Theoretical Computer Science 412 (2011) 3035–3044

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

On the axiomatizability of priority II

Luca Aceto [a], Taolue Chen [b], Anna Ingolfsdottir [a], Bas Luttik [c,d,*], Jaco van de Pol [b]

[a] ICE-TCS, School of Computer Science, Reykjavik University, Iceland

[b] Formal Methods and Tools, Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, The Netherlands

[c] Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands

[d] Department of Computer Science, Vrije Universiteit Amsterdam, The Netherlands

ARTICLE INFO

Keywords:
Bisimilarity
Equational logic
Process algebra
Priority operator
Non-finitely based algebras

ABSTRACT

This paper contributes to the study of the equational theory of the priority operator of Baeten, Bergstra and Klop in the setting of the process algebra BCCSP. It is shown that, in the presence of at least two actions, the collection of process equations over BCCSP with the priority operator that are valid modulo bisimilarity, irrespective of the chosen priority order over actions, is not finitely based. This holds true even if one restricts oneself to the collection of valid process equations that do not contain occurrences of process variables.

1. Introduction

The well-known priority operator Θ was introduced by Baeten, Bergstra and Klop in the context of process algebra in [2]. (See, e.g., [7–9] and the references therein for later accounts of the notion of priority in the setting of process description languages.) The priority operator Θ gives certain actions priority over others based on an irreflexive partial ordering relation $>$ over the set of actions. Intuitively, $b > a$ is interpreted as "action b has priority over action a". This means that, in the context of the priority operator Θ, action a is preempted by action b. For example, if p is some process that can initially perform both a and b, then $\Theta(p)$ will initially only be able to execute the action b.

In their classic paper [2], Baeten et al. provided a sound and ground-complete axiomatization for this operator modulo bisimulation equivalence [14,15]. (An axiomatization is ground complete if it can be used to prove all the valid equations relating terms without variables.) Their axiomatization uses predicates on actions (to express priorities between actions) and one extra auxiliary operator. Bergstra showed in the earlier paper [3] that, in case of a finite alphabet of actions, there exists a finite equational axiomatization for Θ, without action predicates and auxiliary operators. So, if the set of actions is finite, neither equations with action predicates as conditions nor auxiliary operators, as used in [2], are actually necessary to obtain a finite axiomatization of bisimulation equivalence over basic process description languages enriched with the priority operator.

A study of the equational theory of the priority operator modulo bisimilarity in the presence of an infinite collection of actions was carried out in [1] in the setting of the process algebra BCCSP, a process description language that contains only basic process algebraic operators from CCS [14] and CSP [11], but is sufficiently powerful to express all finite synchronization trees. In that paper, Aceto, Chen, Fokkink and Ingolfsdottir showed that, in the presence of a non-trivial priority order, there is no finite, purely equational axiomatization for BCCSP enriched with the priority operator. This result even applies if one is allowed to add an arbitrary collection of auxiliary operators to the syntax, and indicates that the use of equations with action predicates as conditions is essential for axiomatizing Θ if the set of actions is infinite. The aforementioned

* Corresponding author at: Department of Mathematics and Computer Science, Eindhoven University of Technology, The Netherlands. Tel.: +31 40 2475142.

E-mail address: s.p.luttik@tue.nl (B. Luttik).

0304-3975/$ – see front matter
doi:10.1016/j.tcs.2011.02.033

reference also exhibits a specific priority order with respect to which bisimilarity affords no finite, sound and ground-complete axiomatization in terms of equations with action predicates as conditions. This result indicates that, in general, the use of auxiliary operators is necessary to axiomatize bisimilarity finitely, even using equations with action predicates as conditions. In some sense, the main theorems offered in [1] may be seen as providing a technical justification for the choices made by Baeten et al. in [2].

All of the aforementioned references investigate the equational theory of the priority operator with respect to a given priority order $>$. This leaves open the natural question of whether the collection of equations that are valid modulo bisimilarity over the language BCCSP enriched with the priority operator *irrespective of the chosen priority order* affords a finite equational axiomatization. The aim of this study is to provide an answer to this question. We shall prove that the collection of equations over our language that are valid modulo bisimilarity irrespective of the chosen priority order affords no finite equational axiomatization in the presence of at least two distinct actions. (Note that the priority operator is only of interest when the collection of actions satisfies this assumption.) Moreover, this negative result holds true even if we restrict ourselves to the collection of valid equations that contain no occurrences of process variables.

Our negative result is perhaps surprising, because one might expect that there are very few non-trivial identities involving the priority operator that hold irrespective of the priority order. To see that this is not the case, however, consider the following infinite family of valid equations ($n \geq 0$)

$$a^n.(b+c) + a^n.b + a^n.c \approx a^n.(b+c) + a^n.b + a^n.c + a^n.\Theta(b+c),$$

where a, b and c are actions with at least b and c distinct, and a^n denotes a sequence of n occurrences of an a-prefixing operator. We exploit this infinite family in the proof of our non-finite axiomatizability result by showing that no finite collection of valid equations can prove all of the identities in the family. The crux of the proof of our main result is a statement to the effect that, when n is "large enough", the property of having a "successor at depth n" whose set of initial actions depends on the chosen priority order is preserved by equational derivations from a finite set of valid equations E. This means that if E proves an equation $p \approx q$ and one of the two terms has this property, then so does the other one. Note that the process term $a^n.\Theta(b+c)$ has a successor at depth n, namely $\Theta(b+c)$, whose set of initial actions depends on the priority order, whereas $a^n.(b+c) + a^n.b + a^n.c$ does not.

The proof of the validity of the equations in the above family uses a case analysis on the possible relation between the actions b and c, with respect to a priority order, at arbitrary depth in the behavior of process terms. As the proof of our main technical result, namely Theorem 22 to follow, shows, this case analysis cannot be implemented equationally by means of a finite collection of valid equations.

The rest of the paper is organized as follows. Section 2 is devoted to preliminary definitions and results. We present the proof of our main result in Section 3. We conclude the paper with a summary of its main results and some directions for future research in Section 4.

2. Preliminaries

Syntax. Let $\mathcal{A}$ be a set of *actions*, and let $\mathcal{V}$ be a countably infinite set of *variables*. The set of *process terms* is generated by the following grammar:

$$t ::= \mathbf{0} \mid \alpha.t \mid t+t \mid \Theta(t) \mid x,$$

with α ranging over $\mathcal{A}$ and x ranging over $\mathcal{V}$. We let t and u (possibly with subscripts or superscripts) range over process terms. A process term is *closed* if it does not contain any variables. We shall typically denote closed process terms by p and q (possibly with subscripts or superscripts). As usual in the literature on process algebras, we sometimes write α instead of $\alpha.\mathbf{0}$.

In this paper we denote the set of natural numbers by $\mathbb{N}$.

Operational semantics. We proceed to define an $\mathcal{A}$-labeled transition relation on closed process terms using structural operational semantics [16]. The operational rule for Θ presupposes a *priority order*, i.e., a strict partial order $>$ on $\mathcal{A}$; $\alpha > \beta$ is to be read as "α-transitions have priority over β-transitions". We say that an action α has *maximal priority* with respect to $>$ if there does not exist an action β such that $\beta > \alpha$. The $\mathcal{A}$-labeled transition relation $\rightarrow_>$ with respect to the priority order $>$ will be defined as the unique supported model, in the sense of [6], of the rules below. As usual, we write $p \xrightarrow{\alpha}_> p'$ if $(p, \alpha, p') \in \rightarrow_>$.

$$1\ \frac{}{\alpha.p \xrightarrow{\alpha}_> p} \qquad 2\ \frac{p \xrightarrow{\alpha}_> p'}{p+q \xrightarrow{\alpha}_> p'} \qquad 3\ \frac{q \xrightarrow{\alpha}_> q'}{p+q \xrightarrow{\alpha}_> q'}$$

$$4\ \frac{p \xrightarrow{\alpha}_> p' \qquad \forall \beta > \alpha.\ p \stackrel{\beta}{\nrightarrow}_>}{\Theta(p) \xrightarrow{\alpha}_> \Theta(p')}$$

It is well known that the transition relation $\rightarrow_>$ is the one defined by structural induction over closed terms using the above rules.

Let p and p' be closed process terms. We denote by $\mathcal{I}_>(p)$ the set of *initial actions* of p with respect to $>$, i.e., $\mathcal{I}_>(p) = \{\alpha \mid \exists p'.\, p \xrightarrow{\alpha}_> p'\}$. (Note that $\mathcal{I}_>(p)$ is finite for each priority order $>$ and closed term p.) We write $p \longrightarrow_> p'$ if $p \xrightarrow{\alpha}_> p'$ for some $\alpha \in \mathcal{A}$, and $p \longrightarrow^k_> p'$ ($k \in \mathbb{N}$) if there exist $p_0, p_1, \ldots, p_k$ such that $p = p_0 \longrightarrow_> p_1 \longrightarrow_> \cdots \longrightarrow_> p_k = p'$. The *depth* $d_>(p)$ of p with respect to $>$ is the largest $k \in \mathbb{N}$ such that $p \longrightarrow^k_> p'$ for some closed process term p'. The following generalization of $\mathcal{I}_>$ will also be used as an auxiliary notion: we denote by $Acts^k_>(p)$ the set of all actions that are enabled at depth k ($k \in \mathbb{N}$) with respect to $>$, and we denote by $Acts^*_>(p)$ the set of all actions that are enabled at some depth with respect to $>$. That is,

$$Acts^k_>(p) = \bigcup\{\mathcal{I}_>(p') \mid p \longrightarrow^k_> p'\}, \text{ and } \quad Acts^*_>(p) = \bigcup_{k\in\mathbb{N}} Acts^k_>(p).$$

Note that the choice of priority order affects the operational semantics associated with closed process terms by *restricting* the transition relation. For example, the closed process term $\Theta(a.\mathbf{0} + b.\mathbf{0})$ affords both a- and b-labeled transitions if $>$ is empty, but only an a-labeled transition if $> = \{(a, b)\}$. A major part of our technical reasoning later on will, however, be based on the *unrestricted* transition relation $\longrightarrow_\emptyset$ induced by the empty priority order (which, intuitively, gives all actions the same priority). For the sake of succinctness, it is therefore convenient to drop the subscript $\emptyset$ for $\longrightarrow_\emptyset$ and its derived notions $\mathcal{I}_\emptyset$, $\longrightarrow^k_\emptyset$ ($k \in \mathbb{N}$) and $d_\emptyset$. That is, we simply write $p \xrightarrow{\alpha} p'$ instead of $p \xrightarrow{\alpha}_\emptyset p'$, $\mathcal{I}(p)$ instead of $\mathcal{I}_\emptyset(p)$, $p \longrightarrow^k p'$ instead of $p \longrightarrow^k_\emptyset p'$, $d(p)$ instead of $d_\emptyset(p)$, $Acts^k(p)$ instead of $Acts^k_\emptyset(p)$, and $Acts^*(p)$ instead of $Acts^*_\emptyset(p)$.

Furthermore, it will be convenient below to also make use of the $\mathcal{A}$-labeled transition relation and its derived notions as induced on the set of *all* process terms by the operational rules above. (The absence of rules for variables in the operational semantics simply implies that variables do not give rise to any transitions.) In particular, the notion of depth $d(t)$ of a term associated with the unrestricted transition relation will play a crucial role in the proof of the main result in the paper. It is easy to see that it satisfies the following equations: $d(\mathbf{0}) = 0$; $d(\alpha.t) = d(t) + 1$; $d(t_1 + t_2) = \max(d(t_1), d(t_2))$; $d(\Theta(t)) = d(t)$; and $d(x) = 0$.

If σ is a *substitution*, i.e., a mapping from variables to process terms, and t is a process term, then by $\sigma(t)$ we denote the process term obtained by replacing all occurrences of variables x by $\sigma(x)$. The term $\sigma(t)$ is called a *substitution instance* of t. A substitution is *closed* if it maps every variable to a closed term. Below, we shall establish a precise correspondence between the operational behavior of a process term and that of a closed substitution instance. Let us first illustrate some subtleties with an example.

Example 1. Let $t = \Theta(a.x + \Theta(y))$, and let σ be some closed substitution. Then $t \xrightarrow{a} \Theta(x)$. Moreover, $\sigma(t) \xrightarrow{a}_> \sigma(\Theta(x))$, provided that $\mathcal{I}_>(\sigma(\Theta(y)))$ does not contain an action with a higher priority than a. For example, if $\sigma(y) = b.\mathbf{0}$, and a and b are incomparable with respect to $>$, then $\sigma(t) \xrightarrow{a}_> \sigma(\Theta(x)))$. On the other hand, $\sigma(t)$ does not afford an a-labeled transition with respect to $>$ if $b > a$. Of course, transitions of $\sigma(x)$ do not give rise to transitions of $\sigma(t)$, because the occurrence of x is guarded by the action prefix $a.$; we say that x is not *enabled* in t. On the other hand, the variable y is enabled in t (there are no action prefixes on the path from the root of t to the occurrence of y). Therefore, (initial) transitions of $\sigma(y)$ may give rise to (initial) transitions of $\sigma(t)$; e.g., if $\sigma(y) \xrightarrow{b}_> p$, then $\sigma(t) \xrightarrow{b}_> \Theta(\Theta(p))$, unless $a > b$ or $\mathcal{I}_>(\sigma(y))$ contains an action with a higher priority than b.

As illustrated in the above example, it may happen that there is a transition $t \xrightarrow{\alpha} t'$, but not a transition $\sigma(t) \xrightarrow{\alpha}_> \sigma(t')$. If α has maximal priority with respect to $>$, however, then we can be sure that $t \xrightarrow{\alpha}_> t'$ implies $\sigma(t) \xrightarrow{\alpha}_> \sigma(t')$.

Lemma 2. *Let t and t' be process terms, let α be an action with maximal priority with respect to $>$. If $t \xrightarrow{\alpha}_> t'$, then $\sigma(t) \xrightarrow{\alpha}_> \sigma(t')$ for all substitutions σ.*

Proof. Note that if α has maximal priority with respect to $>$, then the premise $\forall\beta > \alpha.\, p \xrightarrow{\beta}\!\!\!\!\!\not\;\;_>$ of Rule 4 is vacuously true. Therefore, the lemma can be established with a straightforward induction on a derivation of $t \xrightarrow{\alpha}_> t'$. □

If t is a process term, x is a variable, and σ is a closed substitution, then transitions of $\sigma(x)$ may induce transitions of $\sigma(t)$ only if $\sigma(x)$ is enabled. As can be seen from the example above, in order to determine the target of a transition from $\sigma(t)$ induced by a transition $\sigma(x)$, we need to know how many Θ-operators are on a path from the root of t to an enabled occurrence of x in t. In the following definition, we shall define relations $\lhd_\ell$ ($\ell \in \mathbb{N}$) between variables and process terms such that $x \lhd_\ell t$ if x has an occurrence in t, and on the path from the root of t to this occurrence there are no applications of an action prefix and there are ℓ applications of the Θ-operator.

Definition 3. We define the relations $\lhd_\ell$ ($\ell \in \mathbb{N}$) between variables and process terms as the least relations satisfying, for all variables x, for all natural numbers ℓ, and for all process terms t and u, the following clauses:

(i) $x \lhd_0 x$;
(ii) if $x \lhd_\ell t$, then $x \lhd_\ell t + u$ and $x \lhd_\ell u + t$; and
(iii) if $x \lhd_\ell t$, then $x \lhd_{\ell+1} \Theta(t)$.

If $x \lhd_\ell t$ for some $\ell \in \mathbb{N}$, then we say that x is *enabled* in t.

Using the relations $\lhd_\ell$ ($\ell \in \mathbb{N}$) we shall now establish a formal correspondence between the transitions of a process term and those of its (closed) substitution instances. First, we prove in Lemma 4 that a transition of a substitution instance $\sigma(x)$ of a variable x enabled in t gives rise to a transition of $\sigma(t)$, under a proviso to ensure that it is not preempted by some other transition of $\sigma(t)$. Then, we prove in Lemma 7 that a transition of a closed substitution instance $\sigma(t)$ of a process term t either stems from a transition of t or from a transition of an occurrence of variable x that is enabled in t. For a succinct formulation of the lemmas, it is convenient to have a notation for repeated application of Θ; we define $\Theta^k(t)$ ($k \in \mathbb{N}$) inductively as follows: $\Theta^0(t) = t$, and $\Theta^{k+1}(t) = \Theta(\Theta^k(t))$.

Lemma 4. *Let t be a process term, let x be a variable, and let ℓ be a natural number such that $x \lhd_\ell t$. Furthermore, let σ be a substitution, and suppose that α is an action with maximal priority with respect to $>$. Then $\sigma(x) \xrightarrow{\alpha}_> p$ implies $\sigma(t) \xrightarrow{\alpha}_> \Theta^\ell(p)$.*

Proof. We prove the implication of the lemma by induction on a derivation of $x \lhd_\ell t$ according to the clauses in Definition 3.

If $x \lhd_\ell t$ according to the first clause, then $\ell = 0$ and $t = x$. So $\sigma(x) \xrightarrow{\alpha}_> p$ implies $\sigma(t) \xrightarrow{\alpha}_> \Theta^\ell(p)$.

If the last clause applied in the derivation of $x \lhd_\ell t$ is the second clause, then there exist process terms t_1 and t_2 such that $t = t_1 + t_2$, and either $x \lhd_\ell t_1$ or $x \lhd_\ell t_2$. In the first case, it holds by the induction hypothesis that $\sigma(x) \xrightarrow{\alpha}_> p$ implies $\sigma(t_1) \xrightarrow{\alpha}_> \Theta^\ell(p)$, and hence, by Rule 2, $\sigma(t) \xrightarrow{\alpha}_> \Theta^\ell(p)$. In the second case the proof is analogous, using Rule 3 instead of Rule 2.

If the last clause applied in the derivation of $x \lhd_\ell t$ is the third clause, then $\ell \geq 1$ and there exists t' such that $t = \Theta(t')$ and $x \lhd_{\ell-1} t'$. If $\sigma(x) \xrightarrow{\alpha}_> p$, then, by the induction hypothesis, $\sigma(t') \xrightarrow{\alpha}_> \Theta^{\ell-1}(p)$, so, since α has maximal priority with respect to $>$, by Rule 4 we have that $\sigma(t) = \Theta(\sigma(t')) \xrightarrow{\alpha}_> \Theta^\ell(p)$. □

Corollary 5. *Let t be a process term, let x be a variable, and let ℓ be a natural number such that $x \lhd_\ell t$. Furthermore, let σ be a substitution, and let k be a positive natural number. Then $\sigma(x) \longrightarrow^k p$ implies $\sigma(t) \longrightarrow^k \Theta^\ell(p)$.*

Proof. If $\sigma(x) \longrightarrow^k p$, then $\sigma(x) \longrightarrow p' \longrightarrow^{k-1} p$ for some p'. Recall that $\longrightarrow$ is based on a priority order that assigns the same priority to every action, so by Lemma 4 $\sigma(t) \longrightarrow \Theta^\ell(p')$, and by Rule 4 and induction on $k-1$ it follows that $\Theta^\ell(p') \longrightarrow^{k-1} \Theta^\ell(p)$. Hence, $\sigma(t) \longrightarrow^k \Theta^\ell(p)$. □

Remark 6. The proviso that k be positive is necessary for the validity of the above statement. Indeed, let $t = x + a.\mathbf{0}$, and let σ be the substitution mapping all variables to $\mathbf{0}$. We have that $x \lhd_0 t$ and $\sigma(x) \longrightarrow^0 \mathbf{0}$. On the other hand, $\sigma(t) \longrightarrow^0 \mathbf{0} = \Theta^0(\mathbf{0})$ does *not* hold.

Lemma 7. *Let t be a process term, let σ be a closed substitution, let α be an action, and let p be a closed process term. If $\sigma(t) \xrightarrow{\alpha}_> p$, then either there exists a process term t' such that $t \xrightarrow{\alpha}_> t'$ and $\sigma(t') = p$, or there exist a variable x, a closed process term p', and a natural number $\ell \in \mathbb{N}$ such that $x \lhd_\ell t$, $\sigma(x) \xrightarrow{\alpha}_> p'$, and $\Theta^\ell(p') = p$.*

Proof. The proof is by structural induction on t.

Case 1: If $t = \mathbf{0}$, then $\sigma(t) = \mathbf{0}$, which, according to the operational semantics, does not admit any transitions. So in this case the lemma vacuously holds.

Case 2: Suppose that $t = \beta.t'$ for some action β and some process term t'. If $\sigma(t) \xrightarrow{\alpha}_> p$, then it is clear from the operational semantics that the only transition of $\sigma(t) = \beta.\sigma(t')$ is $\sigma(t) \xrightarrow{\beta}_> \sigma(t')$. It follows that $\alpha = \beta$, so $t \xrightarrow{\alpha}_> t'$, and $p = \sigma(t')$.

Case 3: Suppose that $t = t_1 + t_2$. If $\sigma(t) \xrightarrow{\alpha}_> p$, then by a straightforward reasoning on the basis of the operational semantics, necessarily either $\sigma(t_1) \xrightarrow{\alpha}_> p$ or $\sigma(t_2) \xrightarrow{\alpha}_> p$. We assume without loss of generality that $\sigma(t_1) \xrightarrow{\alpha}_> p$. Then, according to the induction hypothesis, we only need to consider the following two subcases:

Case 3(a): There exists t_1' such that $t_1 \xrightarrow{\alpha}_> t_1'$ and $\sigma(t_1') = p$. From $t_1 \xrightarrow{\alpha}_> t_1'$ it follows by Rule 2 that $t \xrightarrow{\alpha}_> t_1'$, so the proof is complete for this subcase.

Case 3(b): There exist a variable x, a closed process term p', and a natural number ℓ such that $x \lhd_\ell t_1$, $\sigma(x) \longrightarrow_> p'$, and $\Theta^\ell(p') = p$. From $x \lhd_\ell t_1$ it follows, according to Definition 3, that $x \lhd_\ell t$, so the proof is complete for this subcase.

Case 4: Suppose that $t = \Theta(u)$. If $\sigma(t) \xrightarrow{\alpha}_> p$, then, since $\sigma(t) = \Theta(\sigma(u))$, it follows that $\Theta(\sigma(u)) \xrightarrow{\alpha}_> p$, so with a straightforward reasoning on the basis of the operational semantics we can conclude that there exists q such that $\sigma(u) \xrightarrow{\alpha}_> q$ and $\Theta(q) = p$.

By the induction hypothesis we only need to consider the following two subcases:

Case 4(a): There exists a process term u' such that $u \xrightarrow{\alpha}_> u'$ and $\sigma(u') = q$.

Note that, since $t = \Theta(u)$ and $\sigma(t) \xrightarrow{\alpha}_> p$, for all $\beta > \alpha$ it holds that $\sigma(u) \stackrel{\beta}{\not\longrightarrow}_>$. It follows that $u \stackrel{\beta}{\not\longrightarrow}_>$ for all $\beta > \alpha$. (Indeed, if $u \xrightarrow{\beta}_> u''$, for some $\beta > \alpha$ and process term u'', then either $\sigma(u) \xrightarrow{\beta}_> \sigma(u'')$ or $\sigma(u) \xrightarrow{\beta'}_> r$, for some $\beta' > \alpha$ and closed process term r.) Hence, from $u \xrightarrow{\alpha}_> u'$ and Rule 4, we may infer that $t = \Theta(u) \xrightarrow{\alpha}_> \Theta(u')$. We define $t' = \Theta(u')$ and note that $t \xrightarrow{\alpha}_> t'$ and $\sigma(t') = \sigma(\Theta(u')) = \Theta(\sigma(u')) = \Theta(q) = p$. The proof for this subcase is thereby complete.

CASE 4(b): There exist a variable x, a natural number ℓ, and a closed process term p' such that $x \lhd_\ell u, \sigma(x) \xrightarrow{\alpha}_> p'$, and $\Theta^\ell(p') = q$. From $x \lhd_\ell u$ it follows that $x \lhd_{\ell+1} t$, and from $\Theta^\ell(p') = q$ it follows that $\Theta^{\ell+1}(p') = \Theta(q) = p$. The proof for this subcase is thereby complete.

CASE 5: Suppose that $t = x$ for some variable x. If $\sigma(t) \xrightarrow{\alpha}_> p$, then it is immediate that $x \lhd_0 t, \sigma(x) \xrightarrow{\alpha}_> p$, and $\Theta^0(p) = p$. This completes the proof for this case. □

Lemma 8. *Let t be a process term, let σ be a closed substitution, let n be a natural number, and let p be a closed process term. If $\sigma(t) \longrightarrow^n_> p$, then either*

(i) *there exists a process term t' such that $t \longrightarrow^n_> t'$ and $\sigma(t') = p$, or*
(ii) *there exist a process term t', a variable x, a closed process term p', and natural numbers $k < n$ and ℓ such that $t \longrightarrow^k_> t'$, $x \lhd_\ell t'$, $\sigma(x) \longrightarrow^{n-k}_> p'$, and $\Theta^\ell(p') = p$.*

Proof. Suppose that $\sigma(t) \longrightarrow^n_> p$. We proceed by induction on n.

If $n = 0$, then clearly $t \longrightarrow^n_> t$, and from $\sigma(t) \longrightarrow^n_> p$ it follows that $\sigma(t) = p$.

Suppose that $n > 0$. Then there exists a closed process term q such that $\sigma(t) \longrightarrow_> q \longrightarrow^{n-1}_> p$. According to Lemma 7 we now need to consider two cases:

CASE 1: There exists a process term t' such that $t \longrightarrow_> t'$ and $\sigma(t') = q$.

Then, by the induction hypothesis for $q \longrightarrow^{n-1}_> p$, there are two subcases: If there exist a process term t'' such that $t' \longrightarrow^{n-1}_> t''$ and $\sigma(t'') = p$, then to complete the proof in this subcase it suffices to note that $t \longrightarrow^n_> t''$. If there exist a process term t'', a variable x, a closed process term p', and natural numbers $k < n-1$ and ℓ such that $t' \longrightarrow^k_> t''$, $x \lhd_\ell t''$, $\sigma(x) \longrightarrow^{(n-1)-k}_> p'$, and $\Theta^\ell(p') = p$, then to complete the proof in this subcase it suffices to note that $t \longrightarrow^{k+1}_> t''$ and $k+1 < n$.

CASE 2: There exists a variable x, a closed process term q', and a natural number $\ell \in \mathbb{N}$ such that $x \lhd_\ell t, \sigma(x) \longrightarrow_> q'$, and $\Theta^\ell(q') = q$.

Then from $\Theta^\ell(q') = q \longrightarrow^{n-1}_> p$ it can be established with induction on $n-1$, reasoning on the basis of the operational semantics, that there exists p' such that $q' \longrightarrow^{n-1}_> p'$ and $\Theta^\ell(p') = p$. We define $t' = t$ and $k = 0$. Then $t \longrightarrow^k_> t'$, $x \lhd_\ell t'$, $\sigma(x) \longrightarrow^{n-k}_> p'$, and $\Theta^\ell(p') = p$, so the proof in this case is also complete. □

Bisimilarity. Recall that the operational semantics for closed process terms presupposes a specific priority order $>$. A binary symmetric relation $\mathcal{R}$ on closed process terms is a *bisimulation* [14,15] with respect to $>$ if it satisfies for all closed process terms p and q such that $p \mathrel{\mathcal{R}} q$, and for all actions α, the following condition:

if $p \xrightarrow{\alpha}_> p'$, then there exists q' such that $q \xrightarrow{\alpha}_> q'$ and $p' \mathrel{\mathcal{R}} q'$.

Closed process terms p and q are *bisimilar* with respect to $>$ (notation: $p \leftrightarroweq_> q$) if there exists a bisimulation $\mathcal{R}$ with respect to $>$ such that $p \mathrel{\mathcal{R}} q$. We say that p and q are *order-insensitive bisimilar* (notation: $p \leftrightarroweq_* q$) if they are bisimilar with respect to every priority order on $\mathcal{A}$. *In what follows, by bisimilarity we always mean order-insensitive bisimilarity.*

The relations $\leftrightarroweq_>$ and $\leftrightarroweq_*$ are congruences on the set of closed process terms, i.e., they are equivalences and compatible with the syntactic constructions of our language of closed process terms. (Each of the relations $\leftrightarroweq_>$ is a congruence because the operational rules for the operators in our language are in the GSOS format [6], and $\leftrightarroweq_*$ is a congruence because it is the intersection of a family of congruence relations.)

The following proposition recalls some basic facts pertaining to bisimilarity with respect to $>$.

Proposition 9. *Let p and q be closed process terms such that $p \leftrightarroweq_> q$, and let k be a natural number. Then:*

(i) *for every closed process term p' such that $p \longrightarrow^k_> p'$ there exists a closed process term q' such that $q \longrightarrow^k_> q'$ and $p' \leftrightarroweq_> q'$; and*
(ii) *$Acts^k_>(p) = Acts^k_>(q)$, so, in particular, $\mathcal{I}_>(p) = \mathcal{I}_>(q)$.*

Remark 10. Note that Proposition 9(i) fails for $\leftrightarroweq_*$, which is not a bisimulation. As an example, consider the closed process terms $p = a.(b+c)+a.b+a.c$ and $q = p+a.\Theta(b+c)$. It is not hard to see that $p \leftrightarroweq_* q$ (see also the proof of Proposition 12 to follow) and $q \xrightarrow{a}_> \Theta(b+c)$, for any priority order $>$. On the other hand, as our readers can easily check, there is no closed process term p' such that $p \xrightarrow{a}_> p'$ and $p' \leftrightarroweq_* \Theta(b+c)$.

We shall drop the subscript $\emptyset$ from $\leftrightarroweq_\emptyset$. Note that $\leftrightarroweq$ relates closed process terms p and q if they are bisimilar in the usual sense after removing all occurrences of Θ from p and q.

Let p and q be closed process terms. We say that p is a *(semantic) summand* of q (notation: $p \sqsubseteq_* q$) if there exists some r such that $p + r \leftrightarroweq_* q$. The following proposition states some basic properties of $\sqsubseteq_*$ that will be implicitly used in the technical developments to follow.

Proposition 11. *The relation $\sqsubseteq_*$ is a preorder on closed process terms. Moreover, $p' + p'' \sqsubseteq_* p$ implies $p' \sqsubseteq_* p$ and $p'' \sqsubseteq_* p$, for all closed process terms p, p' and p''.*

Table 1
Some valid process equations.

A1	$x + y$	$\approx y + x$
A2	$(x + y) + z$	$\approx x + (y + z)$
A3	$x + x$	$\approx x$
A4	$x + \mathbf{0}$	$\approx x$
PR1	$\Theta(\mathbf{0})$	$\approx \mathbf{0}$
PR2	$\Theta(\Theta(x) + y)$	$\approx \Theta(x + y)$
PR3	$\Theta(\alpha.x)$	$\approx \alpha.\Theta(x)$
PR4	$\Theta(x) + \Theta(y)$	$\approx \Theta(x) + \Theta(y) + \Theta(x + y)$
PR5	$\Theta(\alpha.x + \alpha.y + z)$	$\approx \Theta(\alpha.x + z) + \Theta(\alpha.y + z)$

Equational basis. A *process equation* is a formula $t \approx u$, with t and u process terms; it is said to be *valid* if $\sigma(t) \leftrightarrow_* \sigma(u)$ for every closed substitution σ. If t and u are process terms such that the process equation $t \approx u$ is valid, then we shall also write $t \leftrightarrow_* u$. Table 1 lists some well-known valid process equations. The depth of a process equation $t \approx u$ is the maximum of the depths of the process terms t and u with respect to the empty priority order.

Let E be a set of process equations, and let $t \approx u$ be a process equation; we write $E \vdash t \approx u$ if the process equation $t \approx u$ is derivable from E by means of the rules of equational logic. In this paper we address the question whether the collection of all valid process equations is *finitely based*, i.e., if there exists a *finite* set E of valid process equations such that

$$E \vdash t \approx u \quad \text{if, and only if,} \quad t \leftrightarrow_* u.$$

A collection E of valid process equations that have the above property is often referred to as a *complete axiomatization* of $\leftrightarrow_*$.

3. Order-insensitive bisimilarity is not finitely based

Our order of business in this section is to prove that the collection of all valid process equations over our language is *not* finitely based. Moreover, we shall show that the above-mentioned negative result holds true even if we restrict ourselves to the collection of valid process equations that do not contain occurrences of process variables—that is, those process equations that relate closed process terms. In order to establish our main result, we shall first isolate an infinite family of valid process equations relating closed process terms. Next we shall show that no finite collection of valid process equations is powerful enough to prove all the process equations in our family.

Before presenting our infinite family of valid process equations that cannot all be derivable from some finite set of valid process equations, we first introduce some auxiliary notations and definitions. Let α be an action and let t be a process term; we define $\alpha^n.t$ inductively by $\alpha^0.t = t$ and $\alpha^{n+1}.t = \alpha.(\alpha^n.t)$ for all $n \in \mathbb{N}$. Now, fix concrete elements a, b and c of $\mathcal{A}$, and suppose that $b \neq c$. (It is not necessary to require that a is distinct from b or c.) We define, for $n \in \mathbb{N}$, the process term P_n by

$$P_n = a^n.(b + c) + a^n.b + a^n.c.$$

Proposition 12. *For every $n \in \mathbb{N}$, the process equation $P_n + a^n.\Theta(b + c) \approx P_n$ is valid.*

Proof. It suffices to show that $a^n.\Theta(b + c) \sqsubseteq_* P_n$, for each n. To this end, let $>$ be a priority order, and let n be a natural number. Observe that

- if $b > c$ then $a^n.\Theta(b + c) \leftrightarrow_> a^n.b$,
- if $c > b$ then $a^n.\Theta(b + c) \leftrightarrow_> a^n.c$, and
- if b and c are incomparable with respect to $>$ then $a^n.\Theta(b + c) \leftrightarrow_> a^n.(b + c)$.

Hence $a^n.\Theta(b + c) + P_n \leftrightarrow_> P_n$ for each priority order $>$ and natural number n. It follows that $a^n.\Theta(b + c) \sqsubseteq_* P_n$, for each n. $\square$

We shall now proceed to prove that there cannot exist a finite set of valid process equations from which the process equations in the family

$$\{P_n + a^n.\Theta(b + c) \approx P_n \mid n \in \mathbb{N}\}$$

of valid process equations are all derivable. Our readers might have noticed that the proof of the validity of the equations in the above family uses a case analysis on the possible relation between the actions b and c, with respect to a priority order, at an arbitrary depth in the behavior of process terms. As the proof of our main technical result, namely Theorem 22 to follow, shows, this case analysis cannot be implemented equationally by means of a finite collection of valid equations.

To formalize the above-mentioned case analysis, we now introduce the notion of Θ-dependent closed process term, which will play a crucial role in subsequent developments.

Definition 13. A closed process term p is *Θ-dependent* if there exist priority orders $>_1$ and $>_2$ such that $\mathcal{I}_{>_1}(p) \neq \mathcal{I}_{>_2}(p)$.

Intuitively, a closed process term is Θ-dependent if its set of initial actions depends on the chosen priority order. For example, $\Theta(b+c)$ is Θ-dependent, whereas $\Theta(b)$ and $\Theta(c)$ are not.

Lemma 14. *If $p \leftrightarrow_* q$ and p is Θ-dependent, then so is q.*

Proof. If p is Θ-dependent, then there exist priority orders $>_1$ and $>_2$ such that $\mathcal{I}_{>_1}(p) \neq \mathcal{I}_{>_2}(p)$. Since $p \leftrightarrow_* q$, it follows that $\mathcal{I}_{>}(p) = \mathcal{I}_{>}(q)$ for all priority orders $>$. So $\mathcal{I}_{>_1}(q) \neq \mathcal{I}_{>_2}(q)$, and hence q is Θ-dependent. $\square$

Let p and p' be closed process terms, and let n be a natural number. If $p \longrightarrow^n p'$, then we call p' an *n-successor* of p. (Note that the notion of n-successor is based on the empty priority order, which, intuitively, assigns to every action the same priority.)

Note that the three n-successors of P_n are Θ-independent, whereas $P_n + a^n.\Theta(b+c)$ has a Θ-dependent n-successor $\Theta(b+c)$. Hence, to prove that there does not exist a finite set of valid process equations from which all the process equations

$$P_n + a^n.\Theta(b+c) \approx P_n \quad (n \in \mathbb{N})$$

are derivable, it suffices to show that the property of having a Θ-dependent n-successor is preserved by equational derivations from a finite set of valid equations E when n is "large enough". To this end, we presuppose a finite set of valid process equations E, of depth less than n, and establish that the following property holds for all closed process terms p and q such that $E \vdash p \approx q$ and $p, q \sqsubseteq_* P_n$:

if p has a Θ-dependent n-successor, then q has a Θ-dependent n-successor too. (†)

The remainder of this section and of the paper will be devoted to a proof of this property, in case $\mathcal{A}$ contains at least two distinct actions.

An important property of the terms P_n is that, intuitively, the only moments of choice in their induced behaviors occur after 0 and after n steps. We formalize this property and establish that it is invariant under $\leftrightarrow_*$.

Definition 15. Let p be a closed process term. We say that p is *determinate* if $|\mathcal{I}(p)| \leq 1$, and for all closed terms p_1 and p_2 such that $p \longrightarrow p_1$ and $p \longrightarrow p_2$ it holds that $p_1 \leftrightarrow_* p_2$. We say that p is *determinate at depth k* if all closed process terms p' such that $p \longrightarrow^k p'$ are determinate.

Remark 16. In [12], Huynh and Tian call a process locally unary if all of its successors can initially perform at most one action.

Note that P_n is determinate at depth k for all $1 \leq k < n$. We shall establish that every semantic summand of P_n is also determinate at all depths $1 \leq k < n$.

Lemma 17. *Let p be a closed process term and let $p \sqsubseteq_* P_n$ for some $n \in \mathbb{N}$. Then p is determinate at depth k for all $1 \leq k < n$.*

Proof. First we prove that $Acts^k(p) \subseteq \{a\}$ for all $0 \leq k < n$. From $p \sqsubseteq_* P_n$ it follows that there exists a closed process term r such that $p + r \leftrightarrow_* P_n$, and hence $p + r \leftrightarrow P_n$. By Proposition 9, $Acts^k(p+r) = Acts^k(P_n) = \{a\}$, and since $Acts^k(p) \subseteq Acts^k(p+r)$, it follows that $Acts^k(p) \subseteq \{a\}$.

Now, to prove that p is determinate at depth k for all $1 \leq k < n$, suppose that $1 \leq k < n$ is the least natural number such that p is *not* determinate at depth k; we derive a contradiction. That p is not determinate at depth k means that there exist closed process terms p', p_1 and p_2, and a priority order $>$ such that $p \longrightarrow^k p'$, $p' \longrightarrow p_1$, $p' \longrightarrow p_2$, and $p_1 \not\leftrightarrow_> p_2$. Since $|Acts^i(p)| = 1$ for all $0 \leq i < k$, from $p \longrightarrow^k p'$ it follows that $p \longrightarrow^k_> p'$. Hence, since $p + r \leftrightarrow_> P_n$, by Proposition 9 there exists P_n' such that $P_n \longrightarrow^k_> P_n'$ and $p' \leftrightarrow_> P_n'$. From $1 \leq k < n$ and the definition of P_n it is clear that there is a unique closed process term P_n'' such that $P_n' \longrightarrow P_n''$. It follows that $p_1 \leftrightarrow_> P_n'' \leftrightarrow_> p_2$, so $p_1 \leftrightarrow_> p_2$, contradicting an immediate consequence of the assumption that p' is not determinate. We conclude that p' is determinate. $\square$

The following lemma roughly states that $\leftrightarrow_*$ does behave like a bisimulation over determinate processes. (Compare with Remark 10.)

Lemma 18. *Let p and q be closed process terms such that $p \leftrightarrow_* q$, and let m be a natural number such that p and q are determinate at all depths $< m$. If there exists p' such that $p \longrightarrow^m p'$, then there exists q' such that $q \longrightarrow^m q'$ and $p' \leftrightarrow_* q'$.*

Proof. The proof is by induction on m.

Suppose $m = 0$. If p' is a closed process term such that $p \longrightarrow^m p'$, then $p' = p$. So we can take $q' = q$ to obtain that $q \longrightarrow^m q'$ and $p' = p \leftrightarrow_* q = q'$.

Suppose $m > 0$. If p' is a closed process term such that $p \longrightarrow^m p'$, then there exists a closed process term p_1 such that $p \longrightarrow p_1 \longrightarrow^{m-1} p'$. Since $p \leftrightarrow_* q$, in particular $p \leftrightarrow q$, so there exists q_1 such that $q \longrightarrow q_1$ and $p_1 \leftrightarrow q_1$.

It remains to argue that $p_1 \leftrightarrow_* q_1$, for then, since p_1 and p_2 are determinate at all depths less than $m-1$, there exists, by the induction hypothesis, a closed process term q' such that $q_1 \longrightarrow^{m-1} q'$ and $p' \leftrightarrow_* q'$. So let us suppose that $p_1 \not\leftrightarrow_* q_1$ and derive a contradiction. If $p_1 \not\leftrightarrow_* q_1$, then there exists $>$ such that $p_1 \not\leftrightarrow_> q_1$. On the other hand, since $p \leftrightarrow_* q$, there exists q_1' such that $q \longrightarrow_> q_1'$ and $p_1 \leftrightarrow_> q_1'$. Clearly, $q \longrightarrow q_1'$, and hence, since q is determinate, $q_1 \leftrightarrow_* q_1'$. We find that $p_1 \leftrightarrow_> q_1' \leftrightarrow_> q_1$, which contradicts our assumption that $>$ is such that $p_1 \not\leftrightarrow_> q_1$. We conclude that $p_1 \leftrightarrow_* q_1$. $\square$

3.1. Proof of the main result

We are now ready to prove that, if E is a finite set of valid process equations of depth less than n, then property (†) holds. Our proof of this claim is based on establishing that the property in question is preserved by the inference rules of equational logic.

An important part of our proof consists in establishing some properties of valid process equations $t \approx u$. In particular, the following lemma plays a crucial role in showing that property (†) holds for closed instantiations of valid process equations whose depth is less than n.

Lemma 19. *Assume that $\mathcal{A}$ contains at least two actions, and let a be an action in $\mathcal{A}$. Let t and u be process terms such that $Acts^*(t) \subseteq \{a\}$, let x be a variable, and suppose that $t \leftrightarroweq_* u$. If there exists t' such that $t \longrightarrow^k t'$ and, for some $\ell \in \mathbb{N}$, $x \lhd_\ell t'$, then there exists u' such that $u \longrightarrow^k u'$ and, for some $m \in \mathbb{N}$, $x \lhd_m u'$. Moreover, $\ell = 0$ if, and only if, $m = 0$.*

Proof. Suppose that $t \leftrightarroweq_* u$, and let t' be a process term such that $t \longrightarrow^k t'$ and, for some $\ell \in \mathbb{N}$, $x \lhd_\ell t'$. Let n be larger than the depths of t and u, and let b be an action in $\mathcal{A}$ that is distinct from a. We let $> \; = \{(a, b)\}$, and define the closed substitution ϱ as follows:

$$\varrho(y) = \begin{cases} a^n.(a.\mathbf{0} + b.\mathbf{0}) & \text{if } y = x; \text{ and} \\ \mathbf{0} & \text{otherwise.} \end{cases}$$

Suppose that $t \longrightarrow^k t'$. Then, since $Acts^*(t) \subseteq \{a\}$, there exist $t_0, \ldots, t_k$ such that

$$t = t_0 \xrightarrow{a} \cdots \xrightarrow{a} t_k = t'.$$

Hence, since a has maximal priority with respect to $>$, by Lemma 2 and induction on k it follows that

$$\varrho(t) = \varrho(t_0) \xrightarrow{a}_> \cdots \xrightarrow{a}_> \varrho(t_k) = \varrho(t'),$$

so $\varrho(t) \longrightarrow^k_> \varrho(t')$. Since $x \lhd_\ell t'$ and a has maximal priority with respect to $>$, by Lemma 4 and the definition of ϱ we have that $\varrho(t') \longrightarrow^n_> \Theta^\ell(a.\mathbf{0} + b.\mathbf{0})$. So $\varrho(t) \longrightarrow^{k+n}_> \Theta^\ell(a.\mathbf{0} + b.\mathbf{0})$.

Now, since $t \leftrightarroweq_* u$, and hence in particular $\varrho(t) \leftrightarroweq_> \varrho(u)$, it follows that there exists a closed process term p such that $\varrho(u) \longrightarrow^{k+n}_> p$ and $\Theta^\ell(a.\mathbf{0} + b.\mathbf{0}) \leftrightarroweq_> p$. Since n is larger than the depth of u, by Lemma 8 there exist a process term u', a variable y, a closed process term p', and natural numbers $h < n$ and m such that $u \longrightarrow^h_> u'$, $y \lhd_m u'$, $\varrho(y) \longrightarrow^{n+k-h}_> p'$ and $\Theta^m(p') = p$. From $h < n$ it follows that $n + k - h > 0$, and therefore, since $\varrho(y) \longrightarrow^{n+k-h}_> p'$, it is clear from the definition of ϱ and $\Theta^\ell(a.\mathbf{0} + b.\mathbf{0}) \leftrightarroweq_> \Theta^m(p'') = p$ that $y = x$, $h = k$, and $p' = a.\mathbf{0} + b.\mathbf{0}$. Moreover, from $\Theta^\ell(a.\mathbf{0} + b.\mathbf{0}) \leftrightarroweq_> p = \Theta^m(p') = \Theta^m(a.\mathbf{0} + b.\mathbf{0})$ and the definition of $>$, it follows that $\ell = 0$ if, and only if, $m = 0$. □

Remark 20. The first part of the lemma can also be established if $\mathcal{A}$ is a singleton set, using a different substitution in its proof, but then second part of the lemma does not hold. In fact, if $\mathcal{A}$ is a singleton set, then the equation $\Theta(x) \approx x$ is valid.

The following lemma establishes that closed substitution instances of valid process equations with a depth less than n preserve the property (†).

Lemma 21. *Let t and u be process terms such that $t \leftrightarroweq_* u$, and let σ be a closed substitution. Suppose that $Acts^*(t) \subseteq \{a\}$ and $d(t), d(u) < n$. If $\sigma(t)$ has a Θ-dependent n-successor, then $\sigma(u)$ also has a Θ-dependent n-successor.*

Proof. Suppose that p is a Θ-dependent term such that $\sigma(t) \longrightarrow^n p$. Then, since $d(t) < n$, there exist, by Lemma 8, a process term t', a variable x, a closed process term p', and natural numbers $k < n$ and ℓ such that $t \longrightarrow^k t'$, $x \lhd_\ell t'$, $\sigma(x) \longrightarrow^{n-k} p'$, and $\Theta^\ell(p') = p$. Observe that, as p is Θ-dependent, the set of actions cannot be a singleton. Therefore, since $Acts^*(t) \subseteq \{a\}$, by Lemma 19 there exist u' and a natural number m such that $u \longrightarrow^k u'$ and $x \lhd_m u'$, so by Corollary 5 $\sigma(u') \longrightarrow^{n-k} \Theta^m(p')$, and hence $\sigma(u) \longrightarrow^n \Theta^m(p')$. Moreover, according to Lemma 19, $\ell = 0$ if, and only if, $m = 0$. Hence, since $\Theta^\ell(p')$ is Θ-dependent, it follows that $\Theta^m(p')$ is Θ-dependent too. □

Theorem 22. *Assume that $\mathcal{A}$ contains at least two actions. Let E be a set of valid process equations of depth less than n, and let p and q be closed process terms such that $p, q \sqsubseteq_* P_n$ and $E \vdash p \approx q$. If p has a Θ-dependent n-successor, then q has a Θ-dependent n-successor too.*

Proof. Without loss of generality, we may assume that E is symmetric in the sense that if the process equation $t \approx u$ is in E, then so is the process equation $u \approx t$. If E satisfies this assumption, then there is a derivation from E of the process equation $p \approx q$ if, and only if, there is a derivation of $p \approx q$ from E without applications of the symmetry rule. This effectively means that we can disregard the symmetry rule in our inductive proof below.

It is well-known that we may, furthermore, assume without loss of generality that all applications of the substitution rule in derivations have a process equation from E as premise. This effectively means that we need not consider the axiom rule – which states that all process equations in E are derivable –, and the substitution rule – which states that if a process equation is derivable, then so are all its substitution instances – separately in our inductive proof below, but instead can consider a new rule stating that all substitution instances of process equations in E are derivable.

We now proceed by induction on a derivation of $p \approx q$ satisfying the above assumptions; we distinguish cases according to the last rule applied in the considered derivation of $p \approx q$. If the last rule is either reflexivity or transitivity, then the claim is immediate or is a direct consequence of the induction hypothesis. Suppose that $p \approx q$ is a closed substitution instance of a process equation $t \approx u$ in E. Then there exists a closed substitution σ such that $p = \sigma(t)$ and $q = \sigma(u)$. From $p \sqsubseteq_* P_n$ it follows that $Acts^*(t) \subseteq \{a\}$, and since $t \approx u$ is in E, it holds that $t \leftrightarrow_* u$ and $d(t), d(u) < n$. Hence, by Lemma 21, if p has a Θ-dependent n-successor, then q has a Θ-dependent n-successor too. It remains to consider the cases in which the rule applied in the derivation is one of the three congruence rules.

Case 1: Suppose that the rule applied last in the considered derivation of $p \approx q$ is the congruence rule for $+$. Then there exist closed process terms p_1, p_2, q_1 and q_2 such that $p = p_1 + p_2$ and $q = q_1 + q_2$, and $p_1 \approx q_1$ and $p_2 \approx q_2$ are the premises of the last rule application in the derivation of $p \approx q$. If p has a Θ-dependent n-successor, then, as $n > 0$, it must be an n-successor of p_1 or of p_2. Assume, without loss of generality, that it is an n-successor of p_1. Then, since clearly $p_1, q_1 \sqsubseteq_* P_n$, we may apply the induction hypothesis to conclude that q_1 has a Θ-dependent n-successor, and hence q has a Θ-dependent n-successor.

Case 2: Suppose that the rule applied last in the considered derivation of $p \approx q$ is the congruence rule for $\alpha.$ ($\alpha \in \mathcal{A}$). Then there exist p' and q' such that $p = \alpha.p'$, $q = \alpha.q'$, and $p' \approx q'$.

Note that, since $p, q \sqsubseteq_* P_n$, by Lemma 17 p' and q' are determinate at all depths $0, \ldots, n-2$. Moreover, since p has a Θ-dependent n-successor, there exists a Θ-dependent closed process term p'' such that $p' \longrightarrow^{n-1} p''$. Hence, since $p' \leftrightarrow_* q'$, there exists by Lemma 18 a closed process term q'' such that $q' \longrightarrow^{n-1} q''$ and $p'' \leftrightarrow_* q''$. Since p'' is Θ-dependent, by Lemma 14 q'' is Θ-dependent too.

Case 3: Suppose that the rule applied last in the considered derivation of $p \approx q$ is the congruence rule for Θ. Then there exist p' and q' such that $p = \Theta(p')$, $q = \Theta(q')$, and $p' \approx q'$.

If p has a Θ-dependent n-successor, then, since $p = \Theta(p')$, p' has an n-successor p'', such that $|\mathcal{I}(p'')| \geq 2$. Hence, since $p' \leftrightarrow_* q'$, and consequently $p' \leftrightarrow q'$, there exists a closed process term q'' such that $q' \longrightarrow^n q''$ and $p'' \leftrightarrow q''$. From $p'' \leftrightarrow q''$ and $|\mathcal{I}(p'')| \geq 2$ it follows that $|\mathcal{I}(q'')| \geq 2$. Clearly, $q = \Theta(q') \longrightarrow^n \Theta(q'')$. Moreover, $|\mathcal{I}(q'')| \geq 2$ implies that $\Theta(q'')$ is Θ-dependent, for if $>$ is a priority order that relates two elements of $\mathcal{I}(q'')$, then $\mathcal{I}(q'') \neq \mathcal{I}_>(q'')$. □

Corollary 23. *If $\mathcal{A}$ contains at least two actions, then there does not exist a finite set of valid process equations E such that all valid process equations are derivable from E.*

Proof. Suppose that E is a finite set of valid process equations. Then there exists $n \in \mathbb{N}$ such that all process equations in E have a depth $< n$. Hence, by Theorem 22, for all p and q such that $p, q \sqsubseteq_* P_n$ it holds that if $E \vdash p \approx q$ and p has a Θ-dependent n-successor, then q has a Θ-dependent n-successor too. Since $P_n + \alpha^n.\Theta(b + c)$ has a Θ-dependent n-successor, but P_n does not, it follows that the valid process equation $P_n + \alpha^n.\Theta(b + c) \approx P_n$ is not derivable from E. □

4. Conclusions

In this study, we have shown that, rather surprisingly, the collection of equations that hold modulo bisimilarity over the language BCCSP enriched with the priority operator irrespective of the chosen priority order is not finitely based. Moreover, this holds true even if we restrict ourselves to the collection of valid ground equations—that is, those equations that do not contain occurrences of variables. As the proof of our main result indicates, the collection of valid (ground) equations does not even afford a complete axiomatization of bounded depth. These results provide further evidence of the weakness of equational logic in axiomatizing the priority operator; see [1] for earlier negative results of this kind and further references.

Since order-insensitive bisimilarity is clearly decidable on the closed process terms over BCCSP, the algebra of closed process terms modulo order-insensitive bisimilarity is computable. Hence, by Theorem 1.1 in the seminal article [5] by Bergstra and Tucker, adding a finite number of auxiliary operations suffices to facilitate a finite ground-complete axiomatization. In their original paper introducing the priority operator [2], Baeten et al. offered a finite, equational, ground-complete axiomatization of bisimilarity using a binary auxiliary operator, the so-called *unless operator*. It is an interesting topic for further research to study whether the unless operator, or some other finite collection of auxiliary operators, can be used to provide a finite (ground-)complete axiomatization of order-insensitive bisimilarity over BCCSP with priority.

We note that Baeten et al. in [2] consider a process theory with action constants and sequential composition instead of action prefixes and a single constant **0**. As Stefan Blom pointed out to us, in the presence of sequential composition the content of the infinite family of equations considered in Section 3 can be expressed with a single equation

$$x \cdot (b + c) + x \cdot b + x \cdot c + x \cdot \Theta(b + c) \approx x \cdot (b + c) + x \cdot b + x \cdot c.$$

Thus, the question arises whether order-insensitive bisimilarity is finitely axiomatizable in the setting of BPA [4] with the priority operator. We believe that also this question can be answered negatively by slightly adapting the infinite family of equations considered in Section 3. First, we define, for an arbitrary process term p, the process terms $A_n(p)$ inductively by: $A_0(p) = p$, and $A_{i+1}(p) = a \cdot A_i(p) + a$. Note that $A_n(p)$ is $a^n \cdot p$ with an additional a-summand at every depth. Due to the extra a-summands, $A_n(p)$ cannot be written as a sequential composition. We conjecture that there cannot exist a finite set of valid BPA_Θ process equations from which all equations in the family

$$\{A_n(b+c) + A_n(b) + A_n(c) + A_n(\Theta(b+c)) \approx A_n(b+c) + A_n(b) + A_n(c) \mid n \in \mathbb{N}\}$$

can be derived, and that the proof is along the same lines as the proof of our main result in Section 3.

Another interesting question for further study is to find an infinite, but finitely described, basis of equations for order-insensitive bisimilarity over BCCSP enriched with the priority operator. In particular, it is natural to wonder what infinite families of equations should be added to the collection of axioms in Table 1 in order to obtain a (ground-)complete axiomatization of order-insensitive bisimilarity.

A classic question in the field of equational logic is the study of the decidability properties of equational theories, and the characterization of the computational complexity of decidable theories. (See, e.g., [13] for an encyclopedic survey in the mathematical literature.) It is an interesting question to determine whether the equational theory of order-insensitive bisimilarity is decidable over the language considered in this paper and, if so, to find out what its structural complexity is.

In this paper, we have focused on studying the equational theory of order-insensitive bisimilarity. To the best of our knowledge, much less is known about the axiomatizability properties of other behavioral semantics in van Glabbeek's spectrum [10] over BCCSP enriched with the priority operator. This is yet another avenue for further research, which requires a study of the congruence properties of the "order-insensitive versions" of the standard behavioral semantics in the spectrum.

Acknowledgements

The work of Aceto and Ingolfsdottir has been partially supported by the projects "New Developments in Operational Semantics" (nr. 080039021) and "Meta-theory of Algebraic Process Theories" (nr. 100014021) of the Icelandic Research Fund.

References

[1] L. Aceto, T. Chen, W. Fokkink, A. Ingolfsdottir, On the axiomatisability of priority, Mathematical Structures in Computer Science 18 (1) (2008) 5–28.
[2] J.C.M. Baeten, J.A. Bergstra, J.W. Klop, Syntax and defining equations for an interrupt mechanism in process algebra, Fundamenta Informaticae IX (2) (1986) 127–168.
[3] J.A. Bergstra, Put and get, primitives for synchronous unreliable message passing, Logic Group Preprint Series Nr. 3, CIF, State University of Utrecht, 1985.
[4] J.A. Bergstra, J.W. Klop, Process algebra for synchronous communication, Information and Control 60 (1–3) (1984) 109–137.
[5] J.A. Bergstra, J.V. Tucker, Equational specifications, complete term rewriting systems, and computable and semicomputable algebras, J. ACM 42 (6) (1995) 1194–1230.
[6] B. Bloom, S. Istrail, A.R. Meyer, Bisimulation can't be traced, J. ACM 42 (1) (1995) 232–268.
[7] J. Camilleri, G. Winskel, CCS with priority choice, Information and Computation 116 (1) (1995) 26–37.
[8] R. Cleaveland, M. Hennessy, Priorities in process algebras, Information and Computation 87 (1–2) (1990) 58–77.
[9] R. Cleaveland, G. Lüttgen, V. Natarajan, Priorities in process algebra, in: J. Bergstra, A. Ponse, S. Smolka (Eds.), Handbook of Process Algebra, Elsevier, 2001, pp. 711–765.
[10] R.J. van Glabbeek, The linear time-branching time spectrum. I. The semantics of concrete, sequential processes, in: Handbook of Process Algebra, Elsevier, Amsterdam, 2001, pp. 3–99.
[11] C.A.R. Hoare, Communicating Sequential Processes, Prentice-Hall International, Englewood Cliffs, 1985.
[12] D.T. Huynh, L. Tian, On deciding readiness and failure equivalences for processes, Information and Computation 117 (2) (1995) 193–205.
[13] O.G. Kharlampovich, M.V. Sapir, Algorithmic problems in varieties, Internat. J. Algebra Comput. 5 (4–5) (1995) 379–602.
[14] R. Milner, Communication and Concurrency, Prentice-Hall International, Englewood Cliffs, 1989.
[15] D. Park, Concurrency and automata on infinite sequences, in: P. Deussen (Ed.), 5th GI Conference, Karlsruhe, Germany, in: Lecture Notes in Computer Science, vol. 104, Springer-Verlag, 1981, pp. 167–183.
[16] G.D. Plotkin, A structural approach to operational semantics, J. Logic Algebraic Program. 60–61 (2004) 17–139.