Centrum voor Wiskunde en Informatica

REPORT RAPPORT

A complete equational axiomatisation for prefix iteration

W.J. Fokkink

Computer Science/Department of Software Technology

Report CS-R9415 February 1994

CWI is the National Research Institute for Mathematics and Computer Science. CWI is part of the Stichting Mathematisch Centrum (SMC), the Dutch foundation for promotion of mathematics and computer science and their applications.
SMC is sponsored by the Netherlands Organization for Scientific Research (NWO). CWI is a member of ERCIM, the European Research Consortium for Informatics and Mathematics.

P.O. Box 94079, 1090 GB Amsterdam (NL)
Kruislaan 413, 1098 SJ Amsterdam (NL)
Telephone +31 20 592 9333
Telefax +31 20 592 4199

A Complete Equational Axiomatisation for Prefix Iteration

Willem Jan Fokkink
CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
e-mail: `wan@cwi.nl`

Abstract

Iteration is added to *Minimal Process Algebra* (MPA_δ), which is a subset of BPA_δ that is equivalent to Milner's basic CCS. We present an equational axiomatisation for MPA_δ^*, and prove that this axiomatisation is complete with respect to strong bisimulation equivalence. To obtain this result, we will set up a term rewriting system, based on the axioms, and show that bisimilar terms have the same normal form.

AMS Subject Classification (1991): 68Q40, 68Q42.
CR Subject Classification (1991): D.3.1, F.3.2, F.4.2.
Keywords and Phrases: Minimal Process Algebra, iteration, complete equational axioms.

1 Introduction

Kleene [Kle56] defined a binary operator * in the context of finite automata, where E^*F denotes the *iterate* of E and F. Milner [Mil84] studied Kleene's star in the setting of (strong) bisimulation equivalence, and raised the question whether there exists a complete axiomatisation for it. Bergstra, Bethke & Ponse [BBP93] incorporated the binary Kleene star into Basic Process Algebra (BPA) [BK84], and they suggested three axioms for iteration. In [FZ93] it was proved that these three axioms, together with the five standard axioms of BPA, are a complete axiomatisation for BPA* with respect to bisimulation.

In this paper, we add the deadlock δ to the syntax. Sewell [Sew93] has proved that there does not exist a complete finite equational axiomatisation for BPA_δ^*. Therefore, we restrict the binary sequential composition $x \cdot y$ to the unary *prefix* sequential composition $a \cdot x$, to obtain *Minimal Process Algebra* MPA_δ, equivalent to basic CCS [Mil80]. Moreover, we add (prefix) iteration a^*x to the syntax, resulting in the algebra MPA_δ^*. We propose two simple axioms for iteration, and prove that these two axioms together with four standard axioms from BPA_δ are a complete axiomatisation for MPA_δ^* with respect to bisimulation.

The proof consists of producing a term rewriting system from the axioms, and showing that bisimilar normal forms are syntactically equal modulo commutativity and associativity of the +. Hence, bisimilarity between terms is decidable.

Acknowledgements: Jan Bergstra and Jos van Wamel are thanked for helpful comments.

Report CS-R9415
ISSN 0169-118X
CWI
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands

2 Minimal Process Algebra with Iteration

We assume an alphabet A of atomic actions. The terms in our algebra $\mathrm{MPA}^*_\delta(A)$ are built from a constant δ, which represents deadlock, together with the binary alternative composition $x+y$, and the unary prefix sequential composition $a \cdot x$ and iteration a^*x, for $a \in A$. Table 1 presents an operational semantics for $\mathrm{MPA}^*_\delta(A)$ in Plotkin style [Plo81]. The special symbol $\surd$ (pronounce 'tick') in this table represents successful termination.

$$\frac{x \xrightarrow{a} x'}{x+y \xrightarrow{a} x' \quad y+x \xrightarrow{a} x'} \qquad \frac{x \xrightarrow{a} \surd}{x+y \xrightarrow{a} \surd \quad y+x \xrightarrow{a} \surd}$$

$$a \cdot x \xrightarrow{a} x \qquad a^*x \xrightarrow{a} a^*x$$

$$\frac{x \xrightarrow{b} x'}{a^*x \xrightarrow{b} x'} \qquad \frac{x \xrightarrow{b} \surd}{a^*x \xrightarrow{b} \surd}$$

Table 1: Action rules for $\mathrm{MPA}^*_\delta(A)$

Our model for $\mathrm{MPA}^*_\delta(A)$ consists of all the closed terms that can be constructed from the atomic actions and the three operators. That is, the BNF grammar for the collection of process terms is as follows, where $a \in A$:

$$p \quad ::= \quad \delta \mid p+p \mid a \cdot p \mid a^*p$$

In the sequel the operator $\cdot$ will often be omitted, so ap denotes $a \cdot p$. As binding convention, $*$ binds stronger than $\cdot$, which in turn binds stronger than $+$.

Process terms are considered modulo *(strong) bisimulation equivalence* [Par81]. Intuitively, two process terms are bisimilar if they have the same branching structure.

Definition 2.1 *Two processes p_0 and q_0 are called* bisimilar, *denoted by $p_0 \leftrightarrow q_0$, if there exists a symmetric relation R between processes such that:*

1. *$R(p_0, q_0)$,*
2. *if $p \xrightarrow{a} p'$ and $R(p,q)$, then there is a transition $q \xrightarrow{a} q'$ such that $R(p',q')$,*
3. *if $p \xrightarrow{a} \surd$ and $R(p,q)$, then $q \xrightarrow{a} \surd$.*

Since the action rules in Table 1 are in *path* format [BV93], it follows that bisimulation equivalence is a *congruence* with respect to all the operators, i.e. if $p \leftrightarrow p'$ and $q \leftrightarrow q'$, then $p+q \leftrightarrow p'+q'$ and $ap \leftrightarrow ap'$ and $a^*p \leftrightarrow a^*p'$.

Table 2 contains an axiom system for $\mathrm{MPA}^*_\delta(A)$, consisting of four axioms from $\mathrm{BPA}_\delta(A)$ together with two axioms for iteration. In the sequel, $p = q$ will mean that this equality can be derived from these axioms. Our axiomatisation for $\mathrm{MPA}^*_\delta(A)$ is *sound* with respect to bisimulation equivalence, i.e. if $p = q$ then $p \leftrightarrow q$. Since bisimulation is a congruence, this can be verified by checking soundness for each axiom separately. In this paper it is proved that the axiomatisation is *complete* with respect to bisimulation, i.e. if $p \leftrightarrow q$ then $p = q$.

A1	$x + y$	$=$	$y + x$
A2	$(x + y) + z$	$=$	$x + (y + z)$
A3	$x + x$	$=$	x
A6	$x + \delta$	$=$	x
MI1	$a \cdot a^*x + x$	$=$	a^*x
MI2	$a^*(a^*x)$	$=$	a^*x

Table 2: Axioms for $\mathrm{MPA}^*_\delta(A)$

3 A Term Rewriting System

We want to define a term rewriting system (TRS) for process terms in that reduces bisimilar terms to the same normal form. However, it is not so easy to construct such a TRS for $\mathrm{MPA}^*_\delta(A)$. Because then the terms $a^*x + x$ and a^*x must reduce to the same normal form, and a rule $a^*x \longrightarrow a^*x + x$ does not terminate, so we need the rules:

$$\begin{array}{rcl} a^*(x + y) + x & \longrightarrow & a^*(x + y) \\ a^*x + x & \longrightarrow & a^*x \end{array}$$

However, these rules are not yet sufficient, because they do not deal with the case $a^*(b^*x) + x \leftrightarrow a^*(b^*x)$. Hence, for this case we must introduce extra rewrite rules. But then, they do not cover the case $a^*(b^*(c^*x)) + x \leftrightarrow a^*(b^*(c^*x))$, etc. So to obtain unique normal forms modulo bisimulation for $\mathrm{MPA}^*_\delta(A)$, apparently we need an infinite number of rewrite rules.

To avoid this complication, we replace iteration by a new, equivalent operator $a^\oplus p$, representing the behaviour of $a \cdot a^*p$. The construct $a^\oplus x$ is called *proper* iteration. (Its standard notation would be a^+x, but we want to avoid ambiguous use of the +.) The operational semantics and the axiomatisation for proper iteration are given in Table 3. They are obtained from the action rules and axioms for $\mathrm{MPA}^*_\delta(A)$, using the obvious equivalence $a^*x \leftrightarrow a^\oplus x + x$. The axiomatisation in Table 3 is complete for $\mathrm{MPA}^\oplus_\delta(A)$ if and only if the axiomatisation in Table 2 is complete for $\mathrm{MPA}^*_\delta(A)$.

3.1 The TRS for $\mathrm{MPA}^\oplus_\delta(A)$

From now on we consider process terms modulo commutativity and associativity of the +.

Table 4 contains a the TRS for $\mathrm{MPA}^\oplus_\delta(A)$. It is easy to see that all rules can be deduced from $\mathrm{MPA}^\oplus_\delta(A)$. In each rule, the term at the left-hand side contains more symbols than the term at the right-hand side. Hence, the TRS is terminating.

$$a^{\oplus}x \xrightarrow{a} a^{\oplus}x + x$$

PMI1	$a(a^{\oplus}x + x)$	$=$	$a^{\oplus}x$
PMI2	$a^{\oplus}(a^{\oplus}x + x)$	$=$	$a^{\oplus}x$

Table 3: Semantics and axioms for proper iteration

1.	$x + x$	$\longrightarrow$	x
2.	$x + \delta$	$\longrightarrow$	x
3.	$a(a^{\oplus}x + x)$	$\longrightarrow$	$a^{\oplus}x$
4.	$a^{\oplus}(a^{\oplus}x + x)$	$\longrightarrow$	$a^{\oplus}x$
5.	$a(a^{\oplus}\delta)$	$\longrightarrow$	$a^{\oplus}\delta$
6.	$a^{\oplus}(a^{\oplus}\delta)$	$\longrightarrow$	$a^{\oplus}\delta$

Table 4: Rewrite rules for $\mathrm{MPA}_{\delta}^{\oplus}(A)$

In order to obtain unique normal forms our TRS must be *weakly confluent*, i.e. if some term has reductions $p'' \longleftarrow p \longrightarrow p'$, then there must exist a term q such that $p'' \twoheadrightarrow q \twoheadleftarrow p'$. Because then Newman's Lemma says that the TRS yields unique normal forms. To obtain the property of weak confluence for our TRS, we have added the extra Rules 5,6. Weak confluence of our TRS can easily be verified by checking this property for all *overlapping redexes*. However, there is no need to do so, because we will not use this property in the proof that bisimilar normal forms are syntactically equal modulo A1,2.

4 Normal Forms Decide Bisimulation Equivalence

In the previous section we have developed a TRS for $\mathrm{MPA}_{\delta}^{\oplus}(A)$ that reduces terms to a normal form. Since all rewrite rules are sound with respect to bisimulation equivalence, it follows that each term is bisimilar with its normal forms. So in order to determine completeness of the axiomatisation for $\mathrm{MPA}_{\delta}^{\oplus}(A)$ with respect to bisimulation, it is sufficient to prove that if two normal forms are bisimilar, then they are provably equal by the axioms A1,2.

Process terms are considered modulo A1,2. From now on, this equivalence is denoted by $p \cong q$, and we say that p and q are of the same form. Each process term p is a sum of terms that are of the form δ or aq or $a^{\oplus}r$; these terms are called the *summands* of p.

Now we present the proof of our main theorem, which is in fact a simplified version of the completeness proof in [FZ93], with some minor extra cases to deal with deadlock. In the proof we apply induction on the following weight function on terms.

$$\begin{array}{rcl} g(\delta) & = & 0 \\ g(p+q) & = & max\{g(p), g(q)\} \\ g(ap) & = & g(p)+1 \\ g(a^{\oplus}p) & = & g(p)+1 \end{array}$$

Theorem 4.1 *If two normal forms p and q are bisimilar, then they are of the same form.*

Proof. We prove the theorem by induction on $g(p)+g(q)$. The case $g(p)+g(q)=0$ is trivial, because then Rule 2 ensures that both p and q must be of the form δ. Now assume that we have already proved the theorem for bisimilar normal forms p and q with $g(p)+g(q)<n$. We prove it for $g(p)+g(q)=n$.

1. Suppose that summands ar of p and as of q are bisimilar. Clearly, $g(r)+g(s)<n$ and $r \leftrightarrow s$. So the induction hypothesis yields $r \cong s$.

2. Next, let summands ar of p and $a^{\oplus}s$ of q be bisimilar, so $r \leftrightarrow a^{\oplus}s+s$. We deduce a contradiction.

   If $s \not\equiv \delta$, then $a^{\oplus}s+s$ is a normal form, because we cannot apply Rules 1 or 2 to it. Moreover, $g(r)+g(a^{\oplus}s+s)<n$, so the induction hypothesis gives $r \cong a^{\oplus}s+s$. But then we can apply Rule 3 to ar, so we have a contradiction.

   And if $s \equiv \delta$, then $r \leftrightarrow a^{\oplus}\delta$, so induction implies $r \cong a^{\oplus}\delta$. But then we can apply Rule 5 to ar, which yields a contradiction again.

3. Finally, assume that summands $a^{\oplus}r$ and $a^{\oplus}s$ are bisimilar. We prove $r \cong s$.

   Clearly $a^{\oplus}r+r \leftrightarrow a^{\oplus}s+s$. If r and s do not contain summands that are bisimilar with $a^{\oplus}s$ and $a^{\oplus}r$ respectively, then $r \leftrightarrow s$, so that induction yields $r \cong s$, and we are done.

   So without loss of generality we may assume that r contains a summand that is bisimilar with $a^{\oplus}s$. According to point 2 this summand cannot be of the form at, so apparently some summand $a^{\oplus}u$ of r is bisimilar with $a^{\oplus}s$. Then induction yields $u \cong s$. According to Rule 1, r can contain only one subterm of the form $a^{\oplus}s$. Its other summands must be bisimilar to summands of s.

   Suppose that s contains a summand bisimilar to $a^{\oplus}r$. Then as before we see that this summand is of the form $a^{\oplus}r$, which indicates that s contains more symbols than r. But on the other hand r has a summand $a^{\oplus}s$, so r contains more symbols than s. This cannot be, so s and $a^{\oplus}r$ can have no behaviour in common.

   And suppose that s contains a summand bisimilar to the summand $a^{\oplus}s$ of r. Then as before it follows that this summand is of the form $a^{\oplus}s$, which indicates that s contains more symbols than itself. Again, this cannot be.

   Hence, the summands of s unequal to δ correspond with the summands of r unequal to $a^{\oplus}s$. We distinguish two possibilities:

- $s \not\equiv \delta$. Then r must be of the form $a^{\oplus}s + r'$, where $r' \underline{\leftrightarrow} s$. Induction implies $r' \cong s$. But then we can apply Rule 4 to $a^{\oplus}r$, so we have a contradiction.
- $s \equiv \delta$. Then $r \cong a^{\oplus}\delta$. But then we can apply Rule 6 to $a^{\oplus}r$, so once more we have a contradiction. □

Corollary 4.2 *The axiomatisation* A1,2,3,6 + MI1,2 *for* $\mathrm{MPA}^*_\delta(A)$ *is complete with respect to bisimulation equivalence.*

Proof. If two terms in $\mathrm{MPA}^{\oplus}_\delta(A)$ are bisimilar, then according to Theorem 4.1 their normal forms are of the same form. Since all the rewrite rules can be deduced from A1,2,3,6 + PMI1,2, it follows that this is a complete axiom system for $\mathrm{MPA}^{\oplus}_\delta(A)$. Then clearly A1,2,3,6 + MI1,2 is a complete axiomatisation for $\mathrm{MPA}^*_\delta(A)$. □

References

[BBP93] J.A. Bergstra, I. Bethke, and A. Ponse. Process algebra with iteration and nesting. Report P9314b, Programming Research Group, University of Amsterdam, 1993. Revised version of Process algebra with iteration, Report P9314.

[BK84] J.A. Bergstra and J.W. Klop. The algebra of recursively defined processes and the algebra of regular processes. In J. Paredaens, editor, *Proceedings* 11^{th} *ICALP,* Antwerp, LNCS 172, pages 82–95. Springer-Verlag, 1984.

[BV93] J.C.M. Baeten and C. Verhoef. A congruence theorem for structured operational semantics with predicates. In E. Best, editor, *Proceedings CONCUR 93,* Hildesheim, LNCS 715, pages 477–492. Springer-Verlag, 1993.

[FZ93] W.J. Fokkink and H. Zantema. Basic process algebra with iteration: completeness of its equational axioms. Report CS-R9368, CWI, Amsterdam, 1993.

[Kle56] S.C. Kleene. Representation of events in nerve nets and finite automata. In *Automata Studies*, pages 3–41. Princeton University Press, 1956.

[Mil80] R. Milner. *A Calculus of Communicating Systems.* LNCS 92. Springer-Verlag, 1980.

[Mil84] R. Milner. A complete inference system for a class of regular behaviours. *Journal of Computer and System Sciences*, 28:439–466, 1984.

[Par81] D.M.R. Park. Concurrency and automata on infinite sequences. In P. Deussen, editor, 5^{th} *GI Conference*, LNCS 104, pages 167–183. Springer-Verlag, 1981.

[Plo81] G.D. Plotkin. A structural approach to operational semantics. Report DAIMI FN-19, Computer Science Department, Aarhus University, 1981.

[Sew93] P. Sewell. Bisimulation is not (first order) equationally axiomatisable. Technical report, Department of Computer Science, Edinburgh University, 1993. Draft.