Algebraic representation of bisimulation for the tagh-format

Citation for published version (APA):

Document status and date:
Published: 01/01/2002

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Algebraic representation of bisimulation for the tagh-format

A. Sokolova\textsuperscript{1,2} & E.P. de Vink\textsuperscript{1,3}

\textsuperscript{1} Department of Mathematics and Computer Science, Eindhoven University of Technology
\textsuperscript{2} Institute of Informatics, Skopje
\textsuperscript{3} LIACS, Leiden University
a.sokolova@tue.nl, evink@win.tue.nl

Abstract

Process behavior is often described in terms of structural operational semantics. For a wide class of such rules, viz. those adhering to the GSOS format or to the tagh-format, it is possible to obtain automatically a finite axiomatization that characterizes strong bisimulation. Here we go a bit further and show for this format how to obtain canonical representatives of bisimulation equivalence classes under a condition of well-foundedness. This we do by constructing the initial algebra in the variety of semi-lattices over the axiomatization of a system.

1 Introduction

Structured operational semantics has become the predominant method of describing the operational semantics of process languages in terms of transition systems. Starting from the founding work of Bloom, Istrael and Meyer on GSOS in [9] and important contributions of De Simone [12] and Groote and Vaandrager [15], various other formats have been studied each with their respective merits, including the property of bisimulation being a congruence. Here, we mention: ntyft/ntyxt [14], panth [26], tree-rules [13], the work in [25], and the tagh-format of [5].

In [1], Aceto, Bloom and Vaandrager provide a method of generating a sound and complete axiomatization of strong bisimulation for transition system specification in the GSOS-format. Recently, in [5] this result has been slightly generalized in the setting of the tagh-format in order to deal with explicit termination. In this paper we focus on the question of deciding the bisimilarity of terms with respect to some given transition system.
specification. In general, this problem is known to be unsolvable (cf. [9]). However, under certain linearity and well-foundedness conditions on the transition rules a decision procedure can be given. More specifically, we address the following problem: Given a transition system specification over a signature Σ and rules confirming to the GSOS or tagh format, determine whether two ground terms in \( T(Σ) \) are bisimilar without constructing a bisimulation itself.

For rules in GSOS-format and tagh-format the resulting transition system is finitely branching and computable. Moreover, as shown in [1, 5], a head normalization property allows one to trade an arbitrary operation in favour of the basic operations of nondeterministic choice and action prefixing. Together with well-foundedness of the underlying transition relation this induces an elimination theorem, stating that any term can be reduced to an equivalent basic term in a basic process algebra.

In [18] a description of the free algebra for a basic process language was presented. Here we adapt their result by dealing with action prefixing instead of sequential composition and incorporating the dagger-construction of [1] and explicit termination as in [4]. In a similar manner as in [18] we provide a technique for the construction of the initial algebra. The construction takes place in the variety \( \mathcal{V}_{t,ɛ} \) defined by \( E_{t,ɛ} \), the equational theory for bisimulation for the basic process language. The free \( \mathcal{V}_{t,ɛ} \)-algebra is given by set-theoretic means as a universe set with suitably defined operations, and an epimorphism is defined that maps each term of the signature of \( \mathcal{V}_{t,ɛ} \) to its canonical unique representative in the free \( \mathcal{V}_{t,ɛ} \)-algebra. Combining this with the results of [1] and [5] we arrive at a procedure for determining whether any two terms with operations defined by a well-founded transition system specification in GSOS-format or tagh format are bisimilar or not, simply comparing their image in the initial algebra.

2 Preliminaries

2.1 Tagh-systems

We assume acquaintance with an SOS system as a pair \( S_S = (Σ_S, R_S) \) of a finite signature \( Σ_S \) and a finite set \( R_S \) of rules. For further details see for instance [23, 2, 7].

The SOS system that we will use as basic in this paper is the system \( S_{t,ɛ} = (Σ_{t,ɛ}, R_{t,ɛ}) \). Here, the signature \( Σ_{t,ɛ} \) is given by \( Σ_{t,ɛ} = \{+, δ, ɛ\} \cup \{a(.) \mid a \in Act\} \cup \{↑B(.) \mid B \subseteq Act\} \), where + is a binary operation, δ and ɛ are constants, there is one unary operation for each action \( a \in Act \), and
also one unary operation $\uparrow_B$ for each subset of $\text{Act}$. We use the notation $ax$ or $a.x$ for $a(x)$ and $x \uparrow B$ for $\uparrow_B(x)$. The set of rules $R_{\uparrow,\epsilon}$ consists of the following.

$$
ax \xrightarrow{a} x 
$$

(1)

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

(2)

$$
\frac{x \xrightarrow{a} x'}{x \uparrow B \xrightarrow{a} x'} (a \notin B)
$$

(3)

$$
\epsilon \downarrow, \quad \frac{x \downarrow}{(x + y) \downarrow}, \quad \frac{y \downarrow}{(x + y) \downarrow}
$$

(4)

$$
\frac{x \downarrow}{x \uparrow B \downarrow}
$$

(5)

The above is in fact the system for the basic process language from [5] with nondeterministic choice $+$, a prefixing operation for every action $a$ in the set of actions $\text{Act}$, a unary one-step restriction, or initial-blocking, operation $\uparrow_B$ for every subset of actions, an explicit termination constant $\epsilon$, and a deadlock constant $\delta$ which has no transitions (cf. [4]). The symbol $\downarrow$ is used for postfix notation of the termination predicate, meaning that a term can terminate immediately.

**Definition 2.1** Let $S_S = (\Sigma_S, R_S)$ be an SOS system with termination. A binary relation $R$ on $T(\Sigma_S)$ is a bisimulation if for all $x, y \in T(\Sigma_S)$ such that $R(x, y)$ the following conditions are met

1. if $x \xrightarrow{a} x'$ then there exists $y'$ such that $y \xrightarrow{a} y'$ and $R(x', y')$;
2. if $y \xrightarrow{a} y'$ then there exists $x'$ such that $x \xrightarrow{a} x'$ and $R(x', y')$;
3. $x \downarrow$ iff $y \downarrow$, i.e. $x$ can terminate if and only if $y$ can.

Two terms $x$ and $y$ are bisimilar (notation $x \sim_S y$) if there exists a bisimulation relation $R$ such that $R(x, y)$.

A familiar result states that bisimulation can be captured equationally, i.e. two terms can be proved to be equal using the equations exactly when the two terms are bisimilar (see [20, 21]).
Theorem 2.2 The equational theory given by the axioms

\[ x + y = y + x \]  
(6)

\[ (x + y) + z = x + (y + z) \]  
(7)

\[ x + x = x \]  
(8)

\[ x + \delta = x \]  
(9)

\[ (x + y) \uparrow B = x \uparrow B + y \uparrow B \]  
(10)

\[ (a.x) \uparrow B = a.x, a \notin B \]  
(11)

\[ (a.x) \uparrow B = \delta, a \in B \]  
(12)

\[ \delta \uparrow B = \delta \]  
(13)

\[ \epsilon \uparrow B = \epsilon \]  
(14)

is sound and complete with respect to bisimulation for \( S_{\uparrow, \epsilon} \).

We will apply the algebraic results, that are presented in the sequel, to a subclass of SOS-systems that are characterized by the format of their rules. Here, we focus on the tagh-format of [5] that extends the well-known GSOS-format of [9] with termination.

Definition 2.3

1. A tagh transition rule for an \( n \)-ary operation \( f \) is a deduction rule of the format

\[
\frac{x_i \xrightarrow{a_{ij}} y_{ip} \mid i \in I, p \in P_i \quad \{x_j \not\rightarrow \mid j \in J, b \in B_j\} \quad \{x_k \downarrow \mid k \in K\} }{f(x_1, \ldots, x_n) \xrightarrow{a} C[x_m, y_{ip}]}
\]

where \( I, J, K \subseteq \{1, \ldots, n\} \), for \( i \in I \), \( P_i \) a nonempty finite index set, for \( j \in J \), \( B_j \) a finite set of actions, and, for \( m \in \{1, \ldots, n\}, i \in I, p \in P_i \), \( x_m, y_{ip} \) are pairwise distinct variables, that are the only variables that may occur in the context \( C[x_m, y_{ip}] \).

The arguments \( x_i, i \in I \) are called active, while the \( x_j, j \in J \) are called negative.

2. A tagh termination rule for an \( n \)-ary operation \( f \) is a deduction rule of the format

\[
\frac{\{x_k \downarrow \mid k \in K\} }{f(x_1, \ldots, x_n) \downarrow}
\]

where \( x_1, \ldots, x_n \) are pairwise distinct variables and \( K \subseteq \{1, \ldots, n\} \).
A tagh-system is an SOS system where all the rules for all the operations, except those of $\Sigma_{\dagger,\epsilon}$, are tagh rules.

Since the tagh-format is an instance of the panth-format, by a general result of [26] we have the following lemma.

**Lemma 2.4** Bisimulation is a congruence for every tagh-system. □

The main result of [5], exploiting similar techniques of [1] for GSOS-systems, is a procedure to automatically obtain for a tagh-system a sound and complete system of equations that characterize bisimulation.

**Theorem 2.5** There exists a procedure to obtain for any tagh-system $S_T$, a disjoint extension $S'_T$ and an equational theory $E'_T$ such that $E'_T$ is head-normalizing, sound and complete with respect to strong bisimulation. □

### 2.2 Free algebras and SOS systems

In this subsection we present basic notions from universal algebra and discuss the connection to our problem of finding a system of representatives of terms modulo bisimulation. For more details on free algebras see, for instance, [11, 22, 24].

An algebra is a pair $A = (A; \Sigma)$ where $A$ is a carrier set and $\Sigma$ is a signature, or a set of basic operations. A variety $V$ is a class of algebras of the same type (i.e., over the same signature) that is closed under homomorphic images, subalgebras and direct products. By Birkhoff’s variety theorem, every variety is an equational class, meaning that it consists of all the algebras that are models of a certain equational theory. Hence we usually think of a variety as a class of algebras defined by identities.

**Definition 2.6** Let $V$ be a variety. A free algebra in $V$ with base set $X$ is an algebra $F = (F; \Sigma)$ such that

- $F$ belongs to $V$.
- The set $X \subseteq F$ generates $F$.
- For every algebra $A = (A, \Sigma) \in V$, any mapping from $X$ to $A$ can be extended to a homomorphism from $F$ to $A$, i.e. $F$ satisfies the universal mapping property.

An initial algebra in $V$ is the free algebra with an empty base.
Let $\Sigma$ be a signature. Let $\mathcal{V}_0$ denote the variety over $\Sigma$ that has no defining identities, i.e. $\mathcal{V}_0$ contains all the algebras of type $\Sigma$. The free algebra in $\mathcal{V}_0$ with base $X$ is the algebra of $\Sigma$-terms built over the base $X$. This is also called the absolutely free algebra with base $X$ of type $\Sigma$, further on denoted by $T(\Sigma_X)$. In case of the initial algebra in $\mathcal{V}_0$ we write $T(\Sigma)$, it exists if and only if the signature contains constants.

For later reference we include here a classical characterization of free algebras in a given variety.

**Theorem 2.7** Let $\mathcal{V}$ be a variety of algebras of type $\Sigma$. Let $E_\mathcal{V}$ denote the underlying equational theory and $=_{\mathcal{V}}$ the equality in $E_\mathcal{V}$. Then the free algebra in $\mathcal{V}$ with base $X$ (up to isomorphism) is $T(\Sigma_X)/=_{\mathcal{V}}$. □

In a free algebra of a given variety only the identities of the whole variety hold and nothing else, hence it is free of any other laws. In an absolutely free algebra no laws hold, all the terms are distinct elements of the absolutely free algebra. For an example of a free algebra consider the variety of semigroups i.e. all algebras with a single binary operation that satisfies the associative law. The free semigroup with base $X$ is isomorphic with $X^+$, all words over the alphabet $X$ with the operation of concatenation. Every group can also be considered a member of this variety but, as more laws hold, is not free in the variety of semigroups.

Having an algorithmic construction of free algebras, with a countable base, in a variety $\mathcal{V}$ is known as solving the word problem for free algebras in $\mathcal{V}$, and is equivalent to the decision problem of the equational theory of $\mathcal{V}$. Namely, it gives a procedure to decide whether two given terms in signature $\Sigma$ (over a countable set of variables) are equal modulo identities of $\mathcal{V}$.

There are various ways of constructing free algebras in a given variety and they vary from rather specific constructions for certain varieties (e.g. [24, 22, 17, 18, 19]) to more general constructions (for example [10, 3]). Roughly speaking, in the sense of Definition 2.6, there are two main approaches. The first consists of constructing a carrier set that consists of a certain kind of objects, defining operations on it of type $\Sigma$ and proving that the conditions of Definition 2.6 are met. The second is by obtaining, via reduction, a system of representatives for $=_{\mathcal{V}}$, i.e. by constructing $F$ as a subset of $T(\Sigma_X)$ containing exactly one element for each $=_{\mathcal{V}}$-congruence class. Note that the first and the second construction could be linked together if we construct the isomorphism between the object $F$ obtained with the first construction and $T(\Sigma_X)/=_{\mathcal{V}}$ or construct the corresponding epimorphism from $T(\Sigma_X)$.
onto $F$. We follow this later approach when constructing the free algebras in Section 3.

By now we have mentioned the main ingredients used in this paper. Throughout the paper we will use the following notation:

- $S$ — an SOS system
- $\Sigma$ — the signature of $S$
- $T(\Sigma_S)$ — the set of ground terms over a signature $\Sigma_S$
- $\sim_S$ — bisimulation w.r.t. $S_S$
- $A_S$ — the algebra $T(\Sigma_S)/\sim_S$
- $E_S$ — equational theory for bisimulation w.r.t. $S_S$
- $=S$ — equality w.r.t. $E_S$
- $\mathcal{V}_S$ — the variety defined by $E_S$
- $F^X_S$ — the free algebra in $\mathcal{V}_S$ with base $X$
- $F_S$ — the initial algebra in $\mathcal{V}_S$

According to the convention above, the equational theory $E^\dagger_{\epsilon}$ consists of the identities of Theorem 2.2, i.e. the laws (6) up to (14).

Next we introduce several subsystems of $S^\dagger_{\epsilon}$ that will be used in the sequel. The simplest one is $S_F$ (with $F$ referring to $FINTREE$ as coined in [1]), with signature $\Sigma_F = \{+, \delta\} \cup \{a(.)| a \in Act\}$ and rules (1) and (2). The system $S_{F,\epsilon}$ with signature $\Sigma_{F,\epsilon} = \{+, \delta, \epsilon\} \cup \{a(.)| a \in Act\}$ and rules (1),(2) and (4) is an extension of $S_F$ with explicit termination. What remains of $S^\dagger_{\epsilon}$ without termination is the system $S^\dagger$ with signature $\Sigma^\dagger = \Sigma^\dagger \setminus \{\epsilon\}$ and rules (1), (2) and (3).

Note that $E_F$ consists of the identities (6) thru (9); $E_{F,\epsilon}$ has the same identities, while $E^\dagger_{\epsilon}$ has the identities (6) thru (13). For systems that do not include termination the definition of bisimulation is restricted to the first two conditions of Definition 2.1 as usual.

Since $E^\dagger_{\epsilon}$ is a disjoint extension of $E_{F,\epsilon}$ we have the following fact.

**Lemma 2.8** For $x, y \in T(\Sigma_{F,\epsilon})$ it holds that $x =^\dagger_{\epsilon} y$ iff $x =_{F,\epsilon} y$.

The following lemma explains the connection between free algebras and the issue of finding canonical representatives for bisimulation equivalence classes.

**Lemma 2.9** Let $S_S$ be an SOS system with signature $\Sigma_S$ such that bisimulation $\sim_S$ is a congruence. Assume that $E_S$ is a sound and complete equational theory for $S_S$. Let $\mathcal{V}_S$ be the variety induced by $E_S$. Then the initial $\mathcal{V}_S$-algebra $F_S$ is isomorphic to $A_S$, i.e. to $T(\Sigma_S)/\sim_S$. 7
Proof Since $E_S$ is sound and complete for $\sim_S$, we have that $x \sim_S y \iff x =_S y$ for any $x, y \in T(\Sigma_S)$. Hence $\sim_S$ is in fact the congruence generated by the defining identities of $\mathcal{V}_S$ and $F_S \cong T(\Sigma_S)/\sim_S$. □

Thus, under the conditions of Lemma 2.9, the problem of obtaining canonical representatives for bisimulation is equivalent to the problem of constructing the initial algebra in the corresponding variety.

3 Free $\mathcal{V}_{\dagger,\varepsilon}$ algebras

In this section we focus on the construction of the free algebra with a given base in the variety $\mathcal{V}_{\dagger,\varepsilon}$ induced by the laws of Theorem 2.2. By Lemma 2.9 above this amounts to constructing the quotient algebra $A_{\dagger,\varepsilon}$.

First, we focus on the representation of $A_{F,\varepsilon}$ for which we borrow some ideas from [18]. The construction of [18] of a free algebra for BPA, as it is called in [6], is a construction of the first kind explained above. However, following our basic observation, this can easily be turned into a construction of the second kind, by defining a suitable epimorphism.

As mentioned before, $\mathcal{V}_{F,\varepsilon}$ denotes the variety defined by the identities of $E_{F,\varepsilon}$ in the signature $\Sigma_{F,\varepsilon}$. Note that, in view of the laws of $E_{F,\varepsilon}$, any $\mathcal{V}_{F,\varepsilon}$-algebra is a semi-lattice with respect to $+$, with unit $\delta$. We shall use the description of a free semi-lattice with unit for the free objects in $\mathcal{V}_{F,\varepsilon}$.

Let, for the moment, $X$ be some given set such that $X \cap \mathcal{P}(X) = \emptyset$. Put $SL(X) = \{\emptyset\} \cup X \cup \mathcal{P}_{\geq 2}(X)$. We define an operation $*$ on $SL(X)$ as follows:

\[
\begin{align*}
\emptyset * z &= z * \emptyset = z \\
x * x' &= \{x, x'\}, \quad x \neq x' \\
x * x &= x \\
y * y' &= y \cup y' \\
y * x &= x * y = \{x\} \cup y
\end{align*}
\]

where $x, x' \in X, y, y' \in \mathcal{P}_{\geq 2}(X), z \in SL(X)$.

Note that the operation $*$ is essentially the familiar union of sets, only each singleton is replaced by the element itself. This is done since we want $X \subseteq SL(X)$ to hold, as the $SL$-operator will be applied recursively in the sequel. Also note that the operation $*$ is well-defined since we assumed $X \cap \mathcal{P}(X) = \emptyset$.

Lemma 3.1 The structure $(SL(X),*)$ is the free semi-lattice with unit $\emptyset$ and base $X$. 

8
Proof We prove that \( SL(X) \) belongs to the variety of semi-lattices with a unit, that it is generated by \( X \) and that it has the universal mapping property.

Let \( y \in P_{\geq 2}(X) \). Then \( y = \{x_1, \ldots, x_k\} \subseteq X \), for some \( k \geq 2 \). Thus, \( y = x_1 \ast \{x_2, \ldots, x_k\} \) if \( k > 2 \) and \( y = x_1 \ast x_2 \) if \( k = 2 \). This implies that \( P_{\geq 2}(X) \) is generated by \( X \).

From the definition of \( \ast \) it is obvious that \( SL(X) \) is a semi-lattice. Let \((S, \circ)\) be any semi-lattice with unit 0, and \( f: X \rightarrow S \) any map. We extend \( f \) to a homomorphism \( f^*: SL(X) \rightarrow S \) as follows:

\[
\begin{align*}
  f^*(\emptyset) &= 0 \\
  f^*(x) &= f(x) \text{ for } x \in X \\
  f^*(y) &= f^*(x_1) \circ \cdots \circ f^*(x_k) \text{ for } y = \{x_1, \ldots, x_k\} \in P_{\geq 2}(X)
\end{align*}
\]

It is, using the definition of \( \ast \), straightforwardly checked that \( f^* \) is indeed a homomorphism. From this the universal mapping property follows. \( \square \)

For the construction of our free algebras we need to introduce yet another operator.

Define \( \Sigma_p = \Sigma_{F,\epsilon} - \{+, \delta, \epsilon\} \), i.e. \( \Sigma_p \) is the signature consisting of prefixing operations only. For practical reasons we shall denote the absolutely free algebra with base \( X \) over this signature by \( \overline{X} \). Clearly \( \overline{X} \) is the set of all \( \Sigma_p \) terms built from \( X \).

Using the operators \( SL(\cdot) \) and \( \overline{\cdot} \) we construct the universe \( F_{F,\epsilon}^X \) of the free \( \mathcal{V}_{F,\epsilon} \)-algebra with base \( X \). We define a sequence of sets \( F_i, i \geq 0 \) by

\[
\begin{align*}
  F_0 &= X \cup \{\epsilon, \emptyset\} \\
  F_{i+1} &= SL(F_i \setminus P(F_i))
\end{align*}
\]

and we put \( F_{F,\epsilon}^X = \bigcup_{i \geq 0} F_i \). Note that all the sets \( F_i \setminus P(F_i) \) are eligible as argument of \( SL \), since \((F_i \setminus P(F_i)) \cap P(F_i \setminus P(F_i)) = \emptyset \). Also observe that \( F_i \cap P_1(F_i) = \emptyset \), so we can write \( F_i \setminus P(F_i) \) instead of \( F_i \setminus (\{\emptyset\} \cup P_{\geq 2}(F_i)) \).

The next lemma collects some properties that are needed later.

Lemma 3.2

\( (i) \) The set \( F_i \setminus P(F_i) \) does not contain sets as its elements, for \( i \geq 0 \), if \( X \) does not.

\( (ii) \) For \( i \geq 0 \), if an element \( x \) of \( F_{i+1} \) is a set, then \( x \subseteq F_i \setminus P(F_i) \).
(iii) $F_i \setminus \mathcal{P}(F_i) \subseteq F_{i+1} \setminus \mathcal{P}(F_{i+1})$ for $i \geq 0$.

(iv) $F_i \subseteq F_{i+1}$ for $i \geq 0$.

Proof

(i) Clearly, $F_0 \setminus \mathcal{P}(F_0)$ does not contain sets. Let $x \in F_i \setminus \mathcal{P}(F_i)$, for $i \geq 1$. Then $x \in F_i$ and either $x \in SL(F_{i-1} \setminus \mathcal{P}(F_{i-1}))$ or $SL(F_{i-1} \setminus \mathcal{P}(F_{i-1}))$ in which case $x$ is not a set, or $x \in SL(F_{i-1} \setminus \mathcal{P}(F_{i-1}))$. In the later case there are two more possibilities. The first case is that $x \in F_{i-1} \setminus \mathcal{P}(F_{i-1})$ and by the inductive hypothesis $x$ is not a set. The second case is $x = \{x_1, \ldots, x_k\}, k \geq 2$, $x_1, \ldots, x_k \in F_{i-1} \setminus \mathcal{P}(F_{i-1})$ which implies $x_1, \ldots, x_k \in F_i$, i.e. $x \in \mathcal{P}(F_i)$ which is a contradiction.

(ii) Let $x \in F_{i+1}$ be a set, for $i \geq 0$. Then $x \in SL(F_i \setminus \mathcal{P}(F_i))$ and by (i) we have that $x \notin F_i \setminus \mathcal{P}(F_i)$, i.e. $x \subseteq F_i \setminus \mathcal{P}(F_i)$.

(iii) By construction we have $F_i \setminus \mathcal{P}(F_i) \subseteq F_{i+1}$. In addition, by (i) if $x \in F_i \setminus \mathcal{P}(F_i)$ then $x$ is not a set, hence $x \notin \mathcal{P}(F_{i+1})$.

(iv) This is a consequence of (iii) and the definition of the sets $F_i$. 

\[\square\]

From the lemma we obtain as a corollary the following.

Corollary 3.3 The sequence $(SL(F_i \setminus \mathcal{P}(F_i)) \mid i \geq 0)$ is a chain of semilattices, the sequence $(F_i \mid i \geq 0)$ is a chain of algebras of type $\Sigma_p$ and $F^X_{F,\epsilon} = \bigcup_{i \geq 0} F_i = \bigcup_{i \geq 0} SL(F_i \setminus \mathcal{P}(F_i))$. \[\square\]

Thus $F^X_{F,\epsilon}$ inherits interpretations for + and prefixing operations from the $F_i$'s. Additionally, we can take $\emptyset$ is an interpretation for $\delta$. Therefore $F^X_{F,\epsilon}$ can be turned into a $\Sigma_{F,\epsilon}$-algebra, $F^X_{F,\epsilon}$, in a natural way.

Note that the elements of $F^X_{F,\epsilon}$ have the form $a_1a_2 \ldots a_nx$ where $x$ is either $\epsilon$ or $\emptyset$, or $x = \{x_1, \ldots, x_k\}$ for some $x_1, \ldots, x_k \in F^X_{F,\epsilon}$ which are not sets. The length of $a_1a_2 \ldots a_nx$ is the number $n$ of unary prefix operations that appear as top operations in this element, and the depth of $a_1a_2 \ldots a_nx$ is the number of pairs of braces appearing in $x$ ($0$ for $x = \epsilon$ or $x = \emptyset$). In this informal way we have defined two functions, length and depth, that map elements of $F^X_{F,\epsilon}$ to natural numbers.

By now we have the ingredients for our first result.
Theorem 3.4 \( F_X \) is a free \( \mathcal{V}_{F,\varepsilon} \)-algebra with base \( X \).

**Proof** The proof goes by induction on length and depth. First of all \( F_X \in \mathcal{V}_{F,\varepsilon} \), since by construction it is easy to see that the + operation satisfies the laws of a semi-lattice with unit \( \emptyset \). Next we check that \( X \) generates \( F_X \). If \( x = ax' \in F_X \setminus F_0 \) then by length-induction \( x' \) is generated by \( X \) and so is \( x \). Finally if \( x = \{x_1, \ldots, x_k\} \) then by depth-induction all the \( x_i \)'s are generated by \( X \) and so is \( x = x_1 + \cdots + x_k \). Let \( (F, \Sigma_{F,\varepsilon}) \) be any other \( \mathcal{V}_{F,\varepsilon} \)-algebra and \( f : X \rightarrow F \) be any mapping. Then we can extend \( f \) to a homomorphism \( f' : F_X \rightarrow F \) by putting \( f'(\epsilon) = \epsilon, f'(\emptyset) = \delta, f'(ax) = af'(x), f'(\{x_1, \ldots, x_n\}) = f'(x_1) + \cdots + f'(x_n) \). An easy inductive argument on length and depth shows that \( f' \) is indeed a well-defined homomorphism as desired. \( \square \)

For the remainder of this section we work with an empty base, so \( X = \emptyset \). We define the mapping \( e_F : T(\Sigma_{F,\varepsilon}) \rightarrow F_{F,\varepsilon} \) by

\[
\begin{align*}
e_F(\epsilon) &= \epsilon \\
e_F(\emptyset) &= \emptyset \\
e_F(ax') &= a e_F(x') \\
e_F(x' + x'') &= e_F(x') + e_F(x'')
\end{align*}
\]

where the + in the righthand side is the + of \( F_{F,\varepsilon} \). Clearly \( e_F \) is an epimorphism, so \( F_{F,\varepsilon} = e_F(T(\Sigma_{F,\varepsilon})) \). From the general theory we have \( T(\Sigma_{F,\varepsilon})/\ker(e_F) \cong F_{F,\varepsilon} \), hence two terms in \( T(\Sigma_{F,\varepsilon}) \) are equivalent modulo the identities of \( \mathcal{V}_{F,\varepsilon} \) exactly when they have the same image under \( e_F \), i.e.

\[ x =_{F,\varepsilon} y \iff e_F(x) = e_F(y) \quad (15) \]

Therefore the set \( \{e_F(x) \mid x \in T(\Sigma_{F,\varepsilon})\} \) constitutes a system of representatives of equivalence modulo \( E_{F,\varepsilon} \).

So far we have constructed the free \( \mathcal{V}_{F,\varepsilon} \)-algebra and we have shown how to obtain canonical representatives of congruence classes of \( T(\Sigma_{F,\varepsilon}) \). Next we extend the above construction for \( \mathcal{V}_{1,\varepsilon} \). Note that the identities given in Theorem 2.5 for the unary one-step restriction operations allow for elimination of these operations. In order to make this explicit we define mappings
\( e_B : T(\Sigma_{F,\epsilon}) \rightarrow T(\Sigma_{F,\epsilon}) \), for each subset \( B \subseteq Act \), by

\[
\begin{align*}
e_B(\epsilon) &= \epsilon \\
e_B(\delta) &= \delta \\
e_B(x' + x'') &= e_B(x') + e_B(x'') \\
e_B(ax) &= \delta \text{ if } a \in B \\
e_B(ax) &= ax \text{ if } a \notin B
\end{align*}
\]

and a mapping \( e_\uparrow : T(\Sigma_{t,\epsilon}) \rightarrow T(\Sigma_{F,\epsilon}) \) by \( e_\uparrow(x) = x \) if \( x \in T(\Sigma_{F,\epsilon}) \) and

\[
\begin{align*}
e_\uparrow(ax) &= a e_\uparrow(x) \\
e_\uparrow(x' + x'') &= e_\uparrow(x') + e_\uparrow(x'') \\
e_\uparrow(x \uparrow B) &= e_B(e_\uparrow(x))
\end{align*}
\]

otherwise. An inductive argument shows that the mappings \( e_\uparrow \) and \( e_B \), for \( B \subseteq Act \) are well-defined. Moreover, \( e_\uparrow \) is surjective, since for \( x \in T(\Sigma_{F,\epsilon}) \) we have \( x = e_\uparrow(x) \), and respects, by definition, all the operations of \( \Sigma_{F,\epsilon} \).

Additionally we have the following.

**Lemma 3.5** For any \( x \in T(\Sigma_{F,\epsilon}) \) it holds that \( x \uparrow B =_{t,\epsilon} e_B(x) \) and for any \( x \in T(\Sigma_{t,\epsilon}) \) we have \( x =_{t,\epsilon} e_\uparrow(x) \). For any \( x, y \in T(\Sigma_{F,\epsilon}) \) it holds that \( x =_{F,\epsilon} y \implies e_B(x) =_{F,\epsilon} e_B(y) \).

**Proof** By a simple inductive argument on the structure of \( \Sigma_{F,\epsilon} \)-terms it follows that \( x \uparrow B =_{t,\epsilon} e_B(x) \) for all \( x \in T(\Sigma_{F,\epsilon}) \). Then, once again by induction but now on the structure of \( \Sigma_{t,\epsilon} \) terms, we get that \( x =_{t,\epsilon} e_\uparrow(x) \) for all \( x \in T(\Sigma_{t,\epsilon}) \). If for two \( \Sigma_{F,\epsilon} \)-terms \( x, y \) and \( B \subseteq Act \) it holds that \( x =_{F,\epsilon} y \), then, as \( S_{t,\epsilon} \) is a disjoint extension of \( S_{F,\epsilon} \), it also holds that \( x =_{t,\epsilon} y \). Hence \( x \uparrow B =_{t,\epsilon} y \uparrow B \) and, by the above, \( e_B(x) =_{t,\epsilon} e_B(y) \). But, since \( e_B(x), e_B(y) \) are \( \Sigma_{F,\epsilon} \)-terms as well, we obtain \( e_B(x) =_{F,\epsilon} e_B(y) \) by another appeal to the disjointness of the extension of \( S_{t,\epsilon} \) over \( S_{F,\epsilon} \). \( \square \)

The lemma above states that any \( \Sigma_{t,\epsilon} \)-term \( x \) can be reduced to a \( \Sigma_{F,\epsilon} \)-term, viz. \( e_\uparrow(x) \). There are two reasons underlying the well-definedness of the mappings \( e_\uparrow \) and \( e_B \) and the proof of Lemma 3.5. The first is known as a head-normalization property. Namely for each \( \Sigma_{t,\epsilon} \)-term \( x \) there either exist \( \Sigma_{t,\epsilon} \)-terms \( x_1, x' \), and an action \( a_1 \) such that \( x = a_1 x_1 + x' \), or \( x = \epsilon \), or \( x = \delta \). Moreover, the transition relation is finitely branching for \( S_{t,\epsilon} \). The second reason is that the rules for \( \uparrow B \)-operations are well-founded, i.e. no infinite sequence of rewritings is possible. We will exploit this observation below, when generalizing the above result.
So far we have that $T(\Sigma_{t,\epsilon}) \xrightarrow{e_t} T(\Sigma_{F,\epsilon}) \xrightarrow{e_F} F_{F,\epsilon}$, and both $e_t$ and $e_F$ are surjective mappings that respect the operations of $\Sigma_{F,\epsilon}$. Hence $e_F \circ e_t$ is itself surjective, i.e. $(e_F \circ e_t)(T(\Sigma_{t,\epsilon})) = F_{F,\epsilon}$, and it respects the operations of $\Sigma_{F,\epsilon}$. Moreover, the following holds.

**Lemma 3.6** For all $x, y \in T(\Sigma_{t,\epsilon})$ it holds that $x =_{t,\epsilon} y \iff (e_F \circ e_t)(x) = (e_F \circ e_t)(y)$.

**Proof** Let $x, y \in T(\Sigma_{t,\epsilon})$. Then we have that $x =_{t,\epsilon} y$ iff $e_t(x) =_{t,\epsilon} e_t(y)$, by Lemma 3.5, iff $e_t(x) =_{F,\epsilon} e_t(y)$, by Lemma 2.8, iff $e_F(e_t(x)) = e_F(e_t(y))$, by equation (15).

Next we extend the signature of $F_{F,\epsilon}$ by adding the one-step restriction operations. We put, for $x \in F_{F,\epsilon}$, $x \uparrow B = (e_F \circ e_t)(x_0 \uparrow B)$ if $x_0 \in T(\Sigma_{t,\epsilon})$ is such that $(e_F \circ e_t)(x_0) = x$ and $B \subseteq \text{Act}$. We check that this indeed defines operations $(\cdot) \uparrow B$ on $F_{F,\epsilon}$.

**Lemma 3.7** The operations $(\cdot) \uparrow B$ on $F_{F,\epsilon}$ are well-defined.

**Proof** Pick $x \in F_{F,\epsilon}$. Let $x', x'' \in T(\Sigma_{t,\epsilon})$ be such that $e_F(e_t(x')) = x$ and $e_F(e_t(x'')) = x$. Hence $e_F(e_t(x')) = e_F(e_t(x''))$ and, by equation (15), \( e_t(x') =_{F,\epsilon} e_t(x'') \). Therefore, by one of the properties of Lemma 3.5, we obtain $e_B(e_t(x')) =_{F,\epsilon} e_B(e_t(x''))$. This is, by the definition of $e_t$, equivalent to $e_t(x' \uparrow B) =_{F,\epsilon} e_t(x'' \uparrow B)$. Using equation (15) we conclude that $e_F(e_t(x' \uparrow B)) = e_F(e_t(x'' \uparrow B))$, which was to be shown.

Let $F_{t,\epsilon}$ denote the algebra obtained by extending $F_{F,\epsilon}$ to a $\Sigma_{t,\epsilon}$-algebra using the operations $(\cdot) \uparrow B$ as given above.

**Theorem 3.8** The algebra $F_{t,\epsilon}$ is the initial algebra in the variety $\mathcal{V}_{t,\epsilon}$.

**Proof** As stated before $e_F \circ e_t$ is an epimorphism from $T(\Sigma_{t,\epsilon})$ onto $F_{t,\epsilon}$ such that, by Lemma 3.6, $\ker(e_F \circ e_t)$ is equal to $=_{t,\epsilon}$. Hence $F_{t,\epsilon} \cong T(\Sigma_{t,\epsilon})/\ker(e_F \circ e_t) = T(\Sigma_{t,\epsilon})/ =_{t,\epsilon}$ and the statement follows by Theorem 2.7.

The next consequence is of special interest to us as a characterization of bisimulation.

**Corollary 3.9** For any $x, y \in T(\Sigma_{t,\epsilon})$ it holds that $x \sim_{t,\epsilon} y \iff (e_F \circ e_t)(x) = (e_F \circ e_t)(y)$.
Proof Immediate from Theorem 3.8 and the results of Section 2.

Thus the set \( \{ (e_F \circ e_\dagger)(x) \mid x \in T(\Sigma_{1,\epsilon}) \} \) is a system of representatives for bisimulation equivalence for \( S_{1,\epsilon} \). Hence, we have some canonical representatives which are not terms. Such can be achieved by introducing an ordering on terms, but there is no need for doing so; there is already a procedure for determining whether two terms have the same canonical representative. It should also be noted that the application of the mappings \( e_F \) and \( e_\dagger \) can be done mechanically. This is in fact the term rewriting mechanism underlaying the equational theory.

4 More free algebras

In the previous section we discussed the construction of the initial \( V_{F,\epsilon} \)-algebra \( e_F(T(\Sigma_{F,\epsilon})) \) as well as the construction of the initial \( V_{\dagger,\epsilon} \)-algebra \( e_F(e_\dagger(T(\Sigma_{\dagger,\epsilon}))) \). We can do the same for other operations defined by SOS rules. In fact, the procedure applies well for an SOS system in tagh-format under some mild conditions.

The two essential elements for the argumentation above are a head-normalization property and a well-foundedness criterion. Head-normalization is free for SOS system \( S \) in tagh-format; it always holds in an equivalent extension \( S' \) of \( S \). Thus, for tagh-systems that can be shown to be well-founded we can construct a mapping \( e: T(\Sigma_S) \rightarrow T(\Sigma_{\dagger,\epsilon}) \) as before. From this an initial algebra and a system of representatives can be constructed.

Definition 4.1 Let \( S_S \) be an SOS system with signature \( \Sigma_S \). A term \( x \in T(\Sigma_S) \) is well-founded if there is no infinite sequence of \( S_S \)-transitions starting in \( x \), i.e. there does not exist an infinite sequence of terms \( (x_i \mid i \geq 0) \) and an infinite sequence of actions \( (a_i \mid i \geq 0) \) such that \( x = x_0 \) and \( x_i \xrightarrow{a_i} S_S x_{i+1} \) for \( i \geq 0 \). The SOS system \( S_S \) is well-founded if all terms in \( T(\Sigma_S) \) are well-founded.

Unfortunately well-foundedness of SOS rules is undecidable in general. But as shown in [1], it is decidable via syntactical well-foundedness for a rather wide class of \( GSOS \) rules. Since termination does not at all affect well-foundedness of rules, the same holds for tagh-format rules.

A tagh system is called linear if each variable occurs at most once in the target \( (C[x_m, y_{ip}]) \) and for each active argument \( i \), at most one of the following holds: \( x_i \) occurs in the target; at most one of \( y_{ip} \) does. For linear tagh-systems well-foundedness is indeed decidable, in fact it is equivalent to
the problem of determining whether a linear system of diophantine equations
has a solution in the set of natural numbers.

When all the rules that extend \( S^\dagger,\epsilon \) are well founded, then together with
head-normalization property, by Noetherian induction ([11, 16]), it follows
that a counterpart of Lemma 3.6 holds for all extra operations of the tagh-
system \( E_{S'} \).

More concretely, since there are no infinite transitions and there is no
infinite branching, we can define a function "maximal number of transitions
in a row possible", \( mntp \), that maps any term \( x \in T(\Sigma_{S'}) \) to a natural
number \( mntp(x) \) which is the number of transitions in a maximal transition
sequence starting from \( x \). Then by induction on \( mntp \) of terms from the
head-normalization property it follows that for any \( S_{\dagger,\epsilon} \) term \( x \) there is an
\( S_{\dagger,\epsilon} \) term \( y \) such that \( x \) is bisimilar to \( y \) i.e. \( x =_{S'} y \) in \( A_{S'} \).

Thus, in the same manner as before, we can define a surjective mapping \( e \) which respects the \( \Sigma_{\dagger,\epsilon} \) operations and \( e(x) =_{S'} x \) for \( x \in T(\Sigma_{S'}) \).
This way we get that \( T(\Sigma_{S'}) \xrightarrow{e} T(\Sigma_{\dagger,\epsilon}) \xrightarrow{e_F \circ e_\dagger \circ e} F_{\dagger,\epsilon} \). Subsequently, we extend
the signature of \( F_{\dagger,\epsilon} \) to the signature \( \Sigma_{S'} \) thus obtaining the algebra \( F_{S'} \)
with universe \( F_{S'} = F_{\dagger,\epsilon} \). Namely, if \( f \) is an \( n \)-ary operation in \( \Sigma_{S'} \) and
\( e(x_1), \ldots, e(x_n) \in F_{\dagger,\epsilon} \) then we put \( f(e(x_1), \ldots, e(x_n)) = e(f(x_1, \ldots, x_n)) \)
where \( f(x_1, \ldots, x_n) \) in the right hand side denotes an existing term in
\( T(\Sigma_{S'}) \).

In the exactly same way as before one can prove the following.

**Lemma 4.2** For all \( x, y \in T(\Sigma_{S'}) \) it holds \( x =_{S'} y \iff (e_F \circ e_\dagger \circ e)(x) = (e_F \circ e_\dagger \circ e)(y) \). \( \Box \)

The lemma is the main ingredient of the next theorem.

**Theorem 4.3** The structure \( F_{S'} \) is the initial algebra in the variety \( \mathcal{V}_{S'} \). \( \Box \)

From the theorem, in turn, it follows that

**Corollary 4.4** For any \( x, y \in T(\Sigma_{S}) \), terms in the original signature, it
holds that \( x \sim_{S} y \iff (e_F \circ e_\dagger \circ e)(x) = (e_F \circ e_\dagger \circ e)(y) \). \( \Box \)

So, we start of from a transition system specification \( S \) in tagh-format. By
exploiting the result from [1, 5] we obtain a disjoint extension \( S' \) of which the
accompanying equational theory \( E_{S'} \) is sound and complete for bisimulation
with respect to \( S' \). The theory developed in the present paper, enables us
to construct the initial $V_{S'}$ algebra via a suitable epimorphism. Via this mapping we are given a system of canonical representatives for bisimulation for the starting tagh system $S$. Thus, in the end, the problem whether two $S$-terms $x, y$ are bisimilar can be answered: they are if $e_F(e_\dagger(e(x))) = e_F(e_\dagger(e(y)))$.

References


