A Complete Axiomatisation of Branching Bisimulation for Process Algebras with Alternative Quantification over Data

J.F. Groote, S.P. Luttik

Software Engineering (SEN)

SEN-R9830 November 1998

Report SEN-R9830
ISSN 1386-369X

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

A Complete Axiomatisation of Branching Bisimulation for Process Algebras with Alternative Quantification over Data

J.F. Groote[1,2]
JanFriso.Groote@cwi.nl

S.P. Luttik[1,3]
Bas.Luttik@cwi.nl

[1] CWI, P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

[2] Computing Science Department, Eindhoven University of Technology
P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands

[3] Programming Research Group, University of Amsterdam,
Kruislaan 403, NL-1098 SJ Amsterdam, The Netherlands

ABSTRACT

We define a class of process algebras with silent step and a generalised operation $\sum$ that allows explicit treatment of *alternative quantification* over data, and we investigate the specific subclass formed by the algebras of finite processes modulo rooted branching bisimulation. We give a ground complete axiomatisation for those branching bisimulation algebras of which the data part has built-in equality and Skolem functions.

1991 Mathematics Subject Classification: 03G25; 08A70; 68Q65; 68Q70

1991 Computing Reviews Classification System: D.1.3; F.1.1; F.4.1

Keywords and Phrases: Generalised Algebra, Process Algebra, Algebraic Specification, Alternative Quantification, Input Prefixing, Strong Bisimulation, Branching Bisimulation, Silent Step, Abstraction.

Note: Research supported by the Netherlands Organization for Scientific Research (NWO) under contract SION 612-33-008. Work carried out under project SEN 2.1 Process Specification and Analysis.

1. Introduction

In Groote and Luttik (1998) we proposed an axiomatisation of process algebras with data, conditionals and alternative quantification, that we called pCRL-algebras. We proved that our axiomatisation is complete for strong bisimulation algebras of which the data part has built-in equality and Skolem functions. This seems a rather severe restriction; it implies that the entire first-order theory of the data algebra is decidable. However, in the same paper we argue that one cannot do much better. It turns out that strong bisimulation is not recursively enumerable (it is Π^0_4-hard), and clearly, the existence of a general axiomatisation would contradict this.

The main cause for the complexity of the theory pCRL is the binder $\sum$ that is used to express alternative quantification over data. If $d_0, d_1, d_2, \ldots$ is an enumeration of some data type D and x is a variable that ranges over D, then we let the process term $\sum_{x:D} p$ refer to the (possibly infinite) alternative composition of the processes $p[x := d_0], p[x := d_1], p[x := d_2], \ldots$. Such an operation is a useful specification tool, since it allows us to describe the action that *inputs* an arbitrary element from some data type. For instance the term $\sum_{n:\mathbb{N}} \mathtt{read}(n) \cdot p(n)$ refers to the process that reads an arbitrary natural number and then executes the process p instantiated with this particular natural number.

In this paper we shall address the extension of the theory pCRL with an element τ ('silent step') that represents internal activity, and we study it in branching bisimulation semantics. This extension is operationally conservative (processes that do not involve τ are branching bisimilar if, and only if, they are strongly bisimilar), so the question of whether two processes are branching bisimilar is at least as hard as the question of whether they are strongly bisimilar. Nevertheless, we shall prove that the extension to a branching bisimulation algebra of any strong bisimulation algebra for which we provided an axiomatisation in our previous paper, is axiomatised just by adding the two branching bisimulation laws of Van Glabbeek and Weijland (1996).

As far as we know, the only other investigation of alternative quantification in branching bisimulation semantics is by Klusener (1992). In his work, a (time-stamped) τ is included in a real time process algebra with integration. To the axioms for strong bisimulation Klusener adds a single law to arrive at an axiomatisation of branching bisimulation. It defines the interaction of three concepts (integration, internal activity, and real time) in one go, and is therefore of complicated nature.

Klusener's setting resembles ours in that integration can be viewed as alternative quantification over the datatype of real numbers. On the other hand, our setting is simpler, because there is no interaction between data and sequential composition, and our τ has no parameter. From the results in the present paper we conclude that the complexity of Klusener's law is not caused by the combination of integration and internal activity.

Other extensions of message-passing process algebras with the silent step have been carried out by Hennessy and Lin (1996), Lin (1995), and Parrow and Victor (1998). Contrary to our approach, these extensions take place in a variant of weak bisimulation semantics of Milner (1980). In the first two papers the extension takes place in a setting with input prefixing instead of alternative quantification (we proved in Groote and Luttik (1998) that the input prefix mechanism is in general a less expressive operation than alternative quantification). In both papers it is shown that it suffices to add Milner's τ-laws.

The paper of Parrow and Victor (1998) deals with the extension of the fusion calculus with silent steps. In the fusion calculus there is a single binder that resembles our alternative quantification. The authors argue that, since mismatch operators do not distribute over prefixes in a setting with fusion actions, Milner's third τ-law must be replaced by two schemes.

This paper is organised as follows. In the next section we define the theory of process algebras with data, conditionals, alternative quantification and silent step, and we define the notions of strong and branching bisimulation that give rise to specific models of this theory. In §3 we show that branching bisimulation is first-order definable. This result plays a key role in our completeness proof, which is given in §4.

Acknowledgements We thank Michel Reniers for his careful reading of a draft version of this paper.

2. Process Algebras with Data and Silent Step

In this section we define the class of process algebras with data, conditionals, alternative quantification and the silent step, which we call $p\mathsf{CRL}_\tau$, and we introduce the notions of strong- and branching bisimulation. We assume that the reader has some familiarity with universal algebra, for which we refer to McKenzie *et al.* (1987). We refer to Groote and Luttik (1998) for an account of the generalisation of some of the notions in universal algebra to a setting with infinitary (or: generalised) operations.

2.1 The theory $p\mathsf{CRL}_\tau$

For the purpose of this paper we fix a many-sorted, first-order data signature Δ that contains at least a sort $\mathbb{b}$ of *booleans* with function declarations for $\top$, $\bot$, $\neg$, $\wedge$, and $\vee$. We assume that $\mathcal{A}$ is a set of *action declarations* over Δ, function declarations of the form $\mathsf{a}\colon s_1 \cdots s_n \to \mathbb{p}$ with $s_1, \ldots, s_n \in \Delta$ and $\mathbb{p} \notin \Delta$, which are usually denoted by $\mathsf{a}{:}s_1 \cdots s_n$.

We obtain the $p\mathsf{CRL}_\tau$-signature over Δ and $\mathcal{A}$ by extending Δ with a sort $\mathbb{p}$, the action declarations in $\mathcal{A}$, and

1. nullary function declarations $\delta\colon\lambda \to \mathbb{p}$ (λ refers to the empty sequence) for 'deadlock' and $\tau\colon\lambda \to \mathbb{p}$ for the 'silent step';
2. binary function declarations $(\_ + \_)\colon\mathbb{p}\mathbb{p} \to \mathbb{p}$ for choice and $(\_ \cdot \_)\colon\mathbb{p}\mathbb{p} \to \mathbb{p}$ for sequential composition;
3. a ternary function declaration $(\_ \lhd \_ \rhd \_)\colon\mathbb{p}\mathbb{b}\mathbb{p} \to \mathbb{p}$ for conditionals ($p \lhd b \rhd q$ should be read as 'then p if b else q'); and
4. a binder declaration $\sum : \mathbb{p}$ for alternative quantification over data (if $d_0, d_1, d_2, \ldots$ is an enumeration of elements of some data type and x ranges over the same type, then $\sum_\xi p[x := \xi]$ refers to the (possibly infinite) choice between $p[x := d_0], p[x := d_1], p[x := d_2], \ldots$.)

Let Σ denote such a $p\mathsf{CRL}_\tau$-signature. Terms over Σ are constructed using disjunct, countably infinite families X and Ξ of *free* and *bound* variables for Δ (if $s \in \Delta$ is a sort, then we denote by X_s the set of variables in X of sort s). Terms of sort $\mathbb{p}$ we refer to as $\mathbb{p}$-*ground process terms* (note that X does not contain variables of sort $\mathbb{p}$); they are considered modulo α-conversion. The other terms over Σ and X are *data terms*; these may contain free variables. A process term $a = \mathtt{a}(\bar{t})$, with $\mathtt{a}{:}\bar{s} \in \mathcal{A}$, is called an *action term*; we adopt the convention that the leading action declaration of an action term is denoted by the name of the action term in typewriter font, e.g., $\mathtt{a}$ refers to the leading action declaration of a.

We adopt some notational conventions regarding boolean terms and process terms. Function declarations are usually written in mixfix notation and brackets are omitted where possible. We give the following precedence to the operators: $(\_ + \_) < \sum < (\_ \lhd \_ \rhd \_) < (\_ \cdot \_)$. Terms of the form $p \cdot q$ are usually written pq. If $p = p(x_1, \ldots, x_n)$ is a process term, $x_i \in X_{s_i}$ (for $1 \leq i \leq n$) and ξ is a bound variable of sort s_i, then we shall use $\sum_{x_i:s_i} p$ as an abbreviation for the term $\sum_\xi p[x_i := \xi]$. Note that by α-conversion the specific choice of the bound variable ξ is immaterial and that the free variable x_i does not occur in $\sum_{x_i:s_i} p$. Thus, provided that X_s is infinite for all sorts $s \in \Delta$, we may choose x_i different from all the other free variables that occur in a context; e.g., if $p = \sum_{x_1:s_1} p'$ and $q = \sum_{x_2:s_2} q'$, then we may assume without loss of generality that $x_1 \neq x_2$, $x_1 \notin FV(q)$ and $x_2 \notin FV(p)$. This assumption will often be implicitly present. We use $\sum_{x_1 \cdots x_n : s_1 \cdots s_n} p$ as an abbreviation of $\sum_{x_1:s_n} \cdots \sum_{x_n:s_n} p$.

The axioms of the theory $p\mathsf{CRL}_\tau$ are the axioms for boolean algebras (see e.g. Koppelberg (1989)) and the axioms depicted in Table 1; we obtain the subtheory $p\mathsf{CRL}$ by omitting the axioms B1 and B2. We write $p\mathsf{CRL}_\tau \vdash p \approx q$ if the identity $p \approx q$ is derivable from the axioms of $p\mathsf{CRL}_\tau$ by means of *generalised* equational logic, that is equational logic extended with a congruence rule for binders (cf. Groote and Luttik (1998)). In this particular setting, this rule takes the form

$$\frac{p \approx q}{\sum_\xi (p[x := \xi]) \approx \sum_\xi (q[x := \xi])} \qquad \text{with } x \text{ a variable in } X.$$

In derivations we shall use BA to refer to applications of the boolean axioms.

In the sequel, we shall make liberal use of an element ϵ that acts as a unit for $\cdot$. That is, we assume $p \cdot \epsilon = \epsilon \cdot p = p$. We stress that ϵ is *not* an element of Σ and that it is only used to facilitate notation.

It is well-known that process terms may be thought of as having the form defined below.

Definition 2.1 Let A be the set of action terms, and let B be the set of boolean terms.

We inductively define the set of *basic terms* as follows:

1. δ is a basic term;
2. if p is a basic term or $p = \epsilon$, then $\sum_{\bar{x}:\bar{s}} a \cdot p \lhd b \rhd \delta$ (with $a \in A \cup \{\tau\}$ and $b \in B$) is a basic term; and

(A1)	$x + y$	$\approx y + x$	(Sum1)	$\sum_{v:s} y$	$\approx y$
(A2)	$x + (y + z)$	$\approx (x + y) + z$	(Sum3)	$\sum_{\xi} p[v := \xi]$	$\approx \sum_{\xi} p[v := \xi] + p$
(A3)	$x + x$	$\approx x$	(Sum4)	$\sum_{v:s}(p + q)$	$\approx \sum_{v:s} p + \sum_{v:s} q$
(A4)	$(x + y)z$	$\approx xz + yz$	(Sum5)	$(\sum_{v:s} p)y$	$\approx \sum_{v:s} py$
(A5)	$x(yz)$	$\approx (xy)z$	(Sum12)	$(\sum_{v:s} p) \lhd b \rhd \delta$	$\approx \sum_{v:s} p \lhd b \rhd \delta$
(A6)	$x + \delta$	$\approx x$			
(A7)	δx	$\approx \delta$	(B1)	$x\tau$	$\approx x$
			(B2)	$x(\tau(y + z) + y)$	$\approx x(y + z)$

(Cond1)	$x \lhd \top \rhd y$	$\approx x$
(Cond2)	$x \lhd \bot \rhd y$	$\approx y$
(Cond3)	$x \lhd b \rhd y$	$\approx x \lhd b \rhd \delta + y \lhd \neg b \rhd \delta$
(Cond4)	$(x \lhd b_1 \rhd \delta) \lhd b_2 \rhd \delta$	$\approx x \lhd b_1 \wedge b_2 \rhd \delta$
(Cond5)	$(x \lhd b_1 \rhd \delta) + (x \lhd b_2 \rhd \delta)$	$\approx x \lhd b_1 \vee b_2 \rhd \delta$
(Cond6)	$(x \lhd b \rhd \delta)y$	$\approx xy \lhd b \rhd \delta$
(Cond7)	$(x + y) \lhd b \rhd \delta$	$\approx x \lhd b \rhd \delta + y \lhd b \rhd \delta$

Table 1: The axioms for $p\mathsf{CRL}_\tau$ for a given a $p\mathsf{CRL}_\tau$-signature Σ; the Sum-axioms are schemes in which p and q range over $\mathbb{p}$-ground process terms; the symbols x, y and z are free variables of sort $\mathbb{p}$, and b, b_1 and b_2 are free variables of sort $\mathbb{b}$. (We have kept our numbering consistent with Groote and Ponse (1994): they have an axiom Sum2 that defines α-conversion.)

3. if p and q are basic terms, then $p + q$ is a basic term.

Lemma 2.2 (Basic Term Lemma) For every $\mathbb{p}$-ground process term p there exists a basic term q such that $p\mathsf{CRL} \vdash p \approx q$.

Proof. Straightforward by induction on the number of symbols in p. □

We call a basic term *simple* if it is of the form

$$\textstyle\sum_{\bar{x}:\bar{s}} a \cdot p \lhd b \rhd \delta,$$

where p is a basic term or ϵ. Defining $\delta = \sum_{i \in \emptyset} p_i$, any basic term can be written as

$$\sum_{i \in I} p_i, \qquad p_i \text{ a simple basic term for all } i \in I \ (I \text{ finite}).$$

2.2 The Branching Bisimulation Algebra

For the rest of this paper we fix a Δ-algebra $\mathfrak{D}$ that contains a boolean algebra with precisely *two* elements; we denote $\mathfrak{D}(\top)$ (the interpretation of $\top$ in $\mathfrak{D}$) by T, and $\mathfrak{D}(\bot)$ by $\bot$. We construct models for $p\mathsf{CRL}$ and $p\mathsf{CRL}_\tau$, based on $\mathfrak{D}$, by constructing an algebra of *processes* $\mathfrak{P}$, defining congruences $\leftrightarroweq$ and $\leftrightarroweq_{rb}$ on this algebra, and taking the quotients $\mathfrak{P}/\leftrightarroweq$ and $\mathfrak{P}/\leftrightarroweq_{rb}$; they will turn out to be a $p\mathsf{CRL}$-algebra and a $p\mathsf{CRL}_\tau$-algebra, respectively.

Processes First, define a set A of *atomic actions* by

$$\mathrm{A} = \{ \mathsf{a}\langle d_1, \ldots, d_n \rangle \mid \mathsf{a}{:}s_1 \cdots s_n \in \mathcal{A} \text{ and } d_i \in \mathfrak{D}(s_i) \text{ for } 1 \leq i \leq n \ \}.$$

$$\mathbf{a} \xrightarrow{\mathbf{a}} \epsilon \qquad \text{for all } \mathbf{a} \in \mathrm{A}_{\boldsymbol{\tau}}$$

$$\frac{\mathbf{p} \xrightarrow{\mathbf{a}} \mathbf{p}'}{\mathbf{p}\cdot\mathbf{q} \xrightarrow{\mathbf{a}} \mathbf{p}'\cdot\mathbf{q}} \qquad \frac{\mathbf{p}\xrightarrow{\mathbf{a}}\epsilon}{\mathbf{p}\cdot\mathbf{q}\xrightarrow{\mathbf{a}}\mathbf{q}} \qquad \mathbf{a}\in \mathrm{A}_{\boldsymbol{\tau}} \text{ and } \mathbf{p},\mathbf{p}',\mathbf{q}\in\mathrm{P}$$

$$\frac{\mathbf{p}\xrightarrow{\mathbf{a}}\mathbf{q}}{\sum\mathrm{P}'\xrightarrow{\mathbf{a}}\mathbf{q}} \qquad \mathbf{a}\in\mathrm{A}_{\boldsymbol{\tau}},\ \mathbf{p}\in\mathrm{P}'\subseteq\mathrm{P} \text{ and } \mathbf{q}\in\mathrm{P}^{\epsilon}$$

Table 2: The transition system specification for $\mathfrak{P}$.

Let $\boldsymbol{\delta}$ and $\boldsymbol{\tau}$ be distinct elements such that $\boldsymbol{\delta},\boldsymbol{\tau}\notin\mathrm{A}$; we shall abbreviate $\mathrm{A}\cup\{\boldsymbol{\tau}\}$ by $\mathrm{A}_{\boldsymbol{\tau}}$. The set $\mathrm{P}=\bigcup_{n\in\omega}\mathrm{P}^n$ of *processes* is obtained by the following recursion

$$\begin{array}{lcl} \mathrm{P}^0 & = & \mathrm{A}_{\boldsymbol{\tau}}\cup\{\boldsymbol{\delta}\} \\ \mathrm{P}^{n+1} & = & \mathrm{P}^n\cup\{\mathbf{p}\cdot\mathbf{q},\ \sum\mathrm{P}' \mid \mathbf{p},\mathbf{q}\in\mathrm{P}^n,\ \emptyset\neq\mathrm{P}'\subseteq\mathrm{P}^n\}; \end{array}$$

we shall write $\mathbf{p}+\mathbf{q}$ for $\sum\{\mathbf{p},\mathbf{q}\}$.

Let $\mathfrak{P}$ be the Σ-algebra of which the restriction to Δ is $\mathfrak{D}$, $\mathfrak{P}(\mathbb{p})=\mathrm{P}$, and operations on P defined by

$$\begin{array}{lcll} \mathfrak{P}(\mathsf{a})(d_1,\ldots,d_n) & = & \mathsf{a}\langle d_1,\ldots,d_n\rangle & \text{for each } \mathsf{a}{:}s_1\cdots s_n\in\Sigma; \\ \mathfrak{P}(\tau) & = & \boldsymbol{\tau}; & \\ \mathfrak{P}(\delta) & = & \boldsymbol{\delta}; & \\ \mathfrak{P}(+)(\mathbf{p},\mathbf{q}) & = & \mathbf{p}+\mathbf{q}; & \\ \mathfrak{P}(\cdot)(\mathbf{p},\mathbf{q}) & = & \mathbf{p}\cdot\mathbf{q}; & \\ \mathfrak{P}(\_\lhd\_\rhd\_)(\mathbf{p},\mathbf{b},\mathbf{q}) & = & \left\{\begin{array}{ll}\mathbf{p} & \text{if } \mathbf{b}=\top; \\ \mathbf{q} & \text{if } \mathbf{b}=\bot; \text{ and}\end{array}\right. & \\ \mathfrak{P}(\sum)(\mathrm{P}') & = & \sum\mathrm{P}' & \text{for each } \emptyset\neq\mathrm{P}'\subseteq\mathrm{P}. \end{array}$$

Operational Semantics For convenience of notation we define $\mathrm{P}^{\epsilon}=\mathrm{P}\cup\{\epsilon\}$ and, for any binary relation $\mathcal{R}$ on P, $\mathcal{R}^{\epsilon}=\mathcal{R}\cup\{\langle\epsilon,\epsilon\rangle\}$. The rules in Table 2 define a *transition relation* $\longrightarrow\ \subseteq\mathrm{P}\times\mathrm{A}_{\boldsymbol{\tau}}\times\mathrm{P}^{\epsilon}$ on $\mathfrak{P}$. In the sequel, we shall tacitly assume that $\mathbf{p}$ ranges over P and $\mathbf{p}'$ ranges over P^{ϵ} in $\mathbf{p}\xrightarrow{\mathbf{a}}\mathbf{p}'$. If there is an $\mathbf{a}\in\mathrm{A}_{\boldsymbol{\tau}}$ such that $\mathbf{p}\xrightarrow{\mathbf{a}}\mathbf{p}'$, then we call $\mathbf{p}'$ a *residual* of $\mathbf{p}$.

Let us first recall the definition of strong bisimulation.

Definition 2.3 (Strong Bisimulation) A binary relation $\mathcal{R}\subseteq\mathrm{P}\times\mathrm{P}$ is called a *strong bisimulation relation* if it is symmetric and $\mathcal{R}^{\epsilon}$ satisfies

if $\langle\mathbf{p},\mathbf{q}\rangle\in\mathcal{R}$ and $\mathbf{p}\xrightarrow{\mathbf{a}}\mathbf{p}'$, then there exists $\mathbf{q}'\in\mathrm{P}^{\epsilon}$ such that $\mathbf{q}\xrightarrow{\mathbf{a}}\mathbf{q}'$ and $\langle\mathbf{p}',\mathbf{q}'\rangle\in\mathcal{R}^{\epsilon}$.

If $\mathbf{p},\mathbf{q}\in\mathrm{P}$ and there is a strong bisimulation relation that contains the pair $\langle\mathbf{p},\mathbf{q}\rangle$, then $\mathbf{p}$ and $\mathbf{q}$ are called *strongly bisimilar* (notation: $\mathbf{p}\ \underline{\leftrightarrow}\ \mathbf{q}$).

Strong bisimulation gives rise to a model $\mathfrak{P}/\underline{\leftrightarrow}$ for the axiom system $p\mathsf{CRL}$ (cf. Groote and Luttik (1998)), that we call the *strong bisimulation algebra* for $\mathfrak{D}$ and $\mathcal{A}$:

Theorem 2.4 The family $\underline{\leftrightarrow}=\langle\theta_s|\ s\in\mathcal{S}\rangle$ with $\theta_{\mathbb{p}}=\underline{\leftrightarrow}$ and $\theta_s=\mathsf{id}(\mathfrak{P}(s))$ for $s\neq\mathbb{p}$ is a congruence on $\mathfrak{P}$, and $\mathfrak{P}/\underline{\leftrightarrow}\models p\mathsf{CRL}$.

Let us write $\mathbf{p}_0\Longrightarrow\mathbf{p}_n$ to abbreviate a (possibly empty) sequence of $\boldsymbol{\tau}$-transitions

$$\mathbf{p}_0\xrightarrow{\boldsymbol{\tau}}\mathbf{p}_1\xrightarrow{\boldsymbol{\tau}}\ldots\xrightarrow{\boldsymbol{\tau}}\mathbf{p}_n \qquad n\geq 0.$$

Definition 2.5 (Branching Bisimulation) A binary relation $\mathcal{R} \subseteq \mathrm{P}^{\epsilon} \times \mathrm{P}^{\epsilon}$ is called a *branching bisimulation* if it is symmetric and $\langle \mathbf{p}, \mathbf{q} \rangle \in \mathcal{R}$ implies

i. if $\mathbf{p} \xrightarrow{\mathbf{a}} \mathbf{p}'$, then either $\mathbf{a} = \boldsymbol{\tau}$ and $\langle \mathbf{p}', \mathbf{q} \rangle \in \mathcal{R}$, or there exist $\mathbf{q}^* \in \mathrm{P}$ and $\mathbf{q}' \in \mathrm{P}^{\epsilon}$ such that $\mathbf{q} \Longrightarrow \mathbf{q}^* \xrightarrow{\mathbf{a}} \mathbf{q}'$ and $\langle \mathbf{p}, \mathbf{q}^* \rangle, \langle \mathbf{p}', \mathbf{q}' \rangle \in \mathcal{R}$; and

ii. $\mathbf{p} \Longrightarrow \epsilon$ if, and only if, $\mathbf{q} \Longrightarrow \epsilon$.

If $\mathbf{p}, \mathbf{q} \in \mathrm{P}^{\epsilon}$, and there is a branching bisimulation relation that contains the pair $\langle \mathbf{p}, \mathbf{q} \rangle$, then $\mathbf{p}$ and $\mathbf{q}$ are called *branching bisimilar* (notation: $\mathbf{p} \mathrel{\underline{\leftrightarrow}}_b \mathbf{q}$).

Clearly, in accordance with Table 2 we can associate with every element of A a labeled tree. If we consider the induced set of labeled trees modulo isomorphism, then we obtain a subalgebra of the algebra of graphs of Van Glabbeek and Weijland (1996). Consequently, the following lemma, proved in Van Glabbeek and Weijland (1996), also holds in our setting.

Lemma 2.6 (Stuttering Lemma) *If $\mathbf{p}_0 \xrightarrow{\boldsymbol{\tau}} \mathbf{p}_1 \xrightarrow{\boldsymbol{\tau}} \ldots \xrightarrow{\boldsymbol{\tau}} \mathbf{p}_n$ and $\mathbf{p}_0 \mathrel{\underline{\leftrightarrow}}_b \mathbf{p}_n$, then $\mathbf{p}_i \mathrel{\underline{\leftrightarrow}}_b \mathbf{p}_j$ for all $i, j \leq n$.*

The relation $\mathrel{\underline{\leftrightarrow}}_b$ is not a congruence, for if $\mathbf{a}$ and $\mathbf{b}$ are distinct atomic actions, then $\boldsymbol{\tau} \cdot \mathbf{a} + \mathbf{a} \mathrel{\underline{\leftrightarrow}}_b \mathbf{a}$, but $\boldsymbol{\tau} \cdot \mathbf{a} + \mathbf{a} + \mathbf{b} \not\mathrel{\underline{\leftrightarrow}}_b \mathbf{a} + \mathbf{b}$. This motivates the following definition.

Definition 2.7 A branching bisimulation relation $\mathcal{R}$ is *rooted* with respect to $\mathbf{p}$ if $\langle \mathbf{p}, \mathbf{q} \rangle \in \mathcal{R}$ and $\mathbf{p} \xrightarrow{\mathbf{a}} \mathbf{p}'$ implies that there exists a $\mathbf{q}'$ such that $\mathbf{q} \xrightarrow{\mathbf{a}} \mathbf{q}'$ and $\langle \mathbf{p}', \mathbf{q}' \rangle \in \mathcal{R}$.

If $\mathbf{p}, \mathbf{q} \in \mathrm{P}$ and there is a branching bisimulation relation that contains the pair $\langle \mathbf{p}, \mathbf{q} \rangle$ and that is rooted with respect to $\mathbf{p}$ and $\mathbf{q}$, then $\mathbf{p}$ and $\mathbf{q}$ are called *rooted branching bisimilar* (notation: $\mathbf{p} \mathrel{\underline{\leftrightarrow}}_{rb} \mathbf{q}$).

Lemma 2.8 *The family $\mathrel{\underline{\leftrightarrow}}_{rb} = \langle \theta_s | \ s \in \mathcal{S} \rangle$ with $\theta_{\mathbb{p}} = \mathrel{\underline{\leftrightarrow}}_{rb}$ and $\theta_s = \mathsf{id}(\mathfrak{P}(s))$ for $s \neq \mathbb{p}$ is a congruence on $\mathfrak{P}$, and $\mathfrak{P}/\mathrel{\underline{\leftrightarrow}}_{rb} \models p\mathsf{CRL}_{\tau}$.*

Proof. Basten (1996) has shown that $\mathrel{\underline{\leftrightarrow}}_{rb}$ is an equivalence relation on the set of labeled graphs. By the remarks preceding Lemma 2.6 it is clear that $\mathrel{\underline{\leftrightarrow}}_{rb}$ is then also an equivalence relation on P. In view of the definition of the operations on $\mathfrak{P}$ we only need to show that $\mathrel{\underline{\leftrightarrow}}_{rb}$ has the substitution property for $\cdot$ and $\sum$.

If $\mathbf{p}_i, \mathbf{q}_i \in \mathrm{P}$ and $\mathcal{R}_i$ is a branching bisimulation relation that witnesses $\mathbf{p}_i \mathrel{\underline{\leftrightarrow}}_{rb} \mathbf{q}_i$ (for $i = 1, 2$), then it is straightforward to verify that

$$\{\langle \mathbf{p} \cdot \mathbf{p}_2, \mathbf{q} \cdot \mathbf{q}_2 \rangle, \langle \mathbf{p} \cdot \mathbf{q}_2, \mathbf{q} \cdot \mathbf{p}_2 \rangle \mid \langle \mathbf{p}, \mathbf{q} \rangle \in \mathcal{R}_1\} \cup \mathcal{R}_2$$

is a branching bisimulation relation, and since $\mathcal{R}_1$ is rooted with respect to $\mathbf{p}_1$ and $\mathbf{q}_1$, $\mathcal{R}$ is rooted with respect to $\mathbf{p}_1 \cdot \mathbf{p}_2$ and $\mathbf{q}_1 \cdot \mathbf{q}_2$. Hence $\mathbf{p}_1 \cdot \mathbf{p}_2 \mathrel{\underline{\leftrightarrow}}_{rb} \mathbf{q}_1 \cdot \mathbf{q}_2$.

If $\emptyset \neq \mathrm{P}', \mathrm{P}'' \subseteq \mathrm{P}$ and $\mathrm{P}'/\mathrel{\underline{\leftrightarrow}}_{rb} = \mathrm{P}''/\mathrel{\underline{\leftrightarrow}}_{rb}$, then for all $\mathbf{p} \in \mathrm{P}'$ there exists $\mathbf{q} \in \mathrm{P}''$ and a branching bisimulation $\mathcal{R}_{\mathbf{p}}$ that is rooted with respect to $\mathbf{p}$ and $\mathbf{q}$ and contains $\langle \mathbf{p}, \mathbf{q} \rangle$, and for all $\mathbf{q} \in \mathrm{P}''$ there exists $\mathbf{p} \in \mathrm{P}'$ and a branching bisimulation $\mathcal{R}_{\mathbf{q}}$ that is rooted with respect to $\mathbf{p}$ and $\mathbf{q}$ and contains $\langle \mathbf{q}, \mathbf{p} \rangle$. Arbitrary unions of branching bisimulations are branching bisimulations. Thus it follows that the relation

$$\{\langle \textstyle\sum \mathrm{P}', \sum \mathrm{P}'' \rangle, \langle \sum \mathrm{P}'', \sum \mathrm{P}' \rangle\} \cup \bigcup \{\mathcal{R}_{\mathbf{p}} \mid \mathbf{p} \in \mathrm{P}'\} \cup \bigcup \{\mathcal{R}_{\mathbf{q}} \mid \mathbf{q} \in \mathrm{P}''\}$$

is a branching bisimulation relation, and it is clear that it is rooted with respect to $\sum \mathrm{P}'$ and $\sum \mathrm{P}''$. We conclude that $\mathrel{\underline{\leftrightarrow}}_{rb}$ is a congruence on $\mathfrak{P}$.

Since $\mathrel{\underline{\leftrightarrow}} \subseteq \mathrel{\underline{\leftrightarrow}}_{rb}$, we find by the Second Isomorphism Theorem (p. 149 of McKenzie *et al.* (1987)) that $\mathfrak{P}/\mathrel{\underline{\leftrightarrow}}_{rb}$ is a homomorphic image of $\mathfrak{P}/\mathrel{\underline{\leftrightarrow}}$. Hence, by the HSP Theorem (p. 237 of McKenzie *et al.* (1987)), we conclude that $\mathfrak{P}/\mathrel{\underline{\leftrightarrow}}_{rb} \models p\mathsf{CRL}$. It remains to verify that $\mathfrak{P}/\mathrel{\underline{\leftrightarrow}}_{rb} \models \mathrm{B1}, \mathrm{B2}$; we leave it to the reader to find the witnessing relations. □

We call $\mathfrak{P}/\underline{\leftrightarrow}_{rb}$ the *branching bisimulation algebra* for $\mathfrak{D}$ and $\mathcal{A}$.

Definition 2.9 ($\boldsymbol{\tau}$-inertness and compactness) We call a $\boldsymbol{\tau}$-transition $\mathbf{p} \xrightarrow{\boldsymbol{\tau}} \mathbf{p}'$ *inert* if $\mathbf{p} \mathrel{\underline{\leftrightarrow}_b} \mathbf{p}'$. We call $\mathbf{p} \in \mathrm{P}$ *compact* if $\mathbf{p}$ has no inert $\boldsymbol{\tau}$-transitions and, recursively, all its residuals are compact, that is, if there are no $\mathbf{p}_1, \dots, \mathbf{p}_n, \mathbf{p}' \in \mathrm{P}^\epsilon$ such that

$$\mathbf{p} \xrightarrow{\mathbf{a}_1} \mathbf{p}_1 \xrightarrow{\mathbf{a}_2} \cdots \xrightarrow{\mathbf{a}_n} \mathbf{p}_n \xrightarrow{\boldsymbol{\tau}} \mathbf{p}' \text{ and } \mathbf{p}_n \mathrel{\underline{\leftrightarrow}_b} \mathbf{p}'.$$

Lemma 2.10 (Compactness) If all residuals of $\mathbf{p}$ and $\mathbf{q}$ are compact, then $\mathbf{p} \mathrel{\underline{\leftrightarrow}_{rb}} \mathbf{q}$ iff $\mathbf{p} \mathrel{\underline{\leftrightarrow}} \mathbf{q}$.

Proof. Since any strong bisimulation relation containing the pair $\langle \mathbf{p}, \mathbf{q} \rangle$ is a branching bisimulation relation that is rooted with respect to $\mathbf{p}$ and $\mathbf{q}$, the implication from right to left is immediate. For the other direction observe that, by rootedness, it suffices to show that if $\mathbf{p}$ and $\mathbf{q}$ are compact and $\mathbf{p} \mathrel{\underline{\leftrightarrow}_b} \mathbf{q}$, then $\mathbf{p} \mathrel{\underline{\leftrightarrow}} \mathbf{q}$.

If $\mathbf{p} \mathrel{\underline{\leftrightarrow}_b} \mathbf{q}$ and $\mathbf{p} \xrightarrow{\boldsymbol{\tau}} \mathbf{p}'$ then $\mathbf{p}' \not\mathrel{\underline{\leftrightarrow}_b} \mathbf{q}$; so if $\mathbf{p} \mathrel{\underline{\leftrightarrow}_b} \mathbf{q}$ and $\mathbf{p} \xrightarrow{\mathbf{a}} \mathbf{p}'$, then there exists $\mathbf{q}^*, \mathbf{q}' \in \mathrm{P}$ such that $\mathbf{q} \Longrightarrow \mathbf{q}^* \xrightarrow{\mathbf{a}} \mathbf{q}'$, $\mathbf{p} \mathrel{\underline{\leftrightarrow}_b} \mathbf{q}^*$ and $\mathbf{p}' \mathrel{\underline{\leftrightarrow}_b} \mathbf{q}'$. Using Lemma 2.6 and compactness we may conclude that $\mathbf{q} = \mathbf{q}^*$, so $\mathbf{q} \xrightarrow{\mathbf{a}} \mathbf{q}'$. It follows that $\mathbf{p} \mathrel{\underline{\leftrightarrow}} \mathbf{q}$. □

3. The first-order definability of branching bisimulation

The set of first-order Δ-formulae is the smallest set that contains the Δ-equations and is closed under the connectives $\neg$, $\wedge$, and $\forall$ of first-order logic. We abbreviate $(\top \approx \top)$ by $\top$, $\neg(\top \approx \top)$ by $\bot$ (instead of $(\top \approx \top)$ we could of course have used any other tautology). Moreover, we use $\vee$ and $\exists$ according to their well-known definition in terms of $\neg$, $\wedge$ and $\forall$. We shall use the standard satisfaction relation (see e.g. Chang and Keisler (1990)); if φ is a first-order Δ-formula and α is a valuation of X in $\mathfrak{D}$, then we shall write $\mathfrak{D}, \alpha \models \varphi$ to express that $\bar{\alpha}(\varphi)$ is true of $\mathfrak{D}$ (if α is a valuation of X in $\mathfrak{D}$, then we shall denote by $\bar{\alpha}$ the unique extensions of α to functions from terms to elements of $\mathfrak{D}$ and from formulae to truth values).

Our goal in this section is to associate to every pair $\langle p, q \rangle$ of $\mathbb{p}$-ground process terms a first-order Δ-formula φ such that $\mathfrak{D}, \alpha \models \varphi$ iff $\bar{\alpha}(p) \mathrel{\underline{\leftrightarrow}_b} \bar{\alpha}(q)$. In the sequel, we shall postulate the existence of such a formula by means of the phrase "$p \mathrel{\underline{\leftrightarrow}_b} q$ is first-order definable"; we shall denote the formula by $[p \mathrel{\underline{\leftrightarrow}_b} q]$.

Lemma 3.1 For each $\mathbb{p}$-ground process term p there exists a first-order Δ-formula $p\!\downarrow$ such that $\mathfrak{D}, \alpha \models p\!\downarrow$ iff $\bar{\alpha}(p) \Longrightarrow \epsilon$, for all valuations α of X in $\mathfrak{D}$.

Proof. By structural induction on p; we give the definitions and leave their straightforward correctness proofs to the reader. We define $\delta\!\downarrow := \bot$, $\tau\!\downarrow := \top$, and $a\!\downarrow := \bot$ for all action terms $a \neq \tau$. If there exist formulae $p_i\!\downarrow$ $(i = 1, 2)$ such that $\mathfrak{D}, \alpha \models p_i\!\downarrow$ iff $\bar{\alpha}(p_i) \Longrightarrow \epsilon$, then we can define $(p_1 + p_2)\!\downarrow := (p_1\!\downarrow \vee p_2\!\downarrow)$, $(p_1 \cdot p_2)\!\downarrow := (p_1\!\downarrow \wedge p_2\!\downarrow)$, $(p_1 \lhd b \rhd p_2)\!\downarrow := (((b \approx \top) \wedge p_1\!\downarrow) \vee ((b \approx \bot) \wedge p_2\!\downarrow))$, and $(\sum_{x:s} p_1)\!\downarrow := (\exists x{:}s)(p_1\!\downarrow)$. □

Theorem 3.2 For all $\mathbb{p}$-ground process terms p and q there exists a first-order Δ-formula φ such that $\mathfrak{D}, \alpha \models \varphi$ iff $\bar{\alpha}(q) \mathrel{\underline{\leftrightarrow}_b} \bar{\alpha}(p)$, for all valuations α of X in $\mathfrak{D}$.

Proof. Identities between action terms and τ are first-order definable. Namely, if $a = \mathsf{a}(t_1, \dots, t_m)$ and $a' = \mathsf{a}'(t'_1, \dots, t'_n)$ with $\mathsf{a}{:}\bar{s}, \mathsf{a}'{:}\bar{s}' \in \mathcal{A} \cup \{\tau\}$, then $\bar{\alpha}(a) = \bar{\alpha}(a')$ iff $\mathsf{a} = \mathsf{a}'$ and $\bar{\alpha}(t_k) = \bar{\alpha}(t'_k)$, for all $1 \leq k \leq m = n$; so we can define

$$[a \mathrel{\underline{\leftrightarrow}_b} a'] \leftrightarrow \begin{cases} \bot & \text{if } \mathsf{a} \neq \mathsf{a}' \\ (t_1 \approx t'_1) \wedge \cdots \wedge (t_m \approx t'_m) & \text{otherwise.} \end{cases}$$

For the proof of this theorem we may assume, by Lemmas 2.2 and 2.8, that p and q are basic terms. So let I and J be disjoint finite sets such that

$$p = \sum_{i \in I} p_i, \qquad q = \sum_{j \in J} q_j, \qquad \text{with } p_i \text{ and } q_j \text{ simple.}$$

We proceed by induction on the sum of the complexities $|p|$ and $|q|$ of p and q respectively, defined as the maximal nesting of $\cdot$, with $|\delta| = 0$ and $|a| = 1$ if a is an action term or τ.
If $|p| + |q| = 0$, then $p = q = \delta$; so $\bar{\alpha}(p) = \bar{\alpha}(q)$ for all valuations α of X in $\mathfrak{D}$, and $p \mathrel{\underline{\leftrightarrow}}_b q$ is first-order definable as the formula $\top$.
For the induction step we distinguish two cases: (1) one of $|p|$ and $|q|$ equals 0; and (2) $|p|, |q| > 0$.

1. Suppose $|p| > 0$, $|q| = 0$ and $p_i = \sum_{\bar{x}_i:\bar{s}_i} a_i p'_i \lhd b_i \rhd \delta$. By induction hypothesis we find that $p'_i \mathrel{\underline{\leftrightarrow}}_b \delta$ is first-order definable, whence so is

$$\varphi = \bigwedge_{i \in I} (\forall \bar{x}_i{:}\bar{s}_i)((b_i \approx \bot) \vee ([a_i \mathrel{\underline{\leftrightarrow}}_b \tau] \wedge [p'_i \mathrel{\underline{\leftrightarrow}}_b \delta])).$$

If $\mathfrak{D}, \alpha \models \varphi$, then $\bar{\alpha}(p) \xrightarrow{\mathbf{a}} \mathbf{p}'$ implies that $\mathbf{a} = \boldsymbol{\tau}$ and $\mathbf{p}' \mathrel{\underline{\leftrightarrow}}_b \boldsymbol{\delta}$, whence $\bar{\alpha}(p) \mathrel{\underline{\leftrightarrow}}_b \boldsymbol{\delta}$. Conversely, if $\mathfrak{D}, \alpha \not\models \varphi$, then there exists a sequence $\bar{d}_i$ of elements of $\mathfrak{D}$ such that $\bar{\alpha}_{[\bar{x}_i:=\bar{d}_i]}(b_i) = \mathsf{T}$, and $\bar{\alpha}_{[\bar{x}_i:=\bar{d}_i]}(a_i) \neq \boldsymbol{\tau}$ or $\bar{\alpha}_{[\bar{x}_i:=\bar{d}_i]}(p'_i) \not\mathrel{\underline{\leftrightarrow}}_b \boldsymbol{\delta}$. Hence $\bar{\alpha}(p) \not\mathrel{\underline{\leftrightarrow}}_b \boldsymbol{\delta}$. We conclude that φ indeed defines $[p \mathrel{\underline{\leftrightarrow}}_b q]$.

2. Next, suppose that $|p|, |q| > 0$, each p_i is as in the previous case, and $q_j = \sum_{\bar{x}_j:\bar{s}_j} a_j q'_j \lhd b_j \rhd \delta$. It suffices to prove that $p \mathrel{\underline{\leftrightarrow}}_b p + q_j$ is first-order definable, for all $j \in J$; for then, by symmetry, $q \mathrel{\underline{\leftrightarrow}}_b q + p_i$ is also first-order definable, for all $i \in I$, and by the identity

$$p \mathrel{\underline{\leftrightarrow}}_b p + \sum_{j \in J} q_j = p + q = \sum_{i \in I} p_i + q \mathrel{\underline{\leftrightarrow}}_b q$$

to conclude that $[p \mathrel{\underline{\leftrightarrow}}_b q] := (\bigwedge_{i \in I} [q \mathrel{\underline{\leftrightarrow}}_b q + p_i]) \wedge (\bigwedge_{j \in J} [p \mathrel{\underline{\leftrightarrow}}_b p + q_j])$ is a correct definition. We now define

$$\begin{aligned}
\varphi_\tau &:= [a_j \mathrel{\underline{\leftrightarrow}}_b \tau] \wedge [q'_j \mathrel{\underline{\leftrightarrow}}_b p], \\
\varphi_\subseteq &:= \bigvee_{i \in I} (\exists \bar{x}_i{:}\bar{s}_i)((b_i \approx \top) \wedge [a_i \mathrel{\underline{\leftrightarrow}}_b \tau] \wedge [p'_i \mathrel{\underline{\leftrightarrow}}_b p'_i + q_j]), \text{ and} \\
\varphi_\to &:= \bigvee_{i \in I} (\exists \bar{x}_i{:}\bar{s}_i)((b_i \approx \top) \wedge [a_i \mathrel{\underline{\leftrightarrow}}_b a_j] \wedge [p'_i \mathrel{\underline{\leftrightarrow}}_b q'_j]).
\end{aligned}$$

We complete the proof by verifying that the definition

$$[p \mathrel{\underline{\leftrightarrow}}_b p + q_j] := [(\forall \bar{x}_j{:}\bar{s}_j)(b_j \approx \top) \to (\varphi_\tau \vee \varphi_\subseteq \vee \varphi_\to)] \wedge (q_j \downarrow \to p \downarrow)$$

is correct. Note that by induction hypothesis and Lemma 3.1 it is a well-formed Δ-formula. It remains to verify that $\mathfrak{D}, \alpha \models [p \mathrel{\underline{\leftrightarrow}}_b p + q_j]$ if, and only if, $\bar{\alpha}(p) \mathrel{\underline{\leftrightarrow}}_b \bar{\alpha}(p) \boldsymbol{+} \bar{\alpha}(q_j)$.

($\Rightarrow$) Suppose $\mathfrak{D}, \alpha \models [p \mathrel{\underline{\leftrightarrow}}_b p + q_j]$ and $\bar{\alpha}(q_j) \xrightarrow{\mathbf{a}} \mathbf{q}'$. Then there exists a sequence $\bar{d}_j$ of elements of $\mathfrak{D}$ such that

$$\bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(a_j) = \mathbf{a},\ \bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(q'_j) = \mathbf{q}',\ \text{and } \bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(b_j) = \mathsf{T}.$$

If $\mathfrak{D}, \alpha \models \varphi_\tau$, then $\mathbf{a} = \boldsymbol{\tau}$ and $\mathbf{q}' \mathrel{\underline{\leftrightarrow}}_b \bar{\alpha}(p)$, and if $\mathfrak{D}, \alpha \models (\varphi_\subseteq \vee \varphi_\to)$, then $\bar{\alpha}(p) \Longrightarrow \mathbf{p}^* \xrightarrow{\mathbf{a}} \mathbf{p}'$ such that $\mathbf{p}^* \mathrel{\underline{\leftrightarrow}}_b \mathbf{p}^* + \bar{\alpha}(q_j)$, and $\mathbf{q}' \mathrel{\underline{\leftrightarrow}}_b \mathbf{p}'$. Furthermore, $\bar{\alpha}(p) \Longrightarrow \epsilon$ iff $\bar{\alpha}(p + q_j) \Longrightarrow \epsilon$, since $\mathfrak{D}, \alpha \models (q_j \downarrow \to p \downarrow)$. Hence $\bar{\alpha}(p) \mathrel{\underline{\leftrightarrow}}_b \bar{\alpha}(p) \boldsymbol{+} \bar{\alpha}(q_j)$.

($\Leftarrow$) Fix an arbitrary sequence $\bar{d}_j$ of elements of $\mathfrak{D}$ such that $\bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(b_j) = \mathsf{T}$; then $\bar{\alpha}(q_j) \xrightarrow{\mathbf{a}} \mathbf{q}'$, with $\mathbf{a} = \bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(a_j)$ and $\mathbf{q}' = \bar{\alpha}_{[\bar{x}_j:=\bar{d}_j]}(p'_j)$. So if $\bar{\alpha}(p) \mathrel{\underline{\leftrightarrow}}_b \bar{\alpha}(p) \boldsymbol{+} \bar{\alpha}(q_j)$, then either $\mathbf{a} = \boldsymbol{\tau}$ and $\mathbf{q}' \mathrel{\underline{\leftrightarrow}}_b \bar{\alpha}(p)$, whence $\mathfrak{D}, \alpha \models \varphi_\tau$, or $\bar{\alpha}(p) \Longrightarrow \mathbf{p}^* \xrightarrow{\mathbf{a}} \mathbf{p}'$ such that $\mathbf{q}' \mathrel{\underline{\leftrightarrow}}_b \mathbf{p}'$. In the latter case we apply Lemma 2.6 to conclude that $\mathfrak{D}, \alpha \models (\varphi_\subseteq \vee \varphi_\to)$. Moreover, it is clear that $\mathfrak{D}, \alpha \models (q_j \downarrow \to p \downarrow)$. Hence $\mathfrak{D}, \alpha \models [p \mathrel{\underline{\leftrightarrow}}_b p + q_j]$, and the theorem follows. □

4. Completeness

We shall now prove that, under certain restrictions imposed on $\mathfrak{D}$, the branching bisimulation algebra can be equationally axiomatised. We need that every first-order Δ-formula is logically equivalent to an equation of the form $b \approx \top$. We shall achieve this by assuming that $\mathfrak{D}$ has built-in equality and Skolem functions, as defined below.

Definition 4.1 (Equality) A Δ-algebra $\mathfrak{D}$ has *built-in equality* for sort s iff there exists a boolean Δ-term $[x =_s y]$ in variables x and y of sort s such that for all valuations α of X in $\mathfrak{D}$

$$\bar{\alpha}([x =_s y]) = \begin{cases} \top & \text{if } \alpha(x) = \alpha(y) \\ \bot & \text{otherwise.} \end{cases}$$

A Δ-algebra has *built-in equality* if it has built-in equality for all its sorts.

Definition 4.2 (Skolem Functions) A Δ-algebra $\mathfrak{D}$ has *built-in Skolem functions* if for every Δ-formula φ with $FV(\varphi) = \{x, y_1, \ldots, y_n\}$ there exists a term $t_\varphi = t_\varphi(y_1, \ldots, y_n)$ such that for every valuation α of X in $\mathfrak{D}$

$$\mathfrak{D}, \alpha \models (\exists x{:}s)\varphi \quad \text{implies} \quad \mathfrak{D}, \alpha \models \varphi(t_\varphi(y_1, \ldots, y_n), y_1, \ldots, y_n).$$

The term t_φ shall be called a *Skolem function for x*.

The following proposition follows easily by structural induction on Δ-formulae.

Proposition 4.3 If $\mathfrak{D}$ has built-in equality and Skolem functions, then for every first-order Δ-formula φ there exists a boolean term $\varphi^{\mathbb{b}}$ such that $\mathfrak{D}, \alpha \models \varphi$ iff $\mathfrak{D}, \alpha \models \varphi^{\mathbb{b}} \approx \top$.

(Ae$_{\mathsf{a}}$)	$\mathsf{a}(\bar{x}) \lhd [\bar{x} = \bar{y}] \rhd \delta$	$\approx$	$\mathsf{a}(\bar{y}) \lhd [\bar{x} = \bar{y}] \rhd \delta$
(Sca)	$(xy \lhd b \rhd \delta)$	$\approx$	$(x \lhd b \rhd \delta)(y \lhd b \rhd \delta)$

Table 3: We define Ae $= \{\text{Ae}_{\mathsf{a}} \mid \mathsf{a} \in \mathcal{A}\}$; if $\mathfrak{D}$ has a two-element boolean algebra and built-in equality for $\mathcal{A}$, then Ae and Sca hold in the strong bisimulation algebra for $\mathfrak{D}$ and $\mathcal{A}$.

If $\mathfrak{D}$ has built-in equality and Skolem functions, then the first-order theory of $\mathfrak{D}$ is decidable, and so is its equational theory $EqTh(\mathfrak{D})$ (i.e., the set of all equations that hold of $\mathfrak{D}$). In order to arrive at a complete set of axioms for strong bisimulation we should add the axioms depicted in Table 3. Let $\Pi_\Sigma(\mathfrak{D}) = p\mathsf{CRL} + EqTh(\mathfrak{D}) + \text{Ae} + \text{Sca}$. Groote and Luttik (1998) proved that this set ground axiomatises the strong bisimulation algebra.

Theorem 4.4 If $\mathfrak{D}$ has built-in equality and Skolem functions, then $\mathfrak{P}/\underline{\leftrightarrow} \models p \approx q$ iff $\Pi_\Sigma(\mathfrak{D}) \vdash p \approx q$, for all $\mathbb{p}$-ground process terms p and q.

Below we shall prove that it suffices to add the τ-laws B1–2 to axiomatise the branching bisimulation algebra. Our proof consists in showing that every $\mathbb{p}$-ground process term is provably equal to a basic term the residuals of which are all compact. Then we make use of the fact that $\underline{\leftrightarrow}$ and $\underline{\leftrightarrow}_{rb}$ coincide on the set of processes whose residuals are compact (Lemma 2.10); for this set completeness follows by Theorem 4.4.

First, we prove two useful consequences of the compactness lemma (Lemma 2.10) and the completeness theorem for strong bisimulation (Theorem 4.4): basic terms may be split into a compact and an inert part, and moreover, we may assume that, in certain circumstances, this inert part is of a simple form. First, we define the notions of compactness and inertness for process terms.

We call a $\mathbb{p}$-ground process term p *compact* if $\bar{\alpha}(p)$ is compact, for all valuations α of X in $\mathfrak{D}$. If all residuals of $\bar{\alpha}(p)$ are compact for all valuations α of X in $\mathfrak{D}$, then we shall say that p has compact residuals. If q is a $\mathbb{p}$-ground process term and, for all valuations α of X in $\mathfrak{D}$, $\bar{\alpha}(p) \xrightarrow{\mathbf{a}} \mathbf{p}'$ implies $\mathbf{a} = \boldsymbol{\tau}$ and $\mathbf{p}' \underline{\leftrightarrow}_b \bar{\alpha}(q)$, then we say that p is *q-inert*; we define that p is *ϵ-inert* if $\bar{\alpha}(p) \xrightarrow{\mathbf{a}} \mathbf{p}'$ implies $\mathbf{a} = \boldsymbol{\tau}$ and $\mathbf{p}' \underline{\leftrightarrow}_b \epsilon$.

Lemma 4.5 Suppose that $\mathfrak{D}$ has built-in equality and Skolem functions.
If p is a basic term with compact residuals, then there exist a compact basic term p^c and a p-inert basic term p^τ such that $\Pi_\Sigma(\mathfrak{D}) \vdash p \approx p^c + p^\tau$.

Proof. Let I be a finite set such that

$$p = \sum_{i \in I} p_i, \qquad \text{with } p_i = \textstyle\sum_{\bar{x}_i:\bar{s}_i} a_i p_i' \lhd b_i \rhd \delta.$$

We show that each p_i with $a_i = \tau$ can be split into a compact part p_i^c and a p_i-inert part p_i^τ. By Theorem 3.2 and Proposition 4.3 there exists a boolean term b_i' such that $\bar{\alpha}(b_i') = \mathsf{T}$ iff $\bar{\alpha}(p) \underline{\leftrightarrow}_b \bar{\alpha}(p_i')$, for any valuation α of X in $\mathfrak{D}$. Consequently, the transition $\bar{\alpha}(p_i) \xrightarrow{\boldsymbol{\tau}} \bar{\alpha}(p_i')$ is inert iff $\bar{\alpha}(b_i') = \mathsf{T}$. So $p_i^c = \sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd b_i \wedge \neg b_i' \rhd \delta$ is compact and $p_i^\tau = \sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd b_i \wedge b_i' \rhd \delta$ is p_i-inert, and we derive

$$\begin{aligned}
p_i &\approx \textstyle\sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd (b_i \wedge \neg b_i') \vee (b_i \wedge b_i') \rhd \delta && (\mathsf{BA})\\
&\approx \textstyle\sum_{\bar{x}_i:\bar{s}_i} (\tau p_i' \lhd b_i \wedge \neg b_i' \rhd \delta + \tau p_i' \lhd b_i \wedge b_i' \rhd \delta) && (\textsc{Cond5})\\
&\approx \textstyle\sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd b_i \wedge \neg b_i' \rhd \delta + \sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd b_i \wedge b_i' \rhd \delta && (\textsc{Sum4})\\
&= p_i^c + p_i^\tau.
\end{aligned}$$

□

Lemma 4.6 Suppose $\mathfrak{D}$ has built-in equality and Skolem functions, and let q be a compact process term or ϵ. If p is q-inert, and p has compact residuals, then there exists a boolean term b^τ such that $\Pi_\Sigma(\mathfrak{D}) \vdash p \approx \tau q \lhd b^\tau \rhd \delta$.

Proof. We may assume by Lemma 2.2 that

$$p = \sum_{i \in I} p_i, \qquad \text{with } p_i = \textstyle\sum_{\bar{x}_i:\bar{s}_i} \tau p_i' \lhd b_i \rhd \delta.$$

If $q = \epsilon$, then we may assume that each $p_i' = \epsilon$, while if q is compact then we may assume that $p_i' \neq \epsilon$ for all $i \in I$. By induction on the length of the sequence $\bar{x}_i$, we get a sequence $\bar{t}_i$ of Skolem functions for the $\bar{x}_i$ such that $\bar{\alpha}(b_i[\bar{x}_i := \bar{t}_i]) = \mathsf{T}$ iff $\mathfrak{D}, \alpha \models (\exists \bar{x}_i{:}\bar{s}_i)(b_i \approx \mathsf{T})$.

Claim $\bar{\alpha}(p_i) \underline{\leftrightarrow} \bar{\alpha}(\tau q \lhd b_i[\bar{x}_i := \bar{t}_i] \rhd \delta)$, for any valuation α of X in $\mathfrak{D}$.

Proof. If $\bar{\alpha}(p_i) \xrightarrow{\boldsymbol{\tau}} \mathbf{p}'$, then there exists a sequence $\bar{d}_i$ of elements of $\mathfrak{D}$ such that $\bar{\alpha}_{[\bar{x}_i := \bar{d}_i]}(b_i) = \mathsf{T}$. Hence $\bar{\alpha}(b_i[\bar{x}_i := \bar{t}_i]) = \mathsf{T}$ and $\bar{\alpha}(\tau q \lhd b_i[\bar{x}_i := \bar{t}_i] \rhd \delta) \xrightarrow{\boldsymbol{\tau}} \mathbf{q}$. If $\mathbf{p}' = \mathbf{q} = \epsilon$, then there is nothing to prove; if $\mathbf{p}'$ and $\mathbf{q}$ are compact, then since they are branching bisimilar, we conclude with Lemma 2.10 that $\mathbf{p}' \underline{\leftrightarrow} \mathbf{q}$. Conversely, if $\bar{\alpha}(\tau q \lhd b_i[\bar{x}_i := \bar{t}_i] \rhd \delta) \xrightarrow{\boldsymbol{\tau}} \mathbf{q}$, then there exists a sequence $\bar{d}_i$ of elements of $\mathfrak{D}$ such that $\bar{\alpha}_{[\bar{x}_i := \bar{d}_i]}(b_i) = \mathsf{T}$. Hence $\bar{\alpha}(p_i) \xrightarrow{\boldsymbol{\tau}} \mathbf{p}'$, for some $\mathbf{p}'$. Again, if $\mathbf{p}' = \mathbf{q} = \epsilon$, then we are done, and if $\mathbf{p}'$ and $\mathbf{q}$ are compact, then we find $\mathbf{p}' \underline{\leftrightarrow} \mathbf{q}$ in the same way as above.

Hence it follows by Theorem 4.4 that $\Pi_\Sigma(\mathfrak{D}) \vdash p_i \approx \tau q \lhd b[\bar{x}_i := \bar{t}_i] \rhd \delta$; so we define

$$b^\tau = \bigvee_{i \in I} b_i[\bar{x}_i := \bar{t}_i]$$

and derive with Cond5

$$p = \sum_{i \in I} p_i \approx \sum_{i \in I} \tau q \lhd b_i[\bar{x}_i := \bar{t}_i] \rhd \delta \approx \tau q \lhd \bigvee_{i \in I} b_i[\bar{x}_i := \bar{t}_i] \rhd \delta = \tau q \lhd b^\tau \rhd \delta.$$

□

Let $\Pi^{\tau}_{\Sigma}(\mathfrak{D}) = p\mathsf{CRL}_{\tau}+EqTh(\mathfrak{D})+\textsc{Ae}+\textsc{Sca}$.

Proposition 4.7 The following equalities are derivable from $\Pi^{\tau}_{\Sigma}(\mathfrak{D})$:

i. $x(\tau \lhd b \rhd \delta) \approx x \lhd b \rhd \delta + x\delta \lhd \neg b \rhd \delta$; and

ii. $x(\tau(y+z) \lhd b \rhd \delta + z) \approx x(y \lhd b \rhd \delta + z)$.

Proof. We derive the first equality; the derivation of the second equality goes in a similar fashion (with an application of B2 instead of B1).

$$\begin{aligned} x(\tau \lhd b \rhd \delta) &\approx x(\tau \lhd b \rhd \delta) \lhd b \rhd \delta + x(\tau \lhd b \rhd \delta) \lhd \neg b \rhd \delta && (\textsc{Cond}1,5,\mathsf{BA})\\ &\approx x\tau \lhd b \rhd \delta + x\delta \lhd \neg b \rhd \delta && (\textsc{Sca},\textsc{Cond}2,4,6,\mathsf{BA})\\ &\approx x \lhd b \rhd \delta + x\delta \lhd \neg b \rhd \delta && (\text{B1}) \qquad \Box \end{aligned}$$

Now we are in a position to prove our main theorem.

Theorem 4.8 If $\mathfrak{D}$ has built-in equality and Skolem functions, then for every $\mathbb{p}$-ground process term p there exists a basic term q such that $\Pi^{\tau}_{\Sigma}(\mathfrak{D}) \vdash p \approx q$ and q has compact residuals.

Proof. We may assume, by Lemmas 2.2 and 2.8, that p is a basic term.

We shall prove the theorem by induction on the complexity of $|p|$. Suppose $|p| > 0$; we need to show that each of p's summands is provably equal to a basic term that has compact residuals. Hence, it suffices to consider the case where p is simple, i.e., let

$$p = \textstyle\sum_{\bar{x}:\bar{s}} ap^* \lhd b \rhd \delta.$$

From the induction hypothesis, we get that p^* has compact residuals, so with Lemma 4.5 we can split p^* into a compact part p^c and a p-inert part p^{τ} such that $p^* \approx p^c + p^{\tau}$; let I be a finite set such that

$$p^{\tau} = \sum_{i \in I} p_i, \qquad \text{with } p_i = \textstyle\sum_{\bar{x}_i:\bar{s}_i} \tau p'_i \lhd b_i \rhd \delta.$$

Now it suffices to distinguish two cases: (1) $p^c = \delta$ and $p'_i = \epsilon$ for all $i \in I$, or (2) $p'_i \neq \epsilon$ for all $i \in I$; namely, we may derive

$$p \approx \textstyle\sum_{\bar{x}:\bar{s}} ap^* \lhd b \wedge (p^* \downarrow) \rhd \delta + \textstyle\sum_{\bar{x}:\bar{s}} ap^* \lhd b \wedge \neg(p^* \downarrow) \rhd \delta,$$

and by means of $\textsc{Sca}$, $\textsc{Cond}4,7$, $\textsc{Sum}12$, and BA we cancel all the p_i with $p'_i \neq \epsilon$ in the left summand and those with $p'_i = \epsilon$ in the right summand.

1. If $p^c = \delta$, then p^{τ} is ϵ-inert, so, by A6 and Lemma 4.6, there exists a boolean term b^{τ} such that $p^* \approx \tau \lhd b^{\tau} \rhd \delta$. We apply Proposition 4.7(i) and derive

$$p \approx \textstyle\sum_{\bar{x}:\bar{s}} a(\tau \lhd b^{\tau} \rhd \delta) \lhd b \rhd \delta \approx \textstyle\sum_{\bar{x}:\bar{s}} a \lhd b \wedge b^{\tau} \rhd \delta + \textstyle\sum_{\bar{x}:\bar{s}} a\delta \lhd b \wedge \neg b^{\tau} \rhd \delta,$$

and all residuals of this latter basic term are compact.

2. Suppose that $p'_i \neq \epsilon$ for all $i \in I$. We define

$$p^{\dagger} = p^c + \sum_{i \in I} \textstyle\sum_{\bar{x}_i:\bar{s}_i} p'_i \lhd b_i \rhd \delta.$$

Since $\bar{\alpha}(p^{\dagger}) \leftrightarroweq_b \bar{\alpha}(p^*)$, p^{τ} is $p^{\dagger}$-inert. Moreover, $p^{\dagger}$ is compact, so by Lemma 4.6 there exists a boolean term b^{τ} such that $\Pi_{\Sigma}(\mathfrak{D}) \vdash p^* \approx p^c + \tau p^{\dagger} \lhd b^{\tau} \rhd \delta$. We derive

$$\begin{aligned} p &\approx \textstyle\sum_{\bar{x}:\bar{s}} a(\tau p^{\dagger} \lhd b^{\tau} \rhd \delta + p^c) \lhd b \rhd \delta \\ &\approx \textstyle\sum_{\bar{x}:\bar{s}} a(\tau(p^{\dagger} + p^c) \lhd b^{\tau} \rhd \delta + p^c) \lhd b \rhd \delta && (\text{A1–3})\\ &\approx \textstyle\sum_{\bar{x}:\bar{s}} a(p^{\dagger} \lhd b^{\tau} \rhd \delta + p^c) \lhd b \rhd \delta && (\text{Prop 4.7(ii)}). \end{aligned}$$

All residuals of this latter term are compact, and we can transform it to a basic term according to Lemma 2.2. Since this transformation does not involve applications of B1 and B2, the result is compact, and the proof is complete. □

We now obtain that $\Pi^{\tau}_{\Sigma}(\mathfrak{D})$ is an axiomatisation of $\mathfrak{P}/\underline{\leftrightarrow}_{rb}$ as an easy consequence of Theorems 4.8 and 4.4.

Corollary 4.9 If $\mathfrak{D}$ has built-in equality and Skolem functions, then $\mathfrak{P}/\underline{\leftrightarrow}_{rb} \models p \approx q$ iff $\Pi^{\tau}_{\Sigma}(\mathfrak{D}) \vdash p \approx q$, for all $\mathbb{p}$-ground process terms p and q.

Proof. The implication from right to left is by Lemma 2.8. We may, by Theorem 4.8, assume that p and q have compact residuals, so the other direction follows from Lemma 2.10 and Theorem 4.4. □

References

Basten, T. (1996). Branching bisimilarity is an equivalence indeed! *Information Processing Letters*, **58**(3), 141–147.

Chang, C. C. and Keisler, H. J. (1990). *Model Theory*, volume 73 of *Studies in logic and the foundations of mathematics*. North-Holland, Amsterdam - New York - Oxford - Tokyo, 3rd edition.

Van Glabbeek, R. J. and Weijland, W. P. (1996). Branching time and abstraction in bisimulation semantics. *Journal of the ACM*, **43**(3), 555–600.

Groote, J. F. and Luttik, S. P. (1998). Undecidability and completeness results for process algebras with alternative quantification over data. Report SEN-R9806, CWI, The Netherlands. Available from `http://www.cwi.nl/~luttik/`.

Groote, J. F. and Ponse, A. (1994). Proof theory for μCRL: A language for processes with data. In D. J. Andrews, J. F. Groote, and C. A. Middelburg, editors, *Proceedings of the International Workshop on Semantics of Specification Languages*, Workshops in Computing, pages 232–251, Utrecht, The Netherlands. Springer-Verlag.

Hennessy, M. and Lin, H. (1996). Proof systems for message-passing process algebras. *Formal Aspects of Computing*, **8**(4), 379–407.

Klusener, A. S. (1992). The silent step in time. In W. R. Cleaveland, editor, *Proceedings of CONCUR'92*, volume 630 of *Lecture Notes in Computer Science*, pages 421–435. Springer.

Koppelberg, S. (1989). Elementary arithmetic. In J. D. Monk and R. Bonnet, editors, *Handbook of Boolean Algebras (Vol. I)*, pages 5–46. North-Holland.

Lin, H. (1995). Complete inference systems for weak bisimulation equivalences in the π-calculus. In P. D. Mosses, M. Nielsen, and M. I. Swarzbach, editors, *Proceedings of TAPSOFT'95*, volume 915 of *Lecture Notes in Computer Science*, pages 187–201. Springer.

McKenzie, R. N., McNulty, G. F., and Taylor, W. F. (1987). *Algebras, Lattices, Varieties — Volume I*. Wadsworth & Brooks/Cole, Monterey, California.

Milner, R. (1980). *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer.

Parrow, J. and Victor, B. (1998). The tau-laws of fusion. In D. Sangiorgi and R. De Simone, editors, *Proceedings of CONCUR'98*, volume 1466 of *Lecture Notes in Computer Science*, pages 99–114. Springer.