α is the length of the shortest derivation sequence from α to ϵ not counting τ -moves. We denote by $\|\alpha\|$ the weak norm of α . Also, we follow the convention that $\|\alpha\| = \infty$ for unnormed α , and $\infty > n$ for any number n . A state is totally normed if it is a state of a system $\langle \mathcal{V}, Act_\tau, \Delta \rangle$ where for every variable $X \in \mathcal{V}$, $0 < \|X\| < \infty$. If the norm of a variable X is zero, $\mathcal{N}(X)$ is that we find the length of the shortest transition sequence of the form $X \xrightarrow{a_1} \cdots \xrightarrow{a_n} \epsilon$ not containing $\xrightarrow{\epsilon} \epsilon$, where each $\xrightarrow{a_i}$ is counted as 1, and each $a_i \neq \tau$, else $\mathcal{N}(X) = \|X\|$.

Note that $\|X\| < \infty$ is the same as saying that X is normed, while $\|X\| = 0$ implies that X can terminate silently.

Definition 2.6 Let $Norm : \mathcal{V}^* \longrightarrow \mathbb{N}$ be a length function on strings defined by $Norm(\alpha) = \sum_{i=1}^n \mathcal{N}(X_i)$ where $\alpha \equiv X_1 \cdots X_n$.

It is obvious that weak norm is additive: $\|\alpha\beta\| = \|\alpha\| + \|\beta\|$. Moreover, weak norm is respected by \approx .

For a binary relation Γ on states, we write $\stackrel{\Gamma}{=}$ for the equivalence relation generated by the following four rules:

$$\begin{array}{ll} \text{ref} : & \alpha \stackrel{\Gamma}{=} \alpha \\ \text{axiom} : & \alpha \stackrel{\Gamma}{=} \beta ((\alpha, \beta) \in \Gamma) \\ \text{tran} : & \frac{\alpha \stackrel{\Gamma}{=} \beta, \beta \stackrel{\Gamma}{=} \gamma}{\alpha \stackrel{\Gamma}{=} \gamma} \\ \text{sym} : & \frac{\alpha \stackrel{\Gamma}{=} \beta}{\beta \stackrel{\Gamma}{=} \alpha} \end{array}$$

i.e. $\stackrel{\Gamma}{=}$ is the smallest reflexive, symmetry, and transitive binary relation containing Γ . Clearly if Γ is finite then so is $\stackrel{\Gamma}{=}$, and in this case it is decidable whether $\alpha \stackrel{\Gamma}{=} \beta$.

Definition 2.7 Let $R, \Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$. R is a weak bisimulation w.r.t Γ if for all $(\alpha, \beta) \in R$, then either $\|\alpha\| = \|\beta\| \leq 1$ and $\alpha \stackrel{\Gamma}{=} \beta$, or the following hold:

- (i) whenever $\alpha \xrightarrow{a} \alpha'$, then $\beta \xRightarrow{\hat{a}} \beta'$ for some β' such that $(\alpha', \beta') \in R$,
- (ii) whenever $\beta \xrightarrow{a} \beta'$, then $\alpha \xRightarrow{\hat{a}} \alpha'$ for some α' such that $(\alpha', \beta') \in R$.

For $\alpha, \beta \in \mathcal{V}^*$ and $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$, we say that α is weak bisimilar to β w.r.t. Γ , written $\alpha \approx_\Gamma \beta$, if there is $R \subseteq \mathcal{V}^* \times \mathcal{V}^*$ such that R is a weak bisimulation w.r.t Γ and $(\alpha, \beta) \in R$.

This is a typical co-inductive definition for \approx_Γ . With such definition, the following properties of \approx_Γ are expected and have standard proofs which are omitted

here.

Proposition 2.8 *Let $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$, then*

- (i) \approx_Γ *is the largest weak bisimulation w.r.t Γ ;*
- (ii) \approx_Γ *is an equivalence, i.e. it is reflexive, symmetric, and transitive.*

Remark 2.9 Our definition differs from Hirshfeld’s bisimulation up to as follows. First, bisimulation up to is defined through a series of approximation while \approx_Γ is defined using co-induction technique. Second, in our definition we use the weak norms of α, β as pre-conditions to determine when “bisimulation clauses” 1 and 2 are required to hold when are not, moreover in the case clauses 1 and 2 are not required to hold we then require $\alpha \stackrel{\Gamma}{=} \beta$ instead of simply $(\alpha, \beta) \in \Gamma$. As a result \approx_Γ is an equivalence relation, an important property which is necessary for the tableau method to work correctly. Bisimulation up to is not an equivalence relation in general.

Next we define some special kinds of Γ which have useful properties.

Definition 2.10 Let $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$. We say Γ is uniform if $\|\alpha\| = \|\beta\| \leq 1$ for all $(\alpha, \beta) \in \Gamma$. We say Γ is sound if for all $(\alpha, \beta) \in \Gamma$ the following hold:

- (i) whenever $\alpha \xrightarrow{a} \alpha'$ then $\beta \xRightarrow{\hat{a}} \beta'$ for some β' with $\alpha' \approx_\Gamma \beta'$;
- (ii) whenever $\beta \xrightarrow{a} \beta'$ then $\alpha \xRightarrow{\hat{a}} \alpha'$ for some α' with $\alpha' \approx_\Gamma \beta'$.

Proposition 2.11 *Let $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ be uniform. Then $\stackrel{\Gamma}{=}$ respects weak norms, i.e. if $\alpha \stackrel{\Gamma}{=} \beta$ then either both α, β are un-normed, or both are normed and $\|\alpha\| = \|\beta\|$. Moreover \approx_Γ also respects weak norms.*

The following easy to prove lemma shows that for sound Γ the property holds for each $(\alpha, \beta) \in \Gamma$ also holds for those (α, β) where $\alpha \stackrel{\Gamma}{=} \beta$.

Lemma 2.12 *Let $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ be sound. If $\alpha \stackrel{\Gamma}{=} \beta$ then the following hold:*

- (i) *whenever $\alpha \xrightarrow{a} \alpha'$ then $\beta \xRightarrow{\hat{a}} \beta'$ for some β' with $\alpha' \approx_\Gamma \beta'$;*
- (ii) *whenever $\beta \xrightarrow{a} \beta'$ then $\alpha \xRightarrow{\hat{a}} \alpha'$ for some α' with $\alpha' \approx_\Gamma \beta'$.*

Lemma 2.13 *If $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is both uniform and sound then \approx_Γ is a weak bisimulation.*

Proposition 2.14 *Let $\alpha, \beta, \gamma \in \mathcal{V}^*$, $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$. If $\alpha \approx_\Gamma \beta$ then $\gamma\alpha \approx_\Gamma \gamma\beta$.*

Lemma 2.15 *Let $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$, $\alpha_1, \alpha_2, \beta_1, \beta_2 \in \mathcal{V}^*$ with α_1, β_1 normed and $\|\alpha_1\| \geq \|\beta_1\|$. If $\alpha_1\alpha_2 \approx_\Gamma \beta_1\beta_2$ and $\|\beta_1\beta_2\| > 1$ then there exists $\delta \in \mathcal{V}^*$ such that $\delta\alpha_2 \approx_\Gamma \beta_2$.*

2.3 Generation

We symbolically characterize the weak transition relations of normed BPA. Assume a fixed normed BPA definition with variables set \mathcal{V} , action set Act_τ and transitions

\longrightarrow . The initial step is to stratify the basic transitions in \longrightarrow , by including a numerical index on the transition relation which represents the change in norm produced by the transition [20]. If $X \xrightarrow{a} \alpha \in \longrightarrow$ then we rewrite it as $X \xrightarrow{a}_n \alpha$ where $n = \|\alpha\| - \|X\|$. The index n is bounded, $-1 \leq n \leq |\alpha|M$, where M is the maximum norm of any variable in \mathcal{V} . An important, but simple, observation is that for a stratified τ -transition, $X \xrightarrow{\tau}_n \alpha$, the index n must be nondecreasing, $n \geq 0$. In the example of the previous section is $A \xrightarrow{a}_{-1} \epsilon$, $X \xrightarrow{\tau}_0 XB$, $Y \xrightarrow{c}_1 GD$.

The definition of stratification is extended to the weak transition relations as follows.

- (i) $\alpha \xRightarrow{\epsilon}_0 \beta$ iff $\exists m > 0. \exists \alpha_1, \dots, \alpha_m. \alpha = \alpha_1 \xrightarrow{\tau}_0 \dots \xrightarrow{\tau}_0 \alpha_m = \beta$.
- (ii) $\alpha \xRightarrow{\epsilon}_{n+1} \beta$ iff $\exists \alpha' \beta'. \alpha \xRightarrow{\epsilon}_j \alpha' \xrightarrow{\tau}_{k+1} \beta' \xRightarrow{\epsilon}_l \beta$ where $n = j + k + l$.
- (iii) $\alpha \xRightarrow{a}_n \beta$ iff $\exists \alpha' \beta'. \alpha \xRightarrow{\epsilon}_j \alpha' \xrightarrow{a}_k \beta' \xRightarrow{\epsilon}_l \beta$ where $n = j + k + l$

Proposition 2.16 *If α is totally normed then for all a and n $\{\delta : \alpha \xRightarrow{a}_n \delta\}$ is finite.*

However, when α isn't totally normed then for some a and n $\{\delta : \alpha \xRightarrow{a}_n \delta\}$ may be infinite. The reason is that the transition relation $\xRightarrow{\epsilon}_0$ can be infinite branching. Variables can “generate” other variables. If X generates A with $A \in \mathcal{V}$, and for each variable X , the set of variables generated by X , written $Gen(X)$, is $\{A : X \xRightarrow{\epsilon}_0 XA\}$, A is called Generator. In example of the previous section, $Gen(X) = \{A\}$, $Gen(Y) = \{B\}$ and $Gen(F) = \{D\}$.

Note that we write A^n for n 's concatenation of A where $n \geq 0$ and $A^0 = A^*A^*$, and write A^+ for m 's concatenation of A where $m \geq 1$ and $A^+ = AA^*$.

Proposition 2.17 (i) *If $A \in Gen(X)$ then $\|A\| = 0$;*

(ii) *If $A \in Gen(X)$ then $X \xRightarrow{\epsilon}_0 XA^*$;*

(iii) *If $\alpha \in Gen(X)^*$ then $X \xRightarrow{\epsilon}_0 X\alpha$.*

Since BPA processes have “sequence” characteristic, we can't get $X \approx XA$. Hence, we need some techniques to control infinite branching.

The sets $Gen(X)$ for each variable X is easily computable. The main problem with deciding whether or not $\alpha \approx \beta$ is their infinite branching. The technique for overcoming this is to use the finite characteristic to show that we only need to examine boundedly many transitions of α and β . However we are only able to show this for a subset of normed BPA processes.

We restrict to the subset of normed BPA definitions which obey the following condition.

If $Gen(X) \neq \emptyset$ and $X \xrightarrow{\tau}_0 \alpha$ then $\alpha = X\alpha'$.

Effectively this imposes the constraint that generators are “pure”, there any transition $X \xrightarrow{\tau}_0 \alpha$ is a generating transition or is useless ($X \xrightarrow{\tau}_0 X$). We don't consider more complicated and only consider that the norms of variables except Generators are more than 0. In the Example 1 the normed BPA transition system satisfy the previous restriction. However if we add a transition $X \xrightarrow{\tau}_0 AX$, then the transition

system violate the restriction. The next proposition relies on this constraint.

Proposition 2.18 *If $\text{Gen}(X) \neq \phi$ and $X\alpha \xRightarrow{\epsilon}_0 \beta$ then $X \in \beta$ and β can be finitely presented.*

For example if $A, B \in \text{Gen}(X)$ then $X \xRightarrow{\epsilon}_0 X\alpha$, and α can be written $(A^*B^*)^*$. If α contains $*$, $+$ then α is called infinite string.

3 The Tableau Method for Normed BPA

From now on, we restrict our attention to the subset of normed BPA processes mentioned in previous section, i.e. processes of a sequential labeled rewrite system $\langle \mathcal{V}, \text{Act}_\tau, \Delta \rangle$ where $\infty > \|X\| \geq 0$ for all $X \in \mathcal{V}$, the norm of variables except Generators are more than 0 and Generators are finite states. And throughout the rest of the paper, we assume that all the processes considered are the subset of normed unless stated otherwise.

We show that for normed processes the following are decidable:

- (i) whether $\alpha \approx_\Gamma \beta$, where $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is uniform;
- (ii) whether $\alpha \approx \beta$.

We first show that 1 above is decidable. Then we show 2 is also decidable by showing a reduction to 1.

First we list the following obvious properties of such processes.

Proposition 3.1 *In a subset of normed process system $\langle \mathcal{V}, \text{Act}_\tau, \Delta \rangle$,*

- (i) *for a fixed n , there are only finitely many presented $\alpha \in \mathcal{V}^*$ such that $\|\alpha\| = n$;*
- (ii) *if $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is uniform then Γ is finitely presented;*
- (iii) *there are only finitely many presented $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ which are uniform.*

In the following, we devise a tableau method to decide whether $\alpha \approx_\Gamma \beta$. The rules of each tableau system are built around equations of the form $\alpha =_\Gamma \beta$, where $\alpha, \beta \in \mathcal{V}^*$, $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is uniform. Each rule has the form

$$\text{name} \quad \frac{\alpha =_\Gamma \beta}{\alpha_1 =_{\Gamma_1} \beta_1 \dots \alpha_n =_{\Gamma_n} \beta_n} \quad \text{side condition.}$$

The premise of a rule represents the goal to be achieved while the consequents are the subgoals. There are nine rules altogether. One rule for reducing the weak norms of the states in the goal, one rule for aligning the states, one rule for unfolding, and the last one for substituting.

We now explain the nine rules in turn.

3.1 Reducing Weak Norms

The first rule can be used to reduce the weak norms of the states in the goal. The rule is based on the following observation.

Proposition 3.2 *Let $\alpha, \beta \in \mathcal{V}^*$ with $\|\alpha\| = \|\beta\| > 1$, $X \in \mathcal{V}$, $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ be uniform. Then $\alpha X \approx_\Gamma \beta X$ if and only if there exists $\Gamma' \subseteq \mathcal{V}^* \times \mathcal{V}^*$ such that Γ' is uniform and $\alpha \approx_{\Gamma'} \beta$ and $\alpha' X \approx_\Gamma \beta' X$ for all $(\alpha', \beta') \in \Gamma'$.*

Proof. For the “if” direction, let $R = \{(\alpha' X, \beta' X) \mid \alpha' \approx_{\Gamma'} \beta'\}$, it is not difficult to check that $R \cup \approx_\Gamma$ is a weak bisimulation w.r.t. Γ . Also, obviously $(\alpha X, \beta X) \in R$, thus $(\alpha X, \beta X) \in R \cup \approx_\Gamma$, and so $\alpha X \approx_\Gamma \beta X$.

For the “only if” direction, let $\Gamma' = \{(\alpha', \beta') \mid \|\alpha'\| = \|\beta'\| = 1, \alpha' X \approx_\Gamma \beta' X\}$, $R = \{(\alpha', \beta') \mid \alpha' X \approx_\Gamma \beta' X\}$. Obviously Γ' is uniform and $\alpha' X \approx_\Gamma \beta' X$ for all $(\alpha', \beta') \in \Gamma'$. Also it is easy to check that R is a weak bisimulation w.r.t. Γ' and $(\alpha, \beta) \in R$. \square

This proposition guarantees the soundness and backwards soundness of the following rule:

$$\text{reduc} \frac{\alpha X =_\Gamma \beta X}{\alpha =_{\Gamma'} \beta \quad \{\alpha' X =_\Gamma \beta' X \mid (\alpha', \beta') \in \Gamma'\}} \quad \|\alpha\| = \|\beta\| > 1$$

Note that the states in the subgoals all have smaller weak norms than the states in the original goal. Also note that, by 3. of Proposition 3.1, there are only finitely many possible choices for Γ' . This means there are only finitely many different ways to apply this rule.

3.2 Aligning the States

The next rule can be used to align the states in the goal so that rule **reduc** can be applied to the subgoals. The rule is based on the following observation.

Proposition 3.3 *Let $\alpha_1, \beta_1, \alpha, \beta \in \mathcal{V}^*$ with $\|\alpha_1\| \geq \|\beta_1\| > 1$. Then $\alpha_1 \alpha \approx_\Gamma \beta_1 \beta$ if and only if there exists $\delta \in \mathcal{V}^*$ such that $\delta \alpha \approx_\Gamma \beta$ and $\alpha_1 \alpha \approx_\Gamma \beta_1 \delta \alpha$ and $\|\delta\| = \|\alpha_1\| - \|\beta_1\|$.*

Proof. For the “if” direction, suppose $\alpha_1 \alpha \approx_\Gamma \beta_1 \delta \alpha$ and $\delta \alpha \approx_\Gamma \beta$. Then by Proposition 2.14 $\beta_1 \delta \alpha \approx_\Gamma \beta_1 \beta$. Then since \approx_Γ is an equivalence, by transitivity we obtain $\alpha_1 \alpha \approx_\Gamma \beta_1 \beta$.

For the “only if” direction, suppose $\alpha_1 \alpha \approx_\Gamma \beta_1 \beta$. Since $\|\alpha_1\| \geq \|\beta_1\|$, by Lemma 2.15 there exists $\delta \in \mathcal{V}^*$ with $\delta \alpha \approx_\Gamma \beta$. Then by Proposition 2.14 $\beta_1 \delta \alpha \approx_\Gamma \beta_1 \beta$, and again since \approx_Γ is an equivalence, by transitivity $\alpha_1 \alpha \approx_\Gamma \beta_1 \delta \alpha$. By Proposition 2.11 \approx_Γ respects weak norms, thus $\|\alpha_1\| + \|\alpha\| = \|\beta_1\| + \|\delta\| + \|\alpha\|$, and $\|\delta\| = \|\alpha_1\| - \|\beta_1\|$. \square

This proposition guarantees the soundness and backwards soundness of the following two rules:

$$\text{align} \frac{\alpha_1 \alpha =_{\Gamma} \beta_1 \beta}{\alpha_1 \delta \beta =_{\Gamma} \beta_1 \beta \quad \alpha =_{\Gamma} \delta \beta} \quad \|\delta\| = \|\beta_1\| - \|\alpha_1\|, \|\alpha_1\| > 1$$

Note that by 1. of Proposition 3.1 there are only finitely many presented possible choices for δ . Thus there are only finitely many ways to apply each rule.

Discussion In fact we can refine the rule by imposing more strict restrictions on δ . To do so, for $\alpha, \beta \in \mathcal{V}^*$ with $\|\alpha\| \geq \|\beta\|$, we first define the set $D(\alpha, \beta)$ inductively defined the weak norm of β as follows: if $\|\beta\| = 0$ and $\beta = \epsilon$ then $D(\alpha, \beta) = \{\alpha\}$, otherwise

$$D(\alpha, \beta) = \cup \{D(\alpha', \beta') \mid \exists a \in \text{Act}. \alpha \xrightarrow{a} \alpha', \beta \xrightarrow{a} \beta', \|\alpha'\| < \|\alpha\|, \|\beta'\| < \|\beta\|\}.$$

Note that in the above formula the weak norm of α' (β') is exactly one less than that of α (β). With this in mind it is not difficult to see that $D(\alpha, \beta)$ is finite presented and can be easily computed. Then instead of requiring $\|\delta\| = \|\beta_1\| - \|\alpha_1\|$ in the side condition of rule **align**, we can require $\delta \in D(\beta_1, \alpha_1)$, and the refined rule remains both sound and backwards sound. With the new restriction, we only need to consider fewer choices for δ .

3.3 Unfolding by Matching the Transitions

Definition 3.4 Let $(\alpha, \beta) \in \mathcal{V}^* \times \mathcal{V}^*$. A binary relation $M \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is a match for (α, β) if the following hold:

- (i) whenever $\alpha \xrightarrow{a} \alpha'$ then $\beta \xrightarrow{\hat{a}} \beta'$ for some $(\alpha', \beta') \in M$;
- (ii) whenever $\beta \xrightarrow{a} \beta'$ then $\alpha \xrightarrow{\hat{a}} \alpha'$ for some $(\alpha', \beta') \in M$;
- (iii) whenever $(\alpha', \beta') \in M$ then $\|\alpha'\| = \|\beta'\|$ and either $\alpha \xrightarrow{a} \alpha'$ or $\beta \xrightarrow{a} \beta'$ for some $a \in \Sigma$, where α', β' may be infinite string.

It is easy to see that for a given $(\alpha, \beta) \in \mathcal{V}^* \times \mathcal{V}^*$, there are maybe infinitely many possible $M \subseteq \mathcal{V}^* \times \mathcal{V}^*$ which satisfies 3. above, but they are finitely presented from section 2.3. And for such M it is not difficult to see that it is decidable whether M is a match for (α, β) .

The rule can be used to obtain subgoals by matching transitions, and it is based on the following observation.

Proposition 3.5 Let $\alpha, \beta \in \mathcal{V}^*$. Then $\alpha \approx_{\Gamma} \beta$ if and only if there exists a match M for (α, β) such that $\alpha' \approx_{\Gamma} \beta'$ for all $(\alpha', \beta') \in M$.

Proof. Obvious from Definition 2.7. □

This proposition guarantees the soundness and backwards soundness of the following rule:

$$\text{unfold } \frac{\alpha =_{\Gamma} \beta}{\{\alpha' =_{\Gamma} \beta' \mid (\alpha', \beta') \in M\}} \quad M \text{ is a match for } (\alpha, \beta)$$

As pointed out above there are finitely many presented matches for a given (α, β) , so there are finitely many presented ways to apply this rule on (α, β) .

3.4 Substituting the States

The next four rules can be used to substitute the expressions in the goal. The rules are based on the following observation.

Definition 3.6 (dominate and improve)[9]

- (i) The pair $(\alpha_1\alpha_2, \beta_1\beta_2)$ dominates the pair (α_1, β_1) .
- (ii) The pair (α, β) improves the pair (α_1, β_1) if there is some i_0 such that for $i < i_0$ the (total) number of occurrences of X_i is equal in both pairs while the number of occurrences of X_{i_0} is smaller in (α, β) than in (α_1, β_1) .

Proposition 3.7 *Every sequence of pairs in which every pair improves the previous one is finite.*

Definition 3.8 By \prec we denote the well-founded ordering on \mathcal{V}^* given as follows: $X_1^{k_1} \dots X_n^{k_n} \prec X_1^{l_1} \dots X_n^{l_n}$ iff there exists j such that $k_j < l_j$ and for all $i < j$ we have $k_i = l_i$.

Definition 3.9 $\alpha_1 =_{\Gamma} \alpha_2$ is the dominated node of $\alpha_1\beta_1 =_{\Gamma} \alpha_2\beta_2$ or $\alpha_2\beta_2 =_{\Gamma} \alpha_1\beta_1$, if $\alpha_1 =_{\Gamma} \alpha_2$ occurs above them in the tableau.

It is straightforward to show that \prec is well-founded. We shall rely on the fact that \prec is total in the sense that for any $\alpha, \beta \in \mathcal{V}^*$ such that $\alpha \equiv \beta$ we have $\alpha \prec \beta$ or $\beta \prec \alpha$. Also we shall rely on the fact that $\alpha \prec \beta$ implies $\alpha\alpha' \prec \beta\alpha'$ as well as $\alpha \prec \beta\alpha'$ for any $\alpha' \in \mathcal{V}^*$. All these properties are easily seen to hold for \prec .

When building tableaux basic nodes might dominate other basic nodes; we say a basic node $n : \alpha_1\beta_1 = \alpha_2\beta_2$ or $\alpha_2\beta_2 = \alpha_1\beta_1$ dominates any node $n' : \alpha_1 = \alpha_2$ or $n' : \alpha_1 = \alpha_2$ which appears above n in the tableau. There $n' : \alpha_1 = \alpha_2$ or $n' : \alpha_1 = \alpha_2$ is called the dominated node.

Proposition 3.10 *For every β_1, β_2 , A, B are X, Y 's Generator respectively, if $X\beta_1 \approx_{\Gamma} Y\beta_2$ then $XA\beta_1 \approx_{\Gamma} YB^*\beta_2$ iff $XA\beta_1 \approx_{\Gamma} XB^*\beta_2$ iff $YA\beta_1 \approx_{\Gamma} YB^*\beta_2$.*

Proposition 3.11 *For every α_1, α_2 , if $\beta_1 \approx_{\Gamma} \beta_2$, then $\alpha_1\beta_1 \approx_{\Gamma} \alpha_2\beta_2$ iff $\alpha_1\beta_1 \approx_{\Gamma} \alpha_2\beta_1$ iff $\alpha_1\beta_2 \approx_{\Gamma} \alpha_2\beta_2$.*

This proposition guarantees the soundness and backwards soundness of the following rules:

$$\text{sub1l} \quad \frac{XA\beta_1 =_{\Gamma} YB^*\beta_2}{XA\beta_1 =_{\Gamma} XB^*\beta_2} \quad (\text{if } \|XA\beta_1\| = \|YB^*\beta_2\| \leq 2, A \in \text{Gen}(X), B \in \text{Gen}(Y), X \prec$$

Y and there has been a node labelled $X\beta_1 =_{\Gamma} Y\beta_2$)

$$\text{sub1r} \quad \frac{XA\beta_1 =_{\Gamma} YB^*\beta_2}{YA\beta_1 =_{\Gamma} YB^*\beta_2} \quad (\text{if } \|XA\beta_1\| = \|YB^*\beta_2\| \leq 2, A \in \text{Gen}(X), B \in \text{Gen}(Y), Y \prec$$

X and there has been a node labelled $X\beta_1 =_{\Gamma} Y\beta_2$)

$$\text{sub2l} \quad \frac{\alpha_1\beta_1 =_{\Gamma} \alpha_2\beta_2}{\alpha_1\beta_1 =_{\Gamma} \alpha_2\beta_1} \quad (\text{if } \|\alpha_1\beta_1\| = \|\alpha_2\beta_2\| \leq 1, \text{ there is } \beta_1 \prec \beta_2 \text{ and a dominated node}$$

labeled $\beta_1 =_{\Gamma} \beta_2$ or $\beta_2 =_{\Gamma} \beta_1$)

$$\text{sub2r} \quad \frac{\alpha_1\beta_1 =_{\Gamma} \alpha_2\beta_2}{\alpha_1\beta_2 =_{\Gamma} \alpha_2\beta_2} \quad (\text{if } \|\alpha_1\beta_1\| = \|\alpha_2\beta_2\| \leq 1, \text{ there is } \beta_2 \prec \beta_1 \text{ and a dominated node}$$

labeled $\beta_1 =_{\Gamma} \beta_2$ or $\beta_2 =_{\Gamma} \beta_1$)

In fact in section 2, from Proposition 3.7 we know that every sequence of pairs in which every pair improves the previous one is finite. So this means that there are only finitely many different ways to apply the rules.

3.5 Matching the States

Since the node contain $*(or +)$ which exists uncertain factor, then we want to certain them to match.

Proposition 3.12 *If $XA^* \approx_{\Gamma} YB^*$ with $A \in \text{Gen}(X)$, $B \in \text{Gen}(Y)$, we choose some m, n satisfying $\text{Norm}(XA^m) = \text{Norm}(YB^n)$, then $XA^m \approx_{\Gamma} YB^n$.*

$$\text{star} \quad \frac{\alpha_1 B_1^{\eta_1} \cdots B_n^{\eta_n} \beta_1 =_{\Gamma} \alpha_2 A_1^{\kappa_1} \cdots A_n^{\kappa_n} \beta_2}{\{\alpha_1 B_1^{b_1} \cdots B_n^{b_n} \beta_1 =_{\Gamma} \alpha_2 A_1^{a_1} \cdots A_n^{a_n} \beta_2 \mid \text{Norm}(\alpha_1 B_1^{b_1} \cdots B_n^{b_n} \beta_1) = \text{Norm}(\alpha_2 A_1^{a_1} \cdots A_n^{a_n} \beta_2)\}}$$

$\eta_1, \dots, \eta_n, \kappa_1, \dots, \kappa_n \in \{0, 1, *, +\}$, $b_1 \cdots b_n, a_1 \cdots a_n$ is minimum value to satisfy $\text{Norm}(\alpha_1 B_1^{b_1} \cdots B_n^{b_n} \beta_1) = \text{Norm}(\alpha_2 A_1^{a_1} \cdots A_n^{a_n} \beta_2)$

We call $\alpha_1 B_1^{b_1} \cdots B_n^{b_n} \beta_1 =_{\Gamma} \alpha_2 A_1^{a_1} \cdots A_n^{a_n} \beta_2$ as the instantiation of $\alpha_1 B_1^{\eta_1} \cdots B_n^{\eta_n} \beta_1 =_{\Gamma} \alpha_2 A_1^{\kappa_1} \cdots A_n^{\kappa_n} \beta_2$.

3.6 Constructing Tableau

We determine whether $\alpha \approx_{\Gamma} \beta$ by constructing a tableau. There is each tableau with root $\alpha =_{\Gamma} \beta$ using the nine rules introduced above. A tableau is a finite tree with nodes labeled by equations of the form $\alpha =_{\Gamma} \beta$, where $\alpha, \beta \in \mathcal{V}^*$, $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ is uniform.

Initial $\Gamma = \{(X, Y) \mid \|X\| \leq 1, X, Y \in \mathcal{V}\}$. Moreover if $\alpha =_{\Gamma} \beta$ labels a non-leaf node, then the following are satisfied:

$$(i) \quad \|\alpha\| = \|\beta\|;$$

- (ii) its sons are labeled by $\alpha_1 =_{\Gamma_1} \beta_1 \dots \alpha_n =_{\Gamma_n} \beta_n$ obtained by applying rule **reduc** or **align** or **sub1,2** or **unfold** or **star** to $\alpha =_{\Gamma} \beta$, in that priority order;
- (iii) no other non-leaf node is labeled by $\alpha =_{\Gamma} \beta$.

A tableau is a successful tableau if the labels of all its leaves have either of the following forms:

- (i) $\alpha =_{\Gamma} \beta$ where there is a non-leaf node also labeled $\alpha =_{\Gamma} \beta$;
- (ii) $\alpha =_{\Gamma} \beta$ where $\alpha \stackrel{\Gamma}{=} \beta$;

3.7 Decidability, Soundness, and Completeness

Theorem 3.13 *For $\alpha, \beta \in \mathcal{V}^*$, and uniform $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$, there are finitely many tableaux with root $\alpha =_{\Gamma} \beta$, and all of them can be effectively presented.*

Proof. Note that the only rule in which the weak norms of states in the subgoals can be greater than that in the original goal is rule **unfold**, and the priority rule mentioned above determines that this rule can only be applied when all two other rules are not applicable, and it is easy to see that this can only happen when both states in the goal contains no more than 2 letters which are not Generators. When both states that contain Generators have more than 2 letters, there are **sub1,2** rules to control their expansion. There are the uncertain factor $*, +$ will be certain by the star rules. This fact implies that each state in the nodes of a tableau with root $\alpha \approx_{\Gamma} \beta$ has bounded weak norms. Then by 1. and 3. of Proposition 3.1 there are bounded number of different labels in such a tableau. And since no two non-leaf nodes are labeled the same, a tableau with root $\alpha \approx_{\Gamma} \beta$ can only have bounded number of non-leaf nodes, thus the number of tableaux with root $\alpha \approx_{\Gamma} \beta$ must be finite.

There are finitely many (exactly 3) rules to apply on each node, and each rule with finitely many different ways to apply, thus there is a way to enumerate all different tableaux with a root $\alpha \approx_{\Gamma} \beta$. \square

This theorem gives us a decision procedure for the problem whether there is a successful tableau with root $\alpha =_{\Gamma} \beta$, since we just need to enumerate all tableaux with root $\alpha =_{\Gamma} \beta$, and then test if each of them are successful (this test is also decidable as mentioned earlier).

Definition 3.14 A sound tableau is a tableau such that if $\alpha =_{\Gamma} \beta$ is a label in it then $\alpha \approx_{\Gamma} \beta$.

Theorem 3.15 *A successful tableau is a sound tableau.*

Proof. Let T be a successful tableau. We define $\mathcal{K} = \{(\alpha, \Gamma, \beta) \mid \alpha, \beta \in \mathcal{V}^*, \Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^* \text{ is uniform}\}$ to be the smallest set of triples satisfies the following:

- (i) if $\alpha \stackrel{\Gamma}{=} \beta$ then $(\alpha, \Gamma, \beta) \in \mathcal{K}$;
- (ii) if there is a node in T labeled with $\alpha =_{\Gamma} \beta$ and on which rule **unfold** is applied then $(\alpha, \Gamma, \beta) \in \mathcal{K}$;

- (iii) if $(\alpha, \Gamma, \alpha') \in \mathcal{K}$, $(\gamma\alpha', \Gamma, \beta) \in \mathcal{K}$, and $\|\gamma\| > 1$, then $(\gamma\alpha, \Gamma, \beta) \in \mathcal{K}$;
- (iv) if $(\alpha, \Gamma', \beta) \in \mathcal{K}$, $\|\alpha\| = \|\beta\| > 1$, and moreover $(\alpha', \beta') \in \Gamma'$ implies $(\alpha'X, \Gamma, \beta'X) \in \mathcal{K}$, then $(\alpha X, \Gamma, \beta X) \in \mathcal{K}$.
- (v) if $(XA\beta_1, \Gamma, XB^*\beta_2) \in \mathcal{K}$, $A \in \text{Gen}(X)$, $B \in \text{Gen}(X)$, $(X\beta_1, \Gamma, Y\beta_2) \in \mathcal{K}$ and $X \prec Y$, then $(XA\beta_1, \Gamma, YB^*\beta_2) \in \mathcal{K}$;
- (vi) if $(YA\beta_1, \Gamma, YB^*\beta_2) \in \mathcal{K}$, $A \in \text{Gen}(X)$, $B \in \text{Gen}(X)$, $(X\beta_1, \Gamma, Y\beta_2) \in \mathcal{K}$ and $Y \prec X$, then $(XA\beta_1, \Gamma, YB^*\beta_2) \in \mathcal{K}$;
- (vii) if $(\alpha_1, \Gamma, \alpha_2) \in \mathcal{K}$, $(\alpha_1\beta_1, \Gamma, \alpha_1\beta_2) \in \mathcal{K}$ and $\alpha_1 \prec \alpha_2$, then $(\alpha_1\beta_1, \Gamma, \alpha_2\beta_2) \in \mathcal{K}$;
- (viii) if $(\alpha_1, \Gamma, \alpha_2) \in \mathcal{K}$, $(\alpha_2\beta_1, \Gamma, \alpha_2\beta_2) \in \mathcal{K}$ and $\alpha_2 \prec \alpha_1$, then $(\alpha_1\beta_1, \Gamma, \alpha_2\beta_2) \in \mathcal{K}$;
- (ix) if $(\alpha'_1, \Gamma, \alpha'_2) \in \mathcal{K}$, $\text{Norm}(\alpha'_1) = \text{Norm}(\alpha'_2)$ and $(\alpha'_1, \Gamma, \alpha'_2)$ is the instantiation of $(\alpha_1, \Gamma, \alpha_2)$, then $(\alpha_1, \Gamma, \alpha_2) \in \mathcal{K}$.

We will prove the following properties about \mathcal{K} :

- A. If $\alpha =_{\Gamma} \beta$ labels a node in T then $(\alpha, \Gamma, \beta) \in \mathcal{K}$.
- B. If $(\alpha, \Gamma, \beta) \in \mathcal{K}$, then either $\|\alpha\| = \|\beta\| \leq 1$ and $\alpha \stackrel{\Gamma}{=} \beta$, or $\|\alpha\| > 1$ and $\|\beta\| > 1$ and moreover the following hold:
 - (a) if $\alpha \xrightarrow{a} \alpha'$ then $\beta \xrightarrow{\hat{a}} \beta'$ for some β' such that $(\alpha', \Gamma, \beta') \in \mathcal{K}$;
 - (b) if $\beta \xrightarrow{a} \beta'$ then $\alpha \xrightarrow{\hat{a}} \alpha'$ for some α' such that $(\alpha', \Gamma, \beta') \in \mathcal{K}$.

Here we omit the details. □

This theorem means that the decision procedure for existence of successful tableau with root $\alpha =_{\Gamma} \beta$ is sound for $\alpha \approx_{\Gamma} \beta$.

Theorem 3.16 *Let $\alpha, \beta \in \mathcal{V}^*$, and $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ be uniform. If $\alpha \approx_{\Gamma} \beta$ then there is a successful tableau with root $\alpha =_{\Gamma} \beta$.*

Proof. By using Propositions 3.2, 3.3, 3.5, 3.10, 3.11 and 3.12 we can prove the following basic fact: if a sound tableau T is not successful, then we can construct another sound tableau T' which has the same root as T and which has one more non-leaf node than T .

Repeatedly using this basic fact, we can construct a sequence of sound tableaux T_0, \dots, T_n, \dots such that T_0 is just the single leaf node $\alpha =_{\Gamma} \beta$. However since there are only finitely many tableaux with root $\alpha =_{\Gamma} \beta$, this sequence must end, and obviously the last tableau in the sequence is a successful tableau with root $\alpha =_{\Gamma} \beta$. □

This theorem means that the decision procedure for existence of successful tableau with root $\alpha =_{\Gamma} \beta$ is complete for $\alpha \approx_{\Gamma} \beta$.

At last, the following theorem shows how to use the decidability of \approx_{Γ} to solve the decidability of \approx .

Theorem 3.17 *Let $\alpha, \beta \in \mathcal{V}^*$ be normed. Then $\alpha \approx \beta$ if and only if there exists a sound and uniform $\Gamma \subseteq \mathcal{V}^* \times \mathcal{V}^*$ such that $\alpha \approx_{\Gamma} \beta$.*

Theorem 3.18 *Let $\Gamma \subseteq \mathcal{V} \times \mathcal{V}$ be uniform. Then Γ is sound if and only if for all $(\alpha, \beta) \in \Gamma$, there is a match M for (α, β) such that $\alpha' \approx_{\Gamma} \beta'$ for all $(\alpha', \beta') \in M$.*

4 Conclusions and Directions for Further Work

In this paper we proposed a tableau decision method for weak bisimilarity of the class of a subset of normed BPA processes with silent actions. Along the way, we use the finite characteristic to prove infinite branching problem. There does not say anything about the complexity of the tableau-based decision procedure. Since we only want to present a simple decidable method. Of cause, recent results by Richard Mayr show that the problem is EXPTIME-hard for (general) BPA and even for normed BPA [13].

we have yet to find an extension to the results presented in this paper which will remove our restriction to normed processes. We know that we will have decidability of the whole class of BPA processes if we can prove the unique decomposability theorem. However, we have not yet been able to prove this theorem.

For bisimilarity, we would like to explore how much more expressive we can make our calculus whilst maintaining a decidable theory. And encouraged by the positive results so far obtained on the decidability of bisimulation equivalence, we should start seeking more results which would still allow us to obtain decidability. For future works, one may consider similar tableau method for corresponding problem to lift the restriction of normedness BPA which is much harder challenge.

References

- [1] Moller, F., Infinite results. In: Sassone, V., Montanari, U.(eds.) CONCUR 1996. LNCS, vol.1119, pp. 195-216. Springer, Heidelberg.
- [2] Esparza, J., *Decidability of model checking for infinite-state concurrent systems*. Acta Informatica, 97-34: 85-107.
- [3] Burkart, O., J.Esparza, More infinite results. Electronic Notes in Theoretical computer Science.
- [4] Baeten, J.C.M., Bergstra, J.A., Klop, J.W., *Decidability of bisimulation equivalence for processes generating context-free languages*. Journal of the Association for Computing Machinery, 93(40), 653-682.
- [5] Christensen, S.,H.Hüttel,and C.Stirling, *Bisimulation equivalence is decidable for all context-free processes*. Information and Computation, 95(121), 143-148.
- [6] Burkart, O., Didier Caucal, Bernhard Steffen, An elementary bisimulation decision procedure for arbitrary context-free processes. In *Proc. of the 20th International Symposium on Mathematical Foundations of Computer Science table of contents*, pp.423-433.
- [7] Srba, J., Roadmap of Infinite Results. Bulletin of the European Association for Theoretical Computer Science, vol.78, columns:Concurrency, pp163-175.
- [8] Esparza, J., Petri nets, commutative context-free grammars, and basic parallel processes. In: Reichel, H.(ed.) FCT 1995. LNCS, vol. 965,PP. 221-232. Springer, Heidelberg.
- [9] Hirshfeld, Y., Bisimulation trees and the decidability of weak bisimulations. In: INFINITY 1996. proceedings of the 1st International Workshop on Verification of Infinite State Systems, Germany, vol.5.
- [10] Stirling, C., Decidability of bisimulation equivalence for normed pushdown processes. In Proceedings of Concurrency Theory, Pisa, Italy, pp.217-232.

- [11] Hüttel, H., Stirling, C., Actions speak louder than words. Proving bisimilarity for context-free processes. In: LICS 1991. proceedings of 6th Annual symposium on Logic in Computer Science, Amsterdam, pp 376-386.
- [12] Hüttel Hans, Silence is golden: Branching bisimilarity is decidable for. context-free processes. In Proceedings of CAV91, Denmark, pp.2-12.
- [13] Mayr, R., Weak Bisimilarity and Regularity of Context-free Processes is EXPTIME-hard. In: EXPRESS 2003, 10th International Workshop on Expressiveness in Concurrency, Marseille, France.
- [14] Milner, R., “A Calculus of Communicating Systems”. LNCS, vol. 92. Springer, Heidelberg.
- [15] Milner, R., “Communication and Concurrency”. Prentice-Hall International, Englewood Cliffs.
- [16] Bergstra, J.A., Klop, J.W., Process theory based on bisimulation semantics. In: Proceedings of REX Workshop, Amsterdam, The Netherlands, pp. 50-122.
- [17] Caucal, D., *Graphes canoniques de graphes algébriques. Informatique théorique et Applications (RAIRO)*, 90-24(4), 339-352.
- [18] Hirshfeld, Y., Jerrum, M., Moller, F., Bisimulation equivalence is decidable for normed process algebra. In: Wiedermann, J., van Emde Boas, P., Nielsen, M.(eds.)ICALP 1999, LNCS, vol. 1644, pp. 412-421. Springer, Heidelberg.
- [19] Sénizergues, G., Decidability of bisimulation equivalence for equational graphs of finite out-degree. In *Proc. IEEE FOCS'98*, pp.120-129.
- [20] Stirling, C., Decidability of Weak Bisimilarity for a Subset of Basic Parallel Processes. In Proceedings of the 4th International Conference on Foundations of Software Science and Computation Structures (FOSSACS'01), LNCS, vol. 2030, pp: 379 - 393. Springer, Heidelberg.
- [21] Antonn Kucera, Richard Mayr, Weak Bisimilarity with Infinite-State Systems Can Be Decided in Polynomial Time. In *Proc. CONCUR 1999*, pp.368-382.
- [22] Christensen, S., Hans Hüttel, Colin Stirling, Bisimulation Equivalence is Decidable for all Context-Free Processes. In proceedings of CONCUR 1992, pp:138-147.
- [23] Groote, J.F., *A short proof of the decidability of bisimulation for normed BPA processes.* Inform.process.lett.42,167-171.
- [24] Huynh, D.T., Tian, L., *Deciding bisimilarity of normed context-free processes is in σ_2^P* . Theoretical Computer Science 12, 183-197.
- [25] Liu Xinxin, Chen Haiyan, Deciding weak bisimilarity of normed context-free processes using tableau. In proceeding of the 4th International Colloquium on Theoretical Aspects of Computing (ICTAC 2007), LNCS, vol.4711, pp:337-350.