BRICS RS-07-5 Aceto et al.: On the Existence of a Finite Base for Complete Trace Equivalence over BPA with Interrupt

BRICS

Basic Research in Computer Science

On the Existence of a Finite Base for Complete Trace Equivalence over BPA with Interrupt

Luca Aceto
Silvio Capobianco
Anna Ingólfsdóttir

BRICS Report Series **RS-07-5**

ISSN 0909-0878 **February 2007**

See back inner page for a list of recent BRICS Report Series publications.
Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
IT-parken, Aabogade 34
DK–8200 Aarhus N
Denmark
Telephone: +45 8942 9300
Telefax: +45 8942 5601
Internet: BRICS@brics.dk

BRICS publications are in general accessible through the World Wide Web and anonymous FTP through these URLs:

`http://www.brics.dk`
`ftp://ftp.brics.dk`
This document in subdirectory `RS/07/5/`

On the Existence of a Finite Base for Complete Trace Equivalence over BPA with Interrupt

Luca Aceto Silvio Capobianco
Anna Ingólfsdóttir
Department of Computer Science, Reykjavík University,
Kringlan 1, IS-103, Reykjavík, Iceland
Email: luca@ru.is, silvio@ru.is, annai@ru.is *

Abstract

We study Basic Process Algebra with interrupt modulo complete trace equivalence. We show that, unlike in the setting of the more demanding bisimilarity, a ground complete finite axiomatization exists. We explicitly give such an axiomatization, and extend it to a finite complete one in the special case when a single action is present.

Introduction

Mode switching is a desirable feature of programming and verification languages (see [7, 9, 11, 12, 14]). Actually, interrupts in operating systems and exception handling in virtual machines fall under this category, and similar behaviour is explicitly required for control programs and embedded systems.

From the theoretical viewpoint of process algebra, representation of mode switching translates into the isolation of suitable operators on terms. Baeten and Bergstra [6] (reprising Bergstra [9]) discuss some of these operators for Basic Process Algebra (BPA), enriched with the *deadlock constant* δ (a special process, not doing anything) and the *interrupt* and *disrupt* operators. For that language, they construct a complete axiomatization modulo bisimilarity [13, 17], which is finite if the set of actions is finite. However, that axiomatization is based on the use of four more *auxiliary* operators: hence, it is not immediately clear whether this process algebra, modulo bisimilarity, is finitely axiomatizable *by itself*. This fact is not at

*The work of the authors has been partially supported by the project "The Equational Logic of Parallel Processes" (nr. 060013021) of The Icelandic Research Fund.

all immediate, given the many examples [1, 2, 3, 4, 5, 13, 15, 16, 18] where a finite complete axiomatization does not exist.

In this paper, we deal with the process algebra BPA_{int}, obtained from BPA by adding the interrupt operator and, as for the relation modeling "indistinguishability from an external observer", we choose to work with *complete trace equivalence* (briefly, c.t.e.) instead of the more demanding bisimilarity. Basically, a sequence of actions is a complete trace for a closed term, if it "leads the term to termination"; two terms are c.t.e. if they have exactly the same complete traces.

Since equivalence classes of terms modulo complete trace equivalence can be described in the language of *regular expressions*, it is possible to deal with them via language-theoretical techniques. This is precisely the way we find the first of our main results: interrupt is a derived operator for closed terms, modulo c.t.e; that is, for every closed term t over BPA_{int}, there is a term u over BPA which is c.t.e. to t. Such u can be obtained from t via application of instances of a finite number of axioms. Therefore, since BPA has a finite ground complete axiomatization modulo c.t.e. (as will be shown in the paper), BPA_{int} turns out to have one as well. This theorem is in sharp contrast with the negative result proved in [5], to the effect that bisimilarity has no finite axiomatization over closed BPA_{int} terms even in the presence of a single action.

The technical analysis of c.t.e. becomes more complex when we consider terms including variables. In fact, as in the setting of bisimilarity [5], interrupt is *not* a derived BPA operator modulo complete trace equivalence. This rule has precisely one exception, modulo c.t.e.: *when the set of actions is a singleton.* In this special case, not only we are able to remove every occurrence of the interrupt operator, but we also can reduce each BPA_{int} term to a BPA term with a very special "shape"; and in fact, this "shape" is special enough to be *characterizing*, i.e., two BPA_{int} terms are c.t.e. if and only if they can be reduced to the same "specially shaped" term. Again, this will be achieved syntactically by adding a finite number of axioms to the ones we had found earlier, which yields a finite complete axiomatization for BPA_{int} modulo c.t.e. in the presence of a single action. When the set of actions is not a singleton, we have isolated a collection of valid equations. However, the details involved in (dis)proving the completeness of that set of equations have so far defeated us.

The paper is divided as follows. In Section 1 we sketch the framework we are working with. In Section 2 we prove our results for closed terms. In Section 3 we state and prove our result for general terms with a single action. In Section 4 we introduce some additional sound equations we have found, and give hints and suggestions for future research in the field.

1 Preliminaries

We begin by introducing the basic definitions and results on which the technical developments to follow are based. The interested reader is referred to [6, 10] for more information.

1.1 The Language $\mathrm{BPA}_{\mathbf{int}}$

We assume a nonempty alphabet Act of atomic actions, with typical elements a, b. The language for processes we shall consider in this paper, henceforth referred to as $\mathrm{BPA}_{\mathrm{int}}$, is obtained by adding the interrupt operator from [6] to Bergstra and Klop's BPA [10]. This language is given by the following grammar:

$$t ::= x \mid a \mid t \cdot t \mid t + t \mid t \rhd t \ ,$$

where x is a variable drawn from a countably infinite set Var and a is an action. In the above grammar, we use the symbol $\rhd$ for the *interrupt operator*. We shall use the meta-variables t, u, v to range over process terms, and write $Var(t)$ for the collection of variables occurring in the term t. The *size* of a term is the number of operator symbols in it. A process term is *closed* if it does not contain any variables. As usual, we shall often write tu in lieu of $t \cdot u$, and we assume that $\cdot$ binds stronger than both $+$ and $\rhd$, while $\rhd$ binds stronger than $+$. In this paper we will also consider the language BPA, which is constructed as $\mathrm{BPA}_{\mathrm{int}}$ without the interrupt operator.

A substitution is a mapping from process variables to $\mathrm{BPA}_{\mathrm{int}}$ terms. A substitution σ is closed if $\sigma(x)$ is a closed term for every variable x. For every term t and substitution σ, the term obtained by replacing every occurrence of a variable x in t with the term $\sigma(x)$ will be written $\sigma(t)$. Note that $\sigma(t)$ is closed, if so is σ. In what follows, we shall use the notation $\sigma[x \mapsto p]$, where σ is a closed substitution and p is a closed $\mathrm{BPA}_{\mathrm{int}}$ term, to stand for the substitution mapping x to p, and acting like σ on all of the other variables in Var. If $a \in \mathsf{Act}$, we indicate by σ_a the closed substitution that replaces every variable with a, i.e.,

$$\sigma_a(x) = a \ \ \forall x \in \ Var \ . \tag{1}$$

In the remainder of this paper, we let a^1 denote a, and a^{m+1} denote $a(a^m)$. Moreover, we consider terms modulo associativity and commutativity of $+$. In other words, we do not distinguish $t+u$ and $u+t$, nor $(t+u)+v$ and $t+(u+v)$. This is justified because $+$ is associative and commutative with respect to the notion of equivalence we shall consider over $\mathrm{BPA}_{\mathrm{int}}$. (See axioms A1, A2 in Table 2 on page 7.) In what follows, the symbol $=$ will denote equality modulo associativity and commutativity of $+$.

We say that a term t *has* $+$ *as head operator* if $t = t_1 + t_2$ for some terms t_1 and t_2. For example, $a + b$ has $+$ as head operator, but $(a + b)a$ does not.

For $k \geq 1$, we use a *summation* $\sum_{i \in \{1,\dots,k\}} t_i$ to denote $t_1 + \cdots + t_k$. It is easy to see that every BPA_{int} term t has the form $\sum_{i \in I} t_i$, for some finite, nonempty index set I, and terms t_i ($i \in I$) that do not have $+$ as head operator. The terms t_i ($i \in I$) will be referred to as the *(syntactic) summands* of t. For example, the term $(a + b)a$ has only itself as (syntactic) summand.

The operational semantics for the language BPA_{int} is given by the labeled transition system

$$\left(\text{BPA}_{\text{int}}, \left\{\xrightarrow{a} \mid a \in \mathsf{Act}\right\}, \left\{\xrightarrow{a}\checkmark \mid a \in \mathsf{Act}\right\}\right) ,$$

where the transition relations $\xrightarrow{a}$ and the unary predicates $\xrightarrow{a}\checkmark$ are, respectively, the least subsets of $\text{BPA}_{\text{int}} \times \text{BPA}_{\text{int}}$ and BPA_{int} satisfying the rules in Table 1. Intuitively, a transition $t \xrightarrow{a} u$ means that the system represented by the term t can perform the action a, thereby evolving into u. The special symbol $\checkmark$ stands for (successful) termination; therefore the interpretation of the statement $t \xrightarrow{a}\checkmark$ is that the process term t can terminate by performing a. Note that, for every closed term p, there is some action a for which either $p \xrightarrow{a} p'$ holds for some p', or $p \xrightarrow{a}\checkmark$ does.

$$\frac{}{a \xrightarrow{a}\checkmark}$$

$$\frac{t \xrightarrow{a}\checkmark}{t + u \xrightarrow{a}\checkmark} \qquad \frac{u \xrightarrow{a}\checkmark}{t + u \xrightarrow{a}\checkmark} \qquad \frac{t \xrightarrow{a} t'}{t + u \xrightarrow{a} t'} \qquad \frac{u \xrightarrow{a} u'}{t + u \xrightarrow{a} u'}$$

$$\frac{t \xrightarrow{a}\checkmark}{t \cdot u \xrightarrow{a} u} \qquad \frac{t \xrightarrow{a} t'}{t \cdot u \xrightarrow{a} t' \cdot u}$$

$$\frac{t \xrightarrow{a}\checkmark}{t \rhd u \xrightarrow{a}\checkmark} \qquad \frac{t \xrightarrow{a} t'}{t \rhd u \xrightarrow{a} t' \rhd u} \qquad \frac{u \xrightarrow{a}\checkmark}{t \rhd u \xrightarrow{a} t} \qquad \frac{u \xrightarrow{a} u'}{t \rhd u \xrightarrow{a} u' \cdot t}$$

Table 1: Transition Rules for BPA_{int}

The transition relations $\xrightarrow{a}$ naturally compose to determine the possible effects that performing a sequence of actions may have on a BPA_{int} term.

Definition 1.1 For a sequence of actions $a_1 \cdots a_k$ ($k \geq 0$), and BPA_{int} terms t, t', we write $t \xrightarrow{a_1 \cdots a_k} t'$ iff there exists a sequence of transitions

$$t = t_0 \xrightarrow{a_1} t_1 \xrightarrow{a_2} \cdots \xrightarrow{a_k} t_k = t' \ .$$

Similarly, we say that $a_1 \cdots a_k$ ($k \geq 1$) is a complete trace of a BPA$_{\text{int}}$ term t iff there exists a term t' such that

$$t \stackrel{a_1 \cdots a_{k-1}}{\longrightarrow} t' \stackrel{a_k}{\rightarrow} \checkmark \ .$$

If $t \stackrel{a_1 \cdots a_k}{\longrightarrow} t'$ holds for some BPA$_{\text{int}}$ term t', or $a_1 \cdots a_k$ is a complete trace of t, then $a_1 \cdots a_k$ is a *trace* of t.

The *depth* of a term t, written $depth(t)$, is the length of a longest trace it affords. Observe that such a trace is necessarily a complete trace.

The *norm* of a term t, denoted by $norm(t)$, is the length of its shortest complete trace; this notion stems from [8].

The depth and the norm of closed terms can also be characterized inductively thus:

$$\begin{array}{rcl}
depth(a) & = & 1 \\
depth(p+q) & = & \max\{depth(p), depth(q)\} \\
depth(pq) & = & depth(p) + depth(q) \\
depth(p \rhd q) & = & depth(p) + depth(q) \\
\\
norm(a) & = & 1 \\
norm(p+q) & = & \min\{norm(p), norm(q)\} \\
norm(pq) & = & norm(p) + norm(q) \\
norm(p \rhd q) & = & norm(p) \ .
\end{array}$$

Note that the depth and the norm of each closed BPA$_{\text{int}}$ term are positive.

Lemma 1.1 [Operational Correspondence] Assume that t is a BPA$_{\text{int}}$ term, σ is a closed substitution and a is an action. Then the following statements hold:

1. If $t \stackrel{a}{\rightarrow} \checkmark$, then $\sigma(t) \stackrel{a}{\rightarrow} \checkmark$.

2. If $t \stackrel{a}{\rightarrow} t'$, then $\sigma(t) \stackrel{a}{\rightarrow} \sigma(t')$.

3. Assume that t is a BPA term. If $\sigma(t) \stackrel{a}{\rightarrow} \checkmark$, then either

   (a) $t \stackrel{a}{\rightarrow} \checkmark$, or

   (b) $t = x$ and $\sigma(x) \stackrel{a}{\rightarrow} \checkmark$ for some variable x, or

   (c) $t = x + u$ and $\sigma(x) \stackrel{a}{\rightarrow} \checkmark$ for some variable t and term u.

Proof: Statements 1 and 2 are proved by induction on the proof of the relevant transitions. Statement 3 is proved by induction on the structure of the term t.

The details are lengthy, but straightforward, and we therefore omit them. $\Box$

In this paper, we shall consider the language BPA_{int} modulo complete trace equivalence.

Definition 1.2 Two closed BPA_{int} terms p and q are *complete trace equivalent*, denoted by $p \sim q$, if they have the same complete traces, i.e., if for every nonempty word $w \in \mathsf{Act}^+$, w is a complete trace for p iff it is a complete trace for q.
The relation $\sim$ will be referred to as *complete trace equivalence*.

It is evident that $\sim$ is an equivalence. There is more: $\sim$ is a *congruence* with respect to all the operators in the signature of BPA_{int}, that is, if $t \sim t'$ and $u \sim u'$, then $t+u \sim t'+u'$, $tu \sim t'u'$, and $t \rhd u \sim t' \rhd u'$. This will follow from Lemma 2.1, at the beginning of next section. Observe that complete trace equivalent BPA_{int} terms have the same norm and depth.

Complete trace equivalence is extended to arbitrary BPA_{int} terms thus:

Definition 1.3 Let t, u be BPA_{int} terms. Then $t \sim u$ iff $\sigma(t) \sim \sigma(u)$ for every closed substitution σ.

For instance, we have that

$$x \rhd y \sim (x \rhd y) + yx$$

because, as our readers can easily check, the terms $p \rhd q$ and $(p \rhd q) + qp$ have the same set of initial "capabilities", i.e.,

$$p \rhd q \xrightarrow{a} r \text{ iff } (p \rhd q) + qp \xrightarrow{a} r \ \text{, for each } a \text{ and } r\text{, and}$$
$$p \rhd q \xrightarrow{a} \checkmark \text{ iff } (p \rhd q) + qp \xrightarrow{a} \checkmark\text{, for each } a \ .$$

It is natural to expect that the interrupt operator cannot be defined in the language BPA modulo complete trace equivalence. With a single, remarkable exception, this expectation will be confirmed by Proposition 3.1.

1.2 Equational Logic

An *axiom system* is a collection of equations $t \approx u$ over the language BPA_{int}. An equation $t \approx u$ is derivable from an axiom system E, notation $E \vdash t \approx u$, if it can be proved from the axioms in E using the rules of equational logic (viz. reflexivity, symmetry, transitivity, substitution and closure under BPA_{int} contexts):

$$t \approx t \quad \frac{t \approx u}{u \approx t} \quad \frac{t \approx u \quad u \approx v}{t \approx v} \quad \frac{t \approx u}{\sigma(t) \approx \sigma(u)}$$

$$\frac{t \approx u \quad t' \approx u'}{t + t' \approx u + u'} \quad \frac{t \approx u \quad t' \approx u'}{tt' \approx uu'} \quad \frac{t \approx u \quad t' \approx u'}{t \rhd t' \approx u \rhd u'} \ .$$

A1	$x + y$	$\approx$	$y + x$
A2	$(x + y) + z$	$\approx$	$x + (y + z)$
A3	$x + x$	$\approx$	x
A4.1	$(x + y)z$	$\approx$	$(xz) + (yz)$
A4.2	$x(y + z)$	$\approx$	$(xy) + (xz)$
A5	$(xy)z$	$\approx$	$x(yz)$

Table 2: Some Axioms for $\mathrm{BPA}_{\mathrm{int}}$

Definition 1.4 An equation $t \approx u$ over the language $\mathrm{BPA}_{\mathrm{int}}$ is *sound* with respect to $\sim$ iff $t \sim u$. An axiom system is sound with respect to $\sim$ iff so is each of its equations.

An example of a collection of equations over the language $\mathrm{BPA}_{\mathrm{int}}$ that are sound with respect to $\sim$ is given in Table 2. Those equations stem from [10]. Equations dealing with the interrupt operator in the setting of bisimulation semantics using auxiliary operators are offered in [6].

2 A ground complete finite axiomatization for $\mathrm{BPA}_{\mathrm{int}}$

We start by proving that complete trace equivalence is a congruence over $\mathrm{BPA}_{\mathrm{int}}$. In fact, we give a complete, structural description of the complete traces of $\mathrm{BPA}_{\mathrm{int}}$ terms: congruence of complete trace equivalence will be an easy consequence.

First of all, we observe that, given two closed terms t, u over $\mathrm{BPA}_{\mathrm{int}}$ and a nonempty word w over Act, then w is a complete trace for $t + u$ iff w is a complete trace for either t or u, while w is a complete trace for tu iff $w = xy$ for some words x, y that are complete traces for t and u, respectively. In fact, there is a similar characterization for complete traces of $t \triangleright u$, but it's a bit trickier.

Lemma 2.1 Let t and u be closed terms over $\mathrm{BPA}_{\mathrm{int}}$. Let w be a nonempty word over the alphabet Act. Then w is a complete trace for $t \triangleright u$ iff there exist words x, y, z over Act such that

1. $w = xyz$,
2. z is nonempty,
3. xz is a complete trace for t, and
4. y is either empty or a complete trace for u.

Proof: Suppose $t \rhd u \xrightarrow{w} \checkmark$. Then

- either u does *not* initiate, so that $t \xrightarrow{w} \checkmark$, or
- u initiates before t, so that $u \xrightarrow{y} \checkmark$, then $t \xrightarrow{z} \checkmark$ for some nonempty words y, z such that $w = yz$, or
- u initiates while t is running, so that $t \xrightarrow{x} t'$ for some closed term t', $u \xrightarrow{y} \checkmark$, and $t' \xrightarrow{z} \checkmark$ for some nonempty words x, y, z such that $w = xyz$.

The reverse implication is trivial. □

Lemma 2.1 allows one to give a language-theoretic characterization of closed terms over BPA_{int} modulo complete trace equivalence. Call $CT(t)$ the set of complete traces of closed term t: then Lemma 2.1 states that the equalities

$$\begin{aligned} CT(t+u) &= CT(t) \cup CT(u) , & (2)\\ CT(tu) &= CT(t)CT(u) , \text{ and} & (3)\\ CT(t \rhd u) &= CT(t) \cup \bigcup_{rs \in CT(t), s \neq \varepsilon} \{r\}CT(u)\{s\} & (4) \end{aligned}$$

hold. We recall that $XY = \{w : \exists x \in X, y \in Y : w = xy\}$.

Corollary 2.1 For terms over BPA_{int}, complete trace equivalence is a congruence.

Proof: Suppose $t \sim t'$ and $u \sim u'$. Let σ be a closed substitution: then $\sigma(t)$ and $\sigma(t')$ have the same set of complete traces, and similarly for $\sigma(u)$ and $\sigma(u')$. By (2), $\sigma(t+u) = \sigma(t) + \sigma(u)$ has the same complete traces as $\sigma(t'+u') = \sigma(t') + \sigma(u')$; similarly for $\sigma(tu)$ and $\sigma(t'u')$ because of (3), and for $\sigma(t \rhd u)$ and $\sigma(t' \rhd u')$ because of (4). This is true for every closed substitution σ, thus $t + u \sim t' + u'$, $tu \sim t'u'$, and $t \rhd u \sim t' \rhd u'$. □

As a consequence of Lemma 2.1 and our previous observations, we obtain the following equivalences.

Lemma 2.2 For every action a and closed terms t, u, v over BPA_{int}, the following hold:

1. $t + u \sim u + t$;
2. $t + (u + v) \sim (t + u) + v$;
3. $t + t \sim t$;

4. $(t+u)v \sim tv + uv$;

5. $t(u+v) \sim tu + tv$;

6. $(tu)v \sim t(uv)$;

7. $a \rhd u \sim a + ua$;

8. $at \rhd u \sim a(t \rhd u) + uat$;

9. $(t+u) \rhd v \sim (t \rhd v) + (u \rhd v)$; and

10. $t \rhd (u+v) \sim (t \rhd u) + (t \rhd v)$.

Proof: We must show that, for any of the formulas above and for any word w over Act, w is a complete trace for the left-hand side iff it is for the right-hand side. Thanks to equations (2), (3), and (4), this is basically an exercise in sentence rewriting; only the last four identities require a greater amount of caution.

7. Suppose w is a complete trace for $a \rhd u$. By Lemma 2.1, this is the same as saying that $w = xyz$ so that z is nonempty, $a \stackrel{xz}{\to} \checkmark$ and either y is empty or $u \stackrel{y}{\to} \checkmark$. The first part is possible iff x is empty and $z = a$, thus either $w = a$ or $w = ya$ with $u \stackrel{y}{\to} \checkmark$; by Lemma 2.1, this is the same as saying that w is a complete trace for $a + ua$. On the other hand, if $a + ua \stackrel{w}{\to} \checkmark$, then either $w = a$ or $w = ya$ for some y such that $u \stackrel{y}{\to} \checkmark$; in either case, w is a complete trace for $a \rhd u$ as well.

8. Suppose w is a complete trace for $at \rhd u$. We can write $w = xyz$ with z nonempty, $at \stackrel{xz}{\to} \checkmark$, and either y empty or $u \stackrel{y}{\to} \checkmark$. Two cases are possible: either $x = ax'$, or x is empty and $z = az'$. In the first case $t \stackrel{x'z}{\to} \checkmark$, and either y empty or $u \stackrel{y}{\to} \checkmark$, so that $x'yz$ is a complete trace for $t \rhd u$, and $w = ax'yz$ is a complete trace for $a(t \rhd u)$; in the second case, $w = yaz'$ is a complete trace for uat. On the other hand, let w be a complete trace for $a(t \rhd u) + uat$: then either $a(t \rhd u) \stackrel{w}{\to} \checkmark$, so that $w = axyz$ with $t \stackrel{xz}{\to} \checkmark$ and either y empty or $u \stackrel{y}{\to} \checkmark$; or $w = yax$ with $u \stackrel{y}{\to} \checkmark$ and $t \stackrel{x}{\to} \checkmark$. In either case, $at \rhd u \stackrel{w}{\to} \checkmark$.

9. Suppose w is a complete trace for $(t+u) \rhd v$. This is the same as saying that $w = xyz$ with z nonempty, either $t \stackrel{xz}{\to} \checkmark$ or $u \stackrel{xz}{\to} \checkmark$, and either y empty or $v \stackrel{y}{\to} \checkmark$. This means that either $t \rhd v \stackrel{w}{\to} \checkmark$ or $u \rhd v \stackrel{w}{\to} \checkmark$.

10. Suppose w is a complete trace for $t \rhd (u+v)$. This is the same as saying that $w = xyz$ with z nonempty, $t \stackrel{xz}{\to} \checkmark$, and either y is empty or $u \stackrel{y}{\to} \checkmark$ or $v \stackrel{y}{\to} \checkmark$. This means that either $t \rhd u \stackrel{w}{\to} \checkmark$ or $t \rhd v \stackrel{w}{\to} \checkmark$. $\square$

We can then state

Theorem 2.1 Let a be an action and let x, y, z be variables. The following equations are sound for BPA_{int} modulo complete trace equivalence:

$$
\begin{array}{lrcl}
\text{A1} & x + y & \approx & y + x \\
\text{A2} & x + (y + z) & \approx & (x + y) + z \\
\text{A3} & x + x & \approx & x \\
\text{A4.1} & (x + y)z & \approx & xz + yz \\
\text{A4.2} & x(y + z) & \approx & xy + xz \\
\text{A5} & (xy)z & \approx & x(yz) \\
\text{I1}.a & a \rhd y & \approx & a + ya \\
\text{I2}.a & ax \rhd y & \approx & a(x \rhd y) + yax \\
\text{I3.1} & (x + y) \rhd z & \approx & (x \rhd z) + (y \rhd z) \\
\text{I3.2} & x \rhd (y + z) & \approx & (x \rhd y) + (x \rhd z)
\end{array}
$$

Proof: Let σ be a closed substitution. Apply σ to both sides of any of the equations above: then left-hand and right-hand members are complete trace equivalent by Lemma 2.2. This is true for all closed substitutions, which proves the theorem. □

Observe that, for every action a, there is one equation of the form I1.a and one equation of the form I2.a, so that those equations are infinitely many if Act is infinite.

We now argue that the interrupt operator can be eliminated from closed terms. To be able to support our thesis, we do a little digression, and try to find the "simplest possible form" a BPA term can have, modulo complete trace equivalence.

Definition 2.1 A term t over BPA is in *prenex normal form* if there exists a finite nonempty set $W \subseteq (\mathsf{Act} \cup \mathit{Var})^+$ such that

$$t = \sum_{w \in W} w \ , \tag{5}$$

where the word $\alpha_1 \dots \alpha_n$ is identified with the term $\alpha_1 \cdot \ldots \cdot \alpha_n$.

In other words, a term is in prenex normal form if the nondeterministic choice operator only appears at the topmost level.

Lemma 2.3 Let t be a term over BPA. There exists a term u over BPA in prenex normal form such that $t \sim u$. Moreover, if t is closed, then u is closed as well.

Proof: By structural induction on t. The thesis is trivially true if either $t = a$ for some $a \in \mathsf{Act}$, or $t = x$ for some $x \in \mathit{Var}$.

If $t = t_1 + t_2$ for some terms t_1, t_2, consider u_1, u_2 in prenex normal form such that $t_1 \sim u_1$ and $t_2 \sim u_2$. Put

$$u = \sum_{w \in W} w \ ,$$

where W is the set of all words w that appear as summands in either u_1 or u_2. Observe that u can be constructed from $u_1 + u_2$ by repeatedly applying the idempotence rule A3. It is immediate to check that u is in prenex normal form, and that $t \sim u$; moreover, if t is closed, then t_1 and t_2 are closed, so that u_1 and u_2, and consequently u, are closed by inductive hypothesis.

If $t = t_1 t_2$ for some terms t_1, t_2, consider u_1, u_2 in prenex normal form such that $t_1 \sim u_1$ and $t_2 \sim u_2$. Put

$$u = \sum_{w \in W} w \ ,$$

where W is the set of all words w such that $w = w_1 w_2$ for two nonempty words w_1, w_2 such that w_1 is a summand of u_1 and w_2 is a summand of u_2. Observe that u can be constructed from $u_1 u_2$ by repeatedly applying the associativity laws A4.1 and A4.2, and the idempotence rule A3. It is straightforward to check that u is in prenex normal form, and that $t \sim u$; moreover, if t is closed, then t_1 and t_2 are closed, so that u_1 and u_2, and consequently u, are closed by inductive hypothesis. □

Lemma 2.3 states that, for every closed term t over BPA, there exists a closed term $\nu(t)$ over BPA in prenex normal form, such that $t \sim \nu(t)$. We call $\nu(t)$ the *prenex normal form* of the term t. Observe that $\nu(t)$ is defined up to the order of its summands. Observe also that, to construct u from t in the proof of Lemma 2.3, we have only applied associativity of operators, commutativity and idempotence of nondeterministic choice, and distributivity of nondeterministic choice w.r.t. composition: that is, $\nu(t)$ can be constructed *syntactically* from t by means of axioms in Table 2.

Introduction of prenex normal forms allows us to prove

Lemma 2.4 Let t and u be closed terms over BPA. There exists a closed term v over BPA such that $t \triangleright u \sim v$.

Proof: By induction on the size of t. Because of Lemma 2.3, it is not restrictive to suppose that t is in prenex normal form.

If t has size 1, then $t = a$ for some action a. Then $t \triangleright u = a \triangleright u \sim a + ua$.

Suppose now that the thesis is proved every time that t has at most size n. Let t have size $n + 1$. If $t = t_1 + t_2$, then $t \triangleright u \sim t_1 \triangleright u + t_2 \triangleright u$, with t_1 and

t_2 having size at most n: by inductive hypothesis, $t_1 \rhd u \sim v_1$ and $t_2 \rhd u \sim v_2$ for suitable closed terms v_1, v_2 over BPA, so that $t \rhd u \sim v_1 + v_2 = v$ with v closed term over BPA. Otherwise t has only one summand, so, since it is in prenex normal form, it must have the form $t = at'$ for some action a and closed term t' having size n: by inductive hypothesis, $t' \rhd u \sim v'$ for some closed term v' over BPA, so that $t \rhd u = at' \rhd u \sim a(t' \rhd u) + uat' \sim v$, with $v = av' + uat'$ being a closed term over BPA. □

In turn, Lemma 2.4 paves the way to

Theorem 2.2 Let t be a closed term over BPA_{int}. Then $t \sim u$ for some closed term u over BPA.

In other words: for closed terms over BPA modulo complete trace equivalence, interrupt is a derived operator.

Proof: By induction on the structure of t.

Case 1: $t = a$. This poses no problem: simply put $u = a$.

Case 2: $t = t_1 + t_2$. By inductive hypothesis, there exist closed terms u_1, u_2 over BPA such that $t_1 \sim u_1$ and $t_2 \sim u_2$. Then $u = u_1 + u_2$ is a closed term over BPA such that $t \sim u$.

Case 3: $t = t_1 t_2$. By inductive hypothesis, there exist closed terms u_1, u_2 over BPA such that $t_1 \sim u_1$ and $t_2 \sim u_2$. Then $u = u_1 u_2$ is a closed term over BPA such that $t \sim u$.

Case 4: $t = t_1 \rhd t_2$. By inductive hypothesis, there exist closed terms u_1, u_2 over BPA such that $t_1 \sim u_1$ and $t_2 \sim u_2$. By Lemma 2.4, there exists a closed term u over BPA such that $u_1 \rhd u_2 \sim u$. Then $t = t_1 \rhd t_2 \sim u_1 \rhd u_2 \sim u$. □

Observe that, to prove Lemma 2.4 (and thus Theorem 2.2 as well), we use only leftwise distributivity. This is interesting, because the interrupt operator is *not* associative modulo complete trace equivalence, so that we cannot regroup all of its instances on a single side. As a counterexample, let a be an action: then a^4 is a complete trace for $(a^3 \rhd a^2) \rhd a$, but not for $a^3 \rhd (a^2 \rhd a)$.

Since every closed term over BPA_{int} is c.t.e. to a closed term over BPA in light of Theorem 2.2, we can think of reducing the problem of finding a ground complete axiomatization for BPA_{int}, modulo c.t.e., to that of finding a ground complete axiomatization for BPA, modulo c.t.e. This would be allowed by prenex normal form, if they were *characterizing* for closed terms over BPA modulo complete trace equivalence, that is, if it were true that two closed terms over BPA having the same complete traces, also have the same prenex normal form.

And this is precisely the content of

Lemma 2.5 Let t and u be closed terms over BPA in prenex normal form. Then $t \sim u$ iff t and u have the same summands.

Proof: Suppose $t \sim u$. Let w be a summand of t: then $t \xrightarrow{w} \checkmark$ and $u \xrightarrow{w} \checkmark$ as well. Thus one of the summands w' of u satisfies $w' \xrightarrow{w} \checkmark$: but w' is a closed term without summands, so the only possibility is $w' = w$. This proves that every summand of t appears in u: by swapping the roles of t and u we find that they have the same summands.

The reverse implication is trivial. □

Theorem 2.3 The axioms in Table 2 form a ground complete axiomatization for BPA.

Proof: Let t and u be two closed terms over BPA such that $t \sim u$, and let $\nu(t)$ and $\nu(u)$ be their prenex normal forms. By using the axioms in Table 2 we can prove $t \approx \nu(t)$ and $u \approx \nu(u)$. But two terms in prenex normal form that are complete trace equivalent, are also equal up to the order of their summands: thus, by using the axioms in Table 2 we can also prove $\nu(t) \approx \nu(u)$. This, in turn, allows us to prove $t \approx u$. □

As a consequence of this fact, we obtain the main result of this section.

Theorem 2.4 If $|\mathsf{Act}| < \infty$, then $\mathrm{BPA_{int}}$ has a finite ground complete axiomatization modulo complete trace equivalence.

Proof: Consider the family E of equations from Theorem 2.1: if $|\mathsf{Act}| = n$, then $|E| = 2n + 8$. Let t and u be closed terms over $\mathrm{BPA_{int}}$ such that $t \sim u$: we must show that $E \vdash t \approx u$.

As seen in Theorem 2.2, the equations in E allow us to reduce any closed term over $\mathrm{BPA_{int}}$ to a closed term over BPA: in particular, there exist closed terms t', u' over BPA such that $E \vdash t \approx t'$ and $E \vdash u \approx u'$. By the soundness of E, $t' \sim u'$: since the equations in Table 2 also appear in Theorem 2.1, from Theorem 2.3 we deduce $E \vdash t' \approx u'$. Thus $E \vdash t \approx u$ as well. □

3 The case of general terms

Having proved finite complete axiomatizability for c.t.e. over closed terms in the language $\mathrm{BPA_{int}}$, we want to obtain a similar result for general terms. However, the technique we used to prove Theorem 2.4 does not work in the broader case, because, as we announced in Subsection 1.1, interrupt is not a derived operator, except for a very special case.

Proposition 3.1 Let x and y be variables. Then there exists a term t over BPA such that $t \sim x \rhd y$ if and only if $|\mathsf{Act}| = 1$.

Proof: If $\mathsf{Act} = \{a\}$, then by Lemma 2.3 and Theorem 2.2 the only closed substitutions are, *up to complete trace equivalence*, those of the form

$$\sigma(z) = \sum_{k \in K} a^k \;, \tag{6}$$

where K is a nonempty finite set of positive integers. Therefore, if $|\mathsf{Act}| = 1$, then $x \rhd y \sim x + yx$. In fact, let

$$\sigma(x) = \sum_{i \in I} a^i \text{ and } \sigma(y) = \sum_{j \in J} a^j \;;$$

then

$$\sigma(x + yx) = \sum_{i \in I} a^i + \sum_{j \in J} a^j \sum_{i \in I} a^i \sim \sum_{i \in I} a^i + \sum_{i \in I, j \in J} a^{j+i}$$

and

$$\sigma(x \rhd y) = \left(\sum_{i \in I} a^i\right) \rhd \left(\sum_{j \in J} a^j\right) \sim \sum_{i \in I, j \in J} a^i \rhd a^j$$

both have as complete traces precisely the words of the form a^i for $i \in I$, and those of the form a^{j+i} for $j \in J$ and $i \in I$.

If $\mathsf{Act} = \{a, b, \ldots\}$, then we prove that no BPA term is c.t.e. to $x \rhd y$; closed substitutions of the form (1) will play a key role. Assume, towards a contradiction, that $x \rhd y \sim t$ for some term t over BPA. By complete trace equivalence, $\sigma_a(t) \xrightarrow{a} \checkmark$, which, by Lemma 1.1, is possible if and only if either $t \xrightarrow{a} \checkmark$, or there exists a variable z such that z is a summand of t and $\sigma_a(z) \xrightarrow{a} \checkmark$. But the latter is the only possibility, because if $t \xrightarrow{a} \checkmark$, then $\sigma_a[x \mapsto a^2](t) \xrightarrow{a} \checkmark$ as well, while $\sigma_a[x \mapsto a^2](x \rhd y) = a^2 \rhd a$ has norm 2, contradicting our assumption that t is c.t.e. to $x \rhd y$; moreover, it cannot be just $t = z$, or $\sigma_a[y \mapsto b](z)$ would have ba as a complete trace, which is impossible. So $t = z + u$ for some term u over BPA; if it were $z \neq x$, we would get $\sigma_a[x \mapsto a^2](t) = a + \sigma_a[x \mapsto a^2](u) \xrightarrow{a} \checkmark$, which we know to be a contradiction. Thus, if $t \sim x \rhd y$, then necessarily $t \sim x + u$ for some term u over BPA; it is not restrictive to suppose that u is in prenex normal form, and that the summand x does not occur in u.

We observe that u cannot contain actions. In fact, should u contain action a, let $b \in \mathsf{Act} \setminus \{a\}$: then $\sigma_b(x + u)$ has a complete trace containing a and $\sigma_b(x \rhd y)$ does not, contradicting our assumption that t is c.t.e. to $x \rhd y$. Moreover, u cannot contain variables other than x and y: otherwise, if $\sigma = \sigma_a[x \mapsto b, y \mapsto b]$, then $\sigma(x+u)$ and $\sigma(x \rhd y)$ would yield a similar contradiction.

If $x = y$, then all the summands of u have the form x^n for some $n > 1$. Let then $\sigma(x) = ab$; it follows that $a^2b^2 = a(ab)b$ is a complete trace for $\sigma(x \rhd x) = ab \rhd ab$, but not for $\sigma(x+u)$. This is a contradiction.

If $x \neq y$, then u must contain both x and y. In fact, consider the closed substitution $\sigma_{N,K}$ given by

$$\sigma_{N,K}(x) = a^N \ ; \ \sigma_{N,K}(y) = b^K \ ; \ \sigma(z) = a \ \forall z \notin \{x, y\} \ .$$

Since $x + u \sim x \rhd y$, $b^K a^N$ is a complete trace for $\sigma_{N,K}(x+u)$ for all N and K, which is impossible if either x or y does not occur in u. Thus u is actually a sum of nonempty words over the alphabet $\{x, y\}$; since the only complete traces of $\sigma_a[y \mapsto b](x+u)$ must be a and ba, none of these words can contain xx, xy, or yy as a subword, plus y cannot be a summand of u. Then the only possibility is $u = yx$: however, aba is a complete trace for $\sigma_b[x \mapsto a^2](x \rhd y) = a^2 \rhd b$, but not for $\sigma_b[x \mapsto a^2](x + yx) = a^2 + ba^2$. This is a contradiction as well. □

Proposition 3.1 puts an end to our hopes of finding an easy solution to the finite axiomatization problem for general terms over BPA_{int}; at the same time, however, it opens the way to such a solution in a special case. To better understand the possibilities left, and possibly use an approach based on normal forms for the special case, we need a deeper insight on the properties of prenex normal forms.

We start by observing that, if t is a term over BPA and σ is a closed substitution, then the prenex normal form of $\sigma(t)$, say $\sum_{j \in J} t_j$, is a sum of objects that can be seen both as closed terms over BPA and as words over actions, such that the word is the only complete trace for the term. It follows that, if t and u are terms over BPA such that $t \sim u$, then the "shape" of $\nu(\sigma(t))$ and $\nu(\sigma(u))$ must be the same for every closed substitution σ. The most natural thing to do is to ask oneself whether this "equality of shape" must be true for the terms themselves.

We recall that the *length* of a word w over an alphabet A is the number $|w|$ of characters (i.e., elements of A) occurring in w, while the *number of occurrences* of character a in word w is the number $|w|_a$ of characters in w equal to a. Of course, $|w| = \sum_{a \in A} |w|_a$, and if $w_1 = w_2$, then $|w_1|_a = |w_2|_a$ for every $a \in A$.

Proposition 3.2 Let t and u be nonempty words over $\mathsf{Act} \cup \mathit{Var}$.

1. If $|\mathsf{Act}| > 1$ then $t \sim u$ iff $t = u$.

2. If $\mathsf{Act} = \{a\}$ then $t \sim u$ iff t is a permutation of u.

Proof: First of all, we recall that both points are true for *closed* terms. This fact will be used later on in the proof. Also, substitutions of the form (1) will play a key role.

We now prove point 1 for general terms. Of course, only the "only if" part needs to be proved. Suppose $t \neq u$: then either $|t| \neq |u|$, or $t = \lambda_1 \alpha \lambda_2$, $u = \lambda_1 \beta \lambda_3$, with $\lambda_1, \lambda_2, \lambda_3 \in (\mathsf{Act} \cup \mathit{Var})^*$, $\alpha, \beta \in \mathsf{Act} \cup \mathit{Var}$, $\alpha \neq \beta$. In the former case, $\sigma_a(t)$ and $\sigma_a(u)$ are closed words of different length. In the latter, if one between α and β is action a, and $b \in \mathsf{Act} \setminus \{a\}$, then $\sigma_b(t) \neq \sigma_b(u)$; otherwise, α and β are distinct variables, thus $\sigma_a[\beta \mapsto b](t) \neq \sigma_a[\beta \mapsto b](u)$. Therefore, if $t \neq u$, then there exists a closed substitution σ such that $\sigma(t) \not\sim \sigma(u)$, so that $t \not\sim u$.

We are now left with the proof of point 2 for general terms. Let σ be a closed substitution; let t a term with a single summand, let σ be a closed substitution, and let $w_1, w_2, \ldots, w_n \in \mathsf{Act} \cup \mathit{Var}$ such that

$$t = w_1 w_2 \ldots w_n \ .$$

By Theorem 2.2 and Lemma 2.3, for all $j \in \{1, 2, \ldots, n\}$ there exists a finite set I_j of integers such that

$$\sigma(w_j) \sim \sum_{i_j \in I_j} a^{i_j}$$

Then, since complete trace equivalence is a congruence and distributivity laws apply modulo complete trace equivalence,

$$\begin{aligned}
\sigma(w) &\sim \sigma(w_1)\sigma(w_2)\ldots\sigma(w_n) \\
&\sim \left(\sum_{i_1 \in I_1} a^{i_1}\right)\left(\sum_{i_2 \in I_2} a^{i_2}\right)\cdots\left(\sum_{i_n \in I_n} a^{i_n}\right) \\
&\sim \sum_{i_1 \in I_1, i_2 \in I_2, \ldots, i_n \in I_n} a^{i_1} a^{i_2} \ldots a^{i_n} \\
&= \sum_{i_1 \in I_1, i_2 \in I_2, \ldots, i_n \in I_n} a^{i_1 + i_2 + \ldots + i_n}
\end{aligned}$$

which depends on the nature of the w_j's, but not on their order. Thus, if t is a permutation of u, then $\sigma(t) \sim \sigma(u)$ whatever the closed substitution σ, i.e., $t \sim u$. On the other hand, if t is not a permutation of u, then either $|t| \neq |u|$ or there is a variable x such that $|t|_x \neq |u|_x$, so that either $\sigma_a(t) \neq \sigma_a(u)$ or $\sigma_a[x \mapsto a^2](t) \neq \sigma_a[x \mapsto a^2](u)$; and again, $t \not\sim u$. □

Proposition 3.2 suggests a strategy for finding an axiomatization for the terms over BPA_{int} when $\mathsf{Act} = \{a\}$.

Consider an ordering $\mathsf{Act} \cup \mathit{Var} = \{a, x_1, x_2, \ldots, x_n, \ldots\}$. Let w be a word over $\mathsf{Act} \cup \mathit{Var}$ and let n be the maximum index of a variable occurring in w. We define the *normal form* of w as

$$\nu(w) = a^{|w|_a} x_1^{|w|_{x_1}} x_2^{|w|_{x_2}} \ldots x_n^{|w|_{x_n}} \ . \tag{7}$$

By Proposition 3.2, $w \sim \nu(w)$.

Let now t be a term over BPA_{int}: by Proposition 3.1, t is complete trace equivalent to a term t' over BPA, which, in turn, has a prenex normal form $\sum_{w \in W} w$. We can therefore say that the *normal form* of the term t is

$$\nu(t) = \sum_{w \in W} \nu(w) \ , \tag{8}$$

where each summand in normal form is taken once per occurrence. For instance, the normal form of $t = ya + xax + a \rhd x$ is $\nu(t) = a + ax + ax^2 + ay$, while that of $xy + yx$ is xy. We observe that $\nu(t)$ is unique, up to the order of summands, and that $t \sim \nu(t)$.

Theorem 3.1 Suppose $\mathsf{Act} = \{a\}$. Then two terms u, v over BPA_{int} are complete trace equivalent if and only if $\nu(t) = \nu(u)$ up to the order of summands.

Proof: Let t and u be two terms over BPA_{int} such that $t \sim u$, and let

$$\nu(t) = \sum_{i=1}^{r} p_i \text{ and } \nu(u) = \sum_{j=1}^{s} q_j$$

be their normal forms. Let N be the maximum index of a variable in either the p_i's or the q_j's; then for each i and j we can write

$$p_i = a^{e_{0,i}} x_1^{e_{1,i}} \ldots x_N^{e_{N,i}} \text{ and } q_j = a^{f_{0,j}} x_1^{f_{1,j}} \ldots x_N^{f_{N,j}} \ .$$

Let b be a positive integer greater than all of the $e_{k,i}$'s and the $f_{k,j}$'s; consider the substitution σ defined by

$$\sigma(x_k) = a^{b^k} \quad \forall k \in \mathbb{N} \ . \tag{9}$$

Then, for all i and j, $\sigma(p_i) = a^{\alpha_i}$ and $\sigma(q_j) = a^{\beta_j}$, where

$$\alpha_i = \sum_{k=0}^{N} e_{k,i} b^k \text{ and } \beta_j = \sum_{k=0}^{N} f_{k,j} b^k \ ,$$

I0.1	$x \rhd y$	$\approx$	$x \rhd y + x$
I0.2	$x \rhd y$	$\approx$	$x \rhd y + yx$
I2	$xy \rhd z$	$\approx$	$x(y \rhd z) + (x \rhd z)y$
I4	$(x \rhd y) \rhd z$	$\approx$	$(x \rhd y) \rhd z + x \rhd (y \rhd z)$
I5	$(x \rhd y) \rhd z + x \rhd (z \rhd y)$	$\approx$	$(x \rhd z) \rhd y + x \rhd (y \rhd z)$

Table 3: A list of valid equations for BPA_{int}.

and since b is larger than all of the $e_{k,i}$'s and the $f_{k,j}$'s, the α_i's are pairwise distinct, and so are the β_j's.

Since $t \sim u$, we have $\nu(t) \sim \nu(u)$ as well, so $\sigma(\nu(t)) \sim \sigma(\nu(u))$. But a word $w = a^K$ is a complete trace for $\sigma(\nu(t))$ iff $K = \alpha_i$ for some i, and similarly, w is a complete trace for $\sigma(\nu(u))$ iff $K = \beta_j$ for some j: thus, for every i there must exist a j such that $\alpha_i = \beta_j$, and vice versa. This, in turn, is only possible if for every i there exists j such that $p_i = q_j$, and vice versa; since the p_i's are summands in a normal form, and so are the q_j's, we conclude that $r = s$ and the p_i's are a permutation of the q_j's, that is, $\nu(t) = \nu(u)$ up to the order of summands.

The reverse implication is trivial. □

Theorem 3.2 If $|\mathsf{Act}| = 1$ then BPA_{int} is finitely axiomatizable.

Proof: Consider the set E consisting of the axioms of Theorem 2.1 together with

$$\begin{array}{lrcl} \text{CC} & xy & \approx & yx \\ \text{DI} & x \rhd y & \approx & x + yx \end{array}$$

(Observe that CC and DI are sound modulo complete trace equivalence iff $|\mathsf{Act}| = 1$.) Let t and u be terms over BPA_{int} such that $t \sim u$: we must prove that $E \vdash t \approx u$.

Consider the normal forms $\nu(t)$ and $\nu(u)$. Using equations CC and DI, it is not hard to prove that $E \vdash t \approx \nu(t)$ and $E \vdash u \approx \nu(u)$. But the normal forms of two terms over BPA that are equivalent modulo complete trace equivalence are equal by Theorem 3.1; thus, $E \vdash \nu(t) \approx \nu(u)$. This allows us to conclude that $E \vdash t \approx u$. □

4 Other valid equations

In this section, we will list some equations over BPA_{int}, and prove that they are all valid; plus, we will suggest a kind of "normal forms" for BPA_{int} terms. We use double quotes, because these, as we shall see, are not characterizing.

A list of valid equations for $\mathrm{BPA}_{\mathrm{int}}$ is given in Table 3. We immediately observe that $\mathrm{I0.1}$ and $\mathrm{I0.2}$ are valid indeed, because, whatever t and u are, any complete trace for t or ut is also a complete trace for $t \rhd u$

Proposition 4.1 Equation $\mathrm{I2}$ in Table 3 is sound modulo complete trace equivalence.

Proof: We must show that, for every word w over the alphabet Act and every closed terms t, u, v over $\mathrm{BPA}_{\mathrm{int}}$, $tu \rhd v \xrightarrow{w} \checkmark$ if and only if $t(u \rhd v) + (t \rhd v)u \xrightarrow{w} \checkmark$.

Suppose $tu \rhd v \xrightarrow{w} \checkmark$: this is the same as saying that $w = xyz$ with z nonempty, $tu \xrightarrow{xz} \checkmark$, and either y is empty or $v \xrightarrow{y} \checkmark$. If $t \xrightarrow{x} \checkmark$ and $u \xrightarrow{z} \checkmark$, then $t(u \rhd v) \xrightarrow{w} \checkmark$; otherwise, either $x = x'p$ with $t \xrightarrow{x'} \checkmark$, or $z = qz'$ with $t \xrightarrow{xq} \checkmark$, where p and q are suitable nonempty words. In the former case, $u \rhd v \xrightarrow{pyz} \checkmark$, so that $t(u \rhd v) \xrightarrow{x'pyz} \checkmark$ and $x'pyz = w$; in the latter case, $u \xrightarrow{z'} \checkmark$ and $t \rhd v \xrightarrow{xyq} \checkmark$, so that $(t \rhd v)u \xrightarrow{xyqz'} \checkmark$ and $xyqz' = w$.

On the other hand, suppose $t(u \rhd v) + (t \rhd v)u \xrightarrow{w} \checkmark$: then either $t(u \rhd v) \xrightarrow{w} \checkmark$ or $(t \rhd v)u \xrightarrow{w} \checkmark$. In the first case, $w = rxyz$ with z nonempty, $t \xrightarrow{r} \checkmark$, $u \xrightarrow{xz} \checkmark$, and either y is empty or $v \xrightarrow{y} \checkmark$. Then $tu \xrightarrow{rxz} \checkmark$ and $tu \rhd v \xrightarrow{rxyz} \checkmark$; but $rxyz = w$. In the second case, $w = xyzs$ with z nonempty, $t \xrightarrow{xz} \checkmark$, $u \xrightarrow{s} \checkmark$, and either y is empty or $v \xrightarrow{y} \checkmark$. Then $tu \xrightarrow{xzs} \checkmark$ and $tu \rhd v \xrightarrow{xyzs} \checkmark$; but $xyzs = w$. $\square$

Observe how equation $\mathrm{I2}$ generalizes $\mathrm{I2}.a$ to the case of a general concatenation of terms. In fact,

$$\begin{array}{rcll} at \rhd u & \approx & a(t \rhd u) + (a \rhd u)t & \text{from I2}\,, \\ & \approx & a(t \rhd u) + (a + ua)t & \text{from I1}.a\,, \\ & \approx & a(t \rhd u) + at + uat & \text{from A4.1}\,, \\ & \approx & a(t \rhd u) + uat & \text{from I0.1 and A4.2}\,. \end{array}$$

Proposition 4.2 Equation $\mathrm{I4}$ in Table 3 is sound modulo complete trace equivalence.

Proof: We must show that, for every word w over the alphabet Act and every closed terms t, u, v over $\mathrm{BPA}_{\mathrm{int}}$, if $t \rhd (u \rhd v) \xrightarrow{w} \checkmark$, then $(t \rhd u) \rhd v \xrightarrow{w} \checkmark$.

Let $t \rhd (u \rhd v) \xrightarrow{w} \checkmark$: then $w = xyz$ with z nonempty, $t \xrightarrow{xz} \checkmark$, and either y is empty or $u \rhd v \xrightarrow{y} \checkmark$. If y is empty, then $w = xz$ is a complete trace for $(t \rhd u) \rhd v$; otherwise, $y = pqr$ with $u \xrightarrow{pr} \checkmark$ and either q is empty or $v \xrightarrow{q} \checkmark$. Let then $x' = xp$, $y' = q$, $z' = rz$: then $t \rhd u \xrightarrow{x'z'} \checkmark$ and either y' is empty or $v \xrightarrow{y'} \checkmark$, thus $(t \rhd u) \rhd v \xrightarrow{x'y'z'} \checkmark$. But $x'y'z' = xpqrz = xyz = w$. $\square$

Equation I4 says that $t \rhd (u \rhd v)$ is somewhat "less capable", in terms of "possible terminating executions", than $(t \rhd u) \rhd v$, something we had already seen after Theorem 2.2. The question arises naturally: *how much* is this "less"? Equation I5 provides a possible answer to this question.

Theorem 4.1 Equation I5 in Table 3 is sound modulo complete trace equivalence.

Proof: Suppose $(t \rhd u) \rhd v \xrightarrow{w} \checkmark$. We know from Lemma 2.1 that $w = xyz$ with $t \rhd u \xrightarrow{xz} \checkmark$ and either y is empty or $v \xrightarrow{y} \checkmark$. From the same lemma we get $xz = pqr$ with $t \xrightarrow{pr} \checkmark$ and either q is empty or $u \xrightarrow{q} \checkmark$. Thus four cases must be studied.

Case 1: y and q are both empty. In this case, the transition is entirely due to t, thus $t \rhd (u \rhd v) \xrightarrow{w} \checkmark$.

Case 2: y is empty and q is not. In this case, there is a transition $t \xrightarrow{p} t'$, followed by $u \xrightarrow{q} \checkmark$, then by $t' \xrightarrow{r} \checkmark$; plus, $w = xz = pqr$. This can be mimicked by $t \rhd (u \rhd v)$ as follows:

$$t \rhd (u \rhd v) \xrightarrow{p} t' \rhd (u \rhd v) \xrightarrow{q} t' \xrightarrow{r} \checkmark \ ,$$

because if $u \xrightarrow{q} \checkmark$ then $u \rhd v \xrightarrow{q} \checkmark$ as well.

Case 3: y is nonempty and q is empty. In this case, the transition is of the kind

$$(t \rhd u) \rhd v \xrightarrow{x} (t' \rhd u) \rhd v \xrightarrow{y} t' \rhd u \xrightarrow{z} \checkmark \ ,$$

This cannot in general be mimicked by $t \rhd (u \rhd v)$, because

$$t \rhd (u \rhd v) \xrightarrow{x} t' \rhd (u \rhd v) \xrightarrow{y} ut \, ,$$

and ut might not have z as a complete trace. However, it can be mimicked by

$$t \rhd v \xrightarrow{x} t' \rhd v \xrightarrow{y} t' \xrightarrow{z} \checkmark \ ,$$

which is tolerable, because if $t \rhd v \xrightarrow{w} \checkmark$, then $(t \rhd u) \rhd v \xrightarrow{w} \checkmark$ as well.

Case 4: y and q are both nonempty. This is the most complicated case, so we split it into subcases.

Subcase 4a: $x = pq'$, $z = q''r$. This means that the execution of t is interrupted by that of u, which is in turn interrupted by that of v; that is, for some t', u',

$$(t \rhd u) \rhd v \xrightarrow{p} (t' \rhd u) \rhd v \xrightarrow{q'} u't' \rhd v \xrightarrow{y} u't' \xrightarrow{q''} t' \xrightarrow{r} \checkmark \ .$$

This can be mimicked by

$$t \rhd (u \rhd v) \xrightarrow{p} t' \rhd (u \rhd v) \xrightarrow{q'} (u' \rhd v)t' \xrightarrow{y} u't' \xrightarrow{q''} t' \xrightarrow{r} \checkmark \ .$$

Subcase 4b: $x = pqr'$, $z = r''$. This means that the execution of t is first interrupted by that of u, then resumed, then suspended by that of v; that is,

$$(t \rhd u) \rhd v \xrightarrow{p} (t' \rhd u) \rhd v \xrightarrow{q} t' \rhd v \xrightarrow{r'} t'' \rhd v \xrightarrow{y} t'' \xrightarrow{r''} \checkmark\ .$$

This cannot in general be mimicked by $t \rhd (u \rhd v)$, because

$$t \rhd (u \rhd v) \xrightarrow{p} t' \rhd (u \rhd v) \xrightarrow{q} t'\ ,$$

and t' might not have $r'yr''$ as a complete trace.

Subcase 4c: $x = p'$, $z = p''qr$. This means that the execution of t is first interrupted by that of v, then resumed, then suspended by that of u; that is,

$$(t \rhd u) \rhd v \xrightarrow{p'} (t' \rhd u) \rhd v \xrightarrow{y} t' \rhd u \xrightarrow{p''} t'' \rhd u \xrightarrow{q} t'' \xrightarrow{r} \checkmark\ .$$

This cannot in general be mimicked by $t \rhd (u \rhd v)$, because

$$t \rhd (u \rhd v) \xrightarrow{p'} t' \rhd (u \rhd v) \xrightarrow{y} ut'\ ,$$

and ut' might not have $p''qr$ as a complete trace.

The problems come from subcases 4b and 4c. In fact, in $(t \rhd u) \rhd v$, the execution of t can either be first suspended by u, then resumed, then suspended by v; or be first suspended by v, then resumed, then suspended by u. On the contrary, in $t \rhd (u \rhd v)$, if u interrupts t, then v can only interrupt u, and during this process, the execution of t cannot be resumed; while if v interrupts t, then t cannot be resumed until first v, then u are finished.

However, this behaviour can be mimicked by

$$(t \rhd v) \rhd u$$

by means of

$$(t \rhd v) \rhd u \xrightarrow{p} (t' \rhd v) \rhd u \xrightarrow{q} t' \rhd v \xrightarrow{r'} t'' \rhd v \xrightarrow{y} t'' \xrightarrow{r''} \checkmark$$

for case 4b, and

$$(t \rhd v) \rhd u \xrightarrow{p'} (t' \rhd v) \rhd u \xrightarrow{y} t' \rhd u \xrightarrow{p''} t'' \rhd u \xrightarrow{q} t'' \xrightarrow{r} \checkmark$$

for case 4c; plus, it also works in case 3. This means that we can write down

$$(t \rhd u) \rhd v \preccurlyeq (t \rhd v) \rhd u + t \rhd (u \rhd v)$$

where $A \preccurlyeq B$ is a shortcut for $CT(A) \subseteq CT(B)$. Moreover, by applying I4 to $t \rhd (v \rhd u)$ and $(t \rhd v) \rhd u$, we can refine this inequality into

$$(t \rhd u) \rhd v + t \rhd (v \rhd u) \preccurlyeq (t \rhd v) \rhd u + t \rhd (u \rhd v)$$

But the roles of u an v are symmetrical on either side, thus we can swap these two terms and get the reverse inequality. □

Since, in general, $(t \rhd u) \rhd v$ and $t \rhd (u \rhd v)$ are not c.t.e., they should both be considered when looking for normal forms. For a term t containing a single summand, let us introduce a notion of *leftmost term*:

1. if the interrupt operator does not occur in t, then t is its own leftmost term;
2. if $t = u \rhd v$, then the leftmost term of t is that of u.

For instance, the leftmost term of $x \rhd y$, $x \rhd (y \rhd z)$, and $(x \rhd y) \rhd z$, is always x, while the leftmost term of $xy \rhd z$ is xy.

Lemma 4.1 Every BPA_{int} term t can be written, modulo complete trace equivalence, as a sum of concatenations of singletons and sequences of interrupts where, in every subsequence, the leftmost term is a variable.

Proof: By induction on the structure of t. The cases $t = a$, $t = x$, and $t = u + v$ with the thesis holding for both u and v obviously pose no problem.

Suppose $t = uv$ with the thesis holding for v and v. Then

$$u \approx \sum_{i \in [1 \ldots n]} \prod_{j \in [1 \ldots n_i]} u_{i,j}$$

and

$$v \approx \sum_{r \in [1 \ldots m]} \prod_{s \in [1 \ldots m_r]} v_{r,s}$$

with each $u_{i,j}$ and $v_{r,s}$ being either a singleton, or a sequence of interrupts where, in each subsequence, the leftmost term is a variable. Since concatenation is both left- and right-distributive modulo c.t.e. w.r.t. nondeterministic choice,

$$uv \approx \sum_{\substack{i \in [1 \ldots n] \\ r \in [1 \ldots m]}} \left(\prod_{j \in [1 \ldots n_i]} u_{i,j} \right) \left(\prod_{s \in [1 \ldots s_r]} v_{r,s} \right)$$

with each factor in each summand being either a singleton, or a sequence of interrupts where, in each subsequence, the leftmost term is a variable.

Suppose finally $t = u \rhd v$ with the thesis holding for u and v. Write u and v as in previous case, and put

$$v_r = \prod_{s \in [1 \dots m_r]} v_{r,s} \, .$$

Since interrupt is both left- and right-distributive modulo c.t.e. w.r.t. nondeterministic choice,

$$u \rhd v \approx \sum_{\substack{i \in [1 \dots n] \\ r \in [1 \dots m]}} \left(\left(\prod_{j \in [1 \dots n_i]} u_{i,j} \right) \rhd v_r \right) \, .$$

But by repeatedly applying I2 we find

$$\left(\prod_{j \in [1 \dots n_i]} u_{i,j} \right) \rhd v_r \approx \sum_{j \in [1 \dots n_i]} u_{i,1} \dots u_{i,j-1} \left(u_{i,j} \rhd v_r \right) u_{i,j+1} \dots u_{i,n_i}$$

where each factor in any of the right hand summands is either a singleton or a sequence of interrupts where, in each subsequence, the leftmost term is surely a singleton, but possibly not a variable. However:

- if $u_{i,j}$ is a variable, then the thesis is still satisfied;
- if $u_{i,j}$ is a sequence of interrupts, then again the thesis is satisfied by inductive hypothesis;
- finally, if $u_{i,j}$ is a constant, then $u_{i,j} \rhd v_r = u_{i,j} + v_r u_{i,j}$, and the thesis keeps on being satisfied, again by inductive hypothesis.

From this the thesis follows. □

The terms in the thesis of Lemma 4.1 are tentative candidates as normal forms for $\mathrm{BPA}_{\mathrm{int}}$ terms. However, from that point of view, $x \rhd y$ and $x \rhd y + yx + x$ are different forms; but from what we have seen until now it is obvious that they are complete trace equivalent, so that these forms are unfortunately not characterizing. To solve these problem, a theory of "reduction" of terms should most probably be developed. At the time of the writing, we do not know whether the collection of equations in Theorems 2.1 and 3 is complete for complete trace equivalence over $\mathrm{BPA}_{\mathrm{int}}$.

A thing to do is probably to study the terms of the form $x \rhd t$, which, because of Lemma 4.1, seem to be good candidates for basic constituents of normal forms. The idea should be that, if $x \rhd t \sim y \rhd u$, then $x = y$ and $t \sim u$.

Let t and u be terms over $\mathrm{BPA}_{\mathrm{int}}$, not necessarily closed; let x and y be variables. Suppose $x \rhd t \sim y \rhd u$: then $x = y$, otherwise, if $\sigma(x) = a$ and $\sigma(y) = b$, then $\sigma(x \rhd t) \xrightarrow{a} \checkmark$ and $\sigma(y \rhd u) \not\xrightarrow{a} \checkmark$. Moreover, the maximum length of a complete trace for $\sigma(t)$ and $\sigma(u)$ must be the same: otherwise, by joining two complete traces of maximum length for either $\sigma(t)$ or $\sigma(u)$ and $\sigma(x)$, we would get a complete trace for either $\sigma(x \rhd t)$ or $\sigma(x \rhd u)$, but not both. Lastly, t and u contain the same variables: otherwise, if N is larger than the length of any summand in t and u, then by substituting with b^N a single variable that does not appear in both t and u, and with a all other variables, we get two non-c.t.e. closed terms.

We are now left with the task of checking whether $t \sim u$. This is not immediate, because, in general, $t \rhd u \sim t \rhd v$ does not imply $u \sim v$: as a counterexample, put $t = u = a + a^2$ and $v = a^2$.

Suppose x does not occur in t (and u). Suppose there exists a closed substitution σ such that $\sigma(t)$ and $\sigma(u)$ are not complete trace equivalent. Then $\sigma[x \mapsto a](x \rhd t) \approx a + \sigma(t)a$ and $\sigma[x \mapsto a](x \rhd u) \approx a + \sigma(u)a$ are not complete trace equivalent, because if (for example) w is a complete trace for $\sigma(t)$ and not for $\sigma(u)$, then wa is a complete trace for $a + \sigma(t)a$ and not for $a + \sigma(u)a$.

Suppose now x does occur in t (and u). Again, suppose there exists a closed substitution σ such that $\sigma(t)$ and $\sigma(u)$ are not complete trace equivalent; in particular, let $w \in \mathsf{Act}^+$ be a complete trace for $\sigma(t)$ and not for $\sigma(u)$. How can we prove that there exists a closed substitution σ' such that $\sigma'(x \rhd t)$ and $\sigma'(x \rhd u)$ are not complete trace equivalent?

We can reformulate our problem in terms of language theory. Call X the language of complete traces of $\sigma(x)$, Y that of $\sigma(t)$, and Z that of $\sigma(u)$: then X, Y, and Z are finite languages. Saying that $\sigma(x \rhd t) \sim \sigma(x \rhd u)$, is then equivalent to saying that

$$X \cup \bigcup_{rs \in X, s \neq \varepsilon} rYs = X \cup \bigcup_{rs \in X, s \neq \varepsilon} rZs \tag{10}$$

for all X. We can then state

Conjecture 1 *Suppose that, for every finite language X, equation (10) has a solution (Y, Z) with Y and Z finite. Then, for every finite language X, there is only one such solution.*

If Conjecture 1 is true, then $\sigma(x \rhd t) \sim \sigma(x \rhd u)$ for all σ implies $\sigma(t) = \sigma(u)$ for all σ.

References

[1] L. ACETO, W. FOKKINK, R. VAN GLABBEEK, AND A. INGOLFSDOTTIR, *Nested semantics over finite trees are equationally hard*, Information and Computation, 191 (2004), pp. 203–232.

[2] L. ACETO, W. FOKKINK, AND A. INGOLFSDOTTIR, *A menagerie of non-finitely based process semantics over BPA*—from ready simulation to completed traces*, Mathematical Structures in Computer Science, 8 (1998), pp. 193–230.

[3] L. ACETO, W. FOKKINK, A. INGOLFSDOTTIR, AND B. LUTTIK, *CCS with Hennessy's merge has no finite equational axiomatization*, Theoretical Comput. Sci., 330 (2005), pp. 377–405.

[4] L. ACETO, W. FOKKINK, A. INGOLFSDOTTIR, AND B. LUTTIK, *Finite Equational Bases in Process Algebra: Results and Open Questions*, in Processes, Terms, and Cycles: Steps on the Road to Infinity: Essays Dedicated to Jan Willem Klop on the Occasion of His 60th Birthday A. Middeldorp, V. van Oostrom, F. Raamsdonk, R. Vrijer eds. vol. 3838 of Lecture Notes in Computer Science, Springer Verlag (2005), pp. 338–367.

[5] L. ACETO, W. FOKKINK, A. INGOLFSDOTTIR, AND S. NAIN, *Bisimilarity is not Finitely Based over BPA with Interrupt*, Theoretical Computer Science, 366(1–2):60–81, 2006. An extended abstract of this paper appeared in J. Fiadeiro, N. Harman, M. Roggenbach, and J. Rutten, eds., Proc. 1st Conference on Algebra and Coalgebra in Computer Science (CALCO'05), vol. 3629 of Lecture Notes in Computer Science, Springer Verlag (2005), pp. 52–66.

[6] J. C. BAETEN AND J. BERGSTRA, *Mode transfer in process algebra*, Report CSR 00–01, Technische Universiteit Eindhoven, 2000. This paper is an expanded and revised version of [9].

[7] J. C. BAETEN, J. BERGSTRA, AND J. W. KLOP, *Syntax and defining equations for an interrupt mechanism in process algebra*, Fundamenta Informaticae, IX (1986), pp. 127–168.

[8] ——, *Decidability of bisimulation equivalence for processes generating context-free languages*, J. Assoc. Comput. Mach., 40 (1993), pp. 653–682.

[9] J. BERGSTRA, *A mode transfer operator in process algebra*, Report P8808, Programming Research Group, University of Amsterdam, 1988.

[10] J. BERGSTRA AND J. W. KLOP, *Fixed point semantics in process algebras*, Report IW 206, Mathematisch Centrum, Amsterdam, 1982.

[11] ISO, *Information processing systems – open systems interconnection – LOTOS – a formal description technique based on the temporal ordering of observational behaviour* ISO/TC97/SC21/N DIS8807, 1987.

[12] S. MAUW, *PSF – A Process Specification Formalism*, PhD thesis, University of Amsterdam, Dec. 1991.

[13] R. MILNER, *Communication and Concurrency*, Prentice-Hall International, Englewood Cliffs, 1989.

[14] R. MILNER, M. TOFTE, R. HARPER, AND D. MACQUEEN, *The Definition of Standard ML (Revised)*, MIT Press, 1997.

[15] F. MOLLER, *The importance of the left merge operator in process algebras*, in Proceedings 17^{th} ICALP, Warwick, M. Paterson, ed., vol. 443 of Lecture Notes in Computer Science, Springer-Verlag, July 1990, pp. 752–764.

[16] ——, *The nonexistence of finite axiomatisations for CCS congruences*, in Proceedings 5^{th} Annual Symposium on Logic in Computer Science, Philadelphia, USA, IEEE Computer Society Press, 1990, pp. 142–153.

[17] D. PARK, *Concurrency and automata on infinite sequences*, in 5^{th} GI Conference, Karlsruhe, Germany, P. Deussen, ed., vol. 104 of Lecture Notes in Computer Science, Springer-Verlag, 1981, pp. 167–183.

[18] P. SEWELL, *Nonaxiomatisability of equivalences over finite state processes*, Annals of Pure and Applied Logic, 90 (1997), pp. 163–191.

Recent BRICS Report Series Publications

RS-07-5 **Luca Aceto, Silvio Capobianco, and Anna Ingólfsdóttir.** ***On the Existence of a Finite Base for Complete Trace Equivalence over BPA with Interrupt*****. February 2007. 26 pp.**

RS-07-4 **Kristian Støvring and Søren B. Lassen.** ***A Complete, Co-Inductive Syntactic Theory of Sequential Control and State*****. February 2007. 36 pp. Appears in the proceedings of POPL 2007, p. 161–172.**

RS-07-3 **Luca Aceto, Willem Jan Fokkink, and Anna Ingólfsdóttir.** ***Ready To Preorder: Get Your BCCSP Axiomatization for Free!*** **February 2007. 37 pp.**

RS-07-2 **Luca Aceto and Anna Ingólfsdóttir.** ***Characteristic Formulae: From Automata to Logic*****. January 2007. 18 pp.**

RS-07-1 **Daniel Andersson.** ***HIROIMONO is NP-complete*****. January 2007. 8 pp.**

RS-06-19 **Michael David Pedersen.** ***Logics for The Applied π Calculus*****. December 2006. viii+111 pp.**

RS-06-18 **Małgorzata Biernacka and Olivier Danvy.** ***A Syntactic Correspondence between Context-Sensitive Calculi and Abstract Machines*****. dec 2006. iii+39 pp. Extended version of an article to appear in TCS. Revised version of BRICS RS-05-22.**

RS-06-17 **Olivier Danvy and Kevin Millikin.** ***A Rational Deconstruction of Landin's J Operator*****. December 2006. ii+37 pp. Revised version of BRICS RS-06-4. A preliminary version appears in the proceedings of IFL 2005, LNCS 4015:55–73.**

RS-06-16 **Anders Møller.** ***Static Analysis for Event-Based XML Processing*****. October 2006. 16 pp.**

RS-06-15 **Dariusz Biernacki, Olivier Danvy, and Kevin Millikin.** ***A Dynamic Continuation-Passing Style for Dynamic Delimited Continuations*****. October 2006. ii+28 pp. Revised version of BRICS RS-05-16.**

RS-06-14 **Giorgio Delzanno, Javier Esparza, and Jiří Srba.** ***Monotonic Set-Extended Prefix Rewriting and Verification of Recursive Ping-Pong Protocols*****. July 2006. 31 pp. To appear in ATVA '06.**