# Rule formats for bounded nondeterminism in structural operational semantics*

Luca Aceto, Álvaro García-Pérez, and Anna Ingólfsdóttir

ICE-TCS, School of Computer Science, Reykjavík University,
Menntavegur 1, IS-101, Reykjavík, Iceland.

**Abstract** We present rule formats for structural operational semantics that guarantee that the associated labelled transition system has each of the three following finiteness properties: finite branching, initials finiteness and image finiteness.

**Keywords:** structural operational semantics, labelled transition systems, rule formats, bounded nondeterminism

## 1 Introduction

Structural operational semantics (SOS) [25, 27] is a widely used formalism for defining the formal semantics of computer programs and for proving properties of the corresponding programming languages. In the SOS formalism a transition system specification (TSS) [13], which consists of a signature together with a set of inference rules, specifies a labelled transition system (LTS) [16] whose states (i.e., processes) are closed terms over the signature and whose transitions are those that can be proved using the inference rules.

Rule formats [2, 21] are syntactically checkable restrictions on the inference rules of a TSS that guarantee some useful property of the associated LTS. The properties ensured by such rule formats vary from compositionality of behavioural equivalences [7,13,14,30] to finiteness of the number of outgoing transitions from a given state [6,9,32]. This paper focuses on the finiteness property, which is referred to as bounded nondeterminism in [12]. Broadly, bounded nondeterminism is taken as a synonym of finite branching [9]. Finite branching breaks down into the more elementary properties of initials finiteness and image finiteness [1] (see Section 2 for formal definitions).

Vaandrager [32] introduced the notion of bounded TSS and proved that a bounded TSS in de Simone format [30] induces an LTS that is finite branching. Bloom [6] used a notion of bounded TSS reminiscent of that of Vaandrager and showed that a bounded TSS in his higher-order-GSOS format [6] induces an LTS that is finite branching. Finally, Fokkink and Vu [9] used yet another notion of bounded TSS and introduced a less restrictive rule format that they called

* This research has been supported by the project 'Nominal Structural Operational Semantics' (nr. 141558-051) of the Icelandic Research Fund.

'bounded nondeterminism format'. They adapted the notion of strict stratification from [14] and showed that a bounded TSS in bounded nondeterminism format that has a strict stratification induces an LTS that is finite branching.

In this paper we take Fokkink and Vu's programme further and present rule formats for initials finiteness and for image finiteness. For initials finiteness we relax the requirement that the $\eta$-types of [9] be finitely inhabited and we introduce the initials finite format, which replaces the bounded nondeterminism format of [9]. For image finiteness, we introduce the notion of $\theta$-type. Unlike the $\eta$-types of [9], which carry information about the sources of positive premisses in rules, the $\theta$-types also keep track of the actions that label positive premisses. Moreover, we introduce a uniformity requirement on the targets of positive premisses, which strengthens the requirement in [9] that the variables in a rule have to be used uniformly. We introduce the accompanying notions of initials-bounded TSS and image-bounded TSS and show the following results.

- An initials-bounded TSS in initials finite format that has a strict stratification induces an LTS that is initials finite (Theorem 2).
- An image-bounded TSS in bounded nondeterminism format that has a strict stratification induces an LTS that is image finite (Theorem 3).

The results and the techniques we employ in this paper touch upon some of the main topics in the research of Flemming Nielson and Hanne Riis Nielson over the years, namely operational semantics [23], static analysis [22] and type systems [3]. This study contributes to the development of a general theory of operational semantics based on rule formats, which may be seen as providing some statically checkable, largely syntactic, conditions guaranteeing that the specified languages afford some semantic properties of interest. The various notions of 'types' that we use in the definition of the rule formats discussed in this paper allow us to classify the inference rules in a language specification. Informally, types contribute to guaranteeing that composite processes have the finiteness property of interest, if their components do so.

The rest of the paper is organised as follows. Section 2 revisits preliminaries and basic notions from [9] and adapts some of its definitions. Definition 10 formalises the notion of uniform TSS and Proposition 2 shows that a closed term $p$ unifies only with finitely many rules in a uniform TSS. Section 3 provides an alternative proof of Theorem 1 in [9] that removes the *reductio ad absurdum* argument that is used there. Theorem 1 shows that a bounded TSS in bounded nondeterminism format that has a strict stratification induces an LTS that is finite branching. The proof of Theorem 1 here is direct and fully constructive. Section 4 discusses the variable flow in a transition rule and Definition 18 introduces the initials finite format, which requires that each variable in the source of a positive premiss occur also in the source of the rule. Definition 20 introduces the notion of initials-bounded TSS, which relaxes the $\eta$-types of [9] by requiring that the actions of an $\eta$-type are finite, instead of requiring the $\eta$-type to be finitely inhabited. Theorem 2 shows that an initials-bounded TSS in initials finite format that has a strict stratification induces an LTS that is initials finite. Section 5 discusses the logical content of the $\eta$-types under the prism of intuitionistic logic [19], and shows that the $\eta$-types realise the intuitionistic interpretation of the property of initials finiteness. Definition 21 introduces the $\theta$-types, which are analogous to the $\eta$-types in that they realise the intuitionistic interpretation of the property of image finiteness. Definition 22 introduces uniformity in the targets of positive premisses, which prevents the $\theta$-types to be infinitely many as a result of using infinitely many different names for a variable occurring in the target of some positive premiss, and Definition 23 introduces the notion of image-bounded TSS. Theorem 3 shows that an image-bounded TSS in bounded nondeterminism format that has a strict stratification induces an LTS that is image finite. Section 6 discusses avenues for future work and concludes.

## 2 Preliminaries

We give an overview of the structural operational semantics formalism (SOS for short). We follow the notation and the presentation in [9].

For a set $S$, we write $\mathcal{P}(S)$ for the collection of all the subsets of $S$, and $\mathcal{P}_\omega(S)$ for the collection of all the finite subsets of $S$.

**Definition 1 (Signature and Term).** *We assume a countably infinite set of variables $V$, ranged over by $x$, $y$, $z$. A signature $\Sigma$ is a set of function symbols, disjoint from $V$, together with an arity map that assigns a natural number to each function symbol. We use $f$ to range over $\Sigma$. Function symbols of arity zero, which may be ranged over by $c$, $d$, are called constants. Function symbols of arity one and two are called unary and binary functions respectively.*

*The set $\mathbb{T}(\Sigma)$ of (open) terms over a signature $\Sigma$, ranged over by $t$, $u$, $v$, is the least set such that:*

1. *each variable is a term, and*
2. *if $f$ is a function symbol of arity $n$ and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is a term.*

*The function $\mathrm{var} : \mathbb{T}(\Sigma) \to \mathcal{P}_\omega(V)$ delivers, for a term $t$, the set of variables that occur in $t$. A term $t$ is closed iff $\mathrm{var}(t) = \emptyset$. The set of closed terms over $\Sigma$, ranged over by $p$, $q$, is denoted by $T(\Sigma)$.*

**Definition 2 (Formula).** *We consider a set of actions $A$, ranged over by $a$, $b$ (and $c$ when no confusion arises with the constants). The set of positive formulae over signature $\Sigma$ and actions $A$ is the set of triples $(t, a, t') \in \mathbb{T}(\Sigma) \times A \times \mathbb{T}(\Sigma)$. We use the more suggestive notation $t \xrightarrow{a} t'$ in lieu of $(t, a, t')$. The set of negative formulae over signature $\Sigma$ and actions $A$ is the set of pairs $(t, a) \in \mathbb{T}(\Sigma) \times A$. We use the more suggestive notation $t \stackrel{b}{\not\to}$ in lieu of $(t, b)$.*

**Definition 3 (Substitution).** *A substitution is a partial map $\sigma : V \to \mathbb{T}(\Sigma)$. The substitutions are ranged over by $\sigma$, $\tau$. A substitution is closed if it maps variables to closed terms. A substitution extends to a map from terms to terms in the usual way, i.e., the term $\sigma(t)$ is obtained by replacing the occurrences in* *t of each variable $x$ in the domain of $\sigma$ by $\sigma(x)$. When applying substitutions $\sigma$ and $\tau$ successively, we may abbreviate $\tau(\sigma(t))$ to $\tau\sigma(t)$. We say term $u$ is a substitution instance of $t$ iff there exists a substitution $\sigma$ such that $\sigma(t) = u$.*

*In what follows, we shall sometimes use the notation $\{x_i \mapsto t_i \mid i \in I\}$, where $I$ is an an index set and the $x_i$'s are pairwise distinct variables, to denote the substitution that maps each $x_i$ to the term $t_i$ ($i \in I$).*

A substitution $\sigma$ extends to formulae $t \xrightarrow{a} t'$ and $u \stackrel{b}{\not\rightarrow}$ in the usual way, by applying the substitution to the term components of the formulae, i.e., $\sigma(t) \xrightarrow{a} \sigma(t')$ and $\sigma(u) \stackrel{b}{\not\rightarrow}$ respectively. The notion of substitution instance extends similarly.

**Definition 4 (Labelled transition system).** *Let $\Sigma$ be a signature and $A$ a set of actions. A labelled transition system (LTS for short) is a pair $(T(\Sigma), \rightarrow)$ where $T(\Sigma)$ is the set of processes, i.e., closed terms, and $\longrightarrow \subseteq T(\Sigma) \times A \times T(\Sigma)$ is the set of transitions, i.e., closed positive formulae. We say that $p \xrightarrow{a} p'$ is a transition of the LTS iff $(p, a, p') \in \longrightarrow$.*

Labelled transition systems [16] are a fundamental model of computation and are often used to describe the operational semantics of programming and specification languages—see, for instance, [20, 26, 27, 29]. Transition system specifications, which we now proceed to define, describe the LTS giving the semantics of a language by means of a signature (namely, the collection of term constructors offered by the language) and a set of inference rules that can be used to prove the valid transitions between terms in the language.

**Definition 5 (Transition system specification).** *Let $\Sigma$ be a signature and $A$ a set of actions. A transition rule (a rule, for short) $\rho$ is of the form*

$$\frac{H}{t \xrightarrow{a} t'}$$

*(abbreviated as $H/t \xrightarrow{a} t'$) where $H$ is a set of positive premisses of the form $u \xrightarrow{b} u'$ and negative premisses of the form $v \stackrel{c}{\not\rightarrow}$, and $t \xrightarrow{a} t'$ is the conclusion of the rule (with $t, t', u, u', v \in \mathbb{T}(\Sigma)$ and $a, b, c \in A$). We say $t$ is the source, $a$ is the action, and $t'$ is the target of $\rho$. We say $\rho$ is an axiom iff $\rho$ has an empty set of premisses, i.e., $H = \emptyset$.*

*A transition system specification (TSS for short) is a set of transition rules.*

A substitution map extends to a rule $\rho$ by applying the substitution to the formulae in $\rho$. The notion of substitution instance extends similarly to rules.

**Definition 6 (Unify with a rule).** *Let $R$ be a TSS. We say that transition $p \xrightarrow{a} p'$ unifies with rule $\rho \in R$ iff $\rho$ has conclusion $t \xrightarrow{a} t'$ and $p \xrightarrow{a} p'$ is a substitution instance of $t \xrightarrow{a} t'$.*

**Definition 7 (Proof tree).** *Let $R$ be a TSS without negative premisses. A proof tree in $R$ of a transition $p \xrightarrow{a} p'$ is an upwardly branching tree without paths of infinite length whose nodes are labelled by transitions such that*

1. *the root is labelled by $p \xrightarrow{a} p'$, and*
2. *if $K$ is the set of labels of the nodes directly above a node with label $q \xrightarrow{b} q'$, then $K/q \xrightarrow{b} q'$ is a substitution instance of some rule $H/t \xrightarrow{b} t' \in R$.*

*We say that $p \xrightarrow{a} p'$ is provable in $R$ iff $p \xrightarrow{a} p'$ has a proof tree in $R$.*

The set of provable transitions in $R$ is the least set of transitions that satisfies the rules in $R$. Notice that if $p \xrightarrow{a} p'$ unifies with an axiom (i.e., a rule of the form $\emptyset/t \xrightarrow{a} t'$) then, trivially, $p \xrightarrow{a} p'$ has a proof tree in $R$ which consists of a root node labelled by $p \xrightarrow{a} p'$.

A TSS without negative premisses induces an LTS in a straightforward way.

**Definition 8 (TSS induces LTS).** *Let $R$ be a TSS without negative premisses and $T$ an LTS. $R$ induces $T$ (or $T$ is associated with $R$) iff the set of transitions of $T$ is the set of provable transitions in $R$.*

*The phrases*

1. *$p \xrightarrow{a} p'$ is provable in $R$,*
2. *$p \xrightarrow{a} p'$ is a transition of $T$, and*
3. *$p$ can perform an $a$-transition to $p'$ in $T$*

*are synonyms. For brevity, we may omit the $R$ and/or the $T$ when they are clear from the context.*

In [28], Przymusinsky introduced *three-valued stable models*, which can be used to associate an LTS to a TSS with negative premisses. Each TSS has a least three-valued stable model, which coincides with the well-founded semantics from [11]. We consider the set of *sentences that are certainly true* in the least three-valued stable model, which, for a TSS without negative premisses, coincides with the set of provable transitions in Definition 8. As Fokkink and Vu noticed in [9], if $R$ is a TSS and $R'$ is obtained by removing all the negative premisses from the rules in $R$, then the LTS associated with $R$ is included in the LTS associated with $R'$. In particular, if the LTS associated with $R'$ has any of the finiteness properties considered in this paper, then the LTS associated with $R$ has the property too. We follow [9] and ignore the negative premisses in the TSSs. None of the rule formats that we introduce here impose any restrictions on negative premisses.

The notion of *uniform TSS* stems from [9]. We introduce the notion of *structure of a term*, and provide a formal definition of uniform TSS, to which we refer as *uniform TSS in the sources* because the focus is on the sources of the rules.

**Definition 9 (Structure of a term).** *Let $R$ be a TSS. The terms $t$ and $u$ have the same structure iff*

1. $t = x$ and $u = y$, where $x$ and $y$ are variables, or
2. $t = f(t_1, \ldots, t_n)$ and $u = f(u_1, \ldots, u_n)$, where $f$ is a function symbol of arity $n \geq 0$, and the terms $t_i$ and $u_i$ have the same structure for each $1 \leq i \leq n$.

Intuitively, two terms $t$ and $u$ have the same structure iff their syntax trees differ only in the name of the variables. For example $f(x, y)$ and $f(x, x)$ have the same structure. Two closed terms have the same structure iff they are the same term.

**Definition 10 (Uniform in the sources).** *A TSS $R$ is uniform in the sources iff $t = u$ holds whenever $t$ and $u$ have the same structure and are sources of any two rules in $R$.*

In Section 5 we will introduce the analogous notion of *uniform TSS in the targets of positive premisses*, in which the focus is on the targets of positive premisses. When no confusion arises, we may abbreviate and say 'uniform TSS' for 'uniform TSS in the sources'.

The rationale behind uniformity in [9] is to enforce that in a uniform TSS, each closed term is a substitution instance of the sources of at most finitely many transition rules. In order to show this property, we introduce the notion of *partial term* and the *less-defined-than* relation.

**Definition 11 (Partial term).** *The set $\mathbb{T}_\perp(\Sigma)$ of partial terms over a signature $\Sigma$, ranged over by $r$, $s$, is the set of terms that results by extending $\Sigma$ with the constant symbol $\perp$. The symbol $\perp$, which stands for 'undefined', is different from the other symbols in $\Sigma$.*

Notice that $T(\Sigma) \subset \mathbb{T}(\Sigma) \subset \mathbb{T}_\perp(\Sigma)$. The notion of structure of a term from Definition 9 is extended to partial terms in a straightforward way by considering the symbol $\perp$ as a variable. For instance, $f(x, y)$ has the same structure as $f(\perp, z)$.

**Definition 12 (Less-defined-than relation).** *The relation $\sqsubseteq$ (which we refer to as the less-or-equally-defined-than relation) is the least binary relation over partial terms such that*

1. *$\perp \sqsubseteq r$ for each partial term $r$,*
2. *$x \sqsubseteq x$ for each variable $x$, and*
3. *$f(s_1, \ldots, s_n) \sqsubseteq f(r_1, \ldots, r_n)$ where $f$ is a function symbol of arity $n \geq 0$ iff $s_i \sqsubseteq r_i$ for each $1 \leq i \leq n$.*

*We say $s$ is an approximant of $r$ iff $s \sqsubseteq r$.*

*The less-defined-than relation $\sqsubset$ is the binary relation over partial terms defined thus: $s \sqsubset r$ iff $s \sqsubseteq r$ and $s \neq r$.*

It is easy to see that $\sqsubseteq$ induces a partial order and $\sqsubset$ induces a strict partial order over partial terms.

**Proposition 1.** *The less-defined-than relation, $\sqsubset$, is a well-founded relation.*

*Proof.* We prove that there exists no infinite decreasing chain $r_1 \sqsupset r_2 \sqsupset \ldots$. To this end, we first define the size of a partial term $r$ as follows:

1. the size of $\perp$ is zero,
2. the size of a variable is one, and
3. the size of $f(r_1, \ldots, r_n)$ with $f$ a function symbol of arity $n \geq 0$ is one plus the sum of the sizes of the $r_i$'s with $1 \leq i \leq n$.

Let $r$ and $s$ be partial terms. If $s \sqsubset r$ then $s$ is obtained by replacing by $\perp$ one or more maximally disjoint subterms of $r$ that are different from $\perp$, and hence the size of $s$ is strictly smaller than that of $r$. Since the size of every partial term is finite, each decreasing chain is also finite and we are done. □

**Proposition 2.** *Let $R$ be a uniform TSS. For each closed term $p$ the set of pairs $(t, \sigma)$ with $\sigma : \mathrm{var}(t) \to T(\Sigma)$ such that $\sigma(t) = p$ and $t$ is the source of some rule in $R$ is finite.*

*Proof.* We prove the generalised proposition:

> *Let $R$ be a uniform TSS. For each pair $(p, r)$ where $p$ is a closed term and $r$ is an approximant of $p$ (i.e., $r \sqsubseteq p$), the set of pairs $(t, \sigma)$ with $\sigma : \mathrm{var}(t) \to T(\Sigma)$ such that there exists $r'$ an approximant of $r$ (i.e., $r' \sqsubseteq r$) with the same structure as $t$, and $\sigma(t) = p$ and $t$ is the source of some rule in $R$, is finite.*

The original proposition follows from the generalised proposition by fixing $r = p$. We construct the set $S$ of pairs $(t, \sigma)$ that meet the conditions of the generalised proposition and show that $S$ is finite. We proceed by well-founded induction on the set of approximants of $r$ ordered by the less-defined-than relation ($\sqsubset$).

We first check that the generalised proposition holds for the $\sqsubset$-minimal partial terms in the set of approximants of $r$. The only such partial term is $r_0 = \perp$. There exists only one $r'_0$ an approximant of $r_0$ (i.e., $r'_0 = \perp \sqsubseteq \perp = r_0$) and the only terms that have the same structure as $r'_0$ are the variables. Since $R$ is uniform, all the rules whose source is a variable (if there are any) have the same variable $x$ as source. If there exist no rules whose source is a variable, then the set we are looking for is the empty set. Otherwise, the set we are looking for is $S = \{(x, \{x \mapsto p\})\}$. Both sets are finite.

Now we check that the generalised proposition holds for an arbitrary partial term $r_a \neq \perp$ in the set of approximants of $r$. Notice that the partial term $r_a$ is such that $r_a \sqsubseteq r \sqsubseteq p$. By the induction hypothesis, for every $r_x$ such that $r_x \sqsubset r_a$, the set $S_x$ of pairs $(t, \sigma)$ such that there exists $r'_x$ an approximant of $r_x$ (i.e., $r'_x \sqsubseteq r_x$) with the same structure as $t$, and $\sigma(t) = p$ and $t$ is the source of some rule in $R$, is finite. Since $R$ is uniform, all the rules whose source has the same structure as $r'_x$ (if there are any) have the same source $u$. Since $r'_x \sqsubset r_a$, then $r'_x$ is obtained by replacing by $\perp$ one or more maximally disjoint subterms of $r_a$ that are different from $\perp$. We let $x_j$ (with $j$ ranging over some index set $J$) be the variables that occur in $u$ in the positions corresponding to the occurrences of $\perp$ in $r'_x$. (Notice that no other variables could occur in $u$, since $u$ has the same structure as $r'_x$, and $r'_x \sqsubset p$.) We let $t_j$ be the terms such that $p$ results from replacing respectively the $x_j$ by the $t_j$ in $u$. If there exist no rules with source $u$, then the set we are looking for is $S = S_x$. Otherwise, the set we are looking for is $S = S_x \cup \{(u, \{x_j \mapsto t_j \mid j \in J\})\}$. Both sets are finite. □

The next example shows that Proposition 2 does not hold for TSSs that are not uniform.

*Example 1.* Let $\Sigma$ consist of a constant $c$ and assume $A = \{a\}$. Let the $x_i$ with $i \in \mathbb{N}$ be infinitely many distinct variables. Consider the TSS with rules

$$\frac{}{x_i \xrightarrow{a} c}\,, \qquad i \in \mathbb{N}.$$

All the $x_i$ in the instantiations of the rule template above have the same structure, but $x_j \neq x_k$ for $j, k \in \mathbb{N}$ and $j \neq k$. Therefore, the TSS is not uniform. Notice that for $c$ there exist infinitely many pairs $(x_i, \sigma_i)$ (with $i \in \mathbb{N}$ and $\sigma_i = \{x_i \mapsto c\}$) such that $\sigma_i(x_i) = c$.

We focus on the properties of finite branching, initials finiteness, and image finiteness [1], which we define next.

**Definition 13 (Bounded nondeterminism).** *Let $T$ be an LTS and $p$ a closed term in $T$. We say*

1. *$p$ is finite branching iff the set $\{(a, p') \mid p \xrightarrow{a} p'\}$ is finite,*
2. *$p$ is initials finite iff the set $\{a \mid \exists p'$ s.t. $p \xrightarrow{a} p'\}$ is finite, and*
3. *$p$ is image finite iff for every action $a$, the set $\{p' \mid p \xrightarrow{a} p'\}$ is finite.*

*An LTS $T$ is finite branching (resp. initials finite and image finite) iff every closed term in $T$ is finite branching (resp. initials finite and image finite).*

We call $\{a \mid \exists p' \text{ s.t. } p \xrightarrow{a} p'\}$ the set of initials of $p$. We call $\{p' \mid p \xrightarrow{a} p'\}$ the set of images of $p$ for action $a$.

## 3 Finite branching

The rule format in [9], which restricts a TSS to be bounded, to be in bounded nondeterminism format, and to have a strict stratification, ensures that the associated LTS is finite branching. Intuitively, the restrictions such a format places on the allowed rules ensure that, for each closed term $p$,

1. the rules in the TSS do not allow one to simulate 'unguarded recursion' for $p$,
2. only finitely many rules can be employed to derive transitions from $p$, and
3. each rule can only be used to infer finitely many transitions from $p$.

The third property is checkable for each rule in isolation and is embodied in the requirement that the TSS be in bounded nondeterminism format (see Definition 16 to follow). On the other hand, the first and the second properties are 'global' and need to be checked for sets of rules. The existence of a strict stratification (see Definition 17) enforces the first property, while the second is guaranteed by the requirement that the TSS be bounded (see Definition 15 below). In order to define the notion of bounded TSS, Fokkink and Vu classify the transition rules in a TSS according to their so-called $\eta$-types. Intuitively, rules having the same $\eta$-type are those that could potentially be used to derive transitions from a closed term $p$ that unifies with the source of the rules. The requirement that the TSS be uniform and that the $\eta$-types be finitely inhabited ensures therefore that only finitely many rules can be employed to derive transitions from $p$.

We now adapt the definition of $\eta$-types in [9], on which the notion of bounded TSS is based, and recall the bounded nondeterminism format and the notion of strict stratification in [9].

We let $\eta : \mathbb{T}(\Sigma) \to \mathcal{P}(\mathbb{T}(\Sigma))$ be the maps that parametrise the $\eta$-types of Definition 14 to follow. The maps $\eta$ deliver, for a given term $t$, a predefined set of sources of positive premisses in rules that have source $t$. We say that $\eta(t)$ is the support of the sources for source $t$.[1]

**Definition 14 ($\eta$-type).** *Let $R$ be a TSS, $\rho \in R$ a rule with source $t$ and positive premisses $\{t_i \xrightarrow{a_i} t_i' \mid i \in I\}$, and $\eta$ a map with type $\mathbb{T}(\Sigma) \to \mathcal{P}(\mathbb{T}(\Sigma))$. We define $\psi : \eta(t) \to \mathcal{P}(A)$ as the map that delivers, for each term $u$ in the support of the sources for $t$, i.e., $u \in \eta(t)$, the actions of the positive premisses of $\rho$ with source $u$. More formally,*

$$\psi(u) = \{a_i \mid i \in I \ \wedge \ t_i = u\}.$$

*The tuple $\langle t, \psi \rangle$ is said to be the $\eta$-type of rule $\rho$.*

Differently from [9], our definition of $\eta$-type does not require that each set in the codomain of $\psi$ be finite. This requirement is not necessary for the rule format to ensure finite branching, as we explain in Remark 1 to Theorem 1.

The $\eta$-types distinguish rules based on their source and on the set of actions of their positive premisses whose source belongs to the predefined set specified by the map $\eta$. For instance, all the rules without positive premisses that have the same source belong to the same $\eta$-type, regardless of their action and target.

Intuitively, as mentioned above, rules that have the same $\eta$-type might all be used to derive transitions from a closed instantiation of their source. As the following example indicates, the presence of infinitely many rules with the same $\eta$-type might yield infinite branching.

*Example 2.* Let $A$ be an infinite set of actions and $\Sigma = \{c\}$. Consider the TSS

$$\frac{}{c \xrightarrow{a} c}\,, \qquad a \in A.$$

[1] We beg the reader to bear with us in the repetition of 'sources' and 'source' in sentences like the above. The 'sources' refers to the positive premisses and the 'source' to the conclusion of the rule.

All the infinitely many instantiations of the rule template above have $\eta$-type $\langle c, \psi \rangle$, where $\psi$ maps each term in $\eta(c)$ (if any) to the empty set. Note that $c$ is not finite branching.

**Definition 15 (Bounded).** *A TSS $R$ is bounded iff $R$ is uniform and there exists $\eta$ with codomain $\mathcal{P}_\omega(\mathbb{T}(\Sigma))$ (i.e., the set $\eta(t)$ is finite for each $t$) such that for every rule $\rho \in R$ with $\eta$-type $\langle t, \psi \rangle$, the $\eta$-type $\langle t, \psi \rangle$ is finitely inhabited.*

The requirement that the function $\eta$ have codomain $\mathcal{P}_\omega(\mathbb{T}(\Sigma))$ in Definition 15 means that in a bounded TSS only a finite support of the sources for a source can be distinguished. Consider Example 5 on page 508 of [9], which we reproduce next.

*Example 3.* Let $A$ be an infinite set of actions and $\Sigma$ consist of constants $A \cup \{c\}$ where $c \notin A$. Consider the TSS

$$\frac{}{a \xrightarrow{a} a}\,, \qquad a \in A \qquad\qquad \frac{a \xrightarrow{a} y}{c \xrightarrow{a} y}\,, \qquad a \in A.$$

If we allowed $\eta$ to have codomain $\mathcal{P}(\mathbb{T}(\Sigma))$, e.g., $\eta(a) = \emptyset$ (with $a \in A$) and $\eta(c) = A$, then it would be possible to distinguish the infinite support of the sources in the rule template on the right, and each $\eta$-type $\langle c, \psi_a \rangle$ (with $a \in A$), where $\psi_a(a) = \{a\}$ and $\psi_a(b) = \emptyset$ for $b \neq a$, would correspond to exactly one rule. If instead we require $\eta$ to have codomain $\mathcal{P}_\omega(\mathbb{T}(\Sigma))$, e.g., $\eta(a) = \emptyset$ with $a \in A$ and $\eta(c) = B$ for some $B \in \mathcal{P}_\omega(A)$, then an infinite number of sources of premisses $a \in A \setminus B$ will be excluded from the support for source $c$, i.e., $\eta(c) \cap (A \setminus B) = \emptyset$. The sources $a \in A \setminus B$ cannot be distinguished, and thus the infinitely many instantiations of the rule template on the right with sources of premisses $a \in A \setminus B$ will have the same $\eta$-type $\langle c, \psi \rangle$ where $\psi(t) = \emptyset$ with $t \in B$. Therefore, the TSS is not bounded. Notice that $c$ is not finite branching.

**Definition 16 (Bounded nondeterminism format).** *A rule*

$$\frac{\{u_i \xrightarrow{b_i} u_i' \mid i \in I\}}{t \xrightarrow{a} t'}$$

*is in bounded nondeterminism format iff*

1. $\mathrm{var}(u_i) \subseteq \mathrm{var}(t)$ *for each $i \in I$, that is, all the variables occurring in the source of its positive premisses also occur in its source, and*
2. $\mathrm{var}(t') \subseteq \mathrm{var}(t) \cup \bigcup\{\mathrm{var}(u_i') \mid i \in I\}$*, that is, all the variables occurring in its target also occur in its source, or in the target of some of its positive premisses.*

*A TSS $R$ is in bounded nondeterminism format iff every rule in $R$ is in bounded nondeterminism format.*

The bounded nondeterminism format enforces that the target of a transition ultimately comes from the source, i.e., the rules cannot introduce variables spuriously. The following example illustrates this fact.

*Example 4.* Let $\Sigma$ consist of a constant $c$ and a binary function symbol $f$, and let $A = \{a\}$. Consider the TSS

$$\frac{}{c \xrightarrow{a} c} \qquad\qquad \frac{x \xrightarrow{a} z}{f(x,y) \xrightarrow{a} f(z,y)} \,.$$

The TSS is in bounded nondeterminism format. Note that variable $z$ in the premiss of the rule on the right comes neither from the source of the premiss nor from the source of the rule. However, in every application of that rule in proof trees allowing one to derive transitions from closed terms of the form $f(p,q)$, the variable $z$ will always be instantiated to some closed term $p'$ such that $p \xrightarrow{a} p'$. Therefore, the rule does not introduce variables spuriously.

On the other hand, consider the rule

$$\frac{}{f(x,y) \xrightarrow{a} z} \,.$$

Such a rule is not in bounded nondeterminism format because the variable $z$ in the target of the rule does not appear in its source. The above rule can be used to prove transitions of the form $f(p,q) \xrightarrow{a} r$ for all closed terms $p$, $q$ and $r$, so the target $r$ of a transition does not necessarily stand for a process that can be reached from either $p$ or $q$. Therefore, the rule introduces the variable $z$ spuriously.

**Definition 17 (Strict stratification).** *Let $R$ be a TSS. A strict stratification of $R$ consists of a map $S$ from closed terms $T(\Sigma)$ to ordinal numbers such that for every transition rule $H/t \xrightarrow{a} t' \in R$ and for every closed substitution $\sigma$, $S(\sigma(u)) < S(\sigma(t))$ for every $u \xrightarrow{b} u' \in H$.*

The conditions of Theorem 1 on page 509 of [9] define the rule format for finite branching. We paraphrase Theorem 1 of [9] and its proof, and remove the *reductio ad absurdum* argument that is used there, providing a direct and fully constructive proof.

**Theorem 1 (Theorem 1 of [9]).** *Let $R$ be a bounded TSS in bounded nondeterminism format that has a strict stratification $S$. The LTS associated with $R$ is finite branching.*

*Proof.* We prove that each closed term $p$ in the LTS associated with $R$ is finite branching. Since $R$ is uniform, for a given $p$ there are only finitely many distinct terms $t_i$ and substitutions $\sigma_i : \mathrm{var}(t_i) \to T(\Sigma)$ (i.e., the $i$ ranges over a finite index set $I$) such that $\sigma_i(t_i) = p$ and the rules that unify with transitions from $p$ have some $t_i$ as source. We proceed by induction on $S(p)$.

The initial case is when $S(p) = 0$. Since $S(\sigma_i(t_i)) = 0$, the rules with source $t_i$ are axioms of the form

$$\frac{}{t_i \xrightarrow{a_j} t'_j} \,, \qquad i \in I,\ j \in J_i$$

where the $J_i$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is bounded, there exists $\eta$ such that for each $i$ and for each $j \in J_i$ the instantiation of the rule template above has $\eta$-type $\langle t_i, \psi_j \rangle$, and $\langle t_i, \psi_j \rangle$ is finitely inhabited. By Definition 14, all the $\psi_j$ map each term in $\eta(t_i)$ (if any) to the empty set. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(t'_j) \subseteq \mathrm{var}(t_i)$, and thus the $\sigma_i(t'_j)$ are closed. Since the rules above are axioms, the transitions $\sigma_i(t_i) \xrightarrow{a_j} \sigma_i(t'_j)$ are provable in $R$. Since all the $\psi_j$ in the $\eta$-types $\langle t_i, \psi_j \rangle$ with $j \in J_i$ are equal, and since the $\eta$-types are finitely inhabited, then the $J_i$ are finite. Therefore, for each $i \in I$ the set

$$\{(a_j, \sigma_i(t'_j)) \mid \sigma_i(t_i) \xrightarrow{a_j} \sigma_i(t'_j)\}$$

is finite. By the finiteness of $I$ it follows that the set $\{(a, p') \mid p \xrightarrow{a} p'\}$ is finite and we are done.

The general case is when $S(p) > 0$. The rules with source $t_i$ such that $\sigma_i(t_i) = p$ are of the form

$$\frac{\{u_k \xrightarrow{b_k} u'_k \mid k \in K_j\}}{t_i \xrightarrow{a_j} t'_j}\ , \qquad i \in I,\ j \in J_i$$

where the $J_i$ and the $K_j$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is bounded, there exists $\eta$ such that for each $i$ and for each $j \in J_i$, the instantiation of the rule template above has $\eta$-type $\langle t_i, \psi_j \rangle$, the set $\eta(t_i)$ is finite, and $\langle t_i, \psi_j \rangle$ is finitely inhabited.

For each $i$, we show that there are only finitely many distinct $\psi_j$ with $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$. By Definition 14, each rule of $\eta$-type $\langle t_i, \psi_j \rangle$ contains a premiss of the form $v \xrightarrow{c} v'$ for each $v \in \eta(t_i)$ and each $c \in \psi_j(v)$. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(v) \subseteq \mathrm{var}(t_i)$, and thus the $\sigma_i(v)$ are closed. By Definitions 7 and 8, for each transition in the node of a proof tree, if the transition unifies with a rule of $\eta$-type $\langle t_i, \psi_j \rangle$ then for each $v \in \eta(t_i)$ the process $\sigma_i(v)$ can perform, at least, a $c$-transition for each $c \in \psi_j(v)$. The $\psi_j$ giving rise to transitions from $\sigma_i(t_i)$ are dependent functions of type $\Pi_{v \in \eta(t_i)}\{c \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ with substitutions $\tau : (\mathrm{var}(v') \setminus \mathrm{var}(v)) \to T(\Sigma)$. For each $i$ the refined type of the $\psi_j$ with $j \in J_i$ is finitely inhabited, since the codomain of a dependent function depends on the inputs of the function. Each image of $\psi_j$ cannot be an arbitrary subset of $A$, but only the one that is determined by the input $v$ and by the associated LTS. That is, the only elements in the codomain of $\psi_j$ are the sets $\{c \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ where $v \in \eta(t_i)$. Since the $\eta(t_i)$ are finite sets, both the domain and the codomain of $\psi_j$ are finite. Therefore, for each $i$ there are only finitely many distinct $\psi_j$ with $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$.

Since $R$ is in bounded nondeterminism format, $\mathrm{var}(u_k) \subseteq \mathrm{var}(t_i)$ and therefore the $\sigma_i(u_k)$ are closed terms. As $S$ is a strict stratification, $S(\sigma_i(u_k)) < S(p)$. By the induction hypothesis the $\sigma_i(u_k)$ are finite branching, and therefore for each $i \in I$ the set

$$\{(b_k, \tau_\ell\sigma_i(u'_k)) \mid \sigma_i(u_k) \xrightarrow{b_k} \tau_\ell\sigma_i(u'_k)\}$$

is finite, with $\tau_\ell : ((\bigcup_{k \in K_j} \mathrm{var}(u'_k)) \setminus \mathrm{var}(t_i)) \to T(\Sigma)$ closed substitutions where $\ell$ ranges over some index sets $L_j$ and where $j \in J_i$. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(t'_j) \subseteq (\mathrm{var}(t_i) \cup (\bigcup_{k \in K_j} \mathrm{var}(u'_k)))$ and therefore the $\tau_\ell \sigma_i(t'_j)$ are closed terms. Since for each $i$ there are only finitely many distinct $\psi_j$ with $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$, and since the $\eta$-types $\langle t_i, \psi_j \rangle$ are finitely inhabited, then the $L_j$ are finite. Therefore, for each $i \in I$ the set

$$\{(a_j, \tau_\ell \sigma_i(t'_j)) \mid \sigma_i(t_i) \xrightarrow{a_j} \tau_\ell \sigma_i(t'_j)\} \quad (j \in J_i, \ell \in L_j)$$

is finite. By the finiteness of $I$ it follows that the set $\{(a, p') \mid p \xrightarrow{a} p'\}$ is finite and we are done. ☐

*Remark 1.* The requirement in [9] that each set in the codomain of $\psi$ in an $\eta$-type $\langle t, \psi \rangle$ must be finite (i.e., the codomain of $\psi$ must be $\mathcal{P}_\omega(A)$) is superfluous. In a TSS that induces a finite-branching LTS such that $\eta$ witnesses that the TSS is bounded, there could be rules with $\eta$-type $\langle t, \psi \rangle$ where each set in the codomain of $\psi$ is infinite, but since the LTS is finite branching, the transitions in the nodes of a proof tree will never unify with these rules.

*Remark 2.* The proof above follows that of Theorem 1 in [9], with the most notable difference being that [9] uses a *reductio ad absurdum* argument to show that the distinct $\psi_j$ for a given $i$ are finitely many. The proof in [9] assumes that there exists $m \in I$ such that there are infinitely many $\psi_n$ with $n \in J_m$ such that rules with $\eta$-type $\langle t_m, \psi_n \rangle$ give rise to transitions from $\sigma_m(t_m)$, and then shows that this assumption contradicts the induction hypothesis.

We believe a direct proof is preferable over a proof by contradiction. Our proof not only establishes the desired conclusion above, but also the intermediate conclusion that the $\psi_j$ such that rules with $\eta$-type $\langle t_i, \psi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$ are dependent functions of type $\Pi_{v \in \eta(t_i)} \{c \mid \sigma_i(v) \xrightarrow{c} \tau \sigma_i(v')\}$ with substitutions $\tau : (\mathrm{var}(v') \setminus \mathrm{var}(v)) \to T(\Sigma)$. This is an interesting observation in its own right that could be used to draw further conclusions. Besides, our proof is fully constructive, and thus it is better suited for the purpose of mechanising it.

*Example 5.* Let $A$ consist of an action $a$. Consider a TSS whose signature contains the constants $c_i$, with $i \geq 1$, and whose rules are

$$\frac{x \xrightarrow{a} y \qquad y \xrightarrow{a} z}{x \xrightarrow{a} z} \qquad\qquad \frac{}{c_i \xrightarrow{a} c_{i+1}}\ , \qquad i \geq 1.$$

This TSS is neither in bounded nondeterminism format nor strictly stratified, and therefore does not satisfy the conditions of Theorem 1. It is easy to see that every constant $c_i$ ($i \geq 1$) has infinitely many outgoing transitions. Indeed, $c_i \xrightarrow{a} c_j$ is provable for all $j > i \geq 1$.

Several examples of applications of the rule format defined by the conditions of Theorem 1 can be found in [9]. In the next section we adapt the conditions of Theorem 1 to account for initials finiteness.

## 4 Initials finiteness

As shown by Example 4 in Section 3, the bounded nondeterminism format enforces that no variables are introduced spuriously, thus preventing infinite branching coming from replacing the variables in the target of a rule by infinitely many distinct terms. In a transition rule, there are three kinds of 'variable flow' that it is worth considering:

1. variables from the source of the rule that flow to the sources of the positive premisses,
2. variables from the source of the rule that flow to the target of the rule, and
3. variables from the targets of positive premisses that flow to the target of the rule.

By the bounded nondeterminism format, all the variables in a rule (except for the variables in the source of the rule and in the targets of positive premisses) come from some of the variable flows described above. By induction on the proof tree, it is easy to show that the 'circulation' of the variables is closed in the leaves of the proof tree (i.e., by the second kind of variable flow above) and thus no variables can be introduced spuriously. This requirement is too strong for initials finiteness, which is only concerned with the *actions* of transitions that are provable from a given process. For initials finiteness it is immaterial whether the rules introduce variables in the target spuriously, and the bounded nondeterminism format can be relaxed. However, as the following example shows, dropping all the requirements on the variable flow does not ensure initials finiteness.

*Example 6.* Let $A$ be an infinite set of actions and let $\Sigma = A \cup \{c, f\}$ with $c$ a constant, $f$ a unary function symbol and $f, c \notin A$. Consider the TSS

$$\frac{}{f(a) \xrightarrow{a} f(a)}\,, \qquad a \in A \qquad\qquad \frac{f(x) \xrightarrow{a} y}{c \xrightarrow{a} y}\,, \qquad a \in A.$$

The TSS is uniform and has a strict stratification given by

$$S(c) = 1$$
$$S(f(p)) = 0.$$

We let $\eta(f(a)) = \emptyset$ (with $a \in A$) and $\eta(c) = \{f(x)\}$. For each $a \in A$, the instantiation of the rule template on the left has $\eta$-type $\langle f(a), \emptyset\rangle$, and the instantiation of the rule on the right has $\eta$-type $\langle c, \psi_a\rangle$ where $\psi_a(f(x)) = \{a\}$. However, the associated LTS is not initials finite because the set of initials of $c$ is $A$.

Variable $x$ in the rule template on the right does not come from the source of the rule. Thus, there exist infinitely many substitutions $\tau : \{x\} \to A$ such that the transitions from $\tau(f(x))$ unify with some instantiation of the rule template on the left. For initials finiteness, it is enough to prevent spurious variables in the sources of positive premisses.

We now introduce the initials finite format, which takes care of the first kind of variable flow described above.

**Definition 18 (Initials finite format).** *A rule*

$$\frac{\{u_i \xrightarrow{b_i} u_i' \mid i \in I\}}{t \xrightarrow{a} t'}$$

*is in initials finite format iff all the variables occurring in the sources of its positive premisses also occur in its source, that is,* $\mathrm{var}(u_i) \subseteq \mathrm{var}(t)$ *for each* $i \in I$*.*

*A TSS* $R$ *is in initials finite format iff every rule in* $R$ *is in initials finite format.*

The following example shows that the requirements on the variable flow, except for the one enforced by the initials finite format, can be dropped.

*Example 7.* Let $A = \{a\}$ and $\Sigma$ consists of infinitely many constants $\{c, d, \ldots\}$. Consider the TSS with rule

$$\frac{}{c \xrightarrow{a} x} \ .$$

The system is uniform and has a trivial strict stratification. The rule above has $\eta$-type $\langle c, \psi\rangle$ where $\psi$ maps each term in $\eta(c)$ (if any) to the empty set, and thus the TSS is bounded. Variable $x$ comes neither from the target of any positive premiss, since there are none, nor from the source of the rule, and hence the TSS is not in bounded nondeterminism format. However, the TSS is in initials finite format. Notice that the associated LTS is initials finite.

However, replacing the bounded nondeterminism format by the initials finite format is not enough to cover all the TSSs in which we are interested. Some initials-finite LTSs are induced by TSSs which are not bounded, despite being in initials finite format. This is shown in the following example.

*Example 8.* Let $A = \{a\}$ and let $\Sigma$ consist of infinitely many constants $\{c, d, \ldots\}$. Let $P = \{p_i \mid i \in I\}$ (with the $p_i$ distinct and $I$ an infinite index set) be a proper subset of $T(\Sigma)$, i.e., $P \subset T(\Sigma)$. Consider the TSS with rules

$$\frac{}{c \xrightarrow{a} p_i} \ , \qquad i \in I.$$

All the rules above have $\eta$-type $\langle c, \psi\rangle$ where $\psi$ maps each term in $\eta(c)$ (if any) to the empty set, and hence the $\eta$-type $\langle c, \psi\rangle$ is infinitely inhabited and the TSS is not bounded. However, the associated LTS is initials finite.

Example 8 implements *bounded quantifiers* by means of a rule template and an *ad hoc* infinite index set $I$. The use of bounded quantifiers[2] is different from

[2] Notice that ‘bounded’ in ‘bounded quantifiers’ does not have the connotation of ‘finite’ that is present in ‘bounded nondeterminism’. The bounded quantifiers restrict the range of the quantified variable, but this range could still be infinite. Examples 7 and 8 illustrate the difference between universal quantifiers and bounded quantifiers with an infinite range.

the implicit universal quantifiers for variables in the rules of a TSS, as illustrated in Example 7. The TSS of Example 7 consists of a single rule whose target $x$ ranges over the set of closed terms $T(\Sigma)$. On the contrary, the TSS of Example 8 consists of a rule template such that the targets $p_i$ with $i \in I$ range over an infinite proper subset of the set of closed terms, i.e., $\{p_i \mid i \in I\} \subset T(\Sigma)$. Technically, the sentences $\forall x.\ c \xrightarrow{a} x$ and $\forall x \in \{p_i \mid i \in I\}.\ c \xrightarrow{a} x$ are respectively a $\Pi_1$-sentence and a $\Pi_0$-sentence in the Lévy hierarchy [18].

Bounded quantifiers are conventional and useful, and we wish our rule format to allow for TSSs like the one of Example 8. To this end, we need a more refined notion of bounded TSS, which disregards the cardinality of the set of inhabitants of an $\eta$-type and takes into account the actions of rules.

We now define the actions of an $\eta$-type and introduce the notion of initials-bounded TSS.

**Definition 19 (Actions of an $\eta$-type).** *Let $R$ be a TSS. We define $\chi$ : $\eta$-type $\rightarrow \mathcal{P}(A)$ as the map that delivers, for each $\eta$-type $\langle t, \psi \rangle$, the set of actions of the rules that have $\eta$-type $\langle t, \psi \rangle$. More formally,*

$$\chi(t, \psi) = \{a \mid \rho \text{ has } \eta\text{-type } \langle t, \psi \rangle \text{ and } a \text{ is the action of } \rho\}.$$

*The set $\chi(t, \psi)$ is said to be the actions of $\eta$-type $\langle t, \psi \rangle$.*

**Definition 20 (Initials bounded).** *A TSS $R$ is initials bounded iff $R$ is uniform and there exists $\eta$ with codomain $\mathcal{P}_\omega(\mathbb{T}(\Sigma))$ (i.e., the set $\eta(t)$ is finite for each $t$) such that for every rule $\rho \in R$ with $\eta$-type $\langle t, \psi \rangle$, the $\eta$-type $\langle t, \psi \rangle$ has finitely many actions, i.e., $\chi(t, \psi) \in \mathcal{P}_\omega(A)$.*

In the TSS of Example 8, the $\eta$-type $\langle c, \psi \rangle$ is infinitely inhabited but it has finitely many actions as $\chi(c, \psi) = \{a\}$. Therefore, the TSS of Example 8 is initials bounded.

Intuitively, since rules having the same $\eta$-type are those that could potentially be used to derive transitions from a closed term $p$ that unifies with the source of the rules, requiring that $\eta$-types have finitely many actions can help one to ensure that $p$ be initials finite. However, as the following example shows, having a strict stratification is also needed to ensure initials finiteness as it intuitively disallows ‘unguarded recursion’.

*Example 9.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and let $A = \{a_1, a_2, \ldots\}$ be an infinite set of actions. Consider the TSS with rules

$$\frac{}{f(x) \xrightarrow{a_1} c} \qquad\qquad \frac{f(x) \xrightarrow{a_i} y}{f(x) \xrightarrow{a_{i+1}} y}\,, \qquad i \in \mathbb{N}.$$

The TSS is uniform and in initials finite format. We let $\eta(f(x)) = \{f(x)\}$. The rule on the left has $\eta$-type $\langle f(x), \psi \rangle$ where $\psi(f(x)) = \emptyset$, and for each $i \in \mathbb{N}$, the instantiation of the rule template on the right has $\eta$-type $\langle f(x), \psi_i \rangle$ where $\psi_i(f(x)) = \{a_i\}$. The set of actions of $\langle f(x), \psi \rangle$ is $\{a_1\}$, and for each $i \in \mathbb{N}$

the set of actions of $\langle f(x), \psi_i \rangle$ is $\{a_{i+1}\}$. Therefore, the TSS is initials bounded. However, the TSS does not have a strict stratification. Notice that the associated LTS is not initials finite, since the set of initials of $f(c)$ is $A$.

The conditions of the following theorem define the rule format for initials finiteness.

**Theorem 2.** *Let $R$ be an initials-bounded TSS in initials finite format that has a strict stratification $S$. The LTS associated with $R$ is initials finite.*

*Proof.* We prove that each closed term $p$ in the LTS associated with $R$ is initials finite. Since $R$ is uniform, for a given $p$ there are only finitely many distinct terms $t_i$ and substitutions $\sigma_i : \mathrm{var}(t_i) \to T(\Sigma)$ (i.e., the $i$ ranges over a finite index set $I$) such that $\sigma_i(t_i) = p$ and the rules that unify with transitions from $p$ have some $t_i$ as source. We proceed by induction on $S(p)$.

The initial case is when $S(p) = 0$. Since $S(\sigma_i(t_i)) = 0$, the rules with source $t_i$ are axioms of the form

$$\frac{}{t_i \xrightarrow{a_j} t'_k}\ , \qquad i \in I,\ j \in J_i,\ k \in K_j$$

where the $J_i$ and the $K_j$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is initials bounded, there exists $\eta$ such that for each $i$, for each $j \in J_i$, and for each $k \in K_j$, the instantiation of the rule template above has $\eta$-type $\langle t_i, \psi_k \rangle$, and $\langle t_i, \psi_k \rangle$ has finitely many actions, i.e., $\chi(t_i, \psi_k) \in \mathcal{P}_\omega(A)$. By Definition 14, all the $\psi_k$ map each term in $\eta(t_i)$ (if any) to the empty set, so all the rules above with source $t_i$ have the same $\eta$-type. Since the rules above are axioms, the transitions $\sigma_i(t_i) \xrightarrow{a_j} \tau\sigma_i(t'_k)$ are provable in $R$ for each substitution $\tau : (\mathrm{var}(t'_j) \setminus \mathrm{var}(t_i)) \to T(\Sigma)$. Since all the $\psi_k$ in the $\eta$-types are equal, and since the $\eta$-types have finitely many actions, the sets $J_i$ are finite. Therefore, for each $i \in I$ the set

$$\{a_j \mid \exists p' \text{ s.t. } \sigma_i(t_i) \xrightarrow{a_j} p'\} = \chi(t_i, \psi_k)$$

is finite. By the finiteness of $I$ it follows that the set $\{a \mid \exists p' \text{ s.t. } p \xrightarrow{a} p'\}$ is finite and we are done.

The general case is when $S(p) > 0$. The rules with source $t_i$ such that $\sigma_i(t_i) = p$ are of the form

$$\frac{\{u_\ell \xrightarrow{b_\ell} u'_\ell \mid \ell \in L_k\}}{t_i \xrightarrow{a_j} t'_k}\ , \qquad i \in I,\ j \in J_i,\ k \in K_j$$

where the $J_i$, the $K_j$, and the $L_k$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is initials bounded, there exists $\eta$ such that for each $i$, for each $j \in J_i$, and for each $k \in K_j$, the instantiation of the rule template above has $\eta$-type $\langle t_i, \psi_k \rangle$, the set $\eta(t_i)$ is finite, and $\langle t_i, \psi_k \rangle$ has finitely many actions.

For each $i$, we show that there are only finitely many distinct $\psi_k$ with $k \in K_j$ and $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_k \rangle$ give rise to transitions from $\sigma_i(t_i)$. By Definition 14, each rule of $\eta$-type $\langle t_i, \psi_k \rangle$ contains a premiss of the form $v \xrightarrow{c} v'$ for each $v \in \eta(t_i)$ and each $c \in \psi_k(v)$. Since $R$ is in initials finite format, $\mathrm{var}(v) \subseteq \mathrm{var}(t_i)$, and thus the $\sigma_i(v)$ are closed. By Definitions 7 and 8, for each transition in the node of a proof tree, if the transition unifies with a rule of $\eta$-type $\langle t_i, \psi_k \rangle$, then for each $v \in \eta(t_i)$ the processes $\sigma_i(v)$ can perform, at least, a $c$-transition for each $c \in \psi_k(v)$. The $\psi_k$ giving rise to transitions from $\sigma_i(t_i)$ are dependent functions of type $\Pi_{v \in \eta(t_i)}\{c \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ with substitutions $\tau : (\mathrm{var}(v') \setminus \mathrm{var}(v)) \to T(\Sigma)$. For each $i$ the refined type of the $\psi_k$ with $k \in K_j$ and $i \in J_i$ is finitely inhabited, since the codomain of a dependent function depends on the inputs of the function. Each image of $\psi_k$ cannot be an arbitrary subset of $A$, but only the one determined by the input $v$ and by the associated LTS. That is, the only elements in the codomain of $\psi_k$ are the sets $\{c \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ where $v \in \eta(t_i)$. Since the $\eta(t_i)$ are finite sets, both the domain and the codomain of $\psi_k$ are finite. Therefore for each $i$ there are only finitely many distinct $\psi_k$ with $k \in K_j$ and $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_k \rangle$ give rise to transitions from $\sigma_i(t_i)$.

Since for each $i$ there are only finitely many distinct $\psi_k$ with $k \in K_j$ and $j \in J_i$ such that rules with $\eta$-type $\langle t_i, \psi_k \rangle$ give rise to transitions from $\sigma_i(t_i)$, and since the $\eta$-types $\langle t_i, \psi_k \rangle$ have finitely many actions, then for each $i \in I$ the set

$$\{a_j \mid \exists p' \text{ s.t. } \sigma_i(t_i) \xrightarrow{a_j} p'\}$$

is finite. By the finiteness of $I$ it follows that the set $\{a \mid \exists p' \text{ s.t. } p \xrightarrow{a} p'\}$ is finite and we are done. $\square$

*Remark 3.* In Theorem 2 the TSS $R$ is not required to be in bounded nondeterminism format. The terms $t'_k$ may have variables which are neither in $t_i$ nor in $\bigcup_{\ell \in L_k} \mathrm{var}(u'_\ell)$. Consider $\tau_m : (\mathrm{var}(t'_k) \setminus \mathrm{var}(t_i)) \to T(\Sigma)$ closed substitutions with $m$ ranging over index sets $M_k$ such that $\sigma_i(t_i) \xrightarrow{a_j} \tau_m\sigma_i(t'_k)$. For each $\sigma_i(t_i)$ there may be infinitely many transitions $\sigma_i(t_i) \xrightarrow{a_j} \tau_m\sigma_i(t'_k)$ because the $M_k$ may be infinite. This is illustrated by Example 7.

*Remark 4.* In Theorem 2 the $\eta$-types are not required to be finitely inhabited. For each $\eta$-type $\langle t_i, \psi_k \rangle$ there could be infinitely many rules with conclusions $t_i \xrightarrow{a_j} t'_k$, and the $K_j$ need not be finite. The TSS $R$ could be in bounded nondeterminism format, and then there would be $\tau_m : ((\bigcup_{\ell \in L_k} \mathrm{var}(u'_\ell)) \setminus \mathrm{var}(t_i)) \to T(\Sigma)$ closed substitutions with $m \in M_k$ and $M_k$ are finite index sets such that $\sigma_i(t_i) \xrightarrow{a_j} \tau_m\sigma_i(t'_k)$. But, although the $M_k$ may be finite, for each $\sigma_i(t_i)$ there may be infinitely many transitions $\sigma_i(t_i) \xrightarrow{a_j} \tau_m\sigma_i(t'_k)$, because the $K_j$ may be infinite. This is illustrated by Example 8.

We now present an example of application of the rule format defined by the conditions of Theorem 2.

*Example 10.* Let $\Sigma$ contain constants $c$ and $\mathbf{0}$ and the unary action prefixing operation $a.\_$ from Milner's CCS [20]. Consider the TSS with rules

$$\frac{}{a.x \xrightarrow{a} x} \qquad \frac{}{c \xrightarrow{a} \underbrace{a.\ldots.a.}_{i \text{ times}}\mathbf{0}}\,, \qquad i \geq 0.$$

Intuitively, the constant $c$ is akin to a random assignment [4]. The TSS is uniform and has a trivial strict stratification. We let $\eta(a.x) = \emptyset$ and $\eta(c) = \emptyset$. The rule on the left has $\eta$-type $\langle a.x, \emptyset \rangle$ and each instantiation of the rule template on the right has $\eta$-type $\langle c, \emptyset \rangle$. The rule template on the right implements bounded quantifiers as illustrated in Example 8. Although the $\eta$-type $\langle c, \emptyset \rangle$ is infinitely inhabited, the set of its actions is $\{a\}$. The associated LTS is initials finite.

In the next section we develop a rule format for image finiteness.

## 5 Image finiteness

Consider the properties of an LTS in Definition 13, which we paraphrase here in mathematical notation:

**Finite branching:** $\forall p.\ \{(a, p') \mid p \xrightarrow{a} p'\} \in \mathcal{P}_\omega(A \times T(\Sigma))$.
**Initials finiteness:** $\forall p.\ \{a \mid \exists p'.\ p \xrightarrow{a} p'\} \in \mathcal{P}_\omega(A)$.
**Image finiteness:** $\forall p.\forall a.\ \{p' \mid p \xrightarrow{a} p'\} \in \mathcal{P}_\omega(T(\Sigma))$.

We consider the Brouwer-Heyting-Kolmogorov interpretation of intuitionistic logic (BHK interpretation for short) [15]. According to the BHK interpretation, the proof of any of the properties above consists of a function that takes one argument for each of the universally quantified symbols and returns a proof of the trailing proposition after the quantifiers,[3] which asserts that some set is finite. As a proof of each assertion, it is enough to exhibit the set in point. For example, given a TSS $R$ the proof that the LTS associated with $R$ is initials finite consists of a function that takes an element $p \in T(\Sigma)$ and delivers the finite set of actions $a$ such that $p \xrightarrow{a} p'$ (with $p' \in T(\Sigma)$) is provable in $R$. The BHK interpretation provides a profitable insight on the notion of $\eta$-types. In essence, the $\eta$-types are a sort of syntactic fingerprint of the BHK interpretation of initials finiteness. Recall from Definition 14 that in an $\eta$-type $\langle t, \psi \rangle$ the map $\psi$ takes a term and delivers a set of actions. This map represents the function corresponding to the BHK interpretation. The disciplined focus on the positives premisses (e.g., through the finite support of the sources defined by $\eta$ and with the variable flow enforced by the initials finite format) is only an instrument to construct the intuitionistic proof from $\psi$, by induction on the strict stratification of the TSS. This is exemplified by our proof of Theorem 2.

It may seem odd that the $\eta$-types, which correspond to the BHK interpretation of initials finiteness, are also used in the rule format that ensures finite branching. The map $\psi$ delivers a set of *actions* $b$, instead of the set of *pairs of actions and terms* $(b, u')$ that would be expected from the BHK interpretation of finite branching. The reason lies in the fact that the additional requirements of the rule format make keeping track of the targets $u'$ redundant. To see this, let the positive premisses be of the shape $u \xrightarrow{b} u'$. Since the TSS is required to be in

[3] Recall that in intuitionistic logic a universal quantifier '$\forall x.$' is akin to a big lambda '$\Lambda x.$', i.e., a binding operator at the level of types.

bounded nondeterminism format, then $\mathrm{var}(u) \subseteq \mathrm{var}(t)$ and the $\sigma(u)$ are closed. Thus, for each $u \in \eta(t)$ there are at most finitely many pairs $(b, u')$ such that $b \in \psi(u)$ and $\sigma(u) \xrightarrow{b} \tau\sigma(u')$ with substitutions $\tau : (\mathrm{var}(u') \setminus \mathrm{var}(u)) \to T(\Sigma)$. Therefore, keeping track of the targets $u'$ of positive premisses is redundant because the requirements of the rule format ensure that the associated LTS is finite branching.

The different requirements for bounded TSS and for initials-bounded TSS complete the picture, respectively for the BHK interpretation of finite branching and of initials finiteness. In a bounded TSS, it is required that there exists an $\eta$ such that each $\eta$-type $\langle t, \psi \rangle$ is *finitely inhabited*. This enforces that if for each term $t$ and for each substitution $\sigma$ such that $\sigma(t)$ is closed there are only finitely many $\psi$ such that the rules that give rise to transitions from $\sigma(t)$ have $\eta$-type $\langle t, \psi \rangle$, then the set of pairs $(a, t')$ such that $\sigma(t) \xrightarrow{a} \tau\sigma(t')$ with substitutions $\tau : (\mathrm{var}(t') \setminus \mathrm{var}(t)) \to T(\Sigma)$ is finite. The bounded nondeterminism format ensures that there are only finitely many substitutions $\tau$, and then the set of pairs $(a, \tau\sigma(t'))$ is also finite and the associated LTS is finite branching. In an initials-bounded TSS, it is only required that there exists an $\eta$ such that each $\eta$-type $\langle t, \psi \rangle$ has *finite actions*. This enforces that if for each term $t$ and for each substitution $\sigma$ such that $\sigma(t)$ is closed there are only finitely many $\psi$ such that the rules that give rise to transitions from $\sigma(t)$ have $\eta$-type $\langle t, \psi \rangle$, then the set of actions $a$ such that $\sigma(t) \xrightarrow{a} \tau\sigma(t')$ with substitutions $\tau : (\mathrm{var}(t') \setminus \mathrm{var}(t)) \to T(\Sigma)$ is finite. The bounded nondeterminism format can be replaced by the initials finite format because the number of substitutions $\tau$ is immaterial in order to keep the number of actions $a$ finite, and thus for the associated LTS to be initials finite.

We now introduce the $\theta$-types, which are gleaned from the BHK interpretation of image finiteness. Unlike the $\eta$-types of [9], which carry information about the sources of positive premisses in rules, the $\theta$-types also keep track *of the actions* that label positive premisses.

We let $\theta : (\mathbb{T}(\Sigma) \times A) \to \mathcal{P}(\mathbb{T}(\Sigma) \times A)$ be the maps that parametrise the $\theta$-types of Definition 21 to follow. The maps $\theta$ deliver, for a given term $t$ and action $a$, a predefined set of sources and actions of positive premisses in rules that have source $t$ and action $a$. We say that $\theta(t, a)$ is the support of the sources and of the actions for source and action $(t, a)$.[4]

**Definition 21 ($\theta$-type).** *Let $R$ be a TSS, $\rho \in R$ a rule with source $t$, action $a$, and positive premisses $\{t_i \xrightarrow{a_i} t'_i \mid i \in I\}$, and $\theta$ a map with type $(\mathbb{T}(\Sigma) \times A) \to \mathcal{P}(\mathbb{T}(\Sigma) \times A)$. We define $\phi : \theta(t, a) \to \mathcal{P}(\mathbb{T}(\Sigma))$ as the map that delivers, for each term $u$ and action $b$ in the support of the sources and of the actions for $(t, a)$, i.e., $(u, b) \in \theta(t, a)$, the targets of the positive premisses of $\rho$ with source $u$ and action $b$. More formally,*

$$\phi(u, b) = \{t'_i \mid i \in I \ \wedge \ t_i = u \ \wedge a_i = b\}.$$

[4] We beg the reader to bear with us in the repetition of 'sources', 'actions', 'source', and 'action' in sentences like the above. The 'sources' and 'actions' refer to the positive premisses, and the 'source' and 'action' to the conclusion of the rule.

*The triple $\langle t, a, \phi\rangle$ is said to be the $\theta$-type of rule $\rho$.*

The $\theta$-types distinguish rules based on their source, their action, and on the set of targets of their positive premisses whose source and action belong to the predefined set specified by the map $\theta$. Let us illustrate this with an example.

*Example 11.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$ and let $A$ be an infinite set of actions. Consider the TSS

$$\frac{}{c \xrightarrow{a} c}\ , \qquad a \in A \qquad\qquad \frac{x \xrightarrow{a} y}{f(x) \xrightarrow{a} y}\ , \qquad a \in A.$$

For each $a \in A$, we let $\theta(c, a) = \emptyset$ and $\theta(f(x), a) = \{(x, a)\}$. For each $a \in A$, the $\theta$-types of the instantiations of the rule templates on the left and on the right are $\langle c, a, \emptyset\rangle$ and $\langle f(x), a, \phi_a\rangle$ respectively, where $\phi_a(x, a) = \{y\}$. Notice that the associated LTS is image finite, because the target of every transition is $c$. However, it is neither finite branching nor initials finite, since every process can do an $a$-transition for each $a \in A$.

Intuitively, the $\theta$-types play for image finiteness the role that the $\eta$-types play for finite branching. Rules having the same $\theta$-type $\langle t, a, \phi\rangle$ are those that could potentially be used to derive $a$-transitions from a closed term $p$ that is an instantiation of $t$. In order to ensure that the set of processes that are the targets of $a$-transitions from a closed term $p$ is finite, it is reasonable to require that each $\eta$-type be finitely inhabited. However, for image finiteness, the variables occurring in the targets of positive premisses of rules have to be used uniformly. The following example illustrates this fact.

*Example 12.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and assume $A = \{a\}$. Let the $y_i$ with $i \in \mathbb{N}$ be infinitely many distinct variables. Consider the TSS

$$\frac{}{c \xrightarrow{a} c} \qquad\qquad \frac{x \xrightarrow{a} y_i}{f(x) \xrightarrow{a} f^i(x)}\ , \qquad i \in \mathbb{N}$$

where $f^i$ stands for applying $i$ times the function symbol $f$. The TSS is uniform in the sources (recall Definition 10) and has a strict stratification given by

$$S(c) = 0$$
$$S(f(p)) = 1 + S(p).$$

We let $\theta(c, a) = \emptyset$ and $\theta(f(x), a) = \{(x, a)\}$. The rule on the left has $\theta$-type $\langle c, a, \emptyset\rangle$, and for each $i \in \mathbb{N}$, the instantiation of the rule template on the right has $\theta$-type $\langle f(x), a, \phi_i\rangle$, where $\phi_i(f(x), a) = \{y_i\}$. However, the associated LTS is not image finite, because process $f(c)$ can perform infinitely many $a$-transitions to $f^i(c)$ (with $i \in \mathbb{N}$).

In the TSS of Example 12 there are infinitely many different variables $y_i$, and thus there are infinitely many different $\theta$-types that morally should be the same.

The inhabitants of each of these $\theta$-types give rise, for a given source and action, to transitions with different targets, and the associated LTS is not image finite. To address this issue we introduce the notion of *uniform TSS in the targets of positive premisses*. This notion extends that of *uniform TSS in the source*, which is the *uniform TSS* from [9] that we adapted in Definition 10.

**Definition 22 (Uniform in the targets of positive premisses).** *A TSS $R$ is uniform in the targets of positive premisses iff $t' = t''$ holds whenever $t'$ and $t''$ have the same structure and $t \xrightarrow{a} t'$ and $t \xrightarrow{a} t''$ are positive premisses of any two (not necessarily different) rules.*

The TSS of Example 12 is not uniform in the targets of positive premisses. Indeed, $x \xrightarrow{a} y_1$ and $x \xrightarrow{a} y_2$ are positive premisses of rules and $y_1$ and $y_2$ have the same structure, but $y_1 \neq y_2$. However, the LTS induced by the TSS of Example 12 can be specified by a TSS that is uniform in the targets of positive premisses as follows.

*Example 13.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and assume $A = \{a\}$. Consider the TSS

$$\frac{}{c \xrightarrow{a} c} \qquad\qquad \frac{x \xrightarrow{a} y}{f(x) \xrightarrow{a} f^i(x)}\,, \qquad i \in \mathbb{N}.$$

The TSS is uniform both in the sources of rules and in the targets of their positive premisses and has a strict stratification given by

$$S(c) = 0$$
$$S(f(p)) = 1 + S(p).$$

We let $\theta(c, a) = \emptyset$ and $\theta(f(x), a) = \{(x, a)\}$. The rule on the left has $\theta$-type $\langle c, a, \emptyset\rangle$, and for each $i \in \mathbb{N}$ the instantiation of the rule template on the right has $\theta$-type $\langle f(x), a, \phi\rangle$ where $\phi(x, a) = \{y\}$. Therefore, the $\theta$-type $\langle f(x), a, \phi\rangle$ is infinitely inhabited. Notice that the associated LTS is equal to that in Example 12, which is not image finite.

Next we prove a proposition that resembles Proposition 2 of Section 2, which states that for a uniform TSS in the targets of positive premisses, each transition is a substitution instance of at most finitely many positive premisses of the TSS.

**Proposition 3.** *Let $R$ be a uniform TSS in the targets of positive premisses. For each transition $p \xrightarrow{a} p'$, and for each term $t$ and substitution $\sigma : \mathrm{var}(t) \to T(\Sigma)$ such that $\sigma(t) = p$, the set of pairs $(t', \tau)$ with $\tau : \mathrm{var}(t') \setminus \mathrm{var}(t) \to T(\Sigma)$ such that $\sigma(t) \xrightarrow{a} \tau\sigma(t') = p \xrightarrow{a} p'$ and $t \xrightarrow{a} t'$ is a positive premiss of some rule in $R$ is finite.*

*Proof.* We prove the generalised proposition:

*Let $R$ be a uniform TSS in the targets of positive premisses. For each pair $(p \xrightarrow{a} p', r)$ where $p \xrightarrow{a} p'$ is a transition and $r$ is an approximant of $p'$ (i.e., $r \sqsubseteq p'$) and for each term $t$ and substitution $\sigma : \mathrm{var}(t) \to T(\Sigma)$ such that $\sigma(t) = p$, the set of pairs $(t', \tau)$ with $\tau : \mathrm{var}(t') \setminus \mathrm{var}(t) \to T(\Sigma)$ such that there exists $r'$ an approximant of $r$ (i.e., $r' \sqsubseteq r$) with the same structure as $t'$, and $\sigma(t) \xrightarrow{a} \tau\sigma(t') = p \xrightarrow{a} p'$ and $t \xrightarrow{a} t'$ is a positive premiss of some rule in $R$, is finite.*

The original proposition follows from the generalised proposition by fixing $r = p'$. We fix a $t$ and a $\sigma$ such that $\sigma(t) = p$ and construct the set $S$ of pairs $(t', \tau)$ that meet the conditions of the generalised proposition and show that $S$ is finite. We proceed by well-founded induction on the set of approximants of $r$ ordered by the less-defined-than relation ($\sqsubset$).

We first check that the generalised proposition holds for the $\sqsubset$-minimal partial terms in the set of approximants of $r$. The only such partial term is $r_0 = \bot$. There exists only one $r'_0$ an approximant of $r$ (i.e., $r'_0 = \bot \sqsubseteq \bot = r_0$) and the only terms that have the same structure as $r'_0$ are the variables. Since $R$ is uniform in the targets of positive premisses, all the positive premisses of $R$ with source $t$, action $a$, and whose target is a variable (if there is any) have the same variable $x$ as target. If there exist no positive premisses as described before, then the set we are looking for is the empty set. If there exist positive premisses as described before, we distinguish two cases. If $x \in \mathrm{var}(t)$, then the set we are looking for is $S = \{(x, \emptyset)\}$. Otherwise, the set we are looking for is $S = \{(x, \{x \mapsto p'\})\}$. All three sets are finite.

Now we check that the generalised proposition holds for an arbitrary partial term $r_a \neq \bot$ in the set of approximants of $r$. Notice that the partial term $r_a$ is such that $r_a \sqsubseteq r \sqsubseteq p'$. By the induction hypothesis, for every $r_x$ such that $r_x \sqsubset r_a$ the set $S_x$ of pairs $(t', \tau)$ such that there exists $r'_x$ an approximant of $r_x$ (i.e., $r'_x \sqsubseteq r_x$) with the same structure as $t'$, and $\sigma(t) \xrightarrow{a} \tau\sigma(t') = p \xrightarrow{a} p'$ and $t \xrightarrow{a} t'$ is a positive premiss of some rule in $R$ is finite. Since $R$ is uniform in the targets of positive premisses, all the positive premisses with source $t$, action $a$, and whose target has the same structure as $r'_x$ (if there is any) have the same target $u$. Since $r'_x \sqsubset r_a$, then $r'_x$ is obtained by replacing by $\bot$ one or more maximally disjoint subterms of $r_a$ that are different form $\bot$. We let $x_j$ (with $j$ ranging over some index set $J$) be the variables that occur in $u$ in the positions corresponding to the occurrences of $\bot$ in $r'_x$. (Notice that no other variables could occur in $u$, since $u$ has the same structure as $r'_x$, and $r'_x \sqsubset p'$.) We let $t_j$ be the terms such that $p'$ results from replacing respectively the $x_j$ by the $t_j$ in $u$. If there exist no rules with a positive premiss whose target is $u$, then the set we are looking for is $S_x$. Otherwise, the set we are looking for is $S = S_x \cup \{(u, \{x_j \mapsto t_j \mid j \in J \ \wedge \ x_j \notin \mathrm{var}(t)\})\}$, which is finite. □

The notion of image-bounded TSS, which we introduce next, collects the requirements that we have discussed so far.

**Definition 23 (Image bounded).** *A TSS $R$ is image bounded iff $R$ is uniform in the sources of rules and in the targets of their positive premisses, and there* *exists $\theta$ with codomain $\mathcal{P}_\omega(\mathbb{T}(\Sigma) \times A)$ (i.e., for each pair $(t, a)$ the set $\theta(t, a)$ is finite) such that for every rule $\rho \in R$ with $\theta$-type $\langle t, a, \phi \rangle$, the $\theta$-type $\langle t, a, \phi \rangle$ is finitely inhabited.*

For image finiteness, the restrictions on the variable flow have to be enforced again, and the bounded nondeterminism format is needed. Example 7 in Section 4 shows that the variables in the target of a rule have to occur in either the source of the rule, or in the targets of its positive premisses. The LTS induced by the TSS in Example 7 is not image finite because $c \xrightarrow{a} d$ holds for each of the infinitely many constants $d$. The following example is a variation on Example 6 in Section 4 that shows that for image finiteness, the variables in the sources of positive premisses have to occur in the source of the rules.

*Example 14.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and assume $A = \{a\}$. Consider the TSS

$$\frac{}{f(x) \xrightarrow{a} f(x)} \qquad\qquad \frac{f(x) \xrightarrow{a} y}{c \xrightarrow{a} y} \; .$$

The TSS is uniform both in the sources of rules and in the targets of their positive premisses and has a strict stratification given by

$$\begin{aligned} S(f(p)) &= 0 \\ S(c) &= 1. \end{aligned}$$

We let $\theta(f(x), a) = \emptyset$ and $\theta(c, a) = \{(f(x), a)\}$. The rule on the left has $\theta$-type $\langle f(x), a, \emptyset \rangle$, and the rule on the right has $\theta$-type $\langle c, a, \phi \rangle$ where $\phi(f(x), a) = \{y\}$, and thus the TSS is image bounded. However $c \xrightarrow{a} f(p)$ for every $p \in T(\Sigma)$ and thus the associated LTS is not image finite.

In an image-bounded TSS, it is required that there exists a $\theta$ such that each $\theta$-type $\langle t, a, \phi \rangle$ is finitely inhabited. This enforces that if for each term $t$, for each action $a$, and for each substitution $\sigma$ such that $\sigma(t)$ is closed there are only finitely many $\phi$ such that the rules that give rise to $a$-transitions from $\sigma(t)$ have $\theta$-type $\langle t, a, \phi \rangle$, then the set of targets $t'$ such that $\sigma(t) \xrightarrow{a} \tau\sigma(t')$ with substitutions $\tau : (\mathrm{var}(t') \setminus \mathrm{var}(t)) \to T(\Sigma)$ is finite. The bounded nondeterminism format ensures that only finitely many $\tau$ exist, and thus the set of targets $\tau\sigma(t')$ is also finite and the associated LTS is image finite.

The following example shows that having a strict stratification is needed to ensure image finiteness.

*Example 15.* Let $A$ consist of an action $a$, and let $\Sigma = A \cup \{f\}$ with $f$ a unary function symbol $f$. Consider the TSS

$$\frac{}{f(x) \xrightarrow{a} a} \qquad\qquad \frac{f(x) \xrightarrow{a} y}{f(x) \xrightarrow{a} f(y)} \; .$$

The TSS is uniform in both the sources of rules and the targets of their positive premisses, and it is in bounded nondeterminism format. We let $\theta(f(x), a) =$

$\{(f(x), a)\})$. The rule on the left has $\theta$-type $\langle f(x), a, \phi_1\rangle$ with $\phi_1(f(x), a) = \emptyset$. The rule on the right has $\theta$-type $\langle f(x), a, \phi_2\rangle$ with $\phi_2(f(x), a) = \{y\}$. The TSS is image bounded. However, the TSS does not have a strict stratification. Notice that the associated LTS is not image finite since $f(a)$ can perform an $a$-transition to each of the terms $f^i(a)$.

The conditions of the following theorem define the rule format for image finiteness.

**Theorem 3.** *Let $R$ be an image-bounded TSS that is in bounded nondeterminism format and has a strict stratification $S$. The LTS associated with $R$ is image finite.*

*Proof.* We prove that for each closed term $p$ and for each action $a$ in the LTS associated with $R$ the set $\{p' \mid p \xrightarrow{a} p'\}$ is finite. Since $R$ is uniform in the sources, for a given $p$ there are only finitely many distinct terms $t_i$ and substitutions $\sigma_i : \mathrm{var}(t_i) \to T(\Sigma)$ (i.e., the $i$ ranges over a finite index set $I$) such that $\sigma_i(t_i) = p$ and the rules that unify with transitions from $p$ have some $t_i$ as source. We proceed by induction on $S(p)$.

The initial case is when $S(p) = 0$. Since $S(\sigma_i(t_i)) = 0$, the rules with source $t_i$ and action $a$ are axioms of the form

$$\frac{}{t_i \xrightarrow{a} t'_j}\,, \qquad i \in I,\ j \in J_i$$

where the $J_i$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is image bounded, there exists $\theta$ such that for each $i$ and for each $j \in J_i$, the instantiation of the rule template above has $\theta$-type $\langle t_i, a, \phi_j\rangle$, and $\langle t_i, a, \phi_j\rangle$ is finitely inhabited. By Definition 21, all the $\phi_j$ map each pair in $\theta(t_i, a)$ (if any) to the empty set. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(t'_j) \subseteq \mathrm{var}(t_i)$, and thus the $\sigma_i(t'_j)$ are closed. Since the rules above are axioms, the transitions $\sigma_i(t_i) \xrightarrow{a} \sigma_i(t'_j)$ are provable in $R$. Since all the $\phi_j$ are equal, and since the $\theta$-types are finitely inhabited, then the $J_i$ are finite. Therefore, for each $i \in I$ the set

$$\{\sigma_i(t'_j) \mid \sigma_i(t_i) \xrightarrow{a} \sigma_i(t'_j)\}$$

is finite. By the finiteness of $I$ it follows that the set $\{p' \mid p \xrightarrow{a} p'\}$ is finite and we are done.

The general case is when $S(p) > 0$. The rules with action $a$ and source $t_i$ such that $\sigma_i(t_i) = p$ are of the form

$$\frac{\{u_k \xrightarrow{b_k} u'_k \mid k \in K_j\}}{t_i \xrightarrow{a} t'_j}\,, \qquad i \in I,\ j \in J_i$$

where the $J_i$ and the $K_j$ are taken to be disjoint to avoid proliferation of indices. Since $R$ is image bounded, there exists $\theta$ such that for each $i$ and for each $j \in J_i$, the instantiation of the rule template above has $\theta$-type $\langle t_i, a, \phi_j\rangle$, the set $\theta(t_i, a)$ is finite, and $\langle t_i, a, \phi_j\rangle$ is finitely inhabited.

For each $i$, we show that there are only finitely many distinct $\phi_j$ with $j \in J_i$ such that rules with $\theta$-type $\langle t_i, a, \phi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$. By Definition 21, each rule of $\theta$-type $\langle t_i, a, \phi_j \rangle$ contains a premiss of the form $v \xrightarrow{c} v'$ for each $(v, c) \in \theta(t_i, a)$ and each $v' \in \phi_j(v, c)$. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(v) \subseteq \mathrm{var}(t_i)$ and thus the $\sigma_i(v)$ are closed. By Definitions 7 and 8, for each transition in the node of a proof tree, if the transition unifies with a rule of $\theta$-type $\langle t_i, a, \phi_j \rangle$ then for each pair $(v, c) \in \theta(t_i, a)$ and for each $v' \in \phi_j(v, c)$ the process $\sigma_i(v)$ can perform, at least, a $c$-transition to $\tau\sigma_i(v')$ for some substitution $\tau : (\mathrm{var}(v') \setminus \mathrm{var}(v)) \to T(\Sigma)$. The $\phi_j$ giving rise to transitions from $\sigma_i(t_i)$ are dependent functions of type $\Pi_{(v,c)\in\theta(t_i,a)}\{v' \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ with substitutions $\tau : (\mathrm{var}(v') \setminus \mathrm{var}(v)) \to T(\Sigma)$. For each $i$ the refined type of the $\phi_j$ with $j \in J_i$ is finitely inhabited, since the codomain of a dependent function depends on the inputs of the function. Each image of $\phi_j$ cannot be an arbitrary subset of $\mathbb{T}(\Sigma)$, but only the one determined by the input $(v, c)$ and by the associated LTS. That is, the only elements in the codomain of $\phi_j$ are the sets $\{v' \mid \sigma_i(v) \xrightarrow{c} \tau\sigma_i(v')\}$ where $(v, c) \in \theta(t_i, a)$. Since the $\theta(t_i, a)$ are finite sets, both the domain and the codomain of $\phi_j$ are finite. Therefore, for each $i$, there are only finitely many distinct $\phi_j$ with $j \in J_i$ such that rules with $\theta$-type $\langle t_i, a, \phi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$.

Since $R$ is in bounded nondeterminism format, $\mathrm{var}(u_k) \subseteq \mathrm{var}(t_i)$ with $k \in K_j$ and $j \in J_i$, and therefore the $\sigma_i(u_k)$ are closed terms. Since $S$ is a strict stratification, $S(\sigma_i(u_k)) < S(p)$. By the induction hypothesis the $\sigma_i(u_k)$ are image finite, and for each $i$ and for each $b_k$ the set

$$\{\tau_\ell\sigma_i(u'_k) \mid \sigma_i(u_k) \xrightarrow{b_k} \tau_\ell\sigma_i(u'_k)\}$$

is finite, with $\tau_\ell : ((\bigcup_{k\in K_j} \mathrm{var}(u'_k)) \setminus \mathrm{var}(t_i)) \to T(\Sigma)$ closed substitutions where $\ell$ ranges over some index sets $L_j$. Since $R$ is uniform in the targets of positive premisses and by Proposition 3 the $L_j$ are finite. Since $R$ is in bounded nondeterminism format, $\mathrm{var}(t'_j) \subseteq (\mathrm{var}(t_i) \cup (\bigcup_{k\in K_j} \mathrm{var}(u'_k)))$ and therefore the $\tau_\ell\sigma_i(t'_j)$ are closed terms. Since for each $i$ there are only finitely many distinct $\phi_j$ with $j \in J_i$ such that rules with $\theta$-type $\langle t_i, a, \phi_j \rangle$ give rise to transitions from $\sigma_i(t_i)$, and since the $L_j$ are finite, then for each $i \in I$ the set

$$\{\tau_\ell\sigma_i(t'_j) \mid \sigma_i(t_i) \xrightarrow{a} \tau_\ell\sigma_i(t'_j)\}$$

is finite. By the finiteness of $I$ it follows that the set $\{p' \mid p \xrightarrow{a} p'\}$ is finite and we are done. □

In Theorem 3 the rules of $R$ are not required to have finitely inhabited $\eta$-types. This is illustrated by Example 16 below.

*Example 16.* Let $\Sigma$ consist of infinitely many constants $c_1, c_2, \ldots$ and assume $A = \{a_1, a_2, \ldots\}$. Consider the TSS

$$\frac{}{x \xrightarrow{a_i} c_i}\,, \qquad i \in \mathbb{N}.$$

The TSS is uniform in both the sources of rules and in their targets of premisses, and it is in bounded nondeterminism format and has a strict stratification given by $S(c_i) = 0$, $i \geq 1$. We let $\theta(x, a_i) = \emptyset$ with $i \in \mathbb{N}$. For each $i \in \mathbb{N}$, the instantiation of the rule template above has $\theta$-type $\langle x, a_i, \emptyset \rangle$. The associated LTS is image finite because for each process $p$ and for each $i \in \mathbb{N}$, $p$ can only perform an $a_i$-action to $c_i$. However, the LTS is neither finite branching nor initials finite.

Example 5.3 from page 515 of [9] is an example of application of the rule format defined by the conditions of Theorem 3. We reproduce it next.

*Example 17 (Example 5.3 of [9]).* Let $r \in \mathbb{R}_{>0}$. Consider the operator for deadlock in real-time Basic Process Algebra [17], which can be expressed by the rule

$$\frac{}{\delta[r] \xrightarrow{\delta[s]} \checkmark} \qquad 0 < s < r.$$

Process $\delta[r]$ is infinitely branching and has an uncountable set of initials. However, it is image finite as can be checked using our format. The TSS above is uniform in both the sources of rules and in the targets of their positive premisses, and is in bounded nondeterminism format and has a trivial strict stratification. Take $\theta(\delta[r], \delta[s]) = \emptyset$ for each $r, s \in \mathbb{R}_{>0}$. The $\theta$-type of each instantiation of the rule template above is $\langle \delta[r], \delta[s], \emptyset \rangle$. By Theorem 3, the associated LTS is image finite.

## 6 Future work

We say that the rule formats are *adequate* with respect to the corresponding finiteness property, i.e., the syntactic conditions ensure that the associated LTS has the property. However, the rule formats are not *complete* with respect to the corresponding finiteness property, i.e., not all the LTSs that have the property are induced by TSSs that satisfy the syntactic conditions. One direction for future work is to generalise the rule formats to cover such TSSs. In the following examples we collect some of the cases that we are aware are not covered by the rule formats.

*Example 18.* Consider the following TSS $R_{pc}$ describing a fragment of an instance of the algebra for process creation from [5]. The signature for that TSS contains the following operations:

- constants $a$, $\varepsilon$ and $\delta$,
- the unary process-creation operation new, and
- the binary operations $\cdot$ and $\|\!\lceil$, which we write in infix style.

We set $A = \{a, \checkmark\}$ and use $\alpha$ to range over it. The set of rules of $R_{pc}$, for whose intuition we refer the reader to [5], are:

$$\frac{}{a \xrightarrow{a} \varepsilon} \qquad \frac{}{\varepsilon \xrightarrow{\checkmark} \delta}$$

$$\frac{}{\mathrm{new}(x) \xrightarrow{\checkmark} x \cdot \delta} \qquad \frac{x \xrightarrow{a} x'}{\mathrm{new}(x) \xrightarrow{a} \mathrm{new}(x')}$$

$$\frac{x \xrightarrow{a} x'}{x \cdot y \xrightarrow{a} x' \cdot y} \qquad \frac{x \xrightarrow{\checkmark} x',\ y \xrightarrow{\alpha} y'}{x \cdot y \xrightarrow{\alpha} x' \parallel\!\!\lceil\, y'} \qquad \frac{x \xrightarrow{\checkmark} x',\ x' \xrightarrow{a} x'',\ y \xrightarrow{a} y'}{x \cdot y \xrightarrow{a} x'' \parallel\!\!\lceil\, y'}$$

$$\frac{x \xrightarrow{a} x'}{x \parallel\!\!\lceil\, y \xrightarrow{a} x' \parallel\!\!\lceil\, y} \qquad \frac{y \xrightarrow{\alpha} y'}{x \parallel\!\!\lceil\, y \xrightarrow{\alpha} x \parallel\!\!\lceil\, y'} \qquad \frac{x \xrightarrow{a} x',\ y \xrightarrow{a} y'}{x \parallel\!\!\lceil\, y \xrightarrow{a} x' \parallel\!\!\lceil\, y'} \ .$$

Note that the third rule for the operator $\cdot$ is not in bounded nondeterminism format because of the premise $x' \xrightarrow{a} x''$. Therefore the TSS $R_{pc}$ does not meet the requirements of Theorem 1. On the other hand, it is not too hard to show that the LTS induced by $R_{pc}$ is finite branching. (This is also a consequence of the more general Elimination Theorem from [5, Theorem 4.9].)

*Example 19.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and let $A = \{a\}$. Consider the TSS with rules

$$\frac{}{f(x) \xrightarrow{a} c} \qquad \frac{f(x) \xrightarrow{a} y}{c \xrightarrow{a} y} \ .$$

This TSS is uniform and has a strict stratification given by

$$\begin{aligned} S(f(p)) &= 0 \\ S(c) &= 1. \end{aligned}$$

We let $\theta(f(x), a) = \emptyset$ and $\theta(c, a) = \{(f(x), a)\}$. The rule on the left has $\theta$-type $\langle f(x), a, \emptyset\rangle$, and the rule on the right has $\theta$-type $\langle c, a, \phi\rangle$ where $\phi(f(x), a) = \{y\}$. Variable $x$ in the premiss of the rule on the right does not occur in the source of the rule, and hence the TSS is not in bounded nondeterminism format and does not meet the rule format for image finiteness. However, the set of images of any process $p$ for action $a$ is $\{c\}$, and therefore the associated LTS is image finite. Notice that the TSS does not meet the rule format for finite branching either, but the associated LTS is finite branching.

*Example 20.* Let $A$ consist of infinitely many actions $a_1, a_2, \ldots$ and let $\Sigma = A \cup \{f, g\}$ where $f$ and $g$ are unary function symbols. Consider the TSS with rules

$$\frac{}{g(a_1) \xrightarrow{a_1} a_1} \qquad \frac{g^i(x) \xrightarrow{a_i} x}{f(x) \xrightarrow{a_i} x} \ , \qquad i \in \mathbb{N}$$

where $g^i$ stands for applying the function symbol $g$ to its argument $i$ times. The TSS is uniform, in bounded nondeterminism format, and has a strict stratification given by

$$S(g^i(p)) = 0$$
$$S(f(p)) = 1.$$

Notice that there exists no $\eta$ such that the $\eta$-types are finitely inhabited. No matter how one picks $\eta$, for every finite set $\eta(f(x))$ there would be an infinite number of instances of the rule on the right that have the same $\eta$-type. Thus the TSS is not bounded and the TSS does not meet the rule format for finite branching. However, the associated LTS is finite branching because the only possible transitions are $g(a_1) \xrightarrow{a_1} a_1$ and $f(a_1) \xrightarrow{a_1} a_1$.

*Example 21.* Let $\Sigma$ consist of a constant $c$ and a unary function symbol $f$, and assume $A = \{a_1, a_2, \ldots\}$ with infinitely many actions. Consider the TSS

$$\frac{}{f(x) \xrightarrow{a_1} f(x)} \qquad \frac{f(x) \xrightarrow{a_i} y}{f(x) \xrightarrow{a_{i+1}} y}\,, \qquad i \in \mathbb{N}.$$

The TSS is uniform in both the sources of rules and the targets of their positive premisses, and it is in bounded nondeterminism format. We let $\theta(f(x), a_1) = \emptyset$ and $\theta(f(x), a_{i+1}) = \{(f(x), a_i)\}$ for each $i \in \mathbb{N}$. The rule on the left has $\theta$-type $\langle f(x), a_1, \emptyset\rangle$. For each $i \in \mathbb{N}$, the instantiation of the rule template on the right has $\theta$-type $\langle f(x), a_{i+1}, \phi_i\rangle$, where $\phi_i(f(x), a_i) = \{y\}$. The TSS is image bounded. Notice that the TSS does not have a strict stratification and therefore it does not meet the rule format for image finiteness. However, the associated LTS is image finite, since $c$ has no outgoing transitions and the image of each process of the form $f(p)$ for action $a_i$ (with $i \in \mathbb{N}$) is $f(p)$.

Another direction for future research is the study of algorithmic aspects of the rule formats discussed in this paper. Indeed, whereas the conditions pertaining to single rules, such as those imposed by the bounded nondeterminism format, are purely syntactic and easy to check, those related to the various notions of types have a global nature. It would be interesting to study ways to enforce those global constraints and to develop algorithms for checking them over classes of TSSs.

Nominal structural operational semantics (NoSOS) [8] enriches the SOS formalism by using some of the nominal techniques from [10,24,31] to deal with names and binders within the SOS framework. We are currently investigating how to adapt the results in this paper to NoSOS. The main challenges there are to treat transition labels that may contain variables and the effect that the so-called freshness assertions may have on the finiteness properties of interest. In NoSOS, it is conventional to consider special administrative transitions for freshness conditions, for substitution, and for $\alpha$-conversion [8]. The transitions for freshness conditions in isolation induce an initials-finite LTS. There are two kinds of substitution, atom-for-atom and term-for-atom substitution, which taken in isolation induce image-finite LTSs. The transitions for $\alpha$-conversion taken in isolation induce an initials-finite LTS. One of the problems in extending our results to NoSOS is to abstract from these administrative transitions in order to focus on the finiteness properties of the remaining transitions.

*Acknowledgements* We thank two anonymous referees for their careful reading of our paper and their constructive comments.

## References

1. Abramsky, S.: Domain Theory and the Logic of Observable Properties. Ph.D. thesis, Department of Computer Science, Queen Mary College, University of London (1987)
2. Aceto, L., Fokkink, W., Verhoef, C.: Structural operational semantics. In: Bergstra, J., Ponse, A., Smolka, S. (eds.) Handbook of Process Algebra, chap. 3, pp. 197–292. Elsevier (2001)
3. Amtoft, T., Nielson, F., Nielson, H.R.: Type and Effect Systems. Imperial College Press (1999)
4. Apt, K.R., Plotkin, G.D.: Countable nondeterminism and random assignment. Journal of the ACM 33(4), 724–767 (1986)
5. Baeten, J.C.M., Vaandrager, F.W.: An algebra for process creation. Acta Informatica 29(4), 303–334 (1992), `http://dx.doi.org/10.1007/BF01178776`
6. Bloom, B.: CHOCOLATE: Calculi of Higher Order COmmunication and LAmbda TErms (preliminary report). In: Boehm, H.J., Lang, B., Yellin, D.M. (eds.) Conference Record of the 21st ACM Symposium on Principles of Programming Languages, Portland, Oregon. pp. 339–347. ACM Press (1994)
7. Bloom, B., Istrail, S., Meyer, A.R.: Bisimulation can't be traced. Journal of the ACM 42(1), 232–268 (1995)
8. Cimini, M., Mousavi, M.R., Reniers, M.A., Gabbay, M.J.: Nominal SOS. Electronic Notes in Theoretical Computer Science 286, 103–116 (2012)
9. Fokkink, W., Vu, T.D.: Structural operational semantics and bounded nondeterminism. Acta Informatica 39(6-7), 501–516 (2003)
10. Gabbay, M.J., Pitts, A.M.: A new approach to abstract syntax involving binders. In: Longo, G. (ed.) Proceedings of the 14th Symposium on Logic in Computer Science, Trento, Italy. pp. 214–224. IEEE Computer Society Press (1999)
11. van Gelder, A., Ross, K.A., Schilpf, J.S.: The well-founded semantics for general logic programs. Journal of the ACM 38(3), 620–662 (1991)
12. van Glabbeek, R.J.: Bounded nondeterminism and the approximation induction principle in process algebra. In: Brandenburg, F.J., Vidal-Naquet, G., Wirsing, M. (eds.) Proceedings of the 4th Annual Symposium on Theoretical Aspects of Computer Science, Passau, Germany. Lecture Notes in Computer Science, vol. 247, pp. 336–347. Springer (1987)
13. Groote, J.F., Vaandrager, F.W.: Structured operational semantics and bisimulation as a congruence. Information and Computation 100(2), 202–260 (1992)
14. Groote, J.F.: Transition system specifications with negative premises. Theoretical Computer Science 118(2), 263–299 (1993)
15. Heyting, A. (ed.): Constructivity in Mathematics. North-Holland Publishing Company (1959)
16. Keller, R.M.: Formal verification of parallel programs. Communications of the ACM 19(7), 371–384 (1976)

17. Klusener, A.S.: Models and Axioms for a Fragment of Real Time Process Algebra. Ph.D. thesis, Department of Mathematics and Computing Science, Technical University of Eindhoven (1993)
18. Lévy, A.: A hierarchy of formulas in set theory. Memoirs of the American Mathematical Society 57, 76 (1965)
19. Martin-Löf, P.: Intuitionistic Type Theory. Studies in Proof Theory: Lecture Notes, Bibliopolis, Napoli (1984)
20. Milner, R.: Communication and Concurrency. PHI Series in computer science, Prentice Hall (1989)
21. Mousavi, M.R., Reniers, M.A., Groote, J.F.: SOS formats and meta-theory: 20 years after. Theoretical Computer Science 373(3), 238–272 (2007)
22. Nielson, F., Nielson, H.R., Hankin, C.: Principles of Program Analysis. Springer (2005)
23. Nielson, H.R., Nielson, F.: Semantics with Applications: An Appetizer. Springer (2007)
24. Pitts, A.M.: Nominal Sets: Names and Symmetry in Computer Science, Cambridge Tracts in Theoretical Computer Science, vol. 57. Cambridge University Press (2013)
25. Plotkin, G.D.: A structural approach to operational semantics. Tech. Rep. DAIMI FN-19, Department of Computer Science, Aarhus University, Denmark (1981)
26. Plotkin, G.D.: An operational semantics for CSP. In: Formal Description of Programming Concepts, II (Garmisch-Partenkirchen, 1982), pp. 199–225. North-Holland, Amsterdam (1983)
27. Plotkin, G.D.: A structural approach to operational semantics. Journal of Logic and Algebraic Programming 60-61, 17–139 (2004)
28. Przymusinski, T.C.: The well-founded semantics coincides with the three-valued stable semantics. Fundamenta Informaticae 13(4), 445–463 (1990)
29. Sangiorgi, D., Walker, D.: The pi-calculus: A Theory of Mobile Processes. Cambridge Universtity Press (2001)
30. de Simone, R.: Higher-level synchronising devices in Meije–SCCS. Theoretical Computer Science 37(3), 245–267 (1985)
31. Urban, C., Pitts, A.M., Gabbay, M.J.: Nominal unification. Theoretical Computer Science 323(1-3), 473–497 (2004)
32. Vaandrager, F.W.: Expressiveness results for process algebras. In: de Bakker, J.W., de Roever, W.P., Rozenberg, G. (eds.) Proceedings of the REX Workshop on Semantics: Foundations and Applications, Beekbergen, The Netherlands. Lecture Notes in Computer Science, vol. 666, pp. 609–638. Springer (1993)