A Finite Equational Base for CCS with Left Merge and Communication Merge

LUCA ACETO
Reykjavík University
WAN FOKKINK
Vrije Universiteit Amsterdam and CWI
ANNA INGOLFSDOTTIR
Reykjavík University
and
BAS LUTTIK
Technische Universiteit Eindhoven and CWI

Using the left merge and the communication merge from ACP, we present an equational base (i.e., a ground-complete and ω-complete set of valid equations) for the fragment of CCS without recursion, restriction and relabelling modulo (strong) bisimilarity. Our equational base is finite if the set of actions is finite.

Categories and Subject Descriptors: D.3.1 [**Programming Languages**]: Formal Definitions and Theory—*Semantics*; F.1.2 [**Computation by Abstract Devices**]: Modes of Computation—*Nondeterminism*; *Parallelism and Concurrency*; F.3.2 [**Logics and Meanings of Programs**]: Semantics of Programming Languages; F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic—*Proof Theory*

General Terms: Languages, Theory, Verification

Additional Key Words and Phrases: Bisimilarity, CCS, communication merge, concurrency, finite equational base, handshaking, left merge, parallel composition, process algebra

Author's addresses: L. Aceto, School of Computer Science, Reykjavík University, Kringlan 1, 103 Reykjavík, Iceland, email: luca@ru.is. W. Fokkink, Section Theoretical Computer Science, Department of Computer Science, Vrije Universiteit Amsterdam, De Boelelaan 1081a, 1081 HV Amsterdam, The Netherlands, email: wanf@cs.vu.nl. A. Ingolfsdottir, School of Computer Science, Reykjavík University, Kringlan 1, 103 Reykjavík, Iceland, email: annai@ru.is. B. Luttik, Department of Computer Science, Technische Universiteit Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands, email: s.p.luttik@tue.nl.

The first and third author were partly supported by the project "The Equational Logic of Parallel Processes" (nr. 060013021) of The Icelandic Research Fund.

An extended abstract of this article appeared in *Proceedings of ICALP 2006*, Lecture Notes in Computer Science, vol. 4052. Springer, 492-503.

1. INTRODUCTION

One of the first detailed studies of the equational theory of a process algebra was carried out by Hennessy and Milner [1985]. They considered the equational theory of the process algebra that arises from the recursion-free fragment of CCS [Milner 1989a], and presented a set of equational axioms that is complete in the sense that all valid *closed* equations (i.e., equations in which no variables occur) are derivable from it in equational logic. For the elimination of parallel composition from closed terms, Hennessy and Milner proposed the well-known *Expansion Law*, an axiom schema that generates infinitely many axioms. Thus, the question arose whether a finite complete set of axioms exists. With their axiom system ACP, Bergstra and Klop [1984] demonstrated that it exists if two auxiliary operators are used: the left merge and the communication merge. It was later proved by Moller [1990] that without using at least one auxiliary operator a finite complete set of axioms does not exist.

The aforementioned results pertain to the closed fragments of the equational theories discussed, i.e., to the subsets consisting of the closed valid equations only. Many valid equations, such as, e.g., the equation $(x \parallel y) \parallel z \approx x \parallel (y \parallel z)$ expressing that parallel composition is associative, are not derivable (by means of equational logic) from the axioms in [Bergstra and Klop 1984] or [Hennessy and Milner 1985]. In this paper we shall not neglect the variables and contribute to the study of full equational theories of process algebras. We take the fragment of CCS without recursion, restriction and relabelling, and consider the full equational theory of the process algebra that is obtained by taking the syntax modulo (strong) bisimilarity [Park 1981]. Our goal is then to present an *equational base* (i.e., a set of valid equations from which every other valid equation can be derived) for it, which is finite if the set of actions is finite. Obviously, Moller's result about the unavoidability of the use of auxiliary operations in a finite complete axiomatisation of the closed fragment of the equational theory of CCS a fortiori implies that auxiliary operations are needed to achieve our goal. So we add the left merge and the communication merge from the start.

Moller [1989] considers the equational theory of the same fragment of CCS, except that his parallel operator implements pure interleaving instead of CCS-communication and the communication merge is omitted. He presents a set of valid axiom schemata and proves that it generates an equational base provided that the set of actions is infinite. Groote [1990] does consider the fragment including the communication merge, but, instead of the CCS-communication mechanism, he assumes an uninterpreted communication function. His axiom schemata also generate an equational base provided that the set of actions is infinite. We improve on these results by considering the communication mechanism present in CCS, and by proving that our axiom schemata generate an equational base also if the set of actions is finite. Moreover, our axiom schemata generate a finite equational base if the set of actions is finite.

Our equational base consists of axioms that are mostly well-known. For parallel composition ($\parallel$), left merge ($\mathbin{|\!\!\lfloor}$) and communication merge ($|$) we adapt the axioms of ACP, adding from Bergstra and Tucker [1985] a selection of the axioms for *standard concurrency* and the axiom $(x \mid y) \mid z \approx \mathbf{0}$, which expresses that the communication mechanism is a form of *handshaking communication*.

Our proof follows the classic two-step approach: first we identify a set of normal forms such that every process term has a provably equal normal form, and then we demonstrate that for distinct normal forms there is a distinguishing valuation that proves that they should not be equated. (We refer to the survey [Aceto et al. 2005b] for a discussion of proof techniques and for an overview of results and open problems in the area. We remark in passing that one of our main results in this paper, viz. Corollary 4.10, solves the open problem mentioned in [Aceto et al. 2005b, p. 362].) Since both associating a normal form with a process term and determining a distinguishing valuation for two distinct normal forms are easily seen to be computable, as a corollary to our proof we get the decidability of the equational theory. Another consequence of our result is that our equational base is complete for the set of valid closed equations as well as ω-complete [Heering 1986].

The positive result that we obtain in Corollary 4.10 of this paper stands in contrast with the negative result that we have obtained in [Aceto et al. 2005a]. In that article we proved that there does not exist a finite equational base for CCS if the auxiliary operation $|\!/$ of Hennessy [1988] is added instead of Bergstra and Klop's left merge and communication merge. Furthermore, we conjecture that a finite equational base fails to exist if the unary action prefixes are replaced by binary sequential composition. (We refer to [Aceto et al. 2005b] for an infinite family of valid equations that we believe cannot all be derivable from a single finite set of valid equations.)

The paper is organised as follows. In Sect. 2 we introduce a class of algebras of processes arising from a process calculus à la CCS, present a set of equations that is valid in all of them, and establish a few general properties needed in the remainder of the paper. Our class of process algebras is parametrised by a communication function. It is beneficial to proceed in this generality, because it allows us to first consider the simpler case of a process algebra with pure interleaving (i.e., no communication at all) instead of CCS-like parallel composition. In Sect. 3 we prove that an equational base for the process algebra with pure interleaving is obtained by simply adding the axiom $x \mid y \approx \mathbf{0}$ to the set of equations introduced in Sect. 2. The proof in Sect. 3 extends nicely to a proof that, for the more complicated case of CCS-communication, it is enough to replace $x \mid y \approx \mathbf{0}$ by $x \mid (y \mid z) \approx \mathbf{0}$; this is discussed in Sect. 4. We end the paper in Sect. 5 with some concluding remarks, a discussion of related work, and some comments on the complications that arise when trying to extend our results to fragments of CCS including restriction, relabelling and recursion, and to adapt our proof for CCS modulo observation congruence.

2. ALGEBRAS OF PROCESSES

We fix a set $\mathcal{A}$ of *actions*, and declare a special action τ that we assume is not in $\mathcal{A}$. We denote by $\mathcal{A}_\tau$ the set $\mathcal{A} \cup \{\tau\}$. Generally, we let a and b range over $\mathcal{A}$ and α over $\mathcal{A}_\tau$. We also fix a countably infinite set $\mathcal{V}$ of *variables*. The set $\mathcal{P}$ of *process terms* is generated by the following grammar:

$$P ::= \; x \;\mid\; \mathbf{0} \;\mid\; \alpha.P \;\mid\; P + P \;\mid\; P \mathbin{\|\!\!\lfloor} P \;\mid\; P \,|\, P \;\mid\; P \,\|\, P \;,$$

with $x \in \mathcal{V}$, and $\alpha \in \mathcal{A}_\tau$. We shall often simply write α instead of $\alpha.\mathbf{0}$. Furthermore, to be able to omit some parentheses when writing terms, we adopt the convention

Table I. The operational semantics.

$$\frac{}{\alpha.P \xrightarrow{\alpha} P} \qquad \frac{P \xrightarrow{\alpha} P'}{P + Q \xrightarrow{\alpha} P'} \qquad \frac{Q \xrightarrow{\alpha} Q'}{P + Q \xrightarrow{\alpha} Q'}$$

$$\frac{P \xrightarrow{\alpha} P'}{P \mathbin{\lfloor\!\!\lfloor} Q \xrightarrow{\alpha} P' \parallel Q} \qquad \frac{P \xrightarrow{\alpha} P'}{P \parallel Q \xrightarrow{\alpha} P' \parallel Q} \qquad \frac{Q \xrightarrow{\alpha} Q'}{P \parallel Q \xrightarrow{\alpha} P \parallel Q'}$$

$$\frac{P \xrightarrow{a} P',\ Q \xrightarrow{b} Q',\ \gamma(a,b){\downarrow}}{P \mid Q \xrightarrow{\gamma(a,b)} P' \parallel Q'} \qquad \frac{P \xrightarrow{a} P',\ Q \xrightarrow{b} Q',\ \gamma(a,b){\downarrow}}{P \parallel Q \xrightarrow{\gamma(a,b)} P' \parallel Q'}$$

that $\alpha.$ binds stronger and $+$ binds weaker than all the other operations. A term of the form $P + Q$ is referred to as *alternative composition* and one of the form $P \parallel Q$ as *parallel composition.* We shall also use the generalised summation operator, inductively defined on a sequence of process terms $P_1, \ldots, P_n$ as follows:

$$\sum_{i=1}^{n} P_i = \begin{cases} \mathbf{0} & \text{if } n = 0, \\ P_1 & \text{if } n = 1, \text{ and} \\ \displaystyle\sum_{i=1}^{n-1} P_i + P_n & \text{if } n > 1. \end{cases}$$

A process term is *closed* if it does not contain variables; we denote the set of all closed process terms by $\mathcal{P}_0$. We define on $\mathcal{P}_0$ binary relations $\xrightarrow{\alpha}$ ($\alpha \in \mathcal{A}_\tau$) by means of the transition system specification in Table I. The last two rules in Table I refer to a *communication function* γ, i.e., a commutative and associative partial binary function $\gamma : \mathcal{A} \times \mathcal{A} \rightharpoonup \mathcal{A}_\tau$. We shall abbreviate the statement '$\gamma(a,b)$ is defined' by $\gamma(a,b){\downarrow}$ and the statement '$\gamma(a,b)$ is undefined' by $\gamma(a,b){\uparrow}$. We shall in particular consider the following communication functions:

(1) The *trivial communication function* is the partial function $f : \mathcal{A} \times \mathcal{A} \rightharpoonup \mathcal{A}_\tau$ such that $f(a,b){\uparrow}$ for all $a, b \in \mathcal{A}$.

(2) The CCS *communication function* $h : \mathcal{A} \times \mathcal{A} \rightharpoonup \mathcal{A}_\tau$ presupposes a bijection $\bar{\cdot}$ on $\mathcal{A}$ such that $\bar{\bar{a}} = a$ and $\bar{a} \neq a$ for all $a \in \mathcal{A}$, and is then defined by $h(a,b) = \tau$ if $\bar{a} = b$ and undefined otherwise.

Definition 2.1. A *bisimulation* is a symmetric binary relation $\mathcal{R}$ on $\mathcal{P}_0$ such that $P \mathrel{\mathcal{R}} Q$ implies

if $P \xrightarrow{\alpha} P'$, then there exists $Q' \in \mathcal{P}_0$ such that $Q \xrightarrow{\alpha} Q'$ and $P' \mathrel{\mathcal{R}} Q'$.

Closed process terms $P, Q \in \mathcal{P}_0$ are said to be *bisimilar* (notation: $P \mathrel{\underline{\leftrightarrow}}_\gamma Q$) if there exists a bisimulation $\mathcal{R}$ such that $P \mathrel{\mathcal{R}} Q$.

The relation $\mathrel{\underline{\leftrightarrow}}_\gamma$ is an equivalence relation on $\mathcal{P}_0$; we denote the equivalence class containing P by $[P]$, i.e.,

$$[P] = \{Q \in \mathcal{P}_0 : P \mathrel{\underline{\leftrightarrow}}_\gamma Q\} \ .$$

If, in Table I, P, P', Q and Q' are treated as variables ranging over closed process terms and the last two rules are treated as rule schemata generating a rule for all

a, b such that $\gamma(a,b)\downarrow$, then the rules in Table I are all in the format of de Simone [1985]. Hence, $\leftrightarrows_\gamma$ is compatible with the syntactic constructs of our language of closed process terms, and therefore the constructs induce an algebraic structure on $\mathcal{P}_0/{\leftrightarrows_\gamma}$, with a constant $\mathbf{0}$, unary operations $\alpha.$ ($\alpha \in \mathcal{A}_\tau$) and four binary operations $+$, $\mathbin{\|\!\lfloor}$, $|$ and $\|$ defined by

$$\begin{array}{ll}
\mathbf{0} = [\mathbf{0}] & [P] \mathbin{\|\!\lfloor} [Q] = [P \mathbin{\|\!\lfloor} Q] \\
\alpha.[P] = [\alpha.P] & [P] \mid [Q] = [P \mid Q] \\
[P] + [Q] = [P + Q] & [P] \parallel [Q] = [P \parallel Q] \ .
\end{array}$$

Henceforth, we denote by $\mathbf{P}_\gamma$ (for γ an arbitrary communication function) the algebra obtained by dividing out $\leftrightarrows_\gamma$ on $\mathcal{P}_0$ with constant $\mathbf{0}$ and operations $\alpha.$ ($\alpha \in \mathcal{A}_\tau$), $+$, $\mathbin{\|\!\lfloor}$, $|$, and $\|$ as defined above. The elements of $\mathbf{P}_\gamma$ are called *processes*, and will be ranged over by p, q and r.

2.1 Equational Reasoning

We can use the full language of process expressions to reason about the elements of $\mathbf{P}_\gamma$. A *valuation* is a mapping $\nu : \mathcal{V} \to \mathbf{P}_\gamma$; it induces an *evaluation mapping*

$$[\![\_]\!]_\nu : \mathcal{P} \to \mathbf{P}_\gamma$$

inductively defined by

$$\begin{array}{ll}
[\![x]\!]_\nu = \nu(x) & [\![P \mathbin{\|\!\lfloor} Q]\!]_\nu = [\![P]\!]_\nu \mathbin{\|\!\lfloor} [\![Q]\!]_\nu \\
[\![\mathbf{0}]\!]_\nu = \mathbf{0} & [\![P \mid Q]\!]_\nu = [\![P]\!]_\nu \mid [\![Q]\!]_\nu \\
[\![\alpha.P]\!]_\nu = \alpha.[\![P]\!]_\nu & [\![P \parallel Q]\!]_\nu = [\![P]\!]_\nu \parallel [\![Q]\!]_\nu \\
[\![P + Q]\!]_\nu = [\![P]\!]_\nu + [\![Q]\!]_\nu . &
\end{array}$$

A *process equation* is a formula $P \approx Q$ with P and Q process terms; it is said to be *valid* (in $\mathbf{P}_\gamma$) if $[\![P]\!]_\nu = [\![Q]\!]_\nu$ for all $\nu : \mathcal{V} \to \mathbf{P}_\gamma$. If $P \approx Q$ is valid in $\mathbf{P}_\gamma$, then we shall also write $P \leftrightarrows_\gamma Q$. The *equational theory* of the algebra $\mathbf{P}_\gamma$ is the set of all valid process equations, i.e.,

$$\mathit{EqTh}(\mathbf{P}_\gamma) = \{P \approx Q : [\![P]\!]_\nu = [\![Q]\!]_\nu \text{ for all } \nu : \mathcal{V} \to \mathbf{P}_\gamma\} \ .$$

The precise contents of the set $\mathit{EqTh}(\mathbf{P}_\gamma)$ depend to some extent on the choice of γ. For instance, the process equation $x \mid y \ \approx\ \mathbf{0}$ is only valid in $\mathbf{P}_\gamma$ if γ is the trivial communication function f; if γ is the CCS communication function h, then $\mathbf{P}_\gamma$ satisfies the weaker equation $x \mid (y \mid z) \ \approx\ \mathbf{0}$.

Table II lists process equations that are valid in $\mathbf{P}_\gamma$ independently of the choice of γ. (The equations L2, C2 and C3 are actually axiom schemata; they generate an axiom for all $\alpha \in \mathcal{A}_\tau$ and $a, b \in \mathcal{A}$. Note that if $\mathcal{A}$ is finite, then these axiom schemata generate finitely many axioms.) Henceforth whenever we write an equation $P \approx Q$, we mean that it is derivable from the axioms in Table II by means of equational logic. It is well-known that the rules of equational logic preserve validity. We therefore obtain the following result.

Proposition 2.2. *For all process terms P and Q, if $P \approx Q$, then $P \leftrightarrows_\gamma Q$.*

In the following lemma we give an example of a valid equation that can be derived from Table II using the rules of equational logic.

Table II. Process equations valid in every $\mathbf{P}_\gamma$.

A1	$x + y \approx y + x$	C1	$\mathbf{0} \mid x \approx \mathbf{0}$	
A2	$(x + y) + z \approx x + (y + z)$	C2	$a.x \mid b.y \approx \gamma(a, b).(x \parallel y)$	if $\gamma(a, b)\downarrow$
A3	$x + x \approx x$	C3	$a.x \mid b.y \approx \mathbf{0}$	if $\gamma(a, b)\uparrow$
A4	$x + \mathbf{0} \approx x$	C4	$(x + y) \mid z \approx x \mid z + y \mid z$	
		C5	$x \mid y \approx y \mid x$	
L1	$\mathbf{0} \llfloor x \approx \mathbf{0}$	C6	$(x \mid y) \mid z \approx x \mid (y \mid z)$	
L2	$\alpha.x \llfloor y \approx \alpha.(x \parallel y)$	C7	$(x \llfloor y) \mid z \approx (x \mid z) \llfloor y$	
L3	$(x + y) \llfloor z \approx x \llfloor z + y \llfloor z$			
L4	$(x \llfloor y) \llfloor z \approx x \llfloor (y \parallel z)$	P1	$x \parallel y \approx (x \llfloor y + y \llfloor x) + x \mid y$	
L5	$x \llfloor \mathbf{0} \approx x$			

Lemma 2.3. *The following equation is derivable from the axioms in Table II:*

$$\text{C8} \quad (x \llfloor y) \mid (z \llfloor u) \approx (x \mid z) \llfloor (y \parallel u) \ .$$

Proof. The lemma is proved with the derivation:

$$\begin{aligned}
(x \llfloor y) \mid (z \llfloor u) &\approx (z \llfloor u) \mid (x \llfloor y) && \text{(by C5)} \\
&\approx (z \mid (x \llfloor y)) \llfloor u && \text{(by C7)} \\
&\approx ((x \llfloor y) \mid z) \llfloor u && \text{(by C5)} \\
&\approx ((x \mid z) \llfloor y) \llfloor u && \text{(by C7)} \\
&\approx (x \mid z) \llfloor (y \parallel u) && \text{(by L4).} \quad \Box
\end{aligned}$$

A set of valid process equations is an *equational base* for $\mathbf{P}_\gamma$ if all other valid process equations are derivable from it by means of equational logic. The purpose of this paper is to prove that if we add to the equations in Table II the equation $x \mid y \approx \mathbf{0}$ we obtain an equational base for $\mathbf{P}_f$, and if, instead, we add $x \mid (y \mid z) \approx \mathbf{0}$ we obtain an equational base for $\mathbf{P}_h$. Both these equational bases are finite if the set of actions $\mathcal{A}$ is finite.

For the proofs of these results, we adopt the classic two-step approach [Aceto et al. 2005b]:

(1) In the first step we identify a set of normal forms, and prove that every process term can be rewritten to a normal form by means of the axioms.

(2) In the second step we prove that bisimilar normal forms are identical modulo applications of the axioms A1–A4. This is done by associating with every pair of normal forms a so-called distinguishing valuation, i.e., a valuation that proves that the normal forms are not bisimilar unless they are provably equal modulo the axioms A1–A4.

Many of the proofs to follow will be by induction, using the following syntactic measure on process terms.

Definition 2.4. Let P be a process term. We define the *height* of a process term

P, denoted $h(P)$, inductively as follows:

$$\begin{array}{ll} h(\mathbf{0}) = 0 \ , & h(P \mathbin{\lfloor\!\lfloor} Q) = h(P) + h(Q) \ , \\ h(x) = 1 \ , & h(P \mid Q) = h(P) + h(Q) \ , \\ h(\alpha.P) = h(P) + 1 \ , & h(P \parallel Q) = h(P) + h(Q) \ , \\ h(P + Q) = \max(h(P), h(Q)) \ . & \end{array}$$

Definition 2.5. We call a process term *simple* if it is not $\mathbf{0}$ and not an alternative composition.

Lemma 2.6. *For every process term P there exists a collection of simple process terms $S_1, \ldots, S_n$ $(n \geq 0)$ such that $h(P) \geq h(S_i)$ for all $i = 1, \ldots, n$ and*

$$P \approx \sum_{i=1}^{n} S_i \qquad \textit{(by A1, A2 and A4).}$$

The terms S_i will be called syntactic summands *of P.*

2.2 General Properties of $\mathbf{P}_\gamma$

We collect some general properties of the algebras $\mathbf{P}_\gamma$ that we shall need in the remainder of the paper.

The binary transition relations $\xrightarrow{\alpha}$ $(\alpha \in \mathcal{A}_\tau)$ on $\mathcal{P}_0$, which were used to associate an operational semantics with closed process terms, will play an important rôle in the remainder of the paper. They induce binary relations on $\mathbf{P}_\gamma$, also denoted by $\xrightarrow{\alpha}$, and defined as the least relations such that $P \xrightarrow{\alpha} P'$ implies $[P] \xrightarrow{\alpha} [P']$. Note that we then get, directly from the definition of bisimulation, that for all $P, P' \in \mathcal{P}_0$:

$$[P] \xrightarrow{\alpha} [P'] \text{ iff for all } Q \in [P] \text{ there exists } Q' \in [P'] \text{ such that } Q \xrightarrow{\alpha} Q'.$$

Proposition 2.7. *For all $p, q, r \in \mathbf{P}_\gamma$:*

(*1*) $p = \mathbf{0}$ *iff there do not exist $p' \in \mathbf{P}_\gamma$ and $\alpha \in \mathcal{A}_\tau$ such that $p \xrightarrow{\alpha} p'$;*

(*2*) $\alpha.p \xrightarrow{\beta} r$ *iff $\alpha = \beta$ and $r = p$;*

(*3*) $p + q \xrightarrow{\alpha} r$ *iff $p \xrightarrow{\alpha} r$ or $q \xrightarrow{\alpha} r$;*

(*4*) $p \mathbin{\lfloor\!\lfloor} q \xrightarrow{\alpha} r$ *iff there exists $p' \in \mathbf{P}_\gamma$ such that $p \xrightarrow{\alpha} p'$ and $r = p' \parallel q$; and*

(*5*) $p \mid q \xrightarrow{\alpha} r$ *iff there exist actions $a, b \in \mathcal{A}$ and processes $p', q' \in \mathbf{P}_\gamma$ such that $\alpha = \gamma(a, b)$, $p \xrightarrow{a} p'$, $q \xrightarrow{b} q'$, and $r = p' \parallel q'$; and*

(*6*) $p \parallel q \xrightarrow{\alpha} r$ *iff $p \mathbin{\lfloor\!\lfloor} q \xrightarrow{\alpha} r$ or $q \mathbin{\lfloor\!\lfloor} p \xrightarrow{\alpha} r$ or $p \mid q \xrightarrow{\alpha} r$.*

Let $p, p' \in \mathbf{P}_\gamma$; we write $p \rightarrow p'$ if $p \xrightarrow{\alpha} p'$ for some $\alpha \in \mathcal{A}_\tau$ and call p' a *residual* of p. We write $p \not\rightarrow$ if p has no residual, that is, if $p = \mathbf{0}$ (by Proposition 2.7(1)). We denote by $\rightarrow^*$ the reflexive transitive closure of $\rightarrow$.

It is easy to see from Table I that if $P \xrightarrow{\alpha} P'$, then P' has fewer symbols than P. Consequently, the length of a transition sequence starting with a process $[P]$ is bounded from above by the number of symbols in P.

Definition 2.8. The *depth* $|p|$ of an element $p \in \mathbf{P}_\gamma$ is defined as

$$|p| = \max\{n \geq 0 : \ \exists p_n, \ldots, p_0 \in \mathbf{P}_\gamma \text{ s.t. } p = p_n \rightarrow \cdots \rightarrow p_0\}.$$

The *branching degree* $bdeg(p)$ of an element $p \in \mathbf{P}_\gamma$ is defined as

$$bdeg(p) = |\{(\alpha, p') : p \xrightarrow{\alpha} p'\}| \ .$$

Note that $p = \mathbf{0}$ iff $|p| = 0$.

For the remainder of this section, we focus on properties of parallel composition on $\mathbf{P}_\gamma$. The depth of a parallel composition is the sum of the depths of its components.

Lemma 2.9. *For all $p, q \in \mathbf{P}_\gamma$, $|p \parallel q| = |p| + |q|$.*

Proof. If $p = p_m \rightarrow \cdots \rightarrow p_0$ and $q = q_n \rightarrow \cdots \rightarrow q_0$, then

$$p \parallel q = p_m \parallel q \rightarrow \cdots \rightarrow p_0 \parallel q = p_0 \parallel q_n \rightarrow \cdots \rightarrow p_0 \parallel q_0 \ ,$$

so clearly $|p \parallel q| \geq |p| + |q|$.

It remains to prove that $|p| + |q| \geq |p \parallel q|$. We proceed by induction on the depth of $p \parallel q$.

If $|p \parallel q| = 0$, then clearly $|p| + |q| \geq |p \parallel q|$ (since depth is nonnegative).

Suppose that $|p \parallel q| > 0$. Then $p \parallel q \rightarrow p' \parallel q'$ for some p' and q' with $|p \parallel q| = 1 + |p' \parallel q'|$, and either $p \rightarrow p'$ and $q = q'$, or $p = p'$ and $q \rightarrow q'$, or $p \rightarrow p'$ and $q \rightarrow q'$. In any case, $|p| + |q| \geq 1 + |p'| + |q'|$, so, by the induction hypothesis, $|p| + |q| \geq 1 + |p'| + |q'| \geq 1 + |p' \parallel q'| = |p \parallel q|$. □

According to the following lemma and Proposition 2.2, $\mathbf{P}_\gamma$ is a commutative monoid with respect to $\parallel$, with $\mathbf{0}$ as the identity element.

Lemma 2.10. *The following equations are derivable from the axioms in Table II:*

$$\begin{array}{lll} \text{P2} & (x \parallel y) \parallel z & \approx x \parallel (y \parallel z) \\ \text{P3} & x \parallel y & \approx y \parallel x \\ \text{P4} & x \parallel \mathbf{0} & \approx x \ . \end{array}$$

An element $p \in \mathbf{P}_\gamma$ is *parallel prime* if $p \neq \mathbf{0}$, and $p = q \parallel r$ implies $q = \mathbf{0}$ or $r = \mathbf{0}$. Suppose that p is an arbitrary element of $\mathbf{P}_\gamma$; a *parallel decomposition* of p is a finite multiset $[p_1, \ldots, p_n]$ of parallel primes such that $p = p_1 \parallel \cdots \parallel p_n$. (The process $\mathbf{0}$ has as decomposition the empty multiset, and a parallel prime process p has as decomposition the singleton multiset $[p]$.)

The following unique parallel decomposition result was proved for $\mathbf{P}_f$ by Milner and Moller [1993] and for $\mathbf{P}_h$ by Moller [1989]. In its formulation below, with γ an arbitrary communication function, it is a straightforward consequence of a unique decomposition theorem by Luttik and van Oostrom [2005], which generalises the unique parallel decomposition theorems in [Moller 1989].

Theorem 2.11. *Every element of $\mathbf{P}_\gamma$ has a unique parallel decomposition.*

Proof. In a similar way as in [Luttik and van Oostrom 2005, Sect. 4] it can be established that the inverse of $\rightarrow^*$ is a decomposition order on the commutative monoid $\mathbf{P}_\gamma$ with respect to parallel composition; it then follows from [Luttik and van Oostrom 2005, Theorem 32] that this commutative monoid has unique decomposition. □

The following corollary follows easily from the above unique decomposition result.

Corollary 2.12. *Let $p, q, r \in \mathbf{P}_\gamma$. If $p \parallel q = p \parallel r$, then $q = r$.*

The branching degree of a parallel composition is at least the branching degree of its components.

LEMMA 2.13. *For all $p, q \in \mathbf{P}_\gamma$, $bdeg(p \parallel q) \geq bdeg(p), bdeg(q)$.*

PROOF. First we prove that $bdeg(p \parallel q) \geq bdeg(q)$. By Proposition 2.7, if $q \xrightarrow{\alpha} q'$, then $p \parallel q \xrightarrow{\alpha} p \parallel q'$. Suppose that q_1 and q_2 are distinct processes such that $q \xrightarrow{\alpha} q_1$ and $q \xrightarrow{\alpha} q_2$. Then $p \parallel q \xrightarrow{\alpha} p \parallel q_1$ and $p \parallel q \xrightarrow{\alpha} p \parallel q_2$. Since $p \parallel q_1 = p \parallel q_2$ would imply $q_1 = q_2$ by Corollary 2.12, it follows that $p \parallel q_1$ and $p \parallel q_2$ are distinct. Hence $bdeg(p \parallel q) \geq bdeg(q)$.

By commutativity of $\parallel$, it also follows that $bdeg(p \parallel q) \geq bdeg(p)$. □

In the remainder of the paper we will make use of the following sequence of parallel prime processes:

$$\varphi_i = \tau.\mathbf{0} + \tau.\tau.\mathbf{0} + \cdots + \tau^i.\mathbf{0} \qquad (i \geq 1) \tag{1}$$

(with $\tau^i.\mathbf{0}$ recursively defined by $\tau^i.\mathbf{0} = \mathbf{0}$ if $i = 0$, and $\tau.\tau^{i-1}.\mathbf{0}$ if $i > 0$). The special properties of the processes φ_i, proved in the lemma below, make them very suitable tools in the analysis of the equational theory of parallel composition. They were first used for this purpose by Moller [1990].

LEMMA 2.14. (*1*) *For all $i \geq 1$, the processes φ_i are parallel prime.*

(*2*) *The processes φ_i are all distinct, i.e., $\varphi_k = \varphi_l$ implies that $k = l$.*

(*3*) *For all $i \geq 1$, the process φ_i has branching degree i.*

PROOF. (1) Clearly $\varphi_i \neq \mathbf{0}$. Suppose $\varphi_i = p \parallel q$; to prove that φ_i is parallel prime, we need to establish that either $p = \mathbf{0}$ or $q = \mathbf{0}$. Note that $p \parallel q \xrightarrow{\tau} \mathbf{0}$. There do not exist actions a and b and processes p' and q' such that $\gamma(a, b) = \tau$, $p \xrightarrow{a} p'$ and $q \xrightarrow{b} q'$, for then also $p \parallel q \xrightarrow{a} p' \parallel q$, quod non. Therefore, according to Proposition 2.7, there are only two cases to consider:

(a) If there exists p' such that $p \xrightarrow{\tau} p'$ and $p' \parallel q = \mathbf{0}$, then it follows by Lemma 2.9 that $|q| = 0$, and hence $q = \mathbf{0}$.

(b) If there exists q' such that $q \xrightarrow{\tau} q'$ and $p \parallel q' = \mathbf{0}$, then it follows by Lemma 2.9 that $|p| = 0$, and hence $p = \mathbf{0}$.

(2) If $\varphi_k = \varphi_l$, then $k = |\varphi_k| = |\varphi_l| = l$.

(3) On the one hand, $\varphi_i \xrightarrow{\tau} \tau^j.\mathbf{0}$ for all $0 \leq j < i$ and $\tau^k.\mathbf{0} = \tau^l.\mathbf{0}$ implies $k = l$ for all $0 \leq k, l < i$, so $bdeg(\varphi_i)$ is at least i. On the other hand, if $\varphi_i \xrightarrow{\alpha} p$, then $\alpha = \tau$ and $p = \tau^j.\mathbf{0}$ for some $0 \leq j < i$, so $bdeg(\varphi_i)$ is at most i. □

3. AN EQUATIONAL BASE FOR $\mathbf{P}_f$

In this section, we prove that an equational base for $\mathbf{P}_f$ is obtained if the axiom

$$\text{F} \quad x \mid y \;\approx\; \mathbf{0}$$

is added to the set of axioms generated by the axiom schemata in Table II. The resulting equational base is finite if $\mathcal{A}$ is finite. Henceforth, whenever we write $P \approx_{\text{F}} Q$ we mean that the equation $P \approx Q$ is derivable from the axioms in Table II and the axiom F.

PROPOSITION 3.1. *For all process terms P and Q, if $P \approx_{\text{F}} Q$, then $P \leftrightarroweq_f Q$.*

To prove that adding F to the axioms in Table II suffices to obtain an equational base for $\mathbf{P}_f$, we need to establish that $P \leftrightarroweq_f Q$ implies $P \approx_{\mathrm{F}} Q$ for all process terms P and Q. First, we identify a set of normal forms $\mathcal{N}_{\mathrm{F}}$ such that every process term P can be rewritten to a normal form by means of the axioms.

Definition 3.2. The set $\mathcal{N}_{\mathrm{F}}$ of F-*normal forms* is generated by the following grammar:

$$N \ ::= \ \mathbf{0} \ \mid \ N + N \ \mid \ \alpha.N \ \mid \ x \mathbin{|\!|\!\_} N \ ,$$

with $x \in \mathcal{V}$, and $\alpha \in \mathcal{A}_\tau$.

Lemma 3.3. *For every process term P there is an* F*-normal form N such that $P \approx_{\mathrm{F}} N$ and $h(P) \geq h(N)$.*

Proof. Recall that $h(P)$ denotes the height of P (see Definition 2.4). In this proof we also use another syntactic measure on P: the *length* of P, denoted $\ell(P)$, is the number of symbols occurring in P. Define a partial order $\prec$ on process terms by $P \prec Q$ if the pair $(h(P), \ell(P))$ is smaller than the pair $(h(Q), \ell(Q))$ in the lexicographical order on $\omega \times \omega$; i.e., $P \prec Q$ if $h(P) < h(Q)$ or $h(P) = h(Q)$ and $\ell(P) < \ell(Q)$. It is well-known that the lexicographical order on $\omega \times \omega$, and hence the order $\prec$ on process terms, is well-founded; so we may use $\prec$-induction.

The remainder of the proof consists of a case distinction on the syntactic forms that P may take.

(1) If P is a variable, say $P = x$, then $P \approx x \mathbin{|\!|\!\_} \mathbf{0}$ by L5; the process term $x \mathbin{|\!|\!\_} \mathbf{0}$ is an F-normal form and $h(P) = h(x) = h(x) + 0 = h(x \mathbin{|\!|\!\_} \mathbf{0})$.

(2) If $P = \mathbf{0}$, then P is an F-normal form.

(3) If $P = \alpha.P'$, then, since $h(P') < h(P)$, it holds that $P' \prec P$, and hence by the induction hypothesis there exists an F-normal form N such that $P' \approx_{\mathrm{F}} N$ and $h(P') \geq h(N)$. Then $\alpha.N$ is an F-normal form such that $P \approx_{\mathrm{F}} \alpha.N$ and $h(P) \geq h(\alpha.N)$.

(4) If $P = P_1 + P_2$, then, since $h(P_1), h(P_2) \leq h(P)$ and $\ell(P_1), \ell(P_2) < \ell(P)$, it holds that $P_1, P_2 \prec P$, and hence by the induction hypothesis there exist F-normal forms N_1 and N_2 such that $P_1 \approx_{\mathrm{F}} N_1$, $P_2 \approx_{\mathrm{F}} N_2$, $h(P_1) \geq h(N_1)$ and $h(P_2) \geq h(N_2)$. Then $N_1 + N_2$ is an F-normal form such that $P \approx_{\mathrm{F}} N_1 + N_2$ and $h(P) \geq h(N_1 + N_2)$.

(5) If $P = Q \mathbin{|\!|\!\_} R$, then, since $h(Q) \leq h(P)$ and $\ell(Q) < \ell(P)$, it holds that $Q \prec P$, and hence by the induction hypothesis and Lemma 2.6 there exists a collection $S_1, \ldots, S_n$ of simple F-normal forms such that $Q \approx_{\mathrm{F}} \sum_{i=1}^{n} S_i$ and $h(Q) \geq h(S_i)$ for all $i = 1, \ldots, n$. If $n = 0$, then $P \approx_{\mathrm{F}} \mathbf{0} \mathbin{|\!|\!\_} R \approx \mathbf{0}$ by L1, and clearly $h(P) \geq h(\mathbf{0})$. Otherwise, by L3

$$P \approx_{\mathrm{F}} \sum_{i=1}^{n} (S_i \mathbin{|\!|\!\_} R) \ .$$

So it remains to show, for all $i = 1, \ldots, n$, that $S_i \mathbin{|\!|\!\_} R$ is provably equal to an appropriate F-normal form. We distinguish cases according to the syntactic form of S_i:

(a) If $S_i = \alpha.N_i'$, with N_i' an F-normal form, then by L2

$$S_i \mathbin{|\!\lfloor} R \approx \alpha.(N_i' \parallel R) \ .$$

Since $h(N_i') < h(S_i) \leq h(Q)$, it holds that $N_i' \parallel R \prec P$ and hence by the induction hypothesis there exists an F-normal form N_i such that $N_i' \parallel R \approx_{\mathrm{F}} N_i$ and $h(N_i' \parallel R) \geq h(N_i)$. Clearly, $\alpha.N_i$ is an F-normal form such that $S_i \mathbin{|\!\lfloor} R \approx_{\mathrm{F}} \alpha.N_i$ and $h(S_i \mathbin{|\!\lfloor} R) \geq h(\alpha.N_i)$.

(b) If $S_i = x \mathbin{|\!\lfloor} N_i'$, with N_i' an F-normal form, then by L4

$$(x \mathbin{|\!\lfloor} N_i') \mathbin{|\!\lfloor} R \approx x \mathbin{|\!\lfloor} (N_i' \parallel R) \ .$$

Note that $h(x) = 1$, so $h(N_i') < h(S_i) \leq h(Q)$. It follows that $N_i' \parallel R \prec P$, and hence by the induction hypothesis there exists an F-normal form N_i such that $N_i' \parallel R \approx_{\mathrm{F}} N_i$ and $h(N_i' \parallel R) \geq h(N_i)$. Clearly, $x \mathbin{|\!\lfloor} N_i$ is an F-normal form such that $S_i \mathbin{|\!\lfloor} R \approx_{\mathrm{F}} x \mathbin{|\!\lfloor} N_i$ and $h(S_i \mathbin{|\!\lfloor} R) \geq h(x \mathbin{|\!\lfloor} N_i)$.

(6) If $P = Q \mid R$, then $P \approx_{\mathrm{F}} \mathbf{0}$ according to the axiom F and clearly $h(P) \geq h(\mathbf{0})$.

(7) If $P = Q \parallel R$, then $P \approx (Q \mathbin{|\!\lfloor} R + R \mathbin{|\!\lfloor} Q) + Q \mid R \approx_{\mathrm{F}} Q \mathbin{|\!\lfloor} R + R \mathbin{|\!\lfloor} Q$ by the axioms P1, F and A4. We can now proceed as in case 5 to show that for $Q \mathbin{|\!\lfloor} R$ and $R \mathbin{|\!\lfloor} Q$ there exist F-normal forms N_1 and N_2, respectively, such that $Q \mathbin{|\!\lfloor} R \approx_{\mathrm{F}} N_1$, $R \mathbin{|\!\lfloor} Q \approx_{\mathrm{F}} N_2$, $h(Q \mathbin{|\!\lfloor} R) \geq h(N_1)$ and $h(R \mathbin{|\!\lfloor} Q) \geq h(N_2)$. Then $N_1 + N_2$ is an F-normal form such that $P \approx_{\mathrm{F}} N_1 + N_2$ and $h(P) \geq h(N_1 + N_2)$. □

It remains to prove for every pair of F-normal forms N_1 and N_2 that if $N_1 \leftrightarroweq_f N_2$ (i.e., if $[\![N_1]\!]_\nu = [\![N_2]\!]_\nu$ for all valuations $\nu : \mathcal{V} \to \mathbf{P}_f$), then $N_1 \approx_{\mathrm{F}} N_2$. We shall in fact prove something seemingly stronger by associating with every pair of F-normal forms N_1 and N_2 a special valuation $* : \mathcal{V} \to \mathbf{P}_f$ such that

$$\text{if } [\![N_1]\!]_* = [\![N_2]\!]_*, \text{ then } N_1 \approx_{\mathrm{F}} N_2. \qquad (2)$$

Contrapositively, if N_1 and N_2 are *not* provably equal, then their $*$-interpretations are distinct; this is why we call such a valuation $*$ a *distinguishing valuation*.

The idea is to use a valuation $*$ that assigns processes to variables in such a way that much of the original syntactic structure of N_1 and N_2 can be recovered by analysing the behaviour of $[\![N_1]\!]_*$ and $[\![N_2]\!]_*$. To recognize variables, we shall use the special processes φ_i ($i \geq 1$) defined in Eqn. (1) on p. 9. Recall that the processes φ_i have branching degree i. We are going to assign to every variable a distinct process φ_i. By choosing i larger than the maximal 'branching degrees' occurring in N_1 and N_2, the behaviour contributed by an instantiated variable is distinguished from behaviour already present in the F-normal forms themselves.

Definition 3.4. We define the *width* $w(N)$ of an F-normal form N as follows:

(1) if $N = \mathbf{0}$, then $w(N) = 0$;

(2) if $N = N_1 + N_2$, then $w(N) = w(N_1) + w(N_2)$;

(3) if $N = \alpha.N'$, then $w(N) = \max(w(N'), 1)$; and

(4) if $N = x \mathbin{|\!\lfloor} N'$, then $w(N) = \max(w(N'), 1)$.

The valuation $*$ that we now proceed to define is parametrised with a natural number W; in Theorem 3.8 we shall prove that it serves as a distinguishing valuation (i.e., satisfies Eqn. (2)) for all F-normal forms N_1 and N_2 such that $w(N_1), w(N_2) \leq W$. Let $\ulcorner\_\urcorner$ denote an *injective* function

$$\ulcorner\_\urcorner : \mathcal{V} \to \{n \in \omega : n > W\}$$

that associates with every variable a unique natural number greater than W. We define the valuation $* : \mathcal{V} \to \mathbf{P}_f$ for all $x \in \mathcal{V}$ by

$$*(x) = \tau.\varphi_{\ulcorner x \urcorner} \ .$$

The τ-prefix is to ensure the following property.

Lemma 3.5. *For every* F*-normal form N, the branching degree of $\llbracket N \rrbracket_*$ is at most $w(N)$.*

Proof. Structural induction on N. □

Lemma 3.6. *Let S be a simple* F*-normal form, let $\alpha \in \mathcal{A}_\tau$, and let p be a process such that $\llbracket S \rrbracket_* \xrightarrow{\alpha} p$.*

(1) If $S = \beta.N$, then $\alpha = \beta$ and $p = \llbracket N \rrbracket_*$.
(2) If $S = x \mathbin{|\!\!\lfloor} N$, then $\alpha = \tau$ and $p = \varphi_{\ulcorner x \urcorner} \parallel \llbracket N \rrbracket_*$.

An important property of $*$ is that it allows us to distinguish the different types of simple F-normal forms by classifying their residuals according to the number of parallel components with a branching degree that exceeds W. Let us say that a process p is of *type n* $(n \geq 0)$ if its unique parallel decomposition contains precisely n parallel prime components with a branching degree larger than W.

Corollary 3.7. *Let S be a simple* F*-normal form such that $w(S) \leq W$.*

(1) If $S = \alpha.N$, then the unique residual $\llbracket N \rrbracket_*$ of $\llbracket S \rrbracket_*$ is of type 0.
(2) If $S = x \mathbin{|\!\!\lfloor} N$, then the unique residual $\varphi_{\ulcorner x \urcorner} \parallel \llbracket N \rrbracket_*$ of $\llbracket S \rrbracket_*$ is of type 1.

Proof. On the one hand, by Lemma 3.5, in both cases $\llbracket N \rrbracket_*$ has a branching degree of at most $w(N) \leq w(S) \leq W$, and hence, by Lemma 2.13, its unique parallel decomposition cannot contain parallel prime components with a branching degree that exceeds W. On the other hand, by Lemmas 2.14(1) and 2.14(3), the process $\varphi_{\ulcorner x \urcorner}$ is parallel prime and has a branching degree that exceeds W. So $\llbracket N \rrbracket_*$ is of type 0, and $\varphi_{\ulcorner x \urcorner} \parallel \llbracket N \rrbracket_*$ is of type 1. □

Theorem 3.8. *For every two* F*-normal forms N_1, N_2 such that $w(N_1), w(N_2) \leq W$ it holds that $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$ only if $N_1 \approx N_2$ modulo* A1–A4.

Proof. By Lemma 2.6 we may assume that N_1 and N_2 are summations of collections of simple F-normal forms. We assume $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$ and prove that then $N_1 \approx N_2$ modulo A1–A4, by induction on the sum of the heights of N_1 and N_2.

We first prove that for every syntactic summand S_1 of N_1 there is a syntactic summand S_2 of N_2 such that $S_1 \approx S_2$ modulo A1–A4. To this end, let S_1 be an arbitrary syntactic summand of N_1; we distinguish cases according to the syntactic form of S_1.

(1) Suppose $S_1 = \alpha.N_1'$; then $\llbracket S_1 \rrbracket_* \xrightarrow{\alpha} \llbracket N_1' \rrbracket_*$. Hence, since $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$, there exists a syntactic summand S_2 of N_2 such that $\llbracket S_2 \rrbracket_* \xrightarrow{\alpha} \llbracket N_1' \rrbracket_*$. By Lemma 3.5 the branching degree of $\llbracket N_1' \rrbracket_*$ does not exceed W, so $\llbracket S_2 \rrbracket_*$ has a residual of type 0, and therefore, by Corollary 3.7, there exist $\beta \in \mathcal{A}_\tau$ and an F-normal form N_2' such that $S_2 = \beta.N_2'$. Moreover, since $\llbracket S_2 \rrbracket_* \xrightarrow{\alpha} \llbracket N_1' \rrbracket_*$, it follows by Lemma 3.6(1) that $\alpha = \beta$ and $\llbracket N_1' \rrbracket_* = \llbracket N_2' \rrbracket_*$. Hence, by the induction hypothesis, we conclude that $N_1' \approx N_2'$ modulo A1–A4, so $S_1 = \alpha.N_1' \approx \beta.N_2' = S_2$.

(2) Suppose $S_1 = x \mathbin{\lfloor\!\lfloor} N_1'$; then $\llbracket S_1 \rrbracket_* \xrightarrow{\tau} \varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$. Hence, since $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$, there exists a summand S_2 of N_2 such that $\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$. Since S_2 has a residual of type 1, by Corollary 3.7 there exist a variable y and an F-normal form N_2' such that $S_2 = y \mathbin{\lfloor\!\lfloor} N_2'$. Now, since $\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$, it follows by Lemma 3.6(2) that

$$\varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_* = \varphi_{\ulcorner y \urcorner} \parallel \llbracket N_2' \rrbracket_* \ . \tag{3}$$

Since $\llbracket N_1' \rrbracket_*$ and $\llbracket N_2' \rrbracket_*$ are of type 0, we have that the unique decomposition of $\llbracket N_1' \rrbracket_*$ (see Theorem 2.11) does not contain $\varphi_{\ulcorner y \urcorner}$ and the unique decomposition of $\llbracket N_2' \rrbracket_*$ does not contain $\varphi_{\ulcorner x \urcorner}$. Hence, from (3) it follows that $\varphi_{\ulcorner x \urcorner} = \varphi_{\ulcorner y \urcorner}$ and $\llbracket N_1' \rrbracket_* = \llbracket N_2' \rrbracket_*$. From the former we conclude, by Lemma 2.14(2) and the injectivity of $\ulcorner . \urcorner$, that $x = y$ and from the latter we conclude by the induction hypothesis that $N_1' \approx N_2'$ modulo A1–A4. So $S_1 = x \mathbin{\lfloor\!\lfloor} N_1' \approx y \mathbin{\lfloor\!\lfloor} N_2' = S_2$.

We have established that every syntactic summand of N_1 is provably equal to a syntactic summand of N_2. Similarly, it follows that every syntactic summand of N_2 is provably equal to a syntactic summand of N_1. Hence, modulo A1–A4, $N_1 \approx N_1 + N_2 \approx N_2$, so the proof of the theorem is complete. □

Note that it follows from the preceding theorem that there exists a distinguishing valuation for *every* pair of F-normal forms N_1 and N_2 that are distinct modulo A1–A4; it is obtained by instantiating the parameter W in the definition of $*$ with a sufficiently large value. Hence, we get the following corollary.

Corollary 3.9. *For all process terms P and Q, $P \approx_{\mathrm{F}} Q$ if, and only if, $P \mathrel{\underline{\leftrightarrow}}_f Q$, and hence the axioms generated by the schemata in Table II together with the axiom* F *consitute an equational base for* $\mathbf{P}_f$.

Proof. The implication from left to right is Proposition 3.1. To prove the implication from right to left, suppose $P \mathrel{\underline{\leftrightarrow}}_f Q$. Then, by Lemma 3.3 there exist F-normal forms N_1 and N_2 such that $P \approx_{\mathrm{F}} N_1$ and $Q \approx_{\mathrm{F}} N_2$; from $P \mathrel{\underline{\leftrightarrow}}_f Q$ we conclude by Proposition 3.1 that $N_1 \mathrel{\underline{\leftrightarrow}}_f N_2$. Now choose W large enough such that $w(N_1), w(N_2) \leq W$. From $N_1 \mathrel{\underline{\leftrightarrow}}_f N_2$ it follows that $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$, and hence, by Theorem 3.8 $N_1 \approx N_2$. We may therefore conclude that $P \approx_{\mathrm{F}} N_1 \approx N_2 \approx_{\mathrm{F}} Q$. □

Corollary 3.10. *The equational theory of* $\mathbf{P}_f$ *is decidable.*

Proof. From the proof of Lemma 3.3 it is easy to see that there exists an effective procedure that associates with every process term a provably equivalent F-normal form. Furthermore, from Definition 3.4 it is clear that every F-normal form has an effectively computable width. We now sketch an effective procedure that decides whether a process equation $P \approx Q$ is valid:

(1) Compute F-normal forms N_1 and N_2 such that $P \approx_{\mathrm{F}} N_1$ and $Q \approx_{\mathrm{F}} N_2$.
(2) Compute $w(N_1)$ and $w(N_2)$ and define W as their maximum.
(3) Determine the (finite) set $\mathcal{V}'$ of variables occurring in N_1 and N_2; define an injection $\ulcorner . \urcorner : \mathcal{V}' \rightarrow \{n \in \omega : n > W\}$, and a substitution $* : \mathcal{V}' \rightarrow \mathcal{P}_0$ that assigns to a variable x in $\mathcal{V}'$ the closed process term $\tau.\varphi_{\ulcorner x \urcorner}$. (We may interpret Eqn. (1) as defining a sequence of closed process terms instead of a sequence of processes.)
(4) Let N_1^* and N_2^* be the results from applying $*$ to N_1 and N_2, respectively.
(5) Determine if the closed process terms N_1^* and N_2^* are bisimilar; if they are, then the process equation $P \approx Q$ is valid in $\mathbf{P}_f$, and otherwise it is not. □

4. AN EQUATIONAL BASE FOR $\mathbf{P}_h$

We now consider the algebra $\mathbf{P}_h$. Note that if $\mathcal{A}$ happens to be the empty set, then $\mathbf{P}_h$ satisfies the axiom F, and it is clear from the proof in the previous section that the axioms generated by the axiom schemata in Table II together with F in fact constitute a finite equational base for $\mathbf{P}_h$. We therefore proceed with the assumption that $\mathcal{A}$ is nonempty, and prove that an equational base for $\mathbf{P}_h$ is then obtained if we add the axiom

$$\mathrm{H} \quad x \mid (y \mid z) \approx \mathbf{0}$$

to the set of axioms generated by the axiom schemata in Table II. Again, the resulting equational base is finite if the set $\mathcal{A}$ is finite. Henceforth, whenever we write $P \approx_{\mathrm{H}} Q$, we mean that the equation $P \approx Q$ is derivable from the axioms in Table II and the axiom H.

Proposition 4.1. *For all process terms P and Q, if $P \approx_{\mathrm{H}} Q$, then $P \leftrightarrows_h Q$.*

We proceed to adapt the proof presented in the previous section to establish the converse of Proposition 4.1. Naturally, with H instead of F not every occurrence of $\mid$ can be eliminated from process terms, so the first thing we need to do is to adapt the notion of normal form.

Definition 4.2. The set $\mathcal{N}_{\mathrm{H}}$ of H-*normal forms* is generated by the following grammar:

$$N \ ::= \ \mathbf{0} \ \mid \ N + N \ \mid \ \alpha.N \ \mid \ x \mathbin{|\!\!\lfloor} N \ \mid \ (x \mid a) \mathbin{|\!\!\lfloor} N \ \mid (x \mid y) \mathbin{|\!\!\lfloor} N \ ,$$

with $x, y \in \mathcal{V}$, $\alpha \in \mathcal{A}_\tau$ and $a \in \mathcal{A}$.

In the proof that every process term is provably equal to an H-normal form, we use the following derivable equation.

Lemma 4.3. *The following equation is derivable from the axioms in Table II and the axiom* H*:*

$$\mathrm{C9} \quad \tau.x \mid y \approx_{\mathrm{H}} \mathbf{0} \ .$$

Proof. Let $a \in \mathcal{A}$; then

$$\begin{aligned} \tau.x \mid y &\approx_{\mathrm{H}} \tau.(x \parallel \mathbf{0}) \mid y && \text{by P4 (see Lemma 2.10)} \\ &\approx_{\mathrm{H}} (a.x \mid \overline{a}.\mathbf{0}) \mid y && \text{by C2} \\ &\approx_{\mathrm{H}} \mathbf{0} && \text{by H.} \quad \square \end{aligned}$$

Lemma 4.4. *For every process term P there exists an* H*-normal form N such that $P \approx_{\mathrm{H}} N$ and $h(P) \geq h(N)$.*

Proof. As in the proof of Lemma 3.3 we proceed by $\prec$-induction and do a case distinction on the syntactic form of P. For the first four cases (P is a variable, $P = \mathbf{0}$, $P = \alpha.P'$ and $P = P_1 + P_2$) the proofs are identical to those in Lemma 3.3, so they are omitted.

(5) If $P = Q \mathbin{\|\!\!\lfloor} R$, then, since $h(Q) \leq h(P)$ and $\ell(Q) < \ell(P)$, it holds that $Q \prec P$, and hence by the induction hypothesis and Lemma 2.6 there exists a collection $S_1, \ldots, S_n$ of simple H-normal forms such that $Q \approx_{\mathrm{H}} \sum_{i=1}^{n} S_i$ and $h(Q) \geq h(S_i)$ for all $i = 1, \ldots, n$. If $n = 0$, then $P \approx_{\mathrm{H}} \mathbf{0} \mathbin{\|\!\!\lfloor} R \approx \mathbf{0}$ by L1, and clearly $h(P) \geq h(\mathbf{0})$. Otherwise, by L3

$$P \approx_{\mathrm{H}} \sum_{i=1}^{n} (S_i \mathbin{\|\!\!\lfloor} R) \enspace ,$$

so it remains to show, for all $i = 1, \ldots, n$, that $S_i \mathbin{\|\!\!\lfloor} R$ is provably equal to an appropriate H-normal form. We distinguish cases according to the syntactic form of S_i:

(a) If $S_i = \alpha.N_i'$ (with N_i' an H-normal form), then by L2

$$S_i \mathbin{\|\!\!\lfloor} R \approx_{\mathrm{H}} \alpha.(N_i' \parallel R) \enspace .$$

Since $h(N_i') < h(S_i) \leq h(Q)$, it holds that $N_i' \parallel R \prec P$ and hence by the induction hypothesis there exists an H-normal form N such that $N_i' \parallel R \approx_{\mathrm{H}} N$ and $h(N_i' \parallel R) \geq h(N)$. Clearly, $\alpha.N$ is an H-normal form such that $S_i \mathbin{\|\!\!\lfloor} R \approx_{\mathrm{H}} \alpha.N$ and $h(S_i \mathbin{\|\!\!\lfloor} R) \geq h(\alpha.N)$.

(b) If $S_i = S_i' \mathbin{\|\!\!\lfloor} N_i''$ with $S_i' = x$, $S_i' = (x \mid a)$ or $S_i' = (x \mid y)$, and N_i'' an H-normal form, then by L4

$$S_i \mathbin{\|\!\!\lfloor} R \approx_{\mathrm{H}} S_i' \mathbin{\|\!\!\lfloor} (N_i'' \parallel R) \enspace .$$

Note that $h(S_i') > 0$, so $h(N_i'') < h(S_i) \leq h(Q)$. It follows that $N_i'' \parallel R \prec P$, and hence by the induction hypothesis there exists an H-normal form N such that $N_i'' \parallel R \approx_{\mathrm{H}} N$ and $h(N_i'' \parallel R) \geq h(N)$. Clearly, $S_i' \mathbin{\|\!\!\lfloor} N$ is an H-normal form such that $S_i \mathbin{\|\!\!\lfloor} R \approx_{\mathrm{H}} S_i' \mathbin{\|\!\!\lfloor} N$ and $h(S_i \mathbin{\|\!\!\lfloor} R) \geq h(S_i' \mathbin{\|\!\!\lfloor} N)$.

(6) If $P = Q | R$, then, since $h(Q) \leq h(P)$ and $\ell(Q) < \ell(P)$, it holds that $Q \prec P$, and, for similar reasons, $R \prec P$. Hence, by the induction hypothesis and Lemma 2.6 there exist collections $S_1, \ldots, S_m$ and $T_1, \ldots, T_n$ of simple H-normal forms such that $Q \approx_{\mathrm{H}} \sum_{i=1}^{m} S_i$, $R \approx_{\mathrm{H}} \sum_{j=1}^{n} T_j$, $h(Q) \geq h(S_i)$ for all $i = 1, \ldots, m$, and $h(R) \geq h(T_j)$ for all $j = 1, \ldots, n$. Note that if $m = 0$, then $P \approx_{\mathrm{H}} \mathbf{0} \mid R \approx \mathbf{0}$ by C1, and if $n = 0$, then $P \approx_{\mathrm{H}} Q \mid \mathbf{0} \approx_{\mathrm{H}} \mathbf{0} \mid Q \approx_{\mathrm{H}} \mathbf{0}$ by C5 and C1, and clearly $h(P) \geq h(\mathbf{0})$. Otherwise, by C4 and C5

$$P \approx_{\mathrm{H}} \sum_{i=1}^{m} \sum_{j=1}^{n} (S_i \mid T_j) \enspace ,$$

and it remains to show, for all $i = 1, \ldots, m$ and $j = 1, \ldots, n$, that $S_i \,|\, T_j$ is provably equal to an appropriate H-normal form. We distinguish cases according to the syntactic forms that S_i and T_j may take:

(a) Suppose $S_i = \tau.S'_i$; then $S_i | T_j \approx_{\rm H} \mathbf{0}$ by Lemma 4.3, and clearly $h(S_i | T_j) \geq 0$.

(b) Suppose $T_j = \tau.T'_j$; then we apply C5 and proceed as in the previous case.

(c) Suppose $S_i = S'_i \mathbin{|\!\!\lfloor} S''_i$ with $S'_i = x \mid a$ or $S'_i = x \mid y$; then by C7, C6, H, and L1

$$S_i \mid T_j \approx (S'_i \mid T_j) \mathbin{|\!\!\lfloor} S''_i \approx_{\rm H} \mathbf{0} \mathbin{|\!\!\lfloor} S''_i \approx \mathbf{0} \ ,$$

and clearly $h(S_i \mid T_j) \geq h(\mathbf{0})$.

(d) Suppose $T_j = T'_j \mathbin{|\!\!\lfloor} T''_j$ with $T'_j = x \mid a$ or $T'_j = x \mid y$; then $S_i \mid T_j \approx T_j \mid S_i$ by C5 and we can proceed as in the previous case.

(e) Suppose $S_i = a.S'_i$ and $T_j = b.T'_j$.
If $b \neq \overline{a}$, then $S_i \mid T_j \approx \mathbf{0}$ by C3 and $h(S_i \mid T_j) \geq h(\mathbf{0})$.
On the other hand, if $b = \overline{a}$, then $S_i \mid T_j \approx \tau.(S'_i \parallel T'_j)$ by C2, and, since $h(S'_i) < h(S_i) \leq h(Q)$ and $h(T'_j) < h(T_i) \leq h(R)$, it follows that $S'_i \| T'_j \prec P$. So, by the induction hypothesis there exists an H-normal form N such that $S'_i \parallel T'_j \approx_{\rm H} N$ and $h(S'_i \parallel T'_j) \geq h(N)$. Then clearly $\tau.N$ is an H-normal form such that $S_i \mid T_j \approx_{\rm H} \tau.N$ and $h(S_i \mid T_j) \geq h(\tau.N)$.

(f) Suppose $S_i = a.S'_i$ and $T_j = x \mathbin{|\!\!\lfloor} T'_j$. Then

$$\begin{aligned} a.S'_i \mid (x \mathbin{|\!\!\lfloor} T'_j) &\approx a.(\mathbf{0} \parallel S'_i) \mid (x \mathbin{|\!\!\lfloor} T'_j) && \text{(by P4, P3 in Lemma 2.10)} \\ &\approx (a \mathbin{|\!\!\lfloor} S'_i) \mid (x \mathbin{|\!\!\lfloor} T'_j) && \text{(by L2)} \\ &\approx (x \mid a) \mathbin{|\!\!\lfloor} (S'_i \parallel T'_j) && \text{(by Lemma 2.3 and C5).} \end{aligned}$$

Since $h(S'_i) < h(S_i) \leq h(Q)$ and $h(T'_j) < h(T_i) \leq h(R)$, it follows that $S'_i \parallel T'_j \prec P$, and hence by the induction hypothesis there exists an H-normal form N such that $S'_i \parallel T'_j \approx_{\rm H} N$ and $h(S'_i \parallel T'_j) \geq h(N)$. Then clearly $(x \mid a) \mathbin{|\!\!\lfloor} N$ is an H-normal form such that $S_i \mid T_j \approx_{\rm H} (x \mid a) \mathbin{|\!\!\lfloor} N$ and $h(S_i \mid T_j) \geq h((x \mid a) \mathbin{|\!\!\lfloor} N)$.

(g) If $S_i = x \mathbin{|\!\!\lfloor} S'_i$ and $T_j = a.T'_j$, then the proof is analogous to the previous case.

(h) Suppose $S_i = x \mathbin{|\!\!\lfloor} S'_i$ and $T_j = y \mathbin{|\!\!\lfloor} T'_j$. Then, by the derived equation C8 (see Lemma 2.3)

$$S_i \mid T_j \approx (x \mid y) \mathbin{|\!\!\lfloor} (S'_i \parallel T'_j) \ .$$

Since $h(S'_i) < h(S_i) \leq h(Q)$ and $h(T'_j) < h(T_i) \leq h(R)$, it follows that $S'_i \parallel T'_j \prec P$, and hence by the induction hypothesis there exists an H-normal form N such that $S'_i \parallel T'_j \approx_{\rm H} N$ and $h(S'_i \parallel T'_j) \geq h(N)$. Then clearly $(x \mid y) \mathbin{|\!\!\lfloor} N$ is an H-normal form such that $S_i \mid T_j \approx_{\rm H} (x \mid y) \mathbin{|\!\!\lfloor} N$ and $h(S_i \mid T_j) \geq h((x \mid y) \mathbin{|\!\!\lfloor} N)$.

(7) If $P = Q \parallel R$, then $P \approx Q \mathbin{|\!\!\lfloor} R + R \mathbin{|\!\!\lfloor} Q + Q \mid R$. We can now proceed as in case 5 to show that for $Q \mathbin{|\!\!\lfloor} R$ and $R \mathbin{|\!\!\lfloor} Q$ there exist H-normal forms N_1 and N_2, respectively, such that $Q \mathbin{|\!\!\lfloor} R \approx_{\rm H} N_1$, $R \mathbin{|\!\!\lfloor} Q \approx_{\rm H} N_2$, $h(Q \mathbin{|\!\!\lfloor} R) \geq h(N_1)$ and $h(R \mathbin{|\!\!\lfloor} Q) \geq h(N_2)$. Furthermore, we can proceed as in case 6 to show that for $Q \mid R$ there exists an H-normal form N_3 such that $Q \mid R \approx_{\rm H} N_3$ and $h(Q \mid R) \geq h(N_3)$. Then $N_1 + N_2 + N_3$ is an H-normal form such that $P \approx_{\rm H} N_1 + N_2 + N_3$ and $h(P) \geq h(N_1 + N_2 + N_3)$. □

We proceed to establish that for every two H-normal forms N_1 and N_2 there exists a valuation $* : \mathcal{V} \to \mathbf{P}_h$ such that

$$\text{if } [\![N_1]\!]_* = [\![N_2]\!]_*, \text{ then } N_1 \approx_{\mathrm{H}} N_2. \tag{4}$$

The distinguishing valuations $*$ will have a slightly more complicated definition than before, because of the more complicated notion of normal form.

As in the previous section, the definition of $*$ is parametrised with a natural number W. Since $|$ may now occur in H-normal forms, we also need to make sure that whatever process $*$ assigns to variables has sufficient communication abilities. To achieve this, we also parametrise $*$ with a finite subset $\mathcal{A}' = \{a_1, \ldots, a_n\}$ of $\mathcal{A}$ that is closed under the bijection $\bar{\ }$ on $\mathcal{A}$. (Note that every finite subset of $\mathcal{A}$ has a finite superset with the aforementioned property.) Based on W and $\mathcal{A}'$ we define the valuation $* : \mathcal{V} \to \mathbf{P}_h$ by

$$*(x) = a_1.\varphi_{(1 \cdot \ulcorner x \urcorner)} + \cdots + a_n.\varphi_{(n \cdot \ulcorner x \urcorner)} \ .$$

We shall prove that $*$ satisfies Eqn. (4) if the actions occurring in N_1 and N_2 are in $\mathcal{A}' \cup \{\tau\}$ and the widths of N_1 and N_2, defined below, do not exceed W. We must also be careful to define the injection $\ulcorner\_\urcorner$ in such a way that the extra factors $1, \ldots, n$ in the definition of $*$ do not interfere with the numbers assigned to variables; we let $\ulcorner\_\urcorner$ denote an injection

$$\ulcorner\_\urcorner : \mathcal{V} \to \{m : m \text{ a prime number such that } m > n \text{ and } m > W\}$$

that associates with every variable a prime number greater than the cardinality of $\mathcal{A}'$ and greater than W.

The definition of width also needs to take into account the cardinality of $\mathcal{A}'$ to maintain that the maximal branching degree in $[\![N]\!]_*$ does not exceed $w(N)$.

Definition 4.5. We define the *width* $w(N)$ of an H-normal form N as follows:

(1) if $N = \mathbf{0}$, then $w(N) = 0$;
(2) if $N = N_1 + N_2$, then $w(N) = w(N_1) + w(N_2)$;
(3) if $N = \alpha.N'$, then $w(N) = \max(w(N'), 1)$;
(4) if $N = x \mathbin{|\!\!\lfloor} N'$, then $w(N) = \max(w(N'), n)$;
(5) if $N = (x \mid a) \mathbin{|\!\!\lfloor} N'$, then $w(N) = \max(w(N'), 1)$; and
(6) if $N = (x \mid y) \mathbin{|\!\!\lfloor} N'$, then $w(N) = \max(w(N'), n)$.

Lemma 4.6. *For every* H*-normal form N, the branching degree of $[\![N]\!]_*$ is at most $w(N)$.*

Proof. Structural induction on N. □

Lemma 4.7. *Let S be a simple* H*-normal form, let $\alpha \in \mathcal{A}_\tau$, and let p be a process such that $[\![S]\!]_* \xrightarrow{\alpha} p$. Then the following statements hold:*

(1) if $S = \beta.N$, then $\alpha = \beta$ and $p = [\![N]\!]_*$;
(2) if $S = x \mathbin{|\!\!\lfloor} N$, then $\alpha = a_i$ and $p = \varphi_{i \cdot \ulcorner x \urcorner} \parallel [\![N]\!]_*$ for some $i \in \{1, \ldots, n\}$;
(3) if $S = (x \,|\, a) \mathbin{|\!\!\lfloor} N$, then $\alpha = \tau$ and $p = \varphi_{i \cdot \ulcorner x \urcorner} \parallel [\![N]\!]_*$ for the unique $i \in \{1, \ldots, n\}$ such that $\bar{a} = a_i$; and

(*4*) *if* $S = (x \mid y) \mathbin{\|\!\!\_} N$, *then* $\alpha = \tau$ *and* $p = \varphi_{i\cdot\ulcorner x\urcorner} \parallel \varphi_{j\cdot\ulcorner y\urcorner} \parallel [\![N]\!]_*$ *for some* $i, j \in \{1, \ldots, n\}$ *such that* $\overline{a_i} = a_j$.

As in the previous section, we distinguish H-normal forms by classifying their residuals according to the number of parallel components with a branching degree that exceeds W. Again, we say that a process p is of *type* n ($n \geq 0$) if its unique parallel decomposition contains precisely n parallel prime components with a branching degree larger than W.

Corollary 4.8. *Let* S *be a simple* H*-normal form such that* $w(S) \leq W$ *and such that the actions occurring in* S *are included in* $\mathcal{A}' \cup \{\tau\}$.

(*1*) *If* $S = \alpha.N$, *then the unique residual of* $[\![S]\!]_*$ *is of type* 0.
(*2*) *If* $S = x \mathbin{\|\!\!\_} N$, *then all residuals of* $[\![S]\!]_*$ *are of type* 1.
(*3*) *If* $S = (x \mid a) \mathbin{\|\!\!\_} N$, *then the unique residual of* $[\![S]\!]_*$ *is of type* 1.
(*4*) *If* $S = (x \mid y) \mathbin{\|\!\!\_} N$, *then all residuals of* $[\![S]\!]_*$ *are of type* 2.

Proof. On the one hand, by Lemma 4.6, in each case $[\![N]\!]_*$ has a branching degree of at most $w(N) \leq w(S) \leq W$, and hence, by Lemma 2.13, its unique parallel decomposition cannot contain parallel prime components with a branching degree that exceeds W. On the other hand, by Lemmas 2.14(1) and 2.14(3), the processes $\varphi_{i\cdot\ulcorner x\urcorner}$ and $\varphi_{j\cdot\ulcorner y\urcorner}$ are parallel prime and have a branching degree that exceeds W. Further note that, since the assumption on CCS communication functions that $\overline{a} \neq a$ implies that $i \neq j$, the processes $\varphi_{i\cdot\ulcorner x\urcorner}$ and $\varphi_{j\cdot\ulcorner y\urcorner}$ are distinct. Using these observations it is straightforward to establish the corollary as a consequence of Lemma 4.7. □

Theorem 4.9. *For every two* H*-normal forms* N_1, N_2 *such that* $w(N_1), w(N_2) \leq W$ *and such that the actions occurring in* N_1 *and* N_2 *are included in* $\mathcal{A}' \cup \{\tau\}$ *it holds that* $[\![N_1]\!]_* = [\![N_2]\!]_*$ *only if* $N_1 \approx N_2$ *modulo* A1–A4, C5.

Proof. By Lemma 2.6 we may assume that N_1 and N_2 are summations of collections of simple H-normal forms. We assume $[\![N_1]\!]_* = [\![N_2]\!]_*$ and prove that then $N_1 \approx N_2$ modulo A1–A4, C5, by induction on the sum of the heights of N_1 and N_2.

We first prove that for every syntactic summand S_1 of N_1 there is a syntactic summand S_2 of N_2 such that $S_1 \approx S_2$ modulo A1–A4, C5. To this end, let S_1 be an arbitrary syntactic summand of N_1; we distinguish cases according to the syntactic form of S_1.

(1) Suppose $S_1 = \alpha.N_1'$; then $[\![S_1]\!]_* \xrightarrow{\alpha} [\![N_1']\!]_*$. Hence, since $[\![N_1]\!]_* = [\![N_2]\!]_*$, there exists a syntactic summand S_2 of N_2 such that $[\![S_2]\!]_* \xrightarrow{\alpha} [\![N_1']\!]_*$. By Lemma 4.6 the branching degree of $[\![N_1']\!]_*$ does not exceed W, so $[\![S_2]\!]_*$ has a residual of type 0, and therefore, by Corollary 4.8, there exist $\beta \in \mathcal{A}_\tau$ and an H-normal form N_2' such that $S_2 = \beta.N_2'$. Moreover, since $[\![S_2]\!]_* \xrightarrow{\alpha} [\![N_1']\!]_*$ it follows by Lemma 4.7(1) that $\alpha = \beta$ and $[\![N_1']\!]_* = [\![N_2']\!]_*$. Hence, by the induction hypothesis, we conclude that $N_1' \approx N_2'$ modulo A1–A4, C5. So $S_1 = \alpha.N_1' \approx \beta.N_2' = S_2$.

(2) Suppose $S_1 = x \mathbin{\|\!\!\_} N_1'$; then $[\![S_1]\!]_* \xrightarrow{a_1} \varphi_{\ulcorner x\urcorner} \parallel [\![N_1']\!]_*$. Hence, since $[\![N_1]\!]_* = [\![N_2]\!]_*$, there exists a summand S_2 of N_2 such that $[\![S_2]\!]_* \xrightarrow{a_1} \varphi_{\ulcorner x\urcorner} \parallel [\![N_1']\!]_*$. Since S_2 has

a residual of type 1, by Corollary 4.8(1, 4) it is not of the form $\alpha.N_2'$ for some $\alpha \in \mathcal{A}_\tau$ and H-normal form N_2', or of the form $(y \mid z) \mathbin{\|\!\!\lfloor} N_2'$ for some $y, z \in \mathcal{V}$ and H-normal form N_2'. Moreover, S_2 cannot be of the form $(y \mid a) \mathbin{\|\!\!\lfloor} N_2'$ for some $y \in \mathcal{V}$ and $a \in \mathcal{A}$, for then by Lemma 4.7(3) $\llbracket S_2 \rrbracket_* \xrightarrow{\alpha} p$ would imply $\alpha = \tau \neq a_1$. So, there exists a variable y and an H-normal form N_2' such that $S_2 = y \mathbin{\|\!\!\lfloor} N_2'$. Now, since $\llbracket S_2 \rrbracket_* \xrightarrow{a_1} \varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$, it follows by Lemma 4.7(2) that

$$\varphi_{\ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_* = \varphi_{\ulcorner y \urcorner} \parallel \llbracket N_2' \rrbracket_* \ . \tag{5}$$

Since $\llbracket N_1' \rrbracket_*$ and $\llbracket N_2' \rrbracket_*$ are of type 0, we conclude that the unique decomposition of $\llbracket N_1' \rrbracket_*$ does not contain $\varphi_{\ulcorner y \urcorner}$ and the unique decomposition of $\llbracket N_2' \rrbracket_*$ does not contain $\varphi_{\ulcorner x \urcorner}$. Hence, from (5) it follows that $\varphi_{\ulcorner x \urcorner} = \varphi_{\ulcorner y \urcorner}$ and $\llbracket N_1' \rrbracket_* = \llbracket N_2' \rrbracket_*$. From the former we conclude by the injectivity of $\ulcorner . \urcorner$ that $x = y$, and from the latter we conclude by the induction hypothesis that $N_1' \approx N_2'$ modulo A1–A4, C5. So $S_1 = x \mathbin{\|\!\!\lfloor} N_1' \approx y \mathbin{\|\!\!\lfloor} N_2' = S_2$.

(3) Suppose $S_1 = (x \,|\, a) \mathbin{\|\!\!\lfloor} N_1'$, and let i be such that $\overline{a} = a_i$. Then $\llbracket S_1 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$. Hence, since $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$, there exists a summand S_2 of N_2 such that

$$\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_* \ .$$

Since S_2 has a residual of type 1, by Corollary 4.8(1,4) it is not of the form $\alpha.N_2'$ for some $\alpha \in \mathcal{A}_\tau$ and H-normal form N_2', or of the form $(y \mid z) \mathbin{\|\!\!\lfloor} N_2'$ for some $y, z \in \mathcal{V}$ and H-normal form N_2'. Moreover, S_2 cannot be of the form $y \mathbin{\|\!\!\lfloor} N_2'$ for some $y \in \mathcal{V}$, for then by Lemma 4.7(2) $\llbracket S_2 \rrbracket_* \xrightarrow{\alpha} p$ would imply $\alpha = a_k \neq \tau$ for some $k \in \{1, \ldots, n\}$. So, there exist a variable y, action $b \in \mathcal{A}'$ and an H-normal form N_2' such that $S_2 = (y \,|\, b) \mathbin{\|\!\!\lfloor} N_2'$. Now, since $\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_*$, it follows by Lemma 4.7(3) that

$$\varphi_{i \cdot \ulcorner x \urcorner} \parallel \llbracket N_1' \rrbracket_* = \varphi_{j \cdot \ulcorner y \urcorner} \parallel \llbracket N_2' \rrbracket_* \ , \tag{6}$$

with $j \in \{1, \ldots, n\}$ such that $\overline{b} = a_j$. By Lemma 2.14(1,3) the processes $\varphi_{i \cdot \ulcorner x \urcorner}$ and $\varphi_{j \cdot \ulcorner y \urcorner}$ are parallel prime and have branching degrees that, since $\ulcorner x \urcorner > W$ and $\ulcorner y \urcorner > W$, exceed W. Therefore, since $\llbracket N_1' \rrbracket_*$ and $\llbracket N_2' \rrbracket_*$ are of type 0, it follows that the unique decomposition of $\llbracket N_1' \rrbracket_*$ does not contain $\varphi_{j \cdot \ulcorner y \urcorner}$ and the unique decomposition of $\llbracket N_2' \rrbracket_*$ does not contain $\varphi_{i \cdot \ulcorner x \urcorner}$. Hence, by (6) we have that $\varphi_{i \cdot \ulcorner x \urcorner} = \varphi_{j \cdot \ulcorner y \urcorner}$ and $\llbracket N_1' \rrbracket_* = \llbracket N_2' \rrbracket_*$. From $\varphi_{i \cdot \ulcorner x \urcorner} = \varphi_{j \cdot \ulcorner y \urcorner}$, by Lemma 2.14(2) we infer that $i \cdot \ulcorner x \urcorner = j \cdot \ulcorner y \urcorner$. Since $\ulcorner x \urcorner$ and $\ulcorner y \urcorner$ are prime numbers greater than i and j, it follows that $i = j$, whence $a = b$, and $\ulcorner x \urcorner = \ulcorner y \urcorner$, whence $x = y$ by the injectivity of $\ulcorner . \urcorner$. From $\llbracket N_1' \rrbracket_* = \llbracket N_2' \rrbracket_*$ we conclude by the induction hypothesis that $N_1' \approx N_2'$ modulo A1–A4, C5. So $S_1 = (x \mid a) \mathbin{\|\!\!\lfloor} N_1' \approx (y \mid b) \mathbin{\|\!\!\lfloor} N_2' = S_2$.

(4) Suppose $S_1 = (x \mid y) \mathbin{\|\!\!\lfloor} N_1'$. Then $\llbracket S_1 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel \varphi_{j \cdot \ulcorner y \urcorner} \parallel \llbracket N_1' \rrbracket_*$ with $i, j \in \{1, \ldots, n\}$ such that $\overline{a_i} = a_j$. Hence, since $\llbracket N_1 \rrbracket_* = \llbracket N_2 \rrbracket_*$, there exists a summand S_2 of N_2 such that

$$\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel \varphi_{j \cdot \ulcorner y \urcorner} \parallel \llbracket N_1' \rrbracket_* \ .$$

Since S_2 has a residual of type 2, by Corollary 4.8 there exist $x', y' \in \mathcal{V}$ and an H-normal form N_2' such that $S_2 = (x' \mid y') \mathbin{\|\!\!\lfloor} N_2'$. Now, since $\llbracket S_2 \rrbracket_* \xrightarrow{\tau} \varphi_{i \cdot \ulcorner x \urcorner} \parallel$

$\varphi_{j \cdot \ulcorner y \urcorner} \parallel [\![N_1']\!]_*$ it follows by Lemma 4.7(4) that for some $k, l \in \{1, \ldots, n\}$ such that $\overline{a_k} = a_l$

$$\varphi_{i \cdot \ulcorner x \urcorner} \parallel \varphi_{j \cdot \ulcorner y \urcorner} \parallel [\![N_1']\!]_* = \varphi_{k \cdot \ulcorner x \urcorner} \parallel \varphi_{l \cdot \ulcorner y \urcorner} \parallel [\![N_2']\!]_* \ . \tag{7}$$

By Lemma 2.14(1,3) the processes $\varphi_{i \cdot \ulcorner x \urcorner}$, $\varphi_{j \cdot \ulcorner y \urcorner}$, $\varphi_{k \cdot \ulcorner x \urcorner}$ and $\varphi_{l \cdot \ulcorner y \urcorner}$ are parallel prime and have branching degrees that exceed W. Therefore, since $[\![N_1']\!]_*$ and $[\![N_2']\!]_*$ are of type 0, it follows that the unique decomposition of $[\![N_1']\!]_*$ does not contain $\varphi_{k \cdot \ulcorner x \urcorner}$ and $\varphi_{l \cdot \ulcorner y \urcorner}$, and the unique decomposition of $[\![N_2']\!]_*$ does not contain $\varphi_{i \cdot \ulcorner x \urcorner}$ and $\varphi_{j \cdot \ulcorner y \urcorner}$. Hence, from (7) we infer that $[\![N_1']\!]_* = [\![N_2']\!]_*$ and either $\varphi_{i \cdot \ulcorner x \urcorner} = \varphi_{k \cdot \ulcorner x \urcorner}$ and $\varphi_{j \cdot \ulcorner y \urcorner} = \varphi_{l \cdot \ulcorner y \urcorner}$, or $\varphi_{i \cdot \ulcorner x \urcorner} = \varphi_{l \cdot \ulcorner y \urcorner}$ and $\varphi_{j \cdot \ulcorner y \urcorner} = \varphi_{k \cdot \ulcorner x \urcorner}$. From the former we conclude by the induction hypothesis that $N_1' \approx N_2'$ modulo A1–A4, C5; from the latter it follows reasoning as in case 3 that either $x = x'$ and $y = y'$, or $x = y'$ and $y = x'$. In both cases, $S_1 = (x \mid y) \mathbin{|\!\lfloor} N_1' \approx (x' \mid y') \mathbin{|\!\lfloor} N_2' = S_2$.

We have established that every syntactic summand of N_1 is provably equal to a syntactic summand of N_2. Similarly, it follows that every syntactic summand of N_2 is provably equal to a syntactic summand of N_2. Hence, modulo A1–A4, C5 $N_1 \approx N_1 + N_2 \approx N_2$, and the proof of the theorem is complete. □

Corollary 4.10. *For all process terms P and Q, $P \approx_{\mathrm{H}} Q$ if, and only if, $P \leftrightarroweq_h Q$, and hence the axioms generated by the schemata in Table II together with the axiom* H *consitute an equational base for $\mathbf{P}_h$.*

Proof. The implication from left to right is Proposition 4.1. To prove the implication from right to left, suppose $P \leftrightarroweq_h Q$. Then, by Lemma 4.4 there exist H-normal forms N_1 and N_2 such that $P \approx_{\mathrm{H}} N_1$ and $Q \approx_{\mathrm{H}} N_2$; from $P \leftrightarroweq_h Q$ we conclude by Proposition 4.1 that $N_1 \leftrightarroweq_h N_2$. Now choose W large enough such that $w(N_1), w(N_2) \leq W$, and pick a finite set $\mathcal{A}'$ that is closed under $\bar{\cdot}$ and includes all of the actions occurring in N_1 and N_2. From $N_1 \leftrightarroweq_h N_2$ it follows that $[\![N_1]\!]_* = [\![N_2]\!]_*$, and hence, by Theorem 4.9 $N_1 \approx N_2$. We can therefore conclude $P \approx_{\mathrm{H}} N_1 \approx N_2 \approx_{\mathrm{H}} Q$. □

Corollary 4.11. *The equational theory of $\mathbf{P}_h$ is decidable.*

Proof. From the proof of Lemma 4.4 it is easy to see that there exists an effective procedure that associates with every process term a provably equivalent H-normal. Furthermore, from Definition 4.5 it is clear that, given a set $\mathcal{A}'$, every H-normal form has an effectively computable width. We now sketch an effective procedure that decides whether a process equation $P \approx Q$ is valid:

(1) Compute H-normal forms N_1 and N_2 such that $P \approx_{\mathrm{H}} N_1$ and $Q \approx_{\mathrm{H}} N_2$.

(2) Determine the least set $\mathcal{A}' = \{a_1, \ldots, a_n\}$ of actions that is closed under $\bar{\cdot}$ and contains the actions in $\mathcal{A}$ occurring in N_1 and N_2.

(3) Compute $w(N_1)$ and $w(N_2)$ given $\mathcal{A}'$, and define W as their maximum.

(4) Determine the (finite) set $\mathcal{V}'$ of variables occurring in N_1 and N_2; define an injection

$$\ulcorner . \urcorner : \mathcal{V}' \to \{m \in \omega : m \text{ a prime number such that } m > n \text{ and } m > W\} \ ,$$

and a substitution $* : \mathcal{V}' \to \mathcal{P}_0$ that assigns to a variable x in $\mathcal{V}'$ the closed process term

$$a_1.\varphi_1.\ulcorner x \urcorner + \cdots + a_n.\varphi_n.\ulcorner x \urcorner \ .$$

(Again, we interpret Eqn. (1) as defining a sequence of closed process terms instead of a sequence of processes.)

(5) Let N_1^* and N_2^* be the results from applying $*$ to N_1 and N_2, respectively.

(6) Determine if the closed process terms N_1^* and N_2^* are bisimilar; if they are, then the process equation $P \approx Q$ is valid in $\mathbf{P}_h$, and otherwise it is not. □

5. CONCLUDING REMARKS

We have discussed the equational theories of two process algebras arising from the fragment of CCS without recursion, restriction and relabelling. Moller [1990] has proved that these equational theories are not finitely based. We have shown that if the set of actions is finite and the auxiliary operators left merge and communication merge from Bergstra and Klop [1984] are added, then finite equational bases can be obtained. They consist of (adaptations of) axioms appearing already in [Bergstra and Klop 1984; Bergstra and Tucker 1985; Hennessy and Milner 1985].

Denote by $\mathcal{E}$ the set of the axioms generated by the schemata in Table II on p. 6 together with the axiom $x \mid (y \mid z) \approx \mathbf{0}$, which expresses that the communication mechanism conforms to the *handshaking paradigm*. Our main result (Corollary 4.10) establishes that $\mathcal{E}$ is an equational base for the algebra $\mathbf{P}_h$. Note that an equational base for an algebra is an equational base for every extension of that algebra in which the axioms hold.[1] So, as a consequence of our result, $\mathcal{E}$ is in fact an equational base, e.g., for every algebra of process graphs modulo bisimulation endowed with a distinguished element $\mathbf{0}$ and operations $\alpha.$ ($\alpha \in \mathcal{A}_\tau$), $+$, $\|$, $\mathbin{|\!\lfloor}$ and $|$ according to their standard interpretations. In particular it is clear from the preceding remarks that, although the algebra $\mathbf{P}_h$ contains only finite processes, this is not essential for our result.

As a special case of Corollary 4.10, the axiom system $\mathcal{E}$ is *ground-complete* with respect to bisimilarity (i.e., $\approx_{\mathrm{H}}$ coincides with $\underline{\leftrightarrow}_h$ on the set of *closed* terms $\mathcal{P}_0$). Consequently, the algebra $\mathbf{P}_h$ is isomorphic with the *initial algebra* associated with $\mathcal{E}$, i.e., the quotient of the set of closed terms modulo $\approx_{\mathrm{H}}$. It also follows from Corollary 4.10 that the axiom system $\mathcal{E}$ is *ω-complete.* For suppose that every closed instance of the equation $P \approx Q$ is derivable; then the equation itself is valid in the initial algebra. By ground-completeness, it follows that $P \approx Q$ is valid in $\mathbf{P}_h$, and hence, by Corollary 4.10, it is derivable from $\mathcal{E}$.

As a stepping stone towards our main result, we first considered the process algebra $\mathbf{P}_f$ with a trivial communication mechanism. An equational base for it is obtained if the axiom $x \mid y \approx \mathbf{0}$ is added to the axioms generated by the schemata in Table II on p. 6 (Corollary 3.9). The auxiliary operator $|$ is then actually superfluous. For we can replace P1 by $x \parallel y \approx x \mathbin{|\!\lfloor} y + y \mathbin{|\!\lfloor} x$, and, moreover, transform every equational proof into a proof in which $|$ does not occur by replacing every occurrence of a subexpression $P|\,Q$ by $\mathbf{0}$. It follows that the axiomatisation consisting

[1] The algebra $\mathbf{B}$ is an extension of the algebra $\mathbf{A}$ if there exists an embedding, i.e., an injective homomorphism, from $\mathbf{A}$ into $\mathbf{B}$.

of A1–A4, L1–L5, and the simplified axiom P1 is ω-complete. Thus, we generalise the result of Moller [1989], who establishes ω-completeness of the axiomatisation under the condition that the set of actions is infinite; according to our result the condition can be omitted.

The proofs that the presented axiomatisations are indeed complete for the algebras $\mathbf{P}_f$ and $\mathbf{P}_h$ proceed in two steps. The first step consists of identifying an appropriate collection of normal forms and proving that every process term is provably equal to a normal form. The second step consists of associating with every pair of normal forms a *distinguishing valuation*, i.e., a valuation such that if the two normal forms are equal under this particular valuation, then the normal forms are provably equal. In both cases considered in this article, the first step is fairly straightforward. The second step makes essential use of the property of unique parallel decomposition that holds in the algebras $\mathbf{P}_f$ and $\mathbf{P}_h$.

We now proceed to discuss the extension of our results with restrictions and relabellings, and recursion, and then we comment on the complications that would arise when trying to adapt our proofs for CCS modulo observation congruence.

5.1 Restrictions and Relabellings

In the case of the trivial communication function, restrictions distribute over left merges. If the standard axioms for restriction (see, e.g., [Milner 1980]) and the axiom $(x \mathbin{|\!\lfloor} y)\backslash L \approx x\backslash L \mathbin{|\!\lfloor} y\backslash L$ are added to the axioms in Sect. 3, then restrictions can be pushed all the way down to the variables in a process term. Thus, only a mild adaptation of the notion of F-normal form (cf. Definition 3.2 on p. 10) is needed, assuming that there is a restriction around the variable x in $x \mathbin{|\!\lfloor} N$. It is proved by van Tilburg [2007] that then for any pair of normal forms there exists a distinguishing valuation, which is a refinement of the distinguishing valuation used in this paper. We expect that a similar result can be obtained for the extension with relabellings.

In the case of the handshaking communication function, it is much less obvious what would be an appropriate notion of normal form in the presence of restrictions and relabellings. The reason is that in this case restrictions and relabellings do *not* distribute over parallel compositions, left merges and communication merges. To implement the two-step approach to proving completeness, it will be necessary to add further axioms explaining the relation between parallel compositions (and left merges and communication merges) on the one hand and restrictions and relabellings on the other hand. We refer to [van Tilburg 2007] for (an incomplete set of) additional axioms, and for a more elaborate discussion of the complexity arising from the nondistributivity of restrictions over parallel compositions.

5.2 Recursion

One way of including recursion in CCS is in the form of the fixed-point construction μX. The construction μX binds the free occurrences of the process variable X in the process expression to which it is applied. As a consequence, it does not give rise to a sensible operation on the algebra of closed process expressions modulo strong bisimilarity. Thus, the type of question considered in this paper (Is the algebra of closed process expressions modulo strong bisimilarity finitely based?) does not make sense if recursion is added in the form of a fixed-point construction.

However, the closely related question of whether a certain proof system for deriving equations between process expressions is ω-complete does make sense. In fact, it can be argued that the inference system for the fragment of CCS consisting of $\mathbf{0}$, action prefixing, choice and the fixed-point construction presented by Milner [1984] is ω-complete: whenever all closed substitution instances of an equation with free occurrences of process variables are derivable, then the equation itself is derivable too. Milner's inference system is based on equational logic, but it has an additional inference rule schema expressing the unique existence of certain fixed points, and it has axiom schemata with metavariables ranging over process expressions each generating infinitely many axioms. Due to the presence of the variable-binding construction μX, the metavariables cannot be treated as real algebraic variables (ranging over the elements of some process algebra). Thus, any axiomatisation involving the μX construction will be inherently infinite. Furthermore, [Sewell 1994] has proved that if λ-calculus is used to formally express the axiom schemata in a finite manner, then a finite axiomatisation not requiring the extension of equational logic with additional inference rules is impossible, even when restricting to closed μ-expressions.

Nevertheless, it is an interesting open question how to obtain a (finitely presented) ω-complete inference system for a fragment of CCS including both parallelism and the fixed-point construction. The extension of Milner's inference system with the axioms for parallelism discussed in this paper could be taken as a natural starting point, but most likely, it will be necessary to add further axioms explaining the interplay of the fixed-point construct and parallel composition. The proof that the resulting inference system is indeed ω-complete, will in any case not be a straightforward extension of the proof of Milner [1989a], nor of our proof in the present paper. The proof of Milner [1989a] for the regular fragment of CCS, on the one hand, crucially depends on the property that all behaviours defined by an expression in the considered fragment are finite-state, which is lost by the addition of parallelism. Our proof for the fragment without recursion, on the other hand, crucially depends on the property of unique parallel decomposition, which is lost by the addition of recursion.

An alternative inference system for reasoning about equality in CCS is discussed by Christensen et al. [1994]. They consider a fragment of CCS that includes parallelism, relabelling, restriction and recursion, the latter in the form of a facility for specifying processes by means of a family of recursive process equations. The inference system is sequent based and presupposes a recursive process specification in standard form (all right-hand sides of process equations are sums of prefixes of parallel compositions of variables). In addition, to transform any restriction/relabelling-free or any communication-free process specification into standard form, a collection of equational axioms is provided; it includes a variant of the Expansion Law to deal with parallel compositions.

The sound and complete inference system from Christensen et al. [1994] is for closed terms only; equations do not contain variables ranging over arbitrary processes.[2] Note that when variables are included in the syntax, the notion of stan-

[2] The syntax includes the notion of *process variable*, but it is, in fact, a constant symbol defined by a process equation.

dard form becomes considerably more complicated; in particular, it should allow for unguarded occurrences of parallel compositions (cf. our notion of normal form in Sect. 4).

Baeten and Bravetti [2005] consider a very rich process calculus that includes all the operations of ACP and CCS, and has a facility for specifying processes by means of a family of recursive process equations. They present an inference system for their calculus modulo observation congruence. It is based on equational logic and generalises the inference system of Milner [1989b] for the fragment of CCS consisting of **0**, action prefixing, choice and the fixed-point construction. Baeten and Bravetti [2005] prove that their inference system is ground-complete for a fragment of their calculus that includes parallel compositions and recursion, but disallows process variables at the right-hand side of process equations to occur within the scope of parallel compositions. They do not consider ω-completeness.

5.3 Observation congruence

It would be interesting to try and find also a finite equational base for CCS modulo observation congruence [Milner 1989a]. Of course, for a start, Milner's τ-laws should be added. Then, it will be a challenge to adapt our proofs, if at all possible. For instance, the property of unique parallel decomposition takes a more complicated shape (see [Moller 1989]). Also, our distinguishing valuation would need nontrivial adaptation. Naturally, we can no longer use τ to get a process with a distinguishingly high branching degree or long trace; we should use some observable action a for this. A further complication arises, for the equation $x \,|\, b \approx x \,|\, b + a$ is not valid (take, e.g., $\bar{b}$ for x), while it does hold under any valuation $*$ such that $*(x)$ can communicate with b and then proceed as a process with a summand a (simply replacing τ by a in the valuation we used in Sect. 4 would yield such a valuation). So, in general, the distinguishing valuation cannot have summands at depth 1 that also appear in the normal forms that should be distinguished.

ACKNOWLEDGMENT

We thank the anonymous referees for their careful reading. The detailed suggestions of one of the referees led to improvements of the presentation of the results and a discussion of the complications that arise when trying to extend them.

REFERENCES

Aceto, L., Fokkink, W. J., Ingolfsdottir, A., and Luttik, B. 2005a. CCS with Hennessy's merge has no finite equational axiomatization. *Theor. Comput. Sci. 330,* 3, 377–405.

Aceto, L., Fokkink, W. J., Ingolfsdottir, A., and Luttik, B. 2005b. Finite equational bases in process algebra: Results and open questions. In *Processes, Terms and Cycles: Steps on the Road to Infinity,* A. Middeldorp, V. van Oostrom, F. van Raamsdonk, and R. C. de Vrijer, Eds. Lecture Notes in Computer Science, vol. 3838. Springer, 338–367.

Baeten, J. C. M. and Bravetti, M. 2005. A ground-complete axiomatization of finite state processes in process algebra. In *Proceedings of CONCUR'05*, M. Abadi and L. de Alfaro, Eds. Lecture Notes in Computer Science, vol. 3653. Springer, 248–262.

Bergstra, J. A. and Klop, J. W. 1984. Process algebra for synchronous communication. *Inform. and Control 60,* 1-3, 109–137.

Bergstra, J. A. and Tucker, J. V. 1985. Top-down design and the algebra of communicating processes. *Sci. Comput. Programming 5,* 2, 171–199.

Christensen, S., Hirshfeld, Y., and Moller, F. 1994. Decidable subsets of CCS. *Comput. J. 37,* 4, 233–242.

de Simone, R. 1985. Higher-level synchronising devices in Meije-SCCS. *Theor. Comput. Sci. 37*, 245–267.

Groote, J. F. 1990. A new strategy for proving ω-completeness applied to process algebra. In *Proceedings of CONCUR'90*, J. C. M. Baeten and J. W. Klop, Eds. Lecture Notes in Computer Science, vol. 458. Springer, 314–331.

Heering, J. 1986. Partial evaluation and ω-completeness of algebraic specifications. *Theoret. Comput. Sci. 43,* 2-3, 149–167.

Hennessy, M. 1988. Axiomatising finite concurrent processes. *SIAM J. Comput. 17,* 5, 997–1017.

Hennessy, M. and Milner, R. 1985. Algebraic laws for nondeterminism and concurrency. *J. ACM 32,* 1 (Jan.), 137–161.

Luttik, B. and van Oostrom, V. 2005. Decomposition orders—another proof of the fundamental theorem of arithmetic. *Theor. Comput. Sci. 335,* 2–3, 147–186.

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

Milner, R. 1984. A complete inference system for a class of regular behaviours. *J. Comput. Syst. Sci. 28,* 3, 439–466.

Milner, R. 1989a. *Communication and Concurrency.* Prentice-Hall International.

Milner, R. 1989b. A complete axiomatisation for observational congruence of finite-state behaviors. *Inf. Comput. 81,* 2, 227–247.

Milner, R. and Moller, F. 1993. Unique decomposition of processes. *Theoret. Comput. Sci. 107*, 357–363.

Moller, F. 1989. Axioms for concurrency. Ph.D. thesis, University of Edinburgh.

Moller, F. 1990. The nonexistence of finite axiomatisations for CCS congruences. In *Proceedings of LICS'90.* IEEE Computer Society Press, 142–153.

Park, D. M. R. 1981. Concurrency and automata on infinite sequences. In 5^{th} *GI Conference*, P. Deussen, Ed. Lecture Notes in Computer Science, vol. 104. Springer, 167–183.

Sewell, P. 1994. Bisimulation is not finitely (first order) equationally axiomatisable. In *Proceedings of LICS'94.* IEEE Computer Society, 62–70.

van Tilburg, P. J. A. 2007. Finite equational bases for CCS with restriction. M.S. thesis, Technische Universiteit Eindhoven.

Received August 2006; revised July 2007; accepted July 2007