Semantics and Expressiveness of Ordered SOS [1]

MohammadReza Mousavi [a,b,*], Iain Phillips [c],
Michel A. Reniers [a], Irek Ulidowski [d]

[a] *Eindhoven University of Technology, The Netherlands*

[b] *Reykjavík University, Iceland*

[c] *Imperial College London, United Kingdom*

[d] *University of Leicester, United Kingdom*

Abstract

Structured Operational Semantics (SOS) is a popular method for defining semantics by means of transition rules. An important feature of SOS rules is *negative premises*, which are crucial in the definitions of such phenomena as priority mechanisms and time-outs. However, the inclusion of negative premises in SOS rules also introduces doubts as to the preferred meaning of SOS specifications.

Orderings on SOS rules were proposed by Phillips and Ulidowski as an alternative to negative premises. Apart from the definition of the semantics of positive GSOS rules with orderings, the meaning of more general types of SOS rules with orderings has not been studied hitherto. This paper presents several candidates for the meaning of general SOS rules with orderings and discusses their conformance to our intuition for such rules. We take two general frameworks (rule formats) for SOS with negative premises and SOS with orderings, and present semantics-preserving translations between them with respect to our preferred notion of semantics. Thanks to our semantics-preserving translation, we take existing congruence meta-results for strong bisimilarity from the setting of SOS with negative premises into the setting of SOS with orderings. We further compare the expressiveness of rule formats for SOS with orderings and SOS with negative premises. The paper contains also many examples that illustrate the benefits of SOS with orderings and the properties of the presented definitions of meaning.

---

* Corresponding author. Address: Den Dolech 2, NL-5612 AZ Eindhoven, The Netherlands

Email addresses: `m.r.mousavi@tue.nl` (MohammadReza Mousavi), `iccp@doc.ic.ac.uk` (Iain Phillips), `m.a.reniers@tue.nl` (Michel A. Reniers), `iu3@mcs.le.ac.uk` (Irek Ulidowski).

1 A shorter version of this paper appeared as [15].

1 Introduction

It is well-known that negative premises in Structured Operational Semantics (SOS) [17,1] are useful and non-trivial additions but at the same time they may lead to ambiguities and paradoxical phenomena with respect to the semantics of SOS [10,6,9]. As an alternative to negative premises, [21,22] proposes to furnish SOS deduction rules with an ordering. But to avoid the same pitfalls as those of negative premises, [22] restricts itself to the GSOS subset of SOS, which does not allow for look-ahead or complex terms as sources of premises.

It is also well-known from the term rewriting literature that the introduction of orderings (called priorities) to term rewrite systems introduces challenges for the well-definedness of the semantics of term rewrite systems [14,4,18]. SOS specifications can be seen as conditional term rewrite systems and thus one expects similar or even more difficult challenges when studying the general semantics of SOS with orderings.

However, a fundamental study of the semantics of ordered SOS (in its full generality) has not been carried out to date and even misconceptions exist. In [13, Theorem 4], it is mentioned (without formal proof) that one can generalize a particular rule format for ordered SOS with look-ahead while preserving the congruence property of a probabilistic notion of bisimilarity. However, as we shall show in this paper, the introduction of either look-ahead or complex terms as sources of premises to ordered SOS jeopardizes the well-definedness of the induced transition relation (let alone the congruence result).

The remainder of this paper is as follows. In Section 2, we define the basic concept of Ordered Transition System Specification (OTSS) which is a general framework for ordered SOS. In the same section, we give some examples, both for illustrating the application of ordered SOS and for showing that the semantics of OTSS's is not always clear. Then, in Section 3, following [6,9], we define several alternative approaches for defining the semantics of ordered SOS. In Section 4, we define a rule format, called OTYFT (for ordered TYFT), for congruence of bisimilarity. Subsequently, in Section 5, we give semantics-preserving translations from NTYFT ([10]) to OTYFT and vice versa. In Section 6, we compare the relative expressiveness of the existing rule format for ordered SOS and the OTYFT format introduced in this paper. Section 7 discusses related work and Section 8 concludes the paper.

2 Ordered Transition System Specification

2.1 Basic Concepts

Definition 2.1 (Signature, Term and Substitution) *Assume a countable set of variables V (with typical members x, y, x', y', x_i, $y_i \ldots$). A* signature Σ *is a set of function symbols (operators, with typical members f, g, ...) with fixed arities $ar : \Sigma \to \mathbb{N}$. Functions with zero arity are called constants and are typically denoted by a, b, c and d. Terms $s, t, t_i, \ldots \in \mathcal{T}$ are constructed inductively using variables and function symbols. A list of terms is denoted by $\overrightarrow{t}$. When we write $f(\overrightarrow{t})$, we assume that $\overrightarrow{t}$ has the right size, i.e., $ar(f)$. All terms are considered open terms. Closed terms $p, q, \ldots \in \mathcal{C}$ are terms that do not mention a variable and are typically denoted by $p, q, l, p', p_i, \ldots$. A substitution σ replaces variables in a term with terms. The set of variables appearing in term t is denoted by $vars(t)$. A substitution is called* closed *if its range consists of closed terms.*

Definition 2.2 (Ordered Transition System Specification (OTSS)) *Given a signature and a set of variables, a* Transition System Specification (TSS) *is a set R of deduction rules.*

A deduction rule $r \in R$, is defined as a tuple (H, c) where H is a set of formulae and c is a positive formula. For all $t, t' \in \mathcal{T}$ and $l \in \mathcal{C}$ we define that $\phi = t \xrightarrow{l} t'$ is a positive formula and $\phi' = t \overset{l}{\nrightarrow}$ is a negative formula. A formula is a positive or a negative formula. We denote the set of formulae by Φ and the set of positive formulae by $\Phi_{\mathbf{p}}$. Term t is called the source of both ϕ and ϕ', denoted by $src(\phi)$ and $src(\phi')$, and t' is called its target, denoted by $trg(\phi)$. The formula c is called the conclusion *of r, denoted by $conc(r)$, and the formulae in H are called its* premises *and denoted by $prem(r)$. A positive deduction rule (TSS) is a deduction rule of which all the premises (all the deduction rules) are positive. The notions of source and target generalize to a set of formulae, as expected. Also, the notion of "variables of" is naturally lifted to sets of terms, formulae, sets of formulae and deduction rules.*

An Ordered Transition System Specification (OTSS) *is a pair $(R, <)$ where R is a positive TSS and $< \subseteq R \times R$ is an* arbitrary *relation on the deduction rules. For a rule r, $higher(r)$ is defined as $\{r' \mid r < r'\}$, i.e., the set of rules placed above r by the ordering $<$. We sometimes use $>$ to denote the inverse of $<$.*

The intuition behind the ordering on rules is that a deduction rule r can only be applied when all deduction rules $r' \in higher(r)$ are disabled since they do not have a "reason" (or "proof") for their premises to hold. As we show in the remainder, this notion of "reason" or "proof" is not trivial to define and involves the same complications as those concerning the semantics of TSS's with negative premises [9].

Orderings on positive rules can replace negative premises in rules [22]. In the remainder of this section, we start with two simple examples motivating and illustrating the use of ordering (as an alternative to negative premises). Then we show that our more general definition of ordered TSS extends the applicability of the restricted ordered SOS paradigm from [22] by specifying an example involving look-ahead. Finally, we show that this extension comes at a price, namely, the semantics of general OTSS's (e.g., those containing look-ahead) is not always clear and should be studied more thoroughly.

Example 2.3 (Priority) The priority operator θ [3] may be used to represent such phenomena as time-outs and interrupts. For a given partial order $\prec$ on actions (a set of constants, denoted by a, b, c, $\ldots \in Act$), $\theta(p)$ is a restriction on the behavior of p such that action a can happen only if no b with $a \prec b$ is possible. If $B_a = \{b \mid a \prec b\}$, then θ can be defined by this TSS (where the deduction rule is actually a rule schema which should be repeated for each action $a \in Act$):

$$\frac{x \xrightarrow{a} y \quad \{x \stackrel{b}{\nrightarrow} \mid b \in B_a\}}{\theta(x) \xrightarrow{a} \theta(y)}.$$

Alternatively, θ can be defined by positive deduction rules r_a, equipped with the ordering defined by $r_a < r_b$ whenever $a \prec b$:

$$(r_a)\frac{x \xrightarrow{a} y_a}{\theta(x) \xrightarrow{a} \theta(y_a)}$$

where y_a are distinct variables for all $a \in Act$. (Note that the naming of variables in the rules related by ordering is indeed important; for example, it is essential to assume that y_a and y_b are distinct variables for each a and b such that $a \prec b$. As we show in the remainder of this paper, particularly in Section 4.2, violating this condition ruins intuitive properties such as congruence of bisimilarity.)

Example 2.4 (Timed Parallel Composition) Consider the following TSS defining the semantics of a subset of Hennessy and Regan's Process Algebra for Timed Systems (TPA) [12]. The signature consists of a constant *nil*, unary operators $a.\_$ and $\overline{a}.\_$ (action prefixing, for all $a \in Act$), $\tau.\_$ (internal action prefixing) and $\sigma.\_$ (time step prefixing), and a binary operator $\|$ (parallel composition). (Constants a, $\overline{a}$, τ and σ are also introduced in the signature to model the labels.)

$$(a)\,\frac{}{a.x \xrightarrow{a} x} \qquad (\tau)\,\frac{}{\tau.x \xrightarrow{\tau} x} \qquad (\sigma_0)\,\frac{}{\sigma.x \xrightarrow{\sigma} x} \qquad (\sigma_1)\,\frac{}{a.x \xrightarrow{\sigma} a.x} \qquad (\sigma_2)\,\frac{}{nil \xrightarrow{\sigma} nil}$$

$$(\|_0)\,\frac{x_0 \xrightarrow{a} y_0}{x_0 \parallel x_1 \xrightarrow{a} y_0 \parallel x_1} \qquad (\|_1)\,\frac{x_1 \xrightarrow{a} y_1}{x_0 \parallel x_1 \xrightarrow{a} x_0 \parallel y_1}$$

$$(\tau_0)\,\frac{x_0 \xrightarrow{\tau} y_0}{x_0 \parallel x_1 \xrightarrow{\tau} y_0 \parallel x_1} \qquad (\tau_1)\,\frac{x_1 \xrightarrow{\tau} y_1}{x_0 \parallel x_1 \xrightarrow{\tau} x_0 \parallel y_1}$$

$$\textbf{(comm)}\,\frac{x_0 \xrightarrow{a} y_0 \quad x_1 \xrightarrow{\bar{a}} y_1}{x_0 \parallel x_1 \xrightarrow{\tau} y_0 \parallel y_1} \qquad \textbf{(time)}\,\frac{x_0 \xrightarrow{\sigma} y_0 \quad x_1 \xrightarrow{\sigma} y_1 \quad x_0 \parallel x_1 \stackrel{\tau}{\nrightarrow}}{x_0 \,||\, x_1 \xrightarrow{\sigma} y_0 \,||\, y_1}$$

In the semantics of the parallel composition operator, $p \parallel q$ can pass time (denoted by label σ) if both p and q can pass time, and if they are stable and cannot communicate (i.e. $p \parallel q \stackrel{\tau}{\nrightarrow}$).

The above semantics can be specified in ordered SOS by placing a positive version of the rule **(time)** below the rules (τ_0), (τ_1) and **(comm)** as shown below. All other rules are copied to the following OTSS and are unrelated (in terms of ordering) to the rules below.

$\downarrow$	$(\tau_0)\,\frac{x_0 \xrightarrow{\tau} y_0}{x_0 \parallel x_1 \xrightarrow{\tau} y_0 \parallel x_1} \qquad (\tau_1)\,\frac{x_1 \xrightarrow{\tau} y_1}{x_0 \parallel x_1 \xrightarrow{\tau} x_0 \parallel y_1} \qquad \textbf{(comm)}\,\frac{x_0 \xrightarrow{a} y_0 \quad x_1 \xrightarrow{\bar{a}} y_1}{x_0 \parallel x_1 \xrightarrow{\tau} y_0 \parallel y_1}$
	$\textbf{(time)}\,\frac{x_0 \xrightarrow{\sigma} y_0' \quad x_1 \xrightarrow{\sigma} y_1'}{x_0 \parallel x_1 \xrightarrow{\sigma} y_0' \parallel y_1'}$

We fix the above notation for ordering so that in each column, rules of the upper row have priority over rules of the lower row, i.e., an instance of a rule in the lower row (under a particular substitution σ) can only be "applied" when no rule in the upper row (of the same column and under a substitution agreeing with σ on common variables) can be "applied". Formally, we have the following orderings: $(\tau_0) >$ **(time)**, $(\tau_1) >$ **(time)**, and **(comm)** $>$ **(time)**.

In general, a TSS with rules with negative premises may not contain all the rules that are needed to define orderings that replace negative premises. In such cases, we shall need to extend the TSS with auxiliary rules. If a rule r has a premise $t \stackrel{a}{\nrightarrow}$, the required auxiliary rule is $\frac{t \xrightarrow{a} t'}{t \xrightarrow{a} t'}$ and is placed above r. We shall argue later (Lemma 3.31) that extending TSS's with rules as above is harmless: it does not change the meaning of the transition relation. More precisely, our preferred method for assigning meaning to OTSS's is insensitive to rules of such form.

In the following example, we address the idea of extending ordered SOS [22] with look-ahead as suggested by [13, Theorem 4] and show that it may lead to pathological specifications with an unclear meaning. (The rule format of [13] extends traditional OTSS with probabilities but the problem we address below is orthogonal to the presence or absence of probabilities and hence, we use the plain OTSS setting as defined above.)

Example 2.5 (OSOS with Look-Ahead) Consider the OTSS with the following deduction rules.

$\downarrow$	$\dfrac{x \xrightarrow{b} y \quad y \xrightarrow{d} z}{f(x) \xrightarrow{d} d}$	$\dfrac{x \xrightarrow{a} y \quad y \xrightarrow{c} z}{g(x) \xrightarrow{c} c}$	$\dfrac{}{a \xrightarrow{a} f(a)}$	$\dfrac{}{a \xrightarrow{b} g(a)}$
	$\dfrac{x \xrightarrow{b} y}{f(x) \xrightarrow{c} c}$	$\dfrac{x \xrightarrow{a} y}{g(x) \xrightarrow{d} d}$		

Note that according to the notation fixed before, in the following OTSS, it holds that

$$\frac{x \xrightarrow{b} y \quad y \xrightarrow{d} z}{f(x) \xrightarrow{d} d} > \frac{x \xrightarrow{b} y}{f(x) \xrightarrow{c} c} \quad \text{and} \quad \frac{x \xrightarrow{a} y \quad y \xrightarrow{c} z}{g(x) \xrightarrow{c} c} > \frac{x \xrightarrow{a} y}{g(x) \xrightarrow{d} d}$$

but it does not hold that

$$\frac{}{a \xrightarrow{a} f(a)} > \frac{x \xrightarrow{b} y \quad y \xrightarrow{d} z}{f(x) \xrightarrow{d} d}.$$

At first sight, it is not intuitively clear which of the following three transition relations should be considered as the meaning of the above OTSS.

(1) $\{a \xrightarrow{a} f(a), a \xrightarrow{b} g(a), f(a) \xrightarrow{c} c, g(a) \xrightarrow{c} c\}$, or
(2) $\{a \xrightarrow{a} f(a), a \xrightarrow{b} g(a), f(a) \xrightarrow{d} d, g(a) \xrightarrow{d} d\}$, or
(3) $\{a \xrightarrow{a} f(a), a \xrightarrow{b} g(a)\}$.

So, a convincing semantics for OTSS's should either be neutral about different possibly derivable transitions (in items 1 and 2) or reject the above OTSS altogether due to its ambiguous nature. We present solutions that cater for both possibilities in the remainder of this paper.

Example 2.6 The situation with the following OTSS is even worse.

$\downarrow$	$\dfrac{x \xrightarrow{a} y \quad y \xrightarrow{b} z}{f(x) \xrightarrow{b} a}$	$\dfrac{}{a \xrightarrow{a} f(a)}$
	$\dfrac{x \xrightarrow{a} y}{f(x) \xrightarrow{b} b}$	

If one initially assumes that from rules in the first row one cannot derive any transition with $f(a)$ as its source (which is a legitimate assumption), then the rule below allows for deriving $f(a) \xrightarrow{b} b$. This transition, in turn, enables the premises of the rule above it (leading to the conclusion that $f(a) \xrightarrow{b} a$ should be derivable) and thus the very same rule below must have been disabled and the chain of contradictory conclusions goes on forever. Again, any convincing semantics for OTSS's should either find a way to deal with the contradictory conclusions (e.g., by considering all of them uncertain yet possibly derivable transitions) or reject the above OTSS altogether due to its paradoxical nature. The notions of semantics presented in the remainder allow for both interpretations.

The above examples make the case for a more profound study of the meaning of ordered SOS, which is the subject of the following section.

3 Semantics of OTSS

An OTSS is supposed to induce a unique transition relation on closed terms but as Example 2.5 already suggested, for some OTSS's the way to assign such a transition relation may not be straightforward. This phenomenon has been known in several areas such as logic programming and term rewriting and even inside the SOS meta-theory as the result of introducing negative premises to SOS rules. For TSS's with negative premises, several notions of semantics have been defined and used, of which [9] provides an overview and a comparison. Here, we take the major semantic notions of [9] and re-interpret or re-phrase them for OTSS's. Examples similar to those of [9] are given here both to motivate the notions and to facilitate comparing the two paradigms (i.e., negative premises vs. orderings). Our proofs in this section follow the same structure as those of [9].

The section introduces twelve notions of semantics and assesses their suitability. Our preferred and most general notion is the *least three-valued stable model* (Semantics 7) but, for the purpose of congruence meta-results, we favor semantics defined in terms of *completeness* (Semantics 11).

The semantics of an OTSS is the transition relation it defines on closed terms. Thus, in the remainder of this section, we only have to deal with closed instantiations of deduction rules. To avoid repeating the phrase "an instance of rule r under a closing substitution σ", in this section, we assume that the OTSS's only contain closed terms. To define the semantics of an arbitrary OTSS, one may instantiate the rules and the ordering relation under all closing substitutions and then use the notions of the semantics in the remainder of this section.

Definition 3.1 (Closing (O)TSS's) *Given an arbitrary OTSS* $(R, <)$*, closed*$(R, <) \doteq (R', <')$ *where* $R' \doteq \{\sigma(r) \mid \sigma : V \to \mathcal{C}, r \in R\}$ *and* $<' \doteq \{(\sigma(r), \sigma(r')) \mid \sigma : V \to \mathcal{C}, r, r' \in R, r < r'\}$*. Similarly, for a TSS* R*, closed*$(R) \doteq R'$ *where* R' *is defined above.*

We start with the following notion of provability, which is the usual way of giving semantics to ordinary positive TSS's (i.e., without ordering or negative premises).

Definition 3.2 (Proof) *Given an OTSS* $(R, <)$*, a* proof p *for a formula* ϕ *is a well-founded upwardly branching tree of which*

(1) the nodes are formulae,
(2) the root is ϕ*, and*
(3) if a node is labelled ϕ' *and the nodes above it form the set* K*, then there is a deduction rule* $r \in R$ *such that* $r = \frac{K}{\phi'}$.

An r-proof *for* ϕ *is a proof in which the last step is due to deduction rule* r*. We write* $(R, <) \vdash_p \phi$ *when* p *is a proof in* $(R, <)$ *for* ϕ*. We may drop* $(R, <)$ *when it is clear from the context. We denote the set of deduction rules used in a proof* p *by rules*(p)*. We write* $T \vDash p$ *for a transition relation* T *and a proof* p *to denote that for all nodes* ϕ *in* p*, we have* $\phi \in T$*. A proof* p *is called* minimal *when for each branch of the tree, a formula* ϕ *appears at most once.*

It immediately follows from the above definition that for any two OTSS's $(R, <)$, $(R', <')$, if $R \subseteq R'$ then $\{\phi \mid (R, <) \vdash_p \phi\} \subseteq \{\phi \mid (R', <) \vdash_p \phi\}$. Moreover, for an arbitrary positive formula ϕ, $\vdash_p \phi$ if and only if there exists a minimal proof q such that $\vdash_q \phi$.

The following notion of semantics is the obvious notion when the ordering relation is empty. It serves as a starting point, or first approximation, for several notions of semantics in the remainder of this paper.

Semantics 1 (Provability) *The semantics of an OTSS $(R, <)$ is the set of all formulae ϕ such that $\vdash_p \phi$ for some proof p.*

As said before, the above definition neglects the ordering on rules and thus cannot be used for the semantics of OTSS's, but it can be helpful in finding the right semantics in the following sense. Any proposal for the semantics of OTSS's should coincide with the notion of provability when the ordering relation is taken to be empty. In particular, no semantics of OTSS's should admit transitions that are not provable. Although the above criterion seems trivial, as we observe in the remainder, some notions of semantics for TSS's when applied to the setting with orderings on rules fall short of satisfying it.

Next, we give a few examples which illustrate in more detail our intuitive understanding of the meaning of OTSS's.

Example 3.3

$$\begin{array}{|c||c|}\hline \downarrow & \dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a} \\ \hline & \dfrac{}{b \xrightarrow{b} b} \\ \hline \end{array}$$

In our view, a reasonable notion of semantics should define $\{b \xrightarrow{b} b\}$ as the meaning of the above OTSS. There is no reason to believe that $a \xrightarrow{a} a$ is possible and thus, the rule placed above the axiom $\dfrac{}{b \xrightarrow{b} b}$ is not applicable.

Example 3.4 Consider the following OTSS.

$$\begin{array}{|c||c|c|} \hline \downarrow & \dfrac{a \xrightarrow{a} a}{b \xrightarrow{b} b} & \dfrac{b \xrightarrow{a} b}{a \xrightarrow{b} a} \\ \hline & \dfrac{}{b \xrightarrow{a} b} & \dfrac{}{a \xrightarrow{a} a} \\ \hline \end{array}$$

The meaning of the above OTSS is not intuitively clear, namely, it is not clear how to prefer one of the two transition relations $\{b \xrightarrow{a} b, a \xrightarrow{b} a\}$ or $\{a \xrightarrow{a} a, b \xrightarrow{b} b\}$ over the other.

Example 3.5 Consider the following OTSS in which the symbol $\circlearrowright$ is meant to denote that a deduction rule is placed above itself.

$$\begin{array}{|c||c|} \hline \downarrow & \begin{array}{|c||c|} \hline \circlearrowright & \dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a} \\ \hline \end{array} \\ \hline & \dfrac{}{a \xrightarrow{a} a} \\ \hline \end{array}$$

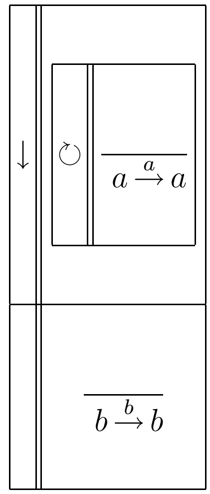
The meaning of the above OTSS is unclear as any decision about applicability of the upper rule is self-contradictory.

Example 3.6

$$\begin{array}{|c||c|} \hline \circlearrowright & \dfrac{}{a \xrightarrow{a} a} \\ \hline \end{array}$$

Any satisfactory semantics for OTSS's should associate the empty set with the above OTSS.

The situation with the following OTSS is more subtle.



One may expect $b \xrightarrow{b} b$ to be included in the transition relation since the transition of $a \xrightarrow{a} a$ is always disabled. In [22] the authors require that for $b \xrightarrow{b} b$ to be included, no higher rule should be applicable (meaning that the premises are in the transition relation). Since $\dfrac{}{a \xrightarrow{a} a}$ can always be applied, the semantics does not allow for deriving $b \xrightarrow{b} b$. Thus, according to [22], an axiom (even with a self-loop) disables any rule placed below it. We aim to be consistent with the same intuition throughout this paper.

3.1 Two-Valued Solutions

In this subsection, we define several ways of assigning a transition relation to an OTSS. These solutions are called two-valued because they define one transition relation (which is a set of transition formulae that, purportedly, can be certainly derived from the OTSS) and reject the other transitions as impossible.

We start with a few model-theoretic notions of semantics. To this end, we define what it means for a rule to be correct with respect to a transition relation, namely that the rules above it cannot find at least one of their premises in the transition relation. Then, we proceed with defining the notions of model and supported model (which actually stand for "correct model" and "supported and correct model").

Definition 3.7 (Correct) *Given an OTSS* $(R, <)$ *and a transition relation* T*, we say that a deduction rule* $r = \dfrac{H}{\phi} \in R$ *is* correct *w.r.t.* T *when for all* $r' = \dfrac{H'}{\psi} \in higher(r)$*,* $H' \nsubseteq T$*. A proof* p *for a formula* ϕ *is* correct *w.r.t.* T *when for all* $r \in rules(p)$*,* r *is correct w.r.t.* T*.*

Lemma 3.8 *If r is correct w.r.t. T then it is correct w.r.t. all $T' \subseteq T$.*

Proof. Trivial from Definition 3.7. ⊠

Using the terminology of [21,22], a rule r is correct, when all rules in *higher*(r) are not *applicable* since they miss at least one of their premises.

Definition 3.9 (Model) *A transition relation T is a* model *for an OTSS $(R, <)$ when for all $r = \dfrac{H}{\phi} \in R$ if $H \subseteq T$ and r is correct w.r.t. T then $\phi \in T$.*

Based on the above definition, one may define the following semantics for OTSS's.

Semantics 2 (Least Model) *An OTSS is meaningful if it has a least model, and its semantics is its least model.*

For the OTSS of Example 3.5, Semantics 2 defines the transition relation $\{a \xrightarrow{a} a\}$ as its semantics, which is counter-intuitive. Similarly, for the following TSS, the above semantics diverges from the intuition.

Example 3.10

$\downarrow$	$\dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a}$
	$\dfrac{}{a \xrightarrow{a} a}$

The semantics of the above TSS is intuitively undefined for there is no base to initially assume $a \xrightarrow{a} a$ and if we apply the second rule and we find a reason for $a \xrightarrow{a} a$ this will in turn enable the upper rule and thus disable the lower rule. However, it has $\{a \xrightarrow{a} a\}$ as its least model (note that $\emptyset$ is not a model of this OTSS, since from the lower rule it will then follow that $a \xrightarrow{a} a$ should be in the model).

Even in cases where the TSS has a model, the least model need not exist. Consider the TSS in Example 3.4; both $\{b \xrightarrow{a} b, a \xrightarrow{b} a\}$ and $\{a \xrightarrow{a} a, b \xrightarrow{b} b\}$ are minimal models of it.

The following notion of supported model makes the requirements on the semantics stricter, and thereby may be helpful in rejecting some of the pathological models.

Definition 3.11 (Supported Model) *A transition relation T is a* supported model *of an OTSS $(R, <)$ when it is a model and for all $\phi \in T$, there exists $r = \dfrac{H}{\phi} \in R$ such that $H \subseteq T$ and r is correct w.r.t. T.*

The following two notions exploit the concept of supported model in order to define a meaning for ordered SOS.

Semantics 3 (Least Supported Model) *An OTSS is meaningful when it has a least supported model and its semantics is its least supported model.*

Semantics 4 (Unique Supported Model) *An OTSS is meaningful when it has a unique supported model and its semantics is its unique supported model.*

However, as the following two examples illustrate, Semantics 3 and 4 are both in some cases counter-intuitive.

Example 3.12

$\downarrow$	$\dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a}$	$\dfrac{b \xrightarrow{b} b}{a \xrightarrow{a} a}$	$\dfrac{b \xrightarrow{b} b}{b \xrightarrow{b} b}$
	$\dfrac{}{a \xrightarrow{a} a}$		

The least supported model for the above OTSS is $\{a \xrightarrow{a} a, b \xrightarrow{b} b\}$, which is counter-intuitive (i.e., there is no reason to believe that $a \xrightarrow{a} a$ or $b \xrightarrow{b} b$ should be included in the transition relation). Similarly, the unique supported model associates the same semantics to the above OTSS, thus giving us a good reason to reject both notions.

Example 3.13

	$\dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a}$

The above TSS (with an empty ordering) has no unique supported model since both $\emptyset$ and $\{a \xrightarrow{a} a\}$ are supported models of it. Therefore the notion of unique supported model does not coincide with provability for unordered TSS's, giving another reason to reject it.

Next, we define the notion of two-valued stable model, which is proposed in [6] and is exploited to define the "most reasonable" two-valued notion of semantics.

Definition 3.14 ((Two-Valued) Stable Model) *Given an OTSS $(R, <)$, transition relation T is* (two-valued) stable *when $\phi \in T$ if and only if $\vdash_p \phi$ for some proof p such that p is correct w.r.t. T.*

The following notion of well-supported model, as shown by the theorem afterwards, is another way of formulating the two-valued stable model.

Definition 3.15 (Well-Supported Model) *Given an OTSS $(R, <)$, transition relation T is* well-supported *when $\phi \in T$ if and only if $\vdash_p \phi$ for some proof p such that $T \vDash p$ and p is correct w.r.t. T.*

Theorem 3.16 *T is a well-supported model for $(R, <)$ if and only if it is stable for it.*

Proof. Well-supported models are clearly stable models. Given a stable model, by an induction on the proof p, one can check that for all nodes ϕ in p, it holds that $T \vDash p$ and thus, stable models are well-supported, as well. ⊠

One may use the notion of well-supported (two-valued stable) model to define a semantics for order SOS.

Semantics 5 (Least Well-Supported Model) *An OTSS is meaningful when it has a least well-supported model, and such a least well-supported model is its semantics.*

Semantics 6 (Unique Well-Supported Model) *An OTSS is meaningful when it has a unique well-supported model, and its semantics is its unique well-supported model.*

The following theorem implies that the above two notions of semantics actually coincide.

Theorem 3.17 *For an OTSS $(R, <)$, any least well-supported model is a unique well-supported model.*

Proof. It suffices to consider stable models, due to Theorem 3.16. We prove that if a stable model T is the least, then it is the unique stable model, i.e., if $T \subseteq T'$ then $T = T'$, for all stable models T'. If $T \subseteq T'$ and $\phi \in T'$, then $(R, <) \vdash_p \phi$ for some proof p such that p is correct w.r.t. T'. Then, it follows from Lemma 3.8 that p is correct w.r.t. T and thus $\phi \in T$. Thus, $T' \subseteq T$ and by assumption $T \subseteq T'$ and hence, $T = T'$. ☒

A unique well-supported model is trivially the least one and hence, it follows from Theorem 3.17 that Semantics 5 and 6 coincide.

Example 3.18 The notion of least (unique) well-supported model improves the counter-intuitive consequences of Semantics 3 and 4 concerning Examples 3.12 and 3.13.

It considers the OTSS in Example 3.12 to be meaningless: Suppose that the OTSS of Example 3.12 admits a well-supported model T. If $a \xrightarrow{a} a \in T$ then there should exist a proof for it and the proof should involve the axiom $\frac{}{a \xrightarrow{a} a}$. But then, for the axiom to be correct, the rule above it should not be applicable and thus $a \xrightarrow{a} a \notin T$. Similarly, if $a \xrightarrow{a} a \notin T$, then the axiom $\frac{}{a \xrightarrow{a} a}$ is correct and thus $a \xrightarrow{a} a \in T$.

It assigns the empty set as the least (and unique) well-supported model of the OTSS in Example 3.13, which is in line with the semantics given in [9] in the setting of general TSS's.

The notion of stratification was suggested in the context of logic programming by Przymusinski in [19] and adopted in the context of SOS by [10]. This notion gives us a syntactic way to check whether a particular semantics is meaningful according to Semantics 5. Note that if a TSS is stratified, it indeed has a unique well-supported model, but stratification is not a necessary condition for the uniqueness of the well-supported model. Next, we adapt this notion to the setting of OTSS's.

Definition 3.19 (Stratification) *An OTSS $(R, <)$ is* stratified *by a function $\mathcal{S} : \Phi_{\mathbf{p}} \to \alpha$, where α is an ordinal, when for all $r = \frac{H}{\phi} \in R$, for all $\psi \in H$, $\mathcal{S}(\psi) \leq \mathcal{S}(\phi)$ and for all $r' = \frac{H'}{\phi'} \in higher(r)$ and for all $\psi' \in H'$, $\mathcal{S}(\psi') < \mathcal{S}(\phi)$.*

Theorem 3.20 *A stratified OTSS $(R, <)$ has a unique well-supported model.*

Proof. Let OTSS $(R, <)$ be *stratified* by $\mathcal{S} : \Phi_{\mathbf{p}} \to \alpha$. We define a transition relation $\to$ as follows:

Definition 3.21 *Define $\to_i$, for each $i < \alpha$ as $\bigcup_j \to_{ij}$ where $\to_{ij}$ is the largest set satisfying $\phi \in \to_{ij}$, when $\exists_{r = \frac{H}{\phi} \in R}$ such that,*

(1) $\mathcal{S}(\phi) = i$,
(2) $H \subseteq \rightarrow_{kl}$, *for some* $k \leq i$ *and* $l < j$, *and*
(3) for all rules $\dfrac{H'}{\phi'} \in \mathit{higher}(r)$, *there exists* $\psi' \in H'$ *such that* $\psi' \notin \rightarrow_{mn}$, *for all* $m < i$, $n < j$.

Then, define $\rightarrow\; = \bigcup_{i<\alpha} \rightarrow_i$.

We claim that $\rightarrow$ is a well-supported model for $(R, <)$. The proof of this claim goes essentially along the same lines as the proof of a similar theorem (Theorem 2.15) in [10]. We prove that $\phi \in\; \rightarrow$ if and only if $\vdash_p \phi$ for a proof p that is correct w.r.t. $\rightarrow$.

$\Rightarrow$ If $\phi \in\; \rightarrow$, then there exists an i such that $\phi \in\; \rightarrow_i$ and $\phi \notin\; \rightarrow_k$ for each $k < i$. Furthermore, there exists a j such that $\phi \in\; \rightarrow_{ij}$ and $\phi \notin\; \rightarrow_{ik}$ for each $k < j$. We prove that if $\phi \in\; \rightarrow_{ij}$ (for such minimal i and j) then $\vdash_p \phi$ for a proof p that is correct w.r.t. $\rightarrow$. We proceed with an induction on i and inside that, an induction on j.

If $\phi \in\; \rightarrow_{ij}$, then there exists a rule $r = \dfrac{H}{\phi}$ such that $H \subseteq\; \rightarrow_{kl}$ for some $k \leq i$ and $l < j$. Thus, by the induction hypothesis, for each $\psi \in H$, $\vdash_q \psi$ for some proof q such that q is correct w.r.t. $\rightarrow$. It only remains to check that r is correct w.r.t. $\rightarrow$. Consider a rule $r' = \dfrac{H'}{\phi'} \in \mathit{higher}(r)$. Since $\phi \in\; \rightarrow_{ij}$, it follows from Definition 3.19 that for all $\psi' \in H'$, $\mathcal{S}(\psi') < i$. Hence, if r is correct w.r.t. $\rightarrow_{mn}$ for all $m \leq i$ and $n < j$, then r is correct w.r.t. $\rightarrow$ since $\psi' \notin \overset{m'n'}{\rightarrow}$ for each $m' \geq i$ by item 1 of Definition 3.21. It follows from item 3 of Definition 3.21 that there exists a $\psi' \in H'$ such that $\psi' \notin\; \rightarrow_{mn}$ for all $m \leq i$ and $n < j$ and this concludes the proof of this item.

$\Leftarrow$ If $\vdash_p \phi$ and p is correct w.r.t. $\rightarrow$, then $\phi \in\; \rightarrow$. The theorem follows by an induction on $\mathcal{S}(\phi)$, and inside that by an induction on the depth of the proof p for ϕ. It is easy to check that $\phi \in\; \rightarrow_{\mathcal{S}(\phi)j}$ where j is the depth of the proof for ϕ: the first item of Definition 3.21 holds vacuously; the second item follows from the induction hypothesis and the third item holds since p is correct with respect to $\rightarrow$ and thus with respect to all $\rightarrow_{ij}$.

It remains to show that this stable model is indeed unique. Assume that there exists another stable model T'. Consider the set $D = (T \setminus T') \cup (T' \setminus T)$; we show that $D = \emptyset$. Define $D_i = \{\phi \mid \phi \in D \wedge \mathcal{S}(\phi) = i\}$ and let j be the smallest j such that $D_j \neq \emptyset$, i.e., for all $k < j$, $D_k = \emptyset$. Hence, there exists a $\phi \in D_j$ and there exists a proof p such that $\vdash_p \phi$ and p is correct with respect to T but not correct w.r.t. T' or vice versa. Therefore, there exist rules $r \in \mathit{rules}(p)$ and $r' = \dfrac{H'}{\phi'} \in \mathit{higher}(r)$ such that $H' \nsubseteq T$ but $H' \subseteq T'$ (or vice versa). But then there exists a $\psi \in H'$ such that $\psi \notin T$ but $\psi \in T$. Since $r' \in \mathit{higher}(r)$, $j > \mathcal{S}(\psi)$ and thus, $D_{\mathcal{S}(\psi)} \neq \emptyset$ which contradicts the minimality of j. $\boxtimes$

The following example gives a reason why one may consider Semantics 5 (and thus, all two-valued semantics for that matter) inappropriate.

Example 3.22

$\downarrow$	$\circlearrowright$ $\dfrac{a \xrightarrow{a} a}{b \xrightarrow{b} b}$	$\circlearrowright$ $\dfrac{b \xrightarrow{b} b}{a \xrightarrow{a} a}$	$\circlearrowright$ $\dfrac{a \xrightarrow{a} a}{a \xrightarrow{a} a}$
	$\dfrac{}{b \xrightarrow{b} b}$	$\dfrac{}{a \xrightarrow{a} a}$	$\dfrac{}{a \xrightarrow{a} a}$

The least well-supported model for the above OTSS is $\{a \xrightarrow{a} a\}$. This is also its least model and least supported model. Note that firstly, $\emptyset$ is not supported (and thus not stable) and not a model since the absence of $a \xrightarrow{a} a$ would result in enabling the axiom $\dfrac{}{a \xrightarrow{a} a}$ in the third column. Secondly, $b \xrightarrow{b} b$ cannot be included in the stable (and even supported) model since then axiom $\dfrac{}{b \xrightarrow{b} b}$ should be enabled and thus, $a \xrightarrow{a} a$ should not be in the model, which we have just shown to lead to a contradiction.

Thus, the notion of two-valued stable model defines a semantics for the intuitively paradoxical OTSS in the above example (due to its third column). Moreover, the third column in the above example, although paradoxical in nature, makes the stable model favor one of the two, otherwise equal, possibilities of including either $a \xrightarrow{a} a$ or $b \xrightarrow{b} b$.

This may be considered counter-intuitive and one may want a solution which rejects such OTSS's due to the paradoxical nature of its third column (when there is no other reason to believe that $a \xrightarrow{a} a$ certainly holds) and moreover, regardless of the third column, treats $a \xrightarrow{a} a$ and $b \xrightarrow{b} b$ equally (i.e., considers both of them possible but not certain). None of the two-valued solutions presented so far can provide us with such a meaning, and thus we proceed with three-valued solutions in the next section.

Figure 1 gives a comparison of the notions of semantics presented so far. The topmost notion is the notion of semantics which only assigns a meaning, namely the set of all provable formulae, to OTSS's with an empty ordering. When there is a solid arrow between two notions of semantics it means that for all OTSS's that the source notion provides a meaning, the target notion provides the same meaning and there are OTSS's for which the source notion does not provide a meaning while the target notion does. The dashed arrows show that the transition relation associated by the semantics in the source of the arrow is a subset of provable transitions, i.e., Semantics 1. All counter-examples showing the differences among unrelated notions are given before. It only remains to prove the arrow relations. The solid arrow between the unique supported model and the least supported model is trivial. Proofs of theorems concerning the other arrows, given below, are very easy. Examples 3.12 and 3.13 show that Semantics' 3 and 4 are unrelated to the rest.

Theorem 3.23 *For an OTSS $(R, <)$ with the least model T, if T' is the set of its provable formulae, then $T \subseteq T'$. Furthermore, if $< = \emptyset$ then $T = T'$.*

Proof. T' is a model and it follows from the assumption (i.e., T is the least model) that $T \subseteq T'$.

If $< = \emptyset$, the set of provable formulae is indeed a model and no proper subset of it constitutes a model (by a proof by contradiction, considering the least proof depth of the element excluded from the subset). ☒

Theorem 3.24 *For an OTSS $(R, <)$ with a unique stable model T, if T' is the set of its provable formulae, then $T \subseteq T'$. Furthermore, if $< = \emptyset$ then $T = T'$.*

Proof. Any formula $\phi \in T$ has a proof and thus $\phi \in T'$.

When $< = \emptyset$, all $r \in R$ are correct w.r.t. any transition relation and thus Semantics 1 and 5 coincide and hence $T = T'$. ☒

3.2 Three-Valued Solutions

Three-valued solutions assign three transition relations to each OTSS: the set of transitions that are *certainly* derivable, denoted by C; transitions that are *possibly* derivable, denoted by P (thus $C \subseteq P$); and the set of transitions that are impossible, denoted by I. Possibly derivable formulae and impossible ones partition the set of formulae. Hence, three-valued solutions may be written as pairs of these sets, i.e., (C, P) or (C, I), with the third component determined easily from the given ones. On such pairs of sets of formulae, the following ordering is used frequently in the remainder:

$$(C, P) \preceq (C', P') \doteq C \subseteq C' \wedge P' \subseteq P$$

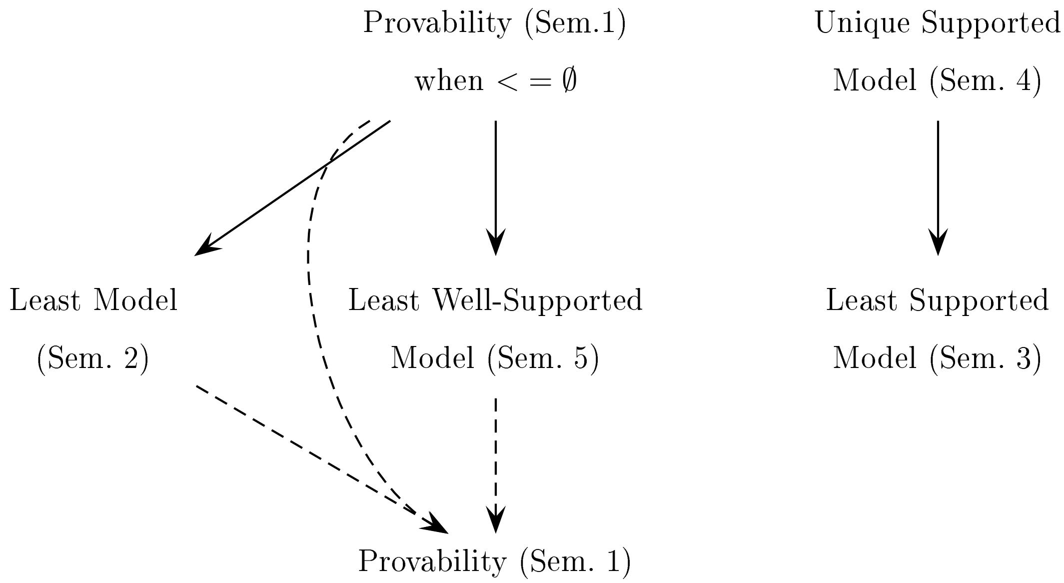


Fig. 1. A Comparison of the Two-Valued Notions of Semantics

The first three-valued solution is based on the following notion of three-valued stable model.

Definition 3.25 (Three-Valued Stable Model) *Given an OTSS $(R, <)$, a pair of transition relations (C, P) is a three-valued stable model when $C \subseteq P$ and*

(1) $\phi \in C \Leftrightarrow \vdash_p \phi$ for some proof p such that p is correct w.r.t. P and
(2) $\phi \in P \Leftrightarrow \vdash_p \phi$ for some proof p such that p is correct w.r.t. C.

The third value of the stable model I, for impossible, is the set of transitions that are not included in P.

Semantics 7 (Least Three-Valued Stable Model) *Any OTSS is meaningful, and its meaning is the least (w.r.t. $\preceq$) three-valued stable model (C, P).*

The following reduction technique [6] is a method to calculate the least three-valued model (thus it shows that such a minimal model indeed exists).

Definition 3.26 (Reduction Technique) *For an ordinal α, define:*

$$C_\alpha \doteq \{\phi \mid \exists_p \ (R, <) \vdash_p \phi \ \wedge \ \exists_{\beta < \alpha} \ p \textit{ is correct w.r.t. } P_\beta\}$$
$$P_\alpha \doteq \{\phi \mid \exists_p \ (R, <) \vdash_p \phi \ \wedge \ \forall_{\beta < \alpha} \ p \textit{ is correct w.r.t. } C_\beta\}$$

Note that it immediately follows from the above definition that $C_0 = \emptyset$ and P_0 is the set of all provable formulae.

Lemma 3.27 *Given an OTSS $(R, <)$, for all ordinals α and β such that $\alpha \leq \beta$, the following statements hold:*

(1) $C_\alpha \subseteq C_\beta$;
(2) $P_\beta \subseteq P_\alpha$;
(3) $C_\alpha \subseteq P_\alpha$.

Proof.

(1) $\phi \in C_\alpha \Rightarrow \vdash_p \phi \wedge \exists_{\alpha'<\alpha}\, p$ is correct w.r.t. $P_{\alpha'} \Rightarrow \vdash_p \phi \wedge \exists_{\alpha'<\beta}\, p$ is correct w.r.t. $P_{\alpha'}$ $\Rightarrow \phi \in C_\beta$;

(2) $\phi \in P_\beta \Rightarrow \vdash_p \phi \wedge \forall_{\alpha'<\beta}\, p$ is correct w.r.t. $C_{\alpha'} \Rightarrow \vdash_p \phi \wedge \forall_{\alpha'<\alpha}\, p$ is correct w.r.t. $C_{\alpha'}$ $\Rightarrow \phi \in P_\alpha$;

(3) By a transfinite induction on α.

Let $\phi \in C_\alpha$. Then, $(R, <) \vdash_p \phi \wedge \exists_{\alpha'<\alpha}\, p$ is correct w.r.t. $P_{\alpha'}$; we need to prove that for all $\gamma < \alpha$, p is correct w.r.t. C_γ. We distinguish the following two cases for γ:

(a) $\alpha' < \gamma < \alpha$: then, it follows from item 2 that p is correct w.r.t. P_γ. It then follows from the induction hypothesis that p is correct w.r.t. C_γ.

(b) $\gamma \leq \alpha'$: then, it follows from the induction hypothesis that p is correct w.r.t. $C_{\alpha'}$ and from item 1 that p is correct w.r.t. C_γ.

⊠

It follows from items 1 and 2 of the above lemma (and Tarski's fixed point theorem) that (C_α, P_α) will reach the least fixed point, which we denote by (C, P). From item 3 and Definition 3.26, it follows that (C, P) is a three-valued stable model of the OTSS under consideration. Furthermore, any three-valued stable model is a fixed point for the equations in Definition 3.26 and hence (C, P) is the least three-valued stable model of the OTSS.

Example 3.28 Example 3.22 gave us a reason to reject all (two-valued) notions of semantics. Semantics 7 does not suffer from such problems. Semantics 7 assigns $(\emptyset, \{a \xrightarrow{a} a, b \xrightarrow{b} b\})$ to the OTSS of Example 3.22, which is sensible because firstly it does not favor either of the two transitions $a \xrightarrow{a} a$ or $b \xrightarrow{b} b$ over the other, and secondly it does not consider either of the two transitions certain.

The proof-theoretic counterpart to the three-valued stable model is the notion of well-supported proof which is introduced later in this section. However, before we go to proof-theoretic solutions, we re-phrase another three-valued model-theoretic notion from [9] called the three-valued supported model.

Definition 3.29 (Three-Valued Supported Model) *Given an OTSS $(R, <)$, a pair of* *transition relations (C,P) is a three-valued supported model when*

(1) $\phi \in C$ if and only if there exists a deduction rule $r = \dfrac{H}{\phi} \in R$ such that $H \subseteq C$ and r is correct w.r.t. P;

(2) $\phi \in P$ if and only if there exists a deduction rule $r = \dfrac{H}{\phi} \in R$ such that $H \subseteq P$ and r is correct w.r.t. C.

As before, the third value of the model, i.e., the set I of impossible transitions, contains precisely those transitions that are not included in P.

Semantics 8 (Least Three-Valued Supported Model) *Any OTSS is meaningful and its meaning is the least (w.r.t. $\preceq$) three-valued supported model (C,P).*

The following example illustrates the difference between the three-valued stable model semantics and its supported counterpart, and motivates why we chose the three-valued stable model as our preferred notion of semantics.

Example 3.30 Consider the OTSS of Example 3.13. The least three-valued stable model of this OTSS is $C = \emptyset$ and $P = \emptyset$. However, its least three-valued supported model is $C = \emptyset$ and $P = \{a \xrightarrow{a} a\}$.

The OTSS of Example 3.3 gives us another reason to chose the three-valued stable model over its supported counterpart. The least three-valued stable model of this OTSS is $C = \{b \xrightarrow{b} b\}$ and $P = \{b \xrightarrow{b} b\}$, which we have already considered intuitive. However, the least three-valued supported model of this OTSS is $C = \emptyset$, $P = \{a \xrightarrow{a} a, b \xrightarrow{b} b\}$, which is rather counter-intuitive.

To summarize the above examples, the notion of least three-valued supported model is sensitive to deduction rules with their conclusion included among their premises, which we do not consider intuitive. The following lemma states that the least three-valued stable model does not suffer from this shortcoming.

Lemma 3.31 *Consider two disjoint sets R and R' of deduction rules and two orderings $< \subseteq R \times R$ and $<' \subseteq R' \times R'$. If for all $r' \in R'$, $conc(r') \in prem(r')$, then the least three-valued stable models of two OTSS's $(R,<)$ and $(R \cup R', < \cup <')$ coincide.*

Proof. For an arbitrary ordinal α, let (C_α, P_α) and (C'_α, P'_α) be the results of the reduction technique of Definition 3.26 for $(R,<)$ and $(R \cup R', < \cup <')$, respectively. It is easy to see, by considering all minimal proofs, that rules in R' cannot contribute any new provable transition formulae to those of R. Hence, by a transfinite induction on α, $C_\alpha = C'_\alpha$ and

$P_\alpha = P'_\alpha$. ☒

For OTSS's placing a rule above itself we have the following general result, which states that adding a set of rules which are placed above themselves to an OTSS can only make the set of certain transitions smaller and the set of possible transitions larger. Later (in Section 3.3), we give a sufficient condition, called *completeness*, under which adding rules with self-loops (i.e., placed above themselves by the ordering) does not change the least three-valued stable model of a TSS.

Lemma 3.32 *Consider two disjoint sets R and R' of deduction rules and two orderings $< \subseteq R \times R$ and $<' \subseteq R' \times R'$. If R' is reflexive and (C, P) and (C', P') are the least three-valued stable models for $(R, <)$ and $(R \cup R', < \cup <')$, respectively, then $(C', P') \preceq (C, P)$.*

Proof. See the first part of the proof of Lemma 3.40. ☒

The following notion of well-supported proof provides a proof-theoretic parallel to the three-valued stable model.

Definition 3.33 (Well-Supported Proof) *Given an OTSS $(R, <)$, a* well-supported r-proof *(or just a* well-supported proof*) for ϕ is a well-founded upwardly branching tree of which*

(1) the nodes are formulae,
(2) the root is ϕ,
(3) if a node is labelled ϕ' and the nodes above it form the set K, then there is a deduction rule $r' \in R$ such that $r' = \frac{K'}{\phi'}$ for some $K' \subseteq K$ (for the root node, $r' = r$), and for all $r'' = \frac{H'}{\psi} \in higher(r')$, there exists a set $D_{\psi'} \subseteq K$ denying some $\psi' \in H'$ by a well-supported proof.

A set D_ϕ denies *a formula ϕ by a well-supported proof if for all proofs p such that $(R, <) \vdash_p \phi$, there exists a rule $r \in rules(p)$ and there exists a rule $r' = \frac{H'}{\phi'} \in higher(r)$ such that $H' \subseteq D_\phi$. The structure providing a well-supported proof for all $\psi \in D_\phi$ is called a* well-supported denial *for ϕ.*

We write $(R, <) \vdash_{\mathrm{ws}} \phi$ ($(R, <) \vdash_{\mathrm{ws}} \neg\phi$) when there is a well-supported proof (denial) for ϕ. We may drop $(R, <)$ in the above two notations when it is clear from the context.

One can use the above notion of well-supported proof to give ordered SOS a semantics as follows.

Semantics 9 (Well-Supported Provability) *Any OTSS is meaningful, and its meaning is the pair (C, P) where C is the set of all formulae that have a well-supported proof and P is the set of all formulae that do not have a well-supported denial.*

The following is an auxiliary lemma that allows us to strip down well-supported proofs to proofs.

Lemma 3.34 *For all $\phi \in \Phi_{\mathbf{p}}$, if $\vdash_{\mathrm{ws}} \phi$ then there exists a proof p such that $\vdash_p \phi$ and furthermore, for all $r \in rules(p)$ and for all $r' = \frac{H'}{\phi'} \in higher(r)$ there exists a $\psi \in H'$ such that $\vdash_{\mathrm{ws}} \neg\psi$ with a well-supported denial that is a sub-structure of the well-supported proof of ϕ.*

Proof. In the above definition of well-supported proof, it suffices to inductively keep $K' \subseteq K$ for the nodes above each node ϕ' such that $\frac{K'}{\phi'} \in R$ and remove $K \setminus K'$. The resulting tree is a proof tree. The rest of the corollary follows from the definition of well-supported proof. ⊠

The following lemma states that the notion of well-supported proof is consistent, i.e., no formula can be both proven and denied.

Lemma 3.35 *The notion of well-supported proof is consistent, i.e., no formula ϕ has both a well-supported proof and a well-supported denial.*

Proof. Assume that ϕ is a formula with the minimal proof and denial structures (i.e., there is no formula ψ that has proof and denial structures that are both parts of the proof and denial structures of ϕ). We show that such a minimal structure does not exist and thus well-supported proof is consistent.

Since ϕ has a well-supported denial, for all proofs p such that $\vdash_p \phi$ there exists a rule $r \in rules(p)$ and there exists a rule $r' = \frac{H'}{\phi'} \in higher(r)$ such that all $\psi \in H'$ have a well-supported proof. However, since ϕ has a well-supported proof, by Lemma 3.34 it has a proof p such that for all rules $r \in rules(p)$ and for all $r' = \frac{H'}{\phi'} \in higher(r)$ there exists a $\psi \in H'$ such that $\vdash_{\mathrm{ws}} \neg\psi$ with a well-supported denial that is a sub-structure of the well-supported proof of ϕ. Thus, there exists ψ that has both a well-supported denial and a well-supported proof, both of which are sub-structures of the well-supported proof and denial for ϕ, respectively. (Contradiction) $\boxtimes$

The following theorem states that the model-theoretic and the proof-theoretic views of the least well-supported semantics, i.e., Semantics 7 and 9, indeed match.

Theorem 3.36 *Given an OTSS $(R, <)$, let (C, P) be the least three-valued stable model of $(R, <)$, C' be the set of all formulae that have a well-supported proof, and I' the set of all formulae that have a well-supported denial. Then, $(C, P) = (C', \Phi_{\mathbf{p}} \setminus I')$.*

Proof. To show simultaneously that $C' \subseteq C$ and $P \subseteq \Phi_{\mathbf{p}} \setminus I'$, we prove, by an induction on the structure of the well-supported proof (denial) for ϕ, that that if $\vdash_{\mathrm{ws}} \phi$ ($\vdash_{\mathrm{ws}} \neg\phi$) then $\exists_{\alpha>0} \phi \in C_\alpha$ ($\exists_{\alpha>0} \phi \notin P_\alpha$).

(1) If $\vdash_{\mathrm{ws}} \phi$ then there exists a proof p such that $\vdash_p \phi$ and $\forall_{r \in rules(p)}$ $\forall_{r' = \frac{H'}{\phi'} \in higher(r)}$, there exists a $\psi_{r'} \in H'$ such that with a smaller well-supported denial structure (than the proof structure of ϕ) it holds that $\vdash_{\mathrm{ws}} \neg\psi_{r'}$. It follows from the induction hypothesis that there exists some $\alpha_{r'}$ such that $\psi_{r'} \notin P_{\alpha_{r'}}$. For each r, let α_r be the maximum of all such $\alpha_{r'}$'s and it follows that r is correct w.r.t. α_r. For all $r \in rules(p)$ take α to be the maximum of all α_r's and $\phi \in C_\alpha \subseteq C$.

(2) If $\vdash_{\mathrm{ws}} \neg\phi$, then if there is no proof for ϕ or for all proofs p such that $\vdash_p \phi$, $\exists_{r \in rules(p)}$ $\exists_{\alpha_r > 0}$ r is not correct w.r.t. C_{α_r} then $\exists_\alpha \phi \notin P_\alpha$ (by taking α to be the maximum of such α_r).

Assume that the above is not the case, i.e., assume that there exists a proof p such that $\vdash_p \phi$, $\forall_{r \in rules(p)}$ $\forall_{\alpha>0}$ r is correct w.r.t. C_α. From $\vdash_{\mathrm{ws}} \neg\phi$, it follows that $\exists_{r \in rules(p)}$ and $\exists_{r' = \frac{H'}{\phi'} \in higher(r)}$ such that all $\psi \in H'$ have a well-supported proof smaller than that of ϕ. Thus, it follows from the induction hypothesis that $H' \subseteq C_\alpha$ for some α. Hence r is not correct w.r.t. C_α. (Contradiction)

For the proof in the other direction, i.e., $C \subseteq C'$ and $\Phi_{\mathbf{p}} \setminus I' \subseteq P$, we use an induction on α and show (again simultaneously) that if $\phi \in C_\alpha$ then $\vdash_{\mathrm{ws}} \phi$ and if $\phi \notin P_\alpha$ then $\vdash_{\mathrm{ws}} \neg\phi$.

(1) If $\phi \in C_\alpha$ then $\vdash_p \phi$ for some p and $\exists_{\beta<\alpha}$ $\forall_{r \in rules(p)}$ such that r is correct w.r.t. P_β. In other words, $\forall_{s = \frac{H'}{\phi'} \in higher(r)}$ $\exists_{\psi_s \in H'}$ such that $\psi_s \notin P_\beta$. It follows then from the induction hypothesis that $\vdash_{\mathrm{ws}} \neg\psi_s$. Extending the premises of each rule r in the proof structure p with all such ψ_s (and their well-supported denials) gives rise to a well-supported proof for ϕ.

(2) If $\phi \notin P_\alpha$ then either there is no proof p such that $\vdash_p \phi$ or for all proofs p such that

$\vdash_p \phi$, $\exists_{r \in rules(p)} \exists_{s=\frac{H'_s}{\phi'_s} \in higher(r)} \exists_{\beta<\alpha} H'_s \subseteq C_\beta$. In the former case, the empty set denies ϕ and thus we have a well-supported denial for ϕ. In the latter case, following the induction hypothesis, all $\psi \in H'_s$ have a well-supported proof and the union of all H'_s for all proofs p of ϕ (together with their well-supported proofs) constitutes a well-supported denial for ϕ.

⊠

Finally and for sake of completeness, we give a proof-theoretic interpretation of three-valued supported models below.

Definition 3.37 (Supported Proof) *Given an OTSS $(R, <)$, a* supported proof *for ϕ is a well-founded upwardly branching tree of which*

(1) the nodes are formulae,
(2) the root is ϕ,
(3) if a node is labelled ϕ' and the nodes above it form the set K, then there is a deduction rule $r \in R$ such that $r = \frac{K'}{\phi'}$ for some $K' \subseteq K$ and for all $r'' = \frac{K''}{\phi''} \in higher(r)$, there exists a $D_{\psi''} \subseteq K$ denying a formula $\psi'' \in K''$ by a supported proof.

A set of formulae D_ϕ denies a formula ϕ by a supported proof when for all deduction rules $r = \frac{H}{\phi}$, there exists a deduction rule $r' = \frac{H'}{\phi'} > r$ such that $H' \subseteq D_\phi$.

A formula ϕ is denied by a supported proof when there exists a set D_ϕ which denies ϕ and all $\phi' \in D_\phi$ have a supported proof.

As before, the above notion of supported proof can be used to give ordered SOS a semantics.

Semantics 10 (Supported Provability) *Any OTSS is meaningful, and its meaning is the pair (C, P) where C is the set of all formulae that have a supported proof and P is the set of all formulae that do not have a supported denial.*

The next theorem shows that Semantics 8 and 10 coincide.

Theorem 3.38 *Given an OTSS $(R, <)$, let C' be the set of all formulae that have a supported proof and I' the set of all formulae that are denied by a supported proof. Let (C, P) be the least three-valued supported model of $(R, <)$. Then, $(C', \Phi_{\mathbf{p}} \setminus I') = (C, P)$.*

Proof. Similar to the proof of Theorem 3.36. ⊠

An important class of OTSS's are the complete OTSS's. They enjoy two important properties: the induced bisimilarity is a congruence, and they can be given a straightforward two-valued semantics.

Definition 3.39 (Completeness) *An OTSS is called* complete *when for its least three-valued stable model* (C, P) *it holds that* $C = P$*.*

For complete OTSS's placing a rule above itself makes the rule inapplicable and does not influence the least three-valued stable model. This is important because it allows us to encode conveniently the full power of negative premises (see an alternative translation from NTYFT to OTYFT at the end of Section 5.1).

Lemma 3.40 *Consider two disjoint sets* R *and* R' *of deduction rules and two orderings* $< \subseteq R \times R$ *and* $<' \subseteq R' \times R'$*. If* $\{(r', r') \mid r' \in R'\} \subseteq <'$ *and* $(R, <)$ *is complete, then the least three-valued stable models of the two OTSS's* $(R, <)$ *and* $(R \cup R', < \cup <')$ *coincide.*

Proof. Let (C_α, P_α) be the result of the reduction technique of Definition 3.26 for $(R, <)$ and (C'_α, P'_α) that for $(R \cup R', < \cup <')$. We prove by an induction on α that the following two statements hold (in fact, this part of the proof does not rely on the completeness of $(R, <)$):

(1) $C'_\alpha \subseteq C_\alpha$;
(2) $P_\alpha \subseteq P'_\alpha$.

(1) Take an arbitrary $\phi \in C'_\alpha$. Then $(R \cup R', < \cup <') \vdash_p \phi$ and $\exists_{\beta<\alpha}\, p$ is correct w.r.t. P'_β and hence, by Lemma 3.8 and by the induction hypothesis ($P_\beta \subseteq P'_\beta$), p is correct w.r.t. P_β. If there is a deduction rule $r' \in R'$ in the proof structure p, then for all $\phi' \in prem(r')$, $\phi' \in C'_\alpha$ and it follows from items 2 and 3 of Lemma 3.27 that $\phi' \in P'_\beta$. Hence, r' is not correct w.r.t. P'_β, and thus a contradiction follows. We can thus assume that all rules in the proof structure p are in R.

We proceed by an induction on the proof structure p. Consider the last deduction rule $r \in R$ applied in the proof structure p. Then by the induction hypothesis, for all $\phi' \in prem(r)$, $\phi' \in C_\alpha$ and hence, $(R, <) \vdash_{p_{\phi'}} \phi'$ for some proof $p_{\phi'}$ which is correct w.r.t. P'_β (and thus, P_β). Hence, by applying the same deduction rule, we get a proof for ϕ for which is correct w.r.t. P_β and thus, $\phi \in C_\alpha$.

(2) Similarly, take a $\phi \in P_\alpha$. Then $(R, <) \vdash_p \phi$ and $\forall_{\beta<\alpha}\ p$ is correct w.r.t. C_β and hence, by the induction hypothesis, p is correct w.r.t. C'_β. Proof p is also a proof w.r.t. $(R \cup R', < \cup <')$ and hence, the thesis follows.

Let (C, P) and (C', P') be the least three-valued stable models of $(R, <)$ and $(R \cup R', < \cup <')$, respectively. We first prove the following two statements.

(1) $\phi \in P' \setminus C \Rightarrow$ for all proofs p such that p is correct w.r.t. C and $(R \cup R', < \cup <') \vdash_p \phi$, there exists a node ϕ' in p labelled by a formula $\phi' \in P \setminus C$;
(2) $\phi \in C \setminus C' \Rightarrow$ for all proofs p if p is correct w.r.t. C and $(R, <) \vdash_p \phi$, there exists a deduction rule $r \in p$ such that for some $r' \in \mathit{higher}(r)$ and a $\phi' \in \mathit{prem}(r')$, $\phi' \in P' \setminus C$.

(1) We proceed by an induction on the proof p. If the latest deduction rule in p is $r \in R$; if all premises of r are in C then since p is correct w.r.t. C, ϕ should also be in C which is a contradiction. Thus, there exists a formula $\phi' \in \mathit{prem}(r)$ such that $\phi \in P' \setminus C$. If $r \in R'$ and all the premises of R are in C, then r is not correct w.r.t. C; hence, there exists a formula such that $\phi' \in \mathit{prem}(r)$ such that $\phi' \in P' \setminus C$ and by induction hypothesis, there is a formula in the sub-proof of p for ϕ' that is in $P \setminus C$.
(2) Suppose that there exists a proof p such that $(R, <) \vdash_p \phi$ and for all $r \in \mathit{rules}(p)$, for all $r' \in \mathit{higher}(r)$, and for all $\phi' \in \mathit{prem}(r')$, $\phi' \notin P' \setminus C$. Then, it follows from the correctness of p w.r.t. C that for all such r', there exists at least one formula ϕ'' such that $\phi'' \notin C$ and since $\phi'' \notin P' \setminus C$, $\phi'' \notin P'$. Thus, p is also correct w.r.t. P' and thus, $\phi \in C'$ which is a contradiction.

It follows from the completeness of $(R, <)$ that $C = P$. Towards a contradiction, assume that $P' \setminus P \neq \emptyset$; then there exists a $\phi \in P' \setminus P$. It then follows from item 1 above, that for all proofs p such that $(R \cup R', < \cup <') \vdash_p \phi$, there exists a formulae $\phi' \in p$ such that $\phi' \in P \setminus C$. But since $C = P$, $\phi' \in P \setminus P$, and hence, $\phi' \in \emptyset$ which is a contradiction. It also follows, by the same line of reasoning, from item 2 and the fact that $P' \setminus C = P' \setminus P = \emptyset$, that $C \setminus C' = \emptyset$. Hence, we conclude that $C \subseteq C'$ and $P' \subseteq P$. From the first part of the proof, we have that $C' \subseteq C$ and $P \subseteq P'$ and thus, we conclude that $C = C' = P = P'$, which was to be proven. ⊠

It follows from the above lemma that the original OTSS is complete if the extended OTSS (with rules containing self-loops) is complete. Lemma 3.40 does not hold if the extended OTSS is not complete. In other words, adding rules with self-loops to an incomplete OTSS may influence its three-valued stable models. The following example illustrates this fact.

Example 3.41 Consider the following (incomplete) OTSS.

$\downarrow$	$\dfrac{b \xrightarrow{b} b}{b \xrightarrow{b} b}$	$\dfrac{c \xrightarrow{c} c}{d \xrightarrow{d} d}$	$\dfrac{d \xrightarrow{d} d}{c \xrightarrow{c} c}$
	$\dfrac{}{a \xrightarrow{a} a}$	$\dfrac{}{d \xrightarrow{d} d}$	$\dfrac{}{c \xrightarrow{c} c}$

Its least three valued stable model is $(\{a \xrightarrow{a} a\}, \{a \xrightarrow{a} a, c \xrightarrow{c} c, d \xrightarrow{d} d\})$. Suppose that we add the following OTSS to the one specified above.

$\circlearrowleft$	$\dfrac{c \xrightarrow{c} c}{b \xrightarrow{b} b}$

Then, the least three-valued stable model of the extended OTSS becomes $(\emptyset, \{b \xrightarrow{b} b, c \xrightarrow{c} c, d \xrightarrow{d} d\})$.

An intuitive way of exploiting the minimum three-valued stable models to give a two-valued semantics to OTSS's is to rule out OTSS's in which the C (certain) part is different from the P (probable) part on the basis that the OTSS does not say anything useful about the formulae in $P \setminus C$, i.e., it neither rejects them nor considers them certain. The following notion of complete OTSS's captures this intuition.

Semantics 11 (Complete) *An OTSS is meaningful when it is complete, i.e., for its least three-valued stable model (C, P) it holds that $C = P$, and its meaning is the least three-valued stable model.*

Example 3.42 Semantics 11 rejects the OTSS of Example 3.22 since the first two components of the least three-valued stable model differ, namely, they are $C = \emptyset$ and $P = \{a \xrightarrow{a} a, b \xrightarrow{b} b\}$ and thus the OTSS cannot unequivocally define a transition relation.

The least three-valued stable model is our preferred notion of semantics, and in order to obtain congruence meta-results (discussed in Section 4) we restrict our attention to complete OTSS's. (Note that stratification, as defined in Definition 3.19, gives us a syntactic criterion to prove completeness.) In our view, for all practical applications the OTSS under consideration should be complete or should be rejected, since the most basic properties such as congruence of bisimilarity cannot be guaranteed. However, one might want to generalize Semantics 11 to the following notion of irrefutability which assigns a two-valued transition relation to all OTSS's.

Semantics 12 (Irrefutable) *All OTSS are meaningful and their meaning is the P component of their least three-valued stable model.*

Example 3.43 Semantics 12 assigns $\{a \xrightarrow{a} a, b \xrightarrow{b} b\}$ as the meaning to the OTSS of Example 3.22.

4 Congruence Rule Formats

A major type of meta-result obtained by the field of rule formats is to guarantee the congruence property, as defined below, for different notions of behavioral equality.

Definition 4.1 (Congruence) *An equivalence relation R is a* congruence *w.r.t. a function symbol f (with an arbitrary arity n), when for all $p_0, q_0, \ldots, p_{n-1}, q_{n-1}$, if $(p_0, q_0), \ldots, (p_{n-1}, q_{n-1}) \in R$ then $(f(p_0, \ldots, p_{n-1}), f(q_0, \ldots, q_{n-1})) \in R$. R is a congruence w.r.t. a signature Σ when it is a congruence for all function symbols $f \in \Sigma$.*

When the signature is clear from the context, we shall just write that R is a congruence.

The notion of behavioral equivalence that is used most throughout the rest of this paper is the following notion of strong bisimilarity.

Definition 4.2 (Strong Bisimilarity) *A symmetric relation $R \subseteq \mathcal{C} \times \mathcal{C}$ is a* strong bisimulation relation *when for all $(p, q) \in R$, $l \in \mathcal{C}$, and $p' \in \mathcal{C}$, if $p \xrightarrow{l} p'$ then there exists a q', $q \xrightarrow{l} q'$ and $(p', q') \in R$. Two closed terms p and q are* strongly bisimilar *(or just bisimilar), denoted by $p \underline{\leftrightarrow} q$, when there exists a strong bisimulation relation R such that $(p, q) \in R$.*

In the remainder of this section, we shall first quote two general congruence rule formats for strong bisimilarity in the setting with negative premises and then present their counterparts in the setting with ordering. A study of their relative expressiveness will follow afterwards.

4.1 Congruence Formats for SOS

In [20], De Simone started a line of research which aims at defining syntactic schema for TSS's which guarantee certain properties such as congruence of strong bisimilarity. Two distinguished examples of such formats are the GSOS [5] and the NTYFT [10] formats, both of which allow for specifying negative transition formulae among premises. Moreover, NTYFT accommodates the challenging feature of look-ahead which as we shall also observe leads to more expressive power. Next, we define the GSOS and the NTYFT formats.

Definition 4.3 (GSOS) *A deduction rule is in the* GSOS *format if and only if it has the following form:*

$$\frac{\{x_i \xrightarrow{l_{ij}} y_{ij} \mid i \in I, 1 \leq j \leq m_i\} \cup \{x_j \stackrel{l'_{jk}}{\nrightarrow} \mid j \in J, 1 \leq k \leq n_j\}}{f(\overrightarrow{x}) \xrightarrow{l} t}$$

where f is a function symbol, x_i's ($1 \leq i \leq ar(f)$) and y_{ij}'s ($i \in I$ and $1 \leq j \leq m_i$) are all distinct variables, I and J are subsets of $\{1, \ldots, ar(f)\}$, m_i and n_j are natural numbers (to set an upper bound on the number of premises), and t is an arbitrary term such that $vars(t) \subseteq \{x_i \mid 1 \leq i \leq ar(f)\} \cup \{y_{ij} \mid i \in I, 1 \leq j \leq m_i\}$. A TSS is in the GSOS *format when it has a finite signature, a finite set of labels, a finite set of deduction rules and all its deduction rules are in the* GSOS *format.*

Definition 4.4 (NTYFT) *A deduction rule is in the* NTYFT *format if and only if it has the following form:*

$$\frac{\{t_i \xrightarrow{l_i} y_i \mid i \in I\} \cup \{t'_j \stackrel{l'_j}{\nrightarrow} \mid j \in J\}}{f(\overrightarrow{x}) \xrightarrow{l} t}$$

where f is a function symbol, x_i ($1 \leq i \leq ar(f)$) and y_i's ($i \in I$) are all distinct variables, I and J are (possibly infinite) sets of indices, and t, t_i's and t'_j's are arbitrary terms. A TSS is in the NTYFT *format when all its deduction rules are. A deduction rule (TSS) is in the* TYFT *format when it is positive and is in the* NTYFT *format.*

Clearly any deduction rule (TSS) in the GSOS rule format is also in the NTYFT rule format. A deduction rule (TSS) is in the *positive* GSOS rule format when it is both in the TYFT and the GSOS rule formats.

A straightforward extension of the NTYFT format allows for variables in the sources of conclusions, leading to the NTYFT/NTYXT format, which does not bring about extra expressive power since rules in the extended format can be coded in the NTYFT format [10]. The following theorem is from [6].

Theorem 4.5 (Congruence for NTYFT) *For a complete TSS in the* NTYFT *format, bisimilarity is a congruence.*

The following example is also taken from [6] and illustrates that completeness of the TSS is essential in obtaining the congruence result.

Example 4.6 Consider the following TSS:

$$\frac{}{a \xrightarrow{a} a} \qquad \frac{}{b \xrightarrow{a} b}$$

$$\frac{x \xrightarrow{a} y \quad f(x) \stackrel{c}{\nrightarrow} \quad f(a) \stackrel{d}{\nrightarrow}}{f(x) \xrightarrow{d} b} \qquad \frac{x \xrightarrow{a} y \quad f(x) \stackrel{d}{\nrightarrow} \quad f(b) \stackrel{c}{\nrightarrow}}{f(x) \xrightarrow{c} a}$$

The above TSS both is in the NTYFT format and induces a unique two-valued stable model, namely $\{a \xrightarrow{a} a,\ b \xrightarrow{a} b,\ f(a) \xrightarrow{c} a,\ f(b) \xrightarrow{d} b\}$. However, it is not complete; its least three-valued valued stable model is $(\{a \xrightarrow{a} a, b \xrightarrow{a} b\}, \{a \xrightarrow{a} a, b \xrightarrow{a} b, f(a) \xrightarrow{d} b, f(b) \xrightarrow{d} b, f(a) \xrightarrow{c} a, f(b) \xrightarrow{c} a\})$. For the unique two-valued stable model of the above transition relation, it holds that $a \underline{\leftrightarrow} b$ but it does not hold that $f(a) \underline{\leftrightarrow} f(b)$ (thus bisimilarity is not a congruence).

4.2 Congruence Formats for Ordered SOS

Our goal is to show that orderings on rules are at least as expressive (and of course complicated in nature) as negative premises. This has already been demonstrated in the case of the GSOS rule format by the OSOS rule format of [22] which is defined below. In [22], Ulidowski and Phillips show that there exists a semantics-preserving translation from the GSOS rule format to the OSOS rule format and vice versa.

Definition 4.7 (OSOS) *An OTSS $(R, <)$ is in the* OSOS *rule format when*

(1) all deduction rules in R are in the positive GSOS *rule format,*
(2) for all deduction rules $r, r' \in R$ if $r \in higher(r')$ then r and r' have the same main operator, and
(3) for all distinct deduction rules $r, r' \in R$, if $r \in higher(r')$ then $vars(trg(prem(r))) \cap vars(r') = \emptyset$ and $vars(src(prem(r))) \subseteq vars(src(conc(r')))$.

Item 3 is not explicitly present in the original definition of [22], but it is implicitly assumed. As we show in the remainder of this paper and illustrate with the examples in this section, the first statement in this item 3, i.e., $vars(trg(prem(r))) \cap vars(r') = \emptyset$, is essential in obtaining the congruence result. The second statement, i.e., $vars(src(prem(r))) \subseteq vars(src(conc(r')))$, is used for obtaining a translation between the GSOS and the OSOS formats and vice versa. Item 2 permits cyclic rules (namely rules that are placed above themselves) in the OSOS format. Cyclic rules are also part of the original OSOS [22]. However, several examples in [22] show that the subset of OSOS without cyclic rules, called *partial OSOS*, is as expressive as OSOS. We decided to include cyclic rules in the definition of OSOS in order to be compatible with the original presentation. As we show in the remainder of this paper, introducing cyclic rules to more general rule formats may cause some complications with respect to their semantics.

In this paper, we introduce the following OTYFT format which will be proved equal in expressiveness to the NTYFT format (in Section 5) and thus, strictly more expressive than the OSOS and the GSOS rule formats, as we shall demonstrate shortly.

Definition 4.8 (OTYFT) *An OTSS $(R, <)$ is in the* OTYFT *format it satisfies the following two conditions.*

(1) For all deduction rules $r \in R$, either r is in the TYFT *format or $conc(r) \in prem(r)$, in the latter case the targets of all premises should be distinct variables.*
(2) For all distinct deduction rules $r, r' \in R$, if $r \in higher(r')$ then $vars(trg(prem(r))) \cap vars(r') = \emptyset$ and $vars(src(prem(r))) \subseteq vars(r')$.

An OTSS is in the acyclic OTYFT *format if it is in the* OTYFT *format and furthermore, for all (possibly identical) deduction rules $r, r' \in R$, item 2 holds.*

Note that the OTYFT format (syntactically) generalizes the OSOS rule format in that any OTSS in the OSOS format is in the OTYFT format but not vice versa. Later, in Section 5, we show that this strict generalization indeed holds in the semantic sense, as well; namely, there are transition systems that can be specified by the OTYFT format but not by the OSOS format.

In the definition of the OTYFT format, we allowed for deduction rules that are not in the TYFT format but have their conclusions among their premises. These rules may seem useless as they cannot lead to a proof for any new transition. However, thanks to the notion of ordering among rules, they can be indeed useful for modeling impossibility of certain transitions (i.e., the idea of negative premises) by placing them above other rules. In other words, although these rules by themselves may not enable the derivation of new transitions, once ordered above other rules, they may indeed disable some. Item 2 of the OTYFT format permits also cyclic rules by the distinctness of r and r' condition. This is in line with the definition of OSOS. Note that in rules placed above others (thus not cyclic rules) lookahead is prohibited by the second statement of item 2. Lookahead is freely allowed, however, for deduction rules which are minimal with respect to the ordering.

We conjecture that the second statement in item 2, i.e., $vars(src(prem(r))) \subseteq vars(r')$, can be dropped without jeopardizing the congruence result. However, our translation to the NTYFT format (in Section 5.2) requires this condition. The following example illustrates that by dropping this statement one can specify transition systems that cannot be specified by the NTYFT format.

Example 4.9

$\downarrow$	$\dfrac{x \xrightarrow{a} y \quad y \xrightarrow{b} z}{x \xrightarrow{a} y}$	$\dfrac{}{a \xrightarrow{a} d}$	$\dfrac{}{b \xrightarrow{a} d}$	$\dfrac{}{b \xrightarrow{a} c}$	$\dfrac{}{c \xrightarrow{b} d}$
	$\dfrac{x \xrightarrow{a} x'}{f(x) \xrightarrow{c} d}$				

The above OTSS is complete and its three-valued stable model is $C = P = \{a \xrightarrow{a} d, b \xrightarrow{a} d, b \xrightarrow{a} c, c \xrightarrow{b} d, f(a) \xrightarrow{c} d\}$. We claim that there is no TSS in the NTYFT format that defines the above three-valued stable model. Assuming that such a TSS does exist (without loss of generality, we can assume that the TSS is pure), consider a minimal well-supported proof for $\vdash_{\mathrm{ws}} f(a) \xrightarrow{c} d$; using the same deduction rule leading to this proof and a new substitution, we prove that $\vdash_{\mathrm{ws}} f(b) \xrightarrow{c} d$ (contradiction).

Assume that the proof for $f(a) \xrightarrow{c} d$ is due to a rule of the following form:

$$(\mathbf{r})\ \frac{\{t_i \xrightarrow{l_i} y_i \mid i \in I\} \quad \{t'_j \stackrel{l'_j}{\nrightarrow} \mid j \in J\}}{f(x) \xrightarrow{c} t}$$

and there exists a substitution σ such that $\sigma(x) = a$ and $\sigma(t) = d$. The premises of such a rule may be of one of the following shapes:

(1) $x \xrightarrow{a} y_i$ or $a \xrightarrow{a} y_i$, for some $i \in I$,
(2) $b \xrightarrow{a} y_i$, for some $i \in I$,
(3) $t_i \xrightarrow{b} y_i$ or $c \xrightarrow{b} y_i$, where $\sigma(t_i) = c$ and $i \in I$,
(4) $t'_j \stackrel{a}{\nrightarrow}$ where t'_j can be any term such that $\sigma(t'_j) \neq a$ and $\sigma(t'_j) \neq b$, and $j \in J$,
(5) $t'_j \stackrel{b}{\nrightarrow}$ where t'_j can be any term such that $\sigma(t'_j) \neq c$, and $j \in J$,
(6) $t'_j \stackrel{c}{\nrightarrow}$ where t'_j can be any term but $f(a)$ or $f(x)$ and $j \in J$, (these two cases are excluded since otherwise, there should exist a well-supported denial for $f(a) \xrightarrow{a}$ and then $f(a) \xrightarrow{c} d$ cannot be included in the C component of the three-valued stable model).

Note that $f(x)$ or $f(a)$ cannot be in the source of a positive premise because the label of such a premise should be a c and then the well-supported proof of $f(a) \xrightarrow{c} d$ due to **(r)** is not minimal and there is a smaller proof which is the proof of such a premise. Also, given the above forms, the target of the conclusion, i.e. t, should either be d or some y_l such that $\sigma(y_l) = d$.

Define σ' as follows: $\sigma'(x) \doteq b$, $\sigma'(y) = \sigma(y)$, for all variables $y \neq x$. Then, all positive premises (items 1 to 3 above) must have a well-supported proof (for they are all included in the C component of the least well-supported model). For the negative premises, there is no case where substituting a b for an a may enable a- or b-transitions. Similarly, substituting a b for an a may disable c-transitions but may not enable them. Hence, we obtain a well-supported proof for $\sigma'(f(x) \xrightarrow{c} t)$, i.e. $f(b) \xrightarrow{c} d$.

Note that the presence of lookahead in the upper rule of the first column plays an essential rôle in establishing this expressiveness result. In essence, the first column states that $f(x)$ can make a c transition to d when x can make an a transition and furthermore, *for all possible a transitions of x* to any y, y cannot make a b transition. The implicit universal quantification in the generalization of the OTYFT format makes it more expressive than the NTYFT format.

Since our translation to the NTYFT format (in Section 5.2) provably preserves the three-valued stable model, we can recast Theorem 4.5 in the setting of ordered SOS, as follows.

Theorem 4.10 (Congruence for OTYFT) *For a complete OTSS in the* OTYFT *format, bisimilarity is a congruence.*

Proof. For a complete OTSS $(R, <)$, $R' = \mathit{ntyft}(R, <)$ is a TSS in the NTYFT format which induces the same least three-valued stable model and thus is complete. Thus, bisimilarity for the induced transition relation of R' is a congruence and it coincides with the bisimilarity for the transition relation induced by $(R, <)$. ⊠

Note that our essential addition to the constraints of the TYFT format is the first constraint in item 2 of Definition 4.8. The following counter-example shows that this constraint on the OTYFT format is indeed useful for obtaining a congruence result, and cannot be dropped.

Example 4.11 The following OTSS is complete and its stable model is $\{a \xrightarrow{a} b, b \xrightarrow{a} a, f(a) \xrightarrow{a} a\}$.

Hence, for the above OTSS a and b are bisimilar while $f(a)$ and $f(b)$ are not.

$\downarrow$	$\dfrac{b \xrightarrow{a} y}{b \xrightarrow{a} y}$	$\dfrac{}{a \xrightarrow{a} b}$	$\dfrac{}{b \xrightarrow{a} a}$
	$\dfrac{x \xrightarrow{a} y}{f(x) \xrightarrow{a} a}$		

The following two counter-examples show that the same condition cannot be dropped for non-TYFT rules, which have their conclusion among their premises, either.

Example 4.12

$\downarrow$	$\dfrac{x \xrightarrow{a} x}{x \xrightarrow{a} x}$	$\dfrac{}{a \xrightarrow{a} a}$	$\dfrac{}{b \xrightarrow{a} a}$
	$\dfrac{}{f(x) \xrightarrow{a} a}$		

The above OTSS is complete and its stable model is $\{a \xrightarrow{a} a, b \xrightarrow{a} a, f(b) \xrightarrow{a} a\}$. Hence, for the above OTSS, a and b are bisimilar while $f(a)$ and $f(b)$ are not.

Example 4.13

$\downarrow$	$\dfrac{a \xrightarrow{a} x}{a \xrightarrow{a} x}$	$\dfrac{}{a \xrightarrow{a} b}$	$\dfrac{}{b \xrightarrow{a} a}$
	$\dfrac{}{f(x) \xrightarrow{a} a}$		

The above OTSS is complete and its stable model is $\{a \xrightarrow{a} b, b \xrightarrow{a} a, f(a) \xrightarrow{a} a\}$. Hence, for the above OTSS, a and b are bisimilar while $f(a)$ and $f(b)$ are not.

In item 1 of Definition 4.8, we require that the targets of premises of non-tyft rules should still contain distinct variables. The following counter-example shows why this requirement cannot be dropped.

Example 4.14

$\downarrow$	$\frac{x \xrightarrow{a} a}{x \xrightarrow{a} a}$	$\frac{}{b \xrightarrow{a} b}$	$\frac{}{a \xrightarrow{a} a}$
	$\frac{}{f(x) \xrightarrow{a} a}$		

The above OTSS is complete and its stable model is $\{a \xrightarrow{a} a, b \xrightarrow{a} b, f(b) \xrightarrow{a} \}$. Hence, for the above OTSS, a and b are bisimilar while $f(a)$ and $f(b)$ are not.

SOS rules in the OSOS format are assumed (implicitly) to have different variables [22]. The last examples shows that in general if OSOS rules share variables, then bisimulation is not a congruence.

Example 4.15

$\downarrow$	$\frac{x \xrightarrow{b} y}{f(x) \xrightarrow{a} c}$	$\frac{}{a \xrightarrow{b} b}$	$\frac{}{b \xrightarrow{b} a}$	$\frac{}{b \xrightarrow{b} b}$
	$\frac{}{f(x) \xrightarrow{a} y}$			

The above OTSS is complete and its stable model is $\{a \xrightarrow{b} b, b \xrightarrow{b} a, b \xrightarrow{b} b, f(a) \xrightarrow{a} a,\ f(a) \xrightarrow{a} c,\ f(a) \xrightarrow{a} f(a),\ f(b) \xrightarrow{a} c,\ f(b) \xrightarrow{a} f(a), \ldots\}$. For the above OTSS a and b are clearly bisimilar. However, $f(a)$ and $f(b)$ are not bisimilar since $f(a)$ can make a transition to a

while $f(b)$ is not capable of making any a-transition to a term that can make a further b-transition.

5 Comparison of Expressiveness of NTYFT and OTYFT Formats

In this section, we present, in both directions, translations between TSS's in the NTYFT format and OTSS's in the OTYFT format that preserve the least three-valued stable models.

5.1 From NTYFT *to* OTYFT

We assume in the remainder that TSS's in the NTYFT format are pure, i.e., all variables appearing in a rule must be drawn from among those used in the source of the conclusion or the targets of the premises. This restriction by no means reduces the expressiveness of the source format; impure TSS's can be transformed to pure ones (while keeping the TSS in the NTYFT format and preserving the least three-valued stable model) by making many copies of rules each instantiating the other variables by a closed term [10].

Definition 5.1 (Pure NTYFT to OTYFT: Translation Scheme) *Given a TSS R in the pure* NTYFT *format, its translation to the* OTYFT *format, denoted by $otyft(R)$, is an OTSS $(R', <)$ where $R' \doteq \{r^+, s_{r,j} \mid r \in R, j \in J_r\}$ and $< \doteq \{(r^+, s_{r,j}) \mid r \in R, j \in J_r\}$ and for each $r \in R$ of the form*

$$(r)\,\frac{\{t_i \xrightarrow{l_i} y_i | i \in I_r\} \cup \{t'_j \overset{l'_j}{\nrightarrow} | j \in J_r\}}{f(\overrightarrow{x}) \xrightarrow{l} t},$$

r^+ and $s_{r,j}$ (for each $j \in J_r$) are defined as follows:

$$(r^+)\,\frac{\{t_i \xrightarrow{l_i} y_i | i \in I_r\}}{f(\overrightarrow{x}) \xrightarrow{l} t}, \quad (s_{r,j})\,\frac{\{t'_j \xrightarrow{l'_j} y'_j\}}{t'_j \xrightarrow{l'_j} y'_j}.$$

In rule $s_{r,j}$, y'_j is a fresh variable not appearing in r^+.

The next theorem states that the diagram in Figure 2 commutes.

Theorem 5.2 (Pure NTYFT to OTYFT: Correctness) *For an arbitrary TSS R in the pure* NTYFT *format, the least three-valued stable models of R and $otyft(R)$ coincide.*

$$\begin{tikzcd} R\ (\text{pure NTYFT}) \arrow[rr, "\text{Translation Definition 5.1}"] \arrow[dr, "\text{Semantics [6]}"'] & & \mathit{otyft}(R)\ (\text{OTYFT}) \arrow[dl, "\text{Semantics 7}"] \\ & \text{Least Three-Valued Stable Model} & \end{tikzcd}$$

Fig. 2. Soundness of the Translation from the pure NTYFT format to the OTYFT format

Proof. Let R be an arbitrary TSS in the pure NTYFT format and $otyft(R)$ be its translation in the OTYFT format. Let S and $(S', <)$ be the closed instantiations of R and $otyft(R)$, respectively.

Next, we quote the least three-valued stable model semantics of TSS's from [6,9] (with a slight change in notation) and prove it equal to our definition of three-valued stable model semantics for the corresponding OTSS. To this end, we first define the following notion of negation (called denial in the literature).

Definition 5.3 (Negation) *A positive formula $p \xrightarrow{l} p'$ negates $p \overset{l}{\nrightarrow}$. A set of positive formulae T does not negate a set of negative formulae N, denoted by $T \vDash N$ when there is no $\phi \in T$ and $\psi \in N$ such that ϕ negates ψ.*

The following reduction procedure is taken from [6].

Definition 5.4 (Reduction Technique for SOS with Negative Premises) *For an ordinal α, define:*

$$C'_\alpha \doteq \{\phi \mid \exists_p \ S \vdash_p \frac{N}{\phi} \wedge \exists_{\beta<\alpha} \ P'_\beta \vDash N\}$$
$$P'_\alpha \doteq \{\phi \mid \exists_p \ S \vdash_p \frac{N}{\phi} \wedge \forall_{\beta<\alpha} \ C'_\beta \vDash N\}$$

where (by abusing the notation), $S \vdash_p \dfrac{N}{\phi}$ refers to the notion of provability as defined below.

A deduction rule $\dfrac{H}{\phi}$ is provable *from a TSS S by means of proof p, denoted by $S \vdash_p \dfrac{H}{\phi}$, when there exists a well-founded upwardly branching tree with formulae as nodes and of which*

- *the root is labelled by ϕ;*
- *if a node is labelled by ψ and the nodes above it form the set K then one of the following two cases hold:*
  - $\cdot$ *$\psi \in H$ and $K = \emptyset$;*
  - $\cdot$ *ψ is a positive transition formula and $s = \dfrac{K}{\psi} \in S$.*

To disambiguate the notation, in the remainder, we use $(S', <) \vdash_p \phi$ to denote that ϕ is provable from OTSS $(S', <)$ and $S \vdash_p \dfrac{H}{\phi}$ to denote that $\dfrac{H}{\phi}$ is provable from TSS S.

Lemma 5.5 *If $S \vdash_p \dfrac{N}{\phi}$ for some set N of negative formulae then there exists a proof q such that $(S', <) \vdash_q \phi$; furthermore, both proofs have the same depth and $rules(q) = \{r^+ \mid r \in rules(p)\}$.*

Proof. By an induction on the structure of proof p. Suppose p is an r-proof for some $r \in S$. Then $r^+ \in S'$ and for positive premises r and r^+ coincide. Thus, all subproofs concerning positive premises of r can be replaced by a proof from S' (of the same depth) using the induction hypothesis, and this way we have a proof q in $(S', <)$ for ϕ which has the same depth as p and comprises positive versions of the rules in q. ⊠

Returning to the proof of Theorem 5.2, we show, by an induction on α, that $C_\alpha = C'_\alpha$ and $P_\alpha = P'_\alpha$ for all ordinals α. Inside the induction on α, we use an induction on the depth of the proof for ϕ (i.e., depth of p such that $S \vdash_p \dfrac{N}{\phi}$ or depth of q such that $(S', <) \vdash_q \phi$).

$C_\alpha \subseteq C'_\alpha$ If $\phi \in C_\alpha$ then there exists a minimal proof p such that $(S', <) \vdash_p \phi$ and there exists a $\beta < \alpha$ such that for all rules $r \in rules(p)$, r is correct w.r.t. P_β. The last step of the proof should be due to a rule $r^+ \in S'$ such that $r \in S$ (rules $s_{r,j} \in S'$, for some $r \in S$ and $j \in J_r$, if at all applicable, cannot be part of a minimal proof). Suppose that deduction rule $r \in S$ (giving rise to $r^+ \in S'$) has the following shape:

$$(r)\,\frac{\{p_i \xrightarrow{l_i} p'_i | i \in I_r\} \cup \{p_j \stackrel{l_j}{\nrightarrow} | j \in J_r\}}{f(\overrightarrow{p}) \xrightarrow{l} p'},$$

It follows then from the induction hypothesis (of the induction on α) that $P_\beta = P'_\beta$ and $C_\beta = C'_\beta$. Since $s_{r,j} > r^+$ for all $j \in J_r$, it follows that $P'_\beta \vDash \{p_j \stackrel{l_j}{\nrightarrow} \mid j \in J_r\}$ (otherwise, r is not correct w.r.t. P_β). Let N' be $\{p_j \stackrel{l_j}{\nrightarrow} \mid j \in J_r\}$.

It also follows from the induction hypothesis (on the depth of the proofs) that for all $i \in I$, $\vdash_{q_i} \dfrac{N_i}{p_i \xrightarrow{l_i} p'_i}$ and $P'_\beta \vDash N_i$ and by using the deduction rule r, we can derive a proof for $\dfrac{N' \cup \bigcup_{i \in I_r} N_i}{\phi}$ and $P'_\beta \vDash N' \cup \bigcup_{i \in I_r} N_i$. Hence, $\phi \in C'_\alpha$.

$C'_\alpha \subseteq C_\alpha$ If $\phi \in C'_\alpha$ then there exists a proof p such that $S \vdash_p \dfrac{N}{\phi}$ and $P'_\beta \vDash N$ for some $\beta < \alpha$. Following Lemma 5.5, there exists a proof q such that $(S', <) \vdash_q \phi$. Consider the rules $r^+ \in \mathit{rules}(q)$; each of them is a positive instance of some rule $r \in S$ (see proof of Lemma 5.5). Suppose that some $r \in \mathit{rules}(p)$ is not correct with respect to P_β; assuming that the original rule r has the following shape:

$$(r)\frac{\{p_i \xrightarrow{l_i} p'_i | i \in I_r\} \cup \{p''_j \overset{l''_j}{\nrightarrow} | j \in J_r\}}{f(\overrightarrow{p}) \xrightarrow{l} p'},$$

then there exists a rule $s_{r,j}$, for some $j \in J$ such that $\mathit{prem}(s_{r,j}) \subseteq P_\beta$. Thus, in particular, $p''_j \xrightarrow{l''_j} q_j \in P_\beta$ for some q_j and since following the induction hypothesis (on α), $C_\beta = C'_\beta$ and $P_\beta = P'_\beta$, it does not hold that $P'_\beta \vDash p''_j \overset{l''_j}{\nrightarrow}$ and thus it does not hold that $P'_\beta \vDash N$. (Contradiction) Thus, we conclude that all rules $r \in \mathit{rules}(p)$ are correct w.r.t. P_β and this concludes the proof for $\phi \in C_\alpha$.

$P_\alpha = P'_\alpha$ Similar to above.

⊠

The following translation scheme is an alternative which preserves the least three-valued supported models. It uses auxiliary rules $s'_{r,j}$ to encode negative premises. These rules are placed above themselves in order to disable them. Also, see Lemma 3.32.

Given a TSS R in the pure NTYFT format, its translation to the OTYFT format, denoted by $\mathit{otyft}'(R)$, is an OTSS $(R', <)$ where $R' \doteq \{r^+, s'_{r,j} \mid r \in R, j \in J_r\}$ and $< \doteq \{(r^+, s'_{r,j}), (s'_{r,j}, s'_{r,j}) \mid r \in R, j \in J_r\}$ and for each $r \in R$ of the form

$$(r)\frac{\{t_i \xrightarrow{l_i} y_i | i \in I_r\} \cup \{t'_j \overset{l'_j}{\nrightarrow} | j \in J_r\}}{f(\overrightarrow{x}) \xrightarrow{l} t},$$

and for each $j \in J_r$, $s'_{r,j}$ is defined as the following rule

$$(s'_{r,j})\frac{\{t'_j \xrightarrow{l'_j} y'_j\}}{f(\overrightarrow{x}) \xrightarrow{l} t},$$

where y'_j is a fresh variable not appearing in r^+, and r^+ is defined as before.

Theorem 5.6 (Pure NTYFT to OTYFT: Correctness) *For an arbitrary TSS R, the least three-valued supported models of R and $\mathit{otyft}'(R)$ coincide.*

Proof. Similar to that of Theorem 5.2. ⊠

In this section, we complete the picture by translating OTSS's in the OTYFT format to TSS's in the NTYFT format. In common with the translation from the NTYFT to the OTYFT format, the translation in this section also preserves the least three-valued stable model.

Definition 5.7 (OTYFT to NTYFT: Translation Scheme) *Given an OTSS $(R, <)$ in the* OTYFT *format, function $S : R \rightarrow \mathcal{I}$, where $\mathcal{I} \doteq \bigcup_{r \in R} I_r$, is a selection function for $r \in R$ when for all $s \in higher(r)$ of the form*

$$\frac{\{t'_i \xrightarrow{l'_i} y'_i | i \in I_s\}}{t_s \xrightarrow{l_s} t'_s},$$

it holds that $S(s) \in I_s$. (Thus, if $I_s = \emptyset$ for some $s \in higher(r)$, then the set of selection functions for r is empty.)

Given an OTSS $(R, <)$ in the OTYFT *format, its translation to the* NTYFT *format, denoted by $ntyft(R, <)$, is defined as $\{r_S \mid r \in tyft(R), S \text{ is a selection function for } r\}$. Here $tyft(R)$ is the subset of R that conforms to the* TYFT *format, and for each $r \in tyft(R)$ of the form $\frac{\{t_i \xrightarrow{l_i} y_i | i \in I_r\}}{f(\overrightarrow{x}) \xrightarrow{l} t}$, r_S is defined as follows:*

$$(r_S)\frac{\{t_i \xrightarrow{l_i} y_i | i \in I_r\} \cup \{t'_{S(s)} \overset{l'_{S(s)}}{\nrightarrow} | s \in higher(r)\}}{f(\overrightarrow{x}) \xrightarrow{l} t}.$$

The idea of the above translation is that for each rule r in R, for all rules placed above r, we negate an arbitrary premise and add the chain of premises up to the negated premise to the premises of r. This way, we can make sure that r is applied precisely when all rules above it are for some reason disabled. The following examples illustrate the idea of this translation.

Example 5.8 Consider the following OTSS in the (acyclic) OTYFT format.

$\downarrow$	$\dfrac{x \xrightarrow{b} y'}{x \xrightarrow{b} y'}$	$\dfrac{}{a \xrightarrow{a} a}$	$\dfrac{}{b \xrightarrow{a} b}$	$\dfrac{}{b \xrightarrow{b} b}$
	$\dfrac{x \xrightarrow{a} y}{f(x) \xrightarrow{a} y}$			

Applying the translation scheme of Definition 5.7 results in the following TSS. Note that the topmost rule is not included since it is not in the TYFT format.

$$\frac{x \xrightarrow{a} y \quad x \stackrel{b}{\not\rightarrow}}{f(x) \xrightarrow{a} y} \qquad \frac{}{a \xrightarrow{a} a} \qquad \frac{}{b \xrightarrow{a} b} \qquad \frac{}{b \xrightarrow{b} b}$$

It is then easy to check that the least three-valued stable model of both the OTSS and the translated TSS is $(\{a \xrightarrow{a} a, b \xrightarrow{a} b, b \xrightarrow{b} b, f(a) \xrightarrow{a} a\}, \{a \xrightarrow{a} a, b \xrightarrow{a} b, b \xrightarrow{b} b, f(a) \xrightarrow{a} a\})$.

Our next milestone is to show that our translation indeed preserves the least three-valued stable models. There is a subtlety concerning cyclic rules. Consider the rule on the left below which is placed above itself. It is translated into the NTYFT rule on the right

$$\frac{\{t_i \xrightarrow{l_i} y_i \mid i \in I\}}{f(x) \xrightarrow{l} t} \qquad\qquad (r)\frac{\{t_i \xrightarrow{l_i} y_i \mid i \in I\} \cup \{t_j \stackrel{l_j}{\not\rightarrow}\}}{f(x) \xrightarrow{l} t}$$

for some $j \in I$. Thus, r contains contradictory premises of the form $t_j \xrightarrow{l_j} y_j$ and $t_j \stackrel{l_j}{\not\rightarrow}$. Such rules may influence the semantics in different ways when the OTSS is not complete. To avoid such complications, we first prove, in the absence of cyclic rules, that our translation preserves the least three-valued stable model even for incomplete OTSS's. Then, we show that including cyclic rules does not endanger the correctness of our translation for complete OTSS's (see Figure 3). Theorem 5.9 proves that the translation presented in Definition 5.7 preserves the least three-valued stable model for the acyclic OTYFT format (presented by dotted lines in the diagram) while Theorem 5.12 proves that the same translation preserves the three-valued stable model for complete OTSS's in the OTYFT format (presented by solid lines).

Theorem 5.9 (Acyclic OTYFT to NTYFT: Correctness) *For an arbitrary OTSS* $(R, <$

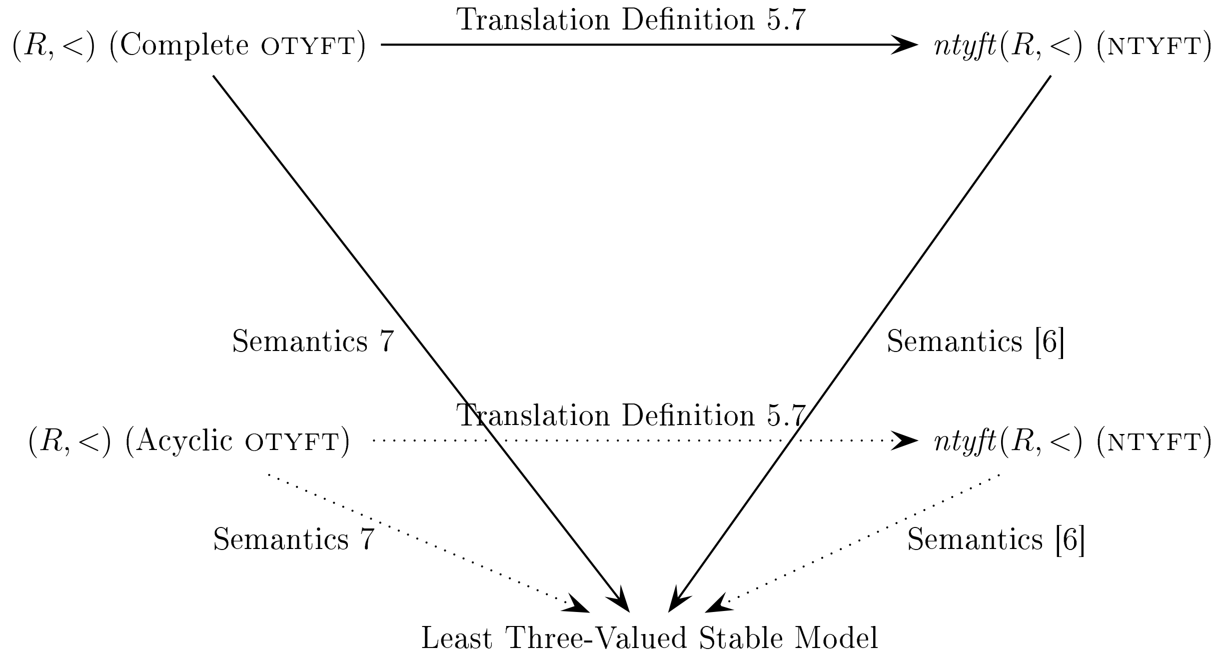


Fig. 3. Soundness of the Translation from the OTYFT format to the NTYFT format

) *in the acyclic* OTYFT *format, the least three-valued stable models of* $(R, <)$ *and* $ntyft(R, <)$ *coincide.*

Proof. The proof is very similar in nature to the proof of Theorem 5.2. Take $(R, <)$ and R' to be the closed instantiations of an arbitrary OTSS in the OTYFT format and its translation to the NTYFT format. Also let C_α and P_α be certain and possible transition relations resulting from applying the reduction technique of Definition 3.26 to $(R, <)$, and let C'_α and P'_α be the result of applying the reduction technique of [6] (see the proof of Theorem 5.2) to R'. Again similarly to the proof of Theorem 5.2, we prove the following auxiliary lemma:

Lemma 5.10 *If* $R' \vdash_p \dfrac{N}{\phi}$ *for some set* N *of negative formulae then there is a proof* q *such that* $(tyft(R), <) \vdash_q \phi$*; furthermore, both proofs have the same depth.*

Proof. By an induction on the structure of proof p. Suppose p is an s-proof for some $s \in R'$. Then s is introduced in R' due to the presence of some rule $r \in tyft(R)$. All positive premises of s are (the same as positive premises of r and are) by induction hypothesis provable from $(tyft(R), <)$. This way, we have a proof q for ϕ in $(tytf(R), <)$. $\boxtimes$

Returning to the proof of Theorem 5.9, we show, by an induction on α, that for all ordinals α, we have $C_\alpha = C'_\alpha$ and $P_\alpha = P'_\alpha$. Inside the induction on α, we use an induction on the depth of the proofs for ϕ in each of the above-mentioned transition relations.

$C_\alpha \subseteq C'_\alpha$ Take $\phi \in C_\alpha$. Let p be an r-proof for ϕ in $(R, <)$ such that p is correct w.r.t. P_β for some $\beta < \alpha$ (without loss of generality, we can assume that p is minimal and hence $r \in \mathit{tyft}(R)$). If there is no $r' > r$ then $r \in R'$. All the premises ϕ_i of r have a smaller proof depth and it follows from the induction hypothesis (on the depth of the proof) that $\phi_i \in C'_\alpha$, that is, $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_{\beta_i} \vDash N_i$ for some $\beta_i < \alpha$. Thus using the same deduction rule $r \in R'$, we obtain $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \bigcup_{i \in I_r} N_i$ and hence, $P'_{\beta'} \vDash N$ where β' is the maximum of all β_i for all $i \in I_r$.

Otherwise, r should be correct w.r.t. P'_β (since $C_\beta = C'_\beta$ and $P_\beta = P'_\beta$ and $\beta < \alpha$) and hence, for all rules $s \in \mathit{higher}(r)$, there exists at least one premise $\phi_{S(s)} = p_{S(s)} \xrightarrow{l'_{S(s)}} p''_{S(s)}$ such that for all q'', $p_{S(s)} \xrightarrow{l'_{S(s)}} q'' \notin P'_\beta$. Otherwise, collect all premises of s that can find a target q''_i such that $p_i \xrightarrow{l'_i} q''_i \in P'_\beta$ (one premise for each $i \in I_s$) and by induction hypothesis $p_i \xrightarrow{l'_i} q''_i \in P_\beta$. Thus, there is a rule $s' > r$ such that it has the set of premises $\{p_i \xrightarrow{l'_i} q''_i \mid i \in I_s\} \subseteq P_\beta$ and thus r is not correct w.r.t. P_β. (Contradiction) Hence, we conclude that there exists a selection function S for r such that $P'_\beta \vDash \{p_{S(s)} \stackrel{l'_{S(s)}}{\nrightarrow} \mid s \in \mathit{higher}(r)\ \}$.

Take the rule r_S of the following shape:

$$(r_S)\,\frac{\{p_i \xrightarrow{l_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \stackrel{l''_{S(s)}}{\nrightarrow} | s \in \mathit{higher}(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}$$

For all its positive premises ϕ_i, $i \in I_r$, we have $\phi_i \in C'_\alpha$ and hence $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_{\beta_i} \vDash N_i$ for some $\beta_i < \alpha$. Also, it follows from the above reasoning that for the negative premises, it holds that $P'_\beta \vDash p''_{S(s)} \stackrel{l''_{S(s)}}{\nrightarrow}$ and hence, using the above deduction rule, we have a proof q for $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \{p''_{S(s)} \stackrel{l''_{S(s)}}{\nrightarrow} | s \in \mathit{higher}(r)\}$ $\cup$ $(\bigcup_{i \in I_r} N_i)$ and $P'_{\beta'} \vDash N$ where β' is the maximum of β and β_i for all $i \in I_r$.

$C'_\alpha \subseteq C_\alpha$ If $\phi \in C'_\alpha$ then there exists a proof q such that $R' \vdash_q \dfrac{N}{\phi}$ and there exists $\beta < \alpha$ such that $P'_\beta \vDash N$. Following Lemma 5.10, there exists a proof p such that $(R, <) \vdash_p \phi$. Consider the rules $r \in \mathit{rules}(p)$ (it follows from the same lemma that $r \in \mathit{tyft}(R)$); each such rule r corresponds to a rule $r_S \in R'$ which has the shape

$$(r_S)\,\frac{\{p_i \xrightarrow{l'_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \stackrel{l''_{S(s)}}{\nrightarrow} | s \in \mathit{higher}(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}.$$

Since $\phi_i = p_i \xrightarrow{l'_i} p'_i \in C'_\alpha$ for all $i \in I_r$, there exist proofs q_i such that $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_{\beta_i} \vDash N_i$, for some $\beta_i < \alpha$. It follows then from the induction hypothesis (on the depth of the proof) that there exist proofs p_i such that $(R, <) \vdash_{p_i} \phi_i$. Furthermore, for all rules $s \in \mathit{higher}(r)$, there exists a premise $\phi_{S(s)} = p''_{S(s)} \stackrel{l''_{S(s)}}{\nrightarrow}$ such that $P'_\beta \vDash \phi_{S(s)}$ or $p''_{S(s)} \xrightarrow{l''_{S(s)}} q'' \notin P_\beta$ for all q'' (following the induction hypothesis on α). Thus, each $r \in \mathit{rules}(p)$ is correct for P_β and this way, we have a proof p for $(R, <) \vdash_p \phi$. Hence, $\phi \in C_\alpha$.

$P_\alpha = P'_\alpha$ Similar to above.

$\boxtimes$

The translation from OTYFT to NTYFT does not generalize trivially to the setting with cyclic rules. The following example illustrates this fact.

Example 5.11 Consider the following OTSS which is in the OTYFT (but not in the acyclic OTYFT) format. Note that all the deduction rules used in this OTSS are in the TYFT format.

$\downarrow$	$\dfrac{b \xrightarrow{b} y}{b \xrightarrow{b} y}$	$\dfrac{b \xrightarrow{b} y}{a \xrightarrow{a} b}$	$\dfrac{}{a \xrightarrow{a} a}$	$\circlearrowright$ $\dfrac{x \xrightarrow{a} y}{f(x) \xrightarrow{a} y}$
	$\dfrac{}{b \xrightarrow{b} b}$			

The least three-valued stable model of this OTSS is $(\{a \xrightarrow{a} a\}, \{a \xrightarrow{a} a, b \xrightarrow{b} b, a \xrightarrow{a} b, f(a) \xrightarrow{a} b\})$. Using the translation scheme of Definition 5.7, the above OTSS is translated into the following TSS.

$$\frac{b \xrightarrow{b} y}{b \xrightarrow{b} y} \qquad \frac{b \stackrel{b}{\nrightarrow}}{b \xrightarrow{b} b} \qquad \frac{b \xrightarrow{b} y}{a \xrightarrow{a} b} \qquad \frac{}{a \xrightarrow{a} a} \qquad \frac{x \xrightarrow{a} y \quad x \stackrel{a}{\nrightarrow}}{f(x) \xrightarrow{a} y}$$

The least three-valued stable model of the above TSS is $(\{a \xrightarrow{a} a\}, \{a \xrightarrow{a} a, b \xrightarrow{b} b, a \xrightarrow{b} b\})$ which is different from the least three-valued stable model of the corresponding OTSS.

The problem displayed in Example 5.11 can be remedied by only considering complete OTSS's, which are the only interesting OTSS's as far as a congruence rule format is concerned. The following theorem shows that including cyclic rules does not endanger the correctness of our translation once the source OTSS is complete.

Theorem 5.12 (Complete OTYFT to NTYFT: Correctness) *For a complete OTSS $(R, <)$ in the* OTYFT *format, the least three-valued stable models of $(R, <)$ and $ntyft(R, <)$ coincide.*

Proof. Take $(R, <)$ and R' to be the closed instantiations of a complete OTSS in the OTYFT format and its translation to the NTYFT format. Also let C_α and P_α be certain and possible transition relations resulting from applying the reduction technique of Definition 3.26 to $(R, <)$, and let C'_α and P'_α be the result of applying the reduction technique of [6] to R'. Furthermore, let (C_λ, P_λ) and (C'_λ, P'_λ) be the least three-valued stable models of $(R, <)$ and R', respectively. (Ordinal λ is the least ordinal for which the construction of both (C_α, P_α) and (C'_α, P'_α) reaches a fixed point.) It follows from the completeness of $(R, <)$ that $C_\lambda = P_\lambda$. By an induction on the ordinal α, we simultaneously prove the following statements.

(1) $C_\alpha \subseteq C'_\lambda$,
(2) $C'_\alpha \subseteq C_\lambda$,
(3) $P_\lambda \subseteq P'_\alpha$, and
(4) $P'_\lambda \subseteq P_\alpha$.

Once we prove the above statements, it follows from items 1 and 2 that $C_\lambda = C'_\lambda$ and from items 3 and 4 that $P_\lambda = P'_\lambda$.

Note that Lemma 5.10 holds in our setting with cyclic rules following the same proof.

Inside the induction on α, we use an induction on the depth of the proofs in each of the mentioned above transition relations.

$C_\alpha \subseteq C'_\lambda$ Take $\phi \in C_\alpha$. Let p be an r-proof for ϕ in $(R, <)$ such that p is correct w.r.t. P_β for some $\beta < \alpha$ (without loss of generality, we can assume that p is minimal and hence $r \in tyft(R)$).

Note that r cannot be cyclic, i.e., it cannot be the case that $r > r$; because otherwise, there exists a premise of r of the form $p_i \xrightarrow{l_i} p'_i$ which is in C_α (since ϕ is provable) but not in P_β (since p is correct); but we know that $C_\alpha \subseteq P_\alpha \subseteq P_\beta$. (Contradiction) Hence, r is not cyclic.

If there is no $r' > r$ then $r \in R'$. All the premises ϕ_i of r have a smaller proof depth and it follows from the induction hypothesis (on the depth of the proof) that $\phi \in C'_\lambda$, that is, $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_\lambda \vDash N_i$. Thus using the same deduction rule $r \in R'$, we obtain $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \bigcup_{i \in I_r} N_i$ and hence, $P'_\lambda \vDash N$.

Otherwise, r should be correct w.r.t. P'_λ (since r is correct w.r.t. P_β and $P'_\lambda \subseteq P_\beta$ by the induction hypothesis of the induction on α) and hence, for all rules $s \in \mathit{higher}(r)$, there exists at least one premise $\phi_{S(s)} = p_{S(s)} \xrightarrow{l'_{S(s)}} p''_{S(s)}$ such that for all q'', $p_{S(s)} \xrightarrow{l'_{S(s)}} q''$ $\notin P'_\lambda$. Otherwise, there is a rule $s' > r$ such that it has the set of premises $\{p_i \xrightarrow{l'_i} q''_i \mid i \in I_s\} \subseteq P'_\lambda \subseteq P_\beta$, and thus r is not correct w.r.t. P_β. (Contradiction) Hence, we conclude that there exists a selection function S for r such that $P'_\lambda \vDash \{p_{S(s)} \overset{l'_{S(s)}}{\not\rightarrow} \mid s \in \mathit{higher}(r)\ \}$.

Take the rule r_S of the following shape:

$$(r_S)\,\frac{\{p_i \xrightarrow{l_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \overset{l''_{S(s)}}{\not\rightarrow} | s \in \mathit{higher}(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}$$

For all its positive premises ϕ_i, $i \in I_r$, we have $\phi_i \in C'_\lambda$ and hence $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_\lambda \vDash N_i$. Also, it follows from the above reasoning that for the negative premises, it holds that $P'_\lambda \vDash p''_{S(s)} \overset{l''_{S(s)}}{\not\rightarrow}$ and hence, using the above deduction rule, we have a proof q for $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \{p''_{S(s)} \overset{l''_{S(s)}}{\not\rightarrow} | s \in \mathit{higher}(r)\} \cup (\bigcup_{i \in I_r} N_i)$ and $P'_\lambda \vDash N$. Hence, $\phi \in C'_\lambda$.

$C'_\alpha \subseteq C_\lambda$ If $\phi \in C'_\alpha$ then there exists a proof q such that $R' \vdash_q \dfrac{N}{\phi}$ and there exists $\beta < \alpha$ such that $P'_\beta \vDash N$. Following Lemma 5.10, there exists a proof p such that $(R, <) \vdash_p \phi$. Consider the rules $r \in \mathit{rules}(p)$ (it follows from the same lemma that $r \in \mathit{tyft}(R)$); each such rule r corresponds to a rule $r_S \in R'$ which has the shape

$$(r_S)\,\frac{\{p_i \xrightarrow{l'_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \overset{l''_{S(s)}}{\not\rightarrow} | s \in \mathit{higher}(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}.$$

Note that r cannot be cyclic. Otherwise, $r \in \mathit{higher}(r)$ and thus r_S would contain a premise $p_i \xrightarrow{l'_i} p'_i$, for some $i \in I_r$ such that $i = S(r)$. Thus, $p'_i \overset{l'_i}{\not\rightarrow}$ would also be a premise of r_S. We have that $p_i \xrightarrow{l'_i} p'_i \in C'_\alpha$, and thus, $p_i \xrightarrow{l'_i} p'_i \in P'_\alpha$. On the other hand, we have that $P'_\beta \vDash p_i \overset{l'_i}{\not\rightarrow}$. It follows from the latter statement and $P'_\alpha \subseteq P'_\beta$ that for no p'_i (including the one mentioned above) $p_i \xrightarrow{l'_i} p'_i \in P'_\alpha$, which contradicts $p_i \xrightarrow{l'_i} p'_i \in P'_\alpha$. Thus, we conclude that r cannot be cyclic.

Since $\phi_i = p_i \xrightarrow{l'_i} p'_i \in C'_\alpha$ for all $i \in I_r$, there exist proofs q_i such that $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$

for some N_i such that $P'_{\beta_i} \vDash N_i$, for some $\beta_i < \alpha$. It follows then from the induction hypothesis (on the depth of the proof) that there exist proofs pr_i such that $(R, <) \vdash_{pr_i} \phi_i$, where all pr_i are correct w.r.t. P_λ. Furthermore, for all rules $s \in \mathit{higher}(r)$, there exists a premise $\phi_{S(s)} = p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow}$ such that $P'_\beta \vDash \phi_{S(s)}$ or $p''_{S(s)} \overset{l''_{S(s)}}{\rightarrow} q'' \notin P_\lambda$ for all q'' (following the induction hypothesis on α). Thus, each $r \in \mathit{rules}(p)$ is correct for P_λ and this way, we obtain a proof p for $(R, <) \vdash_p \phi$ which is correct w.r.t. P_λ. Hence, $\phi \in C_\lambda$.

$P_\lambda \subseteq P'_\alpha$ Assume that $\phi \in P_\lambda$. Let p be an r-proof for ϕ in $(R, <)$ such that p is correct w.r.t. C_λ (without loss of generality, we can assume that p is minimal and hence $r \in \mathit{tyft}(R)$).

Note that r cannot be cyclic; because otherwise, there exists a premise of r of the form $p_i \xrightarrow{l_i} p'_i$ which is in C_λ (since ϕ is provable) but not in P_λ (since p is correct); but we know from the completeness of $(R, <)$ that $C_\lambda = P_\lambda$. (Contradiction) Hence, r is not cyclic.

If there is no $r' > r$ then $r \in R'$. All the premises ϕ_i of r have a smaller proof depth and it follows from the induction hypothesis (on the depth of the proof) that $\phi \in C'_\alpha$, that is, $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $P'_{\beta_i} \vDash N_i$, for some $\beta_i < \alpha$. Thus using the same deduction rule $r \in R'$, we obtain $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \bigcup_{i \in I_r} N_i$ and hence, $P'_\beta \vDash N$, where $\beta < \alpha$ is the maximum of all such β_i.

Otherwise, r is correct w.r.t. C_λ and hence, for all rules $s \in \mathit{higher}(r)$, there exists at least one premise $\phi_{S(s)} = p_{S(s)} \overset{l'_{S(s)}}{\rightarrow} p''_{S(s)}$ such that for all q'', $p_{S(s)} \overset{l'_{S(s)}}{\rightarrow} q'' \notin C_\lambda$. Otherwise, collect all premises of s that can find a target q''_i such that $p_i \xrightarrow{l'_i} q''_i \in C_\lambda$ (one premise for each $i \in I_s$). Thus, there is a rule $s' > r$ such that it has the set of premises $\{p_i \xrightarrow{l'_i} q''_i \mid i \in I_s\} \subseteq C_\lambda$ and thus r is not correct w.r.t. P_β. (Contradiction) Hence, there exists a selection function S for r such that $C_\lambda \vDash \{p_{S(s)} \overset{l'_{S(s)}}{\nrightarrow} \mid s \in \mathit{higher}(r)\}$; it then follows from the induction hypothesis on α that $C'_\beta \vDash \{p_{S(s)} \overset{l'_{S(s)}}{\nrightarrow} \mid s \in \mathit{higher}(r)\}$, for each $\beta < \alpha$.

Take the rule r_S of the following shape:

$$(r_S)\dfrac{\{p_i \xrightarrow{l_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow} | s \in \mathit{higher}(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}$$

For all its positive premises ϕ_i, $i \in I_r$, we have $\phi_i \in P'_\alpha$ and hence $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $C'_\beta \vDash N_i$, for each $\beta < \alpha$. Also, it follows from the above reasoning that for the negative premises, it holds that $P'_\beta \vDash p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow}$ and hence, using the above deduction rule, we have a proof q for $R' \vdash_q \dfrac{N}{\phi}$ where $N \doteq \{p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow} | s \in higher(r)\} \cup (\bigcup_{i \in I_r} N_i)$ and $C'_\beta \vDash N$, for each $\beta < \alpha$. Thus, $\phi \in P'_\alpha$.

$P'_\lambda \subseteq P_\alpha$ Assume that $\phi \in P'_\lambda$. If $\phi \in P'_\lambda$ then there exists a proof q such that $R' \vdash_q \dfrac{N}{\phi}$ and $C'_\lambda \vDash N$. Following Lemma 5.10, there exists a proof p such that $(R, <) \vdash_p \phi$. Consider the rules $r \in rules(p)$ (it follows from the same lemma that $r \in tyft(R)$); each such rule r corresponds to a rule $r_S \in R'$ which has the shape

$$(r_S) \frac{\{p_i \xrightarrow{l'_i} p'_i | i \in I_r\} \cup \{p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow} | s \in higher(r)\}}{f(\overrightarrow{p}) \xrightarrow{l} p'}.$$

Since $\phi_i = p_i \xrightarrow{l'_i} p'_i \in P'_\lambda$ for all $i \in I_r$, there exist proofs q_i such that $R' \vdash_{q_i} \dfrac{N_i}{\phi_i}$ for some N_i such that $C'_\lambda \vDash N_i$. It follows then from the induction hypothesis (on the depth of the proof) that there exist proofs pr_i such that $(R, <) \vdash_{pr_i} \phi_i$, where each pr_i is correct w.r.t. C_β for all $\beta < \alpha$. Furthermore, for all rules $s \in higher(r)$, there exists a premise $\phi_{S(s)} = p''_{S(s)} \overset{l''_{S(s)}}{\nrightarrow}$ such that $C'_\lambda \vDash \phi_{S(s)}$ or $p''_{S(s)} \xrightarrow{l''_{S(s)}} q'' \notin C'_\lambda$ for all q''. It then follows from the induction hypothesis on α (contraposition of item 1) that $p''_{S(s)} \xrightarrow{l''_{S(s)}} q'' \notin C_\beta$ for all q'' and for all $\beta < \alpha$. Thus, each $r \in rules(p)$ is correct w.r.t. C_β, for all $\beta < \alpha$, and this way, we get a proof p for $(R, <) \vdash_p \phi$ which is correct w.r.t. C_β for all $\beta < \alpha$. Hence, $\phi \in P_\alpha$.

$\boxtimes$

6 Relative Expressiveness of Rule Formats

In [11,5,10], the expressiveness of the GSOS and the NTYFT rule formats are studied. The common approach is to characterize the finest trace congruence [5] for image-finite processes induced by operators definable in the respective rule format. Next, we formalize the concepts of trace congruence and image-finiteness.

Definition 6.1 (Trace Congruence) *Given a signature, a* context $C[\,]$ *is a term with one or more appearance of a hole* $[\,]$. $C[p]$ *is then a closed term resulting from replacing all holes in* $C[\,]$ *by* p.

Two closed terms p *and* q *are* (completed) trace equivalent*, denoted by* $p \approx q$*, when the sets of completed traces originating from* p *and* q *coincide. Two closed terms* p *and* q *are* trace congruent, *denoted by* $p \approx_c q$, *when* $C[p] \approx C[q]$ *for all contexts* C.

Definition 6.2 (Image-Finite Transition Systems) *A transition system is called* image-finite, *when for all closed terms* p, *the set* $I_l = \{q \mid p \xrightarrow{l} q\}$ *is finite for each label* l.

One of the consequences of the definition of the GSOS format [5] is that TSS's in the GSOS format are image-finite. The finest trace congruence induced by GSOS-definable operators is ready simulation equivalence as defined below. The same result holds for the OSOS format because [22] provides straightforward translations between GSOS and OSOS specifications.

Definition 6.3 (Ready Simulation Equivalence) *A symmetric relation* $R \subseteq \mathcal{C} \times \mathcal{C}$ *is called a* ready simulation relation, *when for all* $(p, q) \in R$ *and* $l \in \mathcal{C}$:

(1) for all $p' \in \mathcal{C}$, *if* $p \xrightarrow{l} p'$ *then there exists a* q' *such that* $q \xrightarrow{l} q'$ *and* $(p', q') \in R$;
(2) if $p \overset{l}{\nrightarrow}$ *then* $q \overset{l}{\nrightarrow}$.

Two terms p *and* q *are* ready simulation equivalent *if there is some ready simulation relation* R *such that* $(p, q) \in R$.

Next, we show that for image-finite processes, the finest trace congruence induced by the operators definable in the OTYFT format is strong bisimilarity. But before proving this result, we formalize a few notions that are used next.

Definition 6.4 (n-nested Bisimilarity) *A symmetric relation* $R_n \subseteq \mathcal{C} \times \mathcal{C}$ *is an* n-nested bisimulation relation *when for* $n = 0$ *it is the full relation, i.e.,* $\mathcal{C} \times \mathcal{C}$, *and for* $n > 0$, *for all* $(p, q) \in R_n$ *and* $l \in \mathcal{C}$, *for all* $p' \in \mathcal{C}$, *if* $p \xrightarrow{l} p'$ *then there exists a* q' *such that* $q \xrightarrow{l} q'$ *and* $(p', q') \in R_{n-1}$. *Two closed terms are* n-nested bisimilar *when there exists an* n*-nested bisimulation relation relating them.*

It is well-known that for image-finite processes p and q, i.e., processes from which an image-finite transition system originates, p and q are strongly bisimilar if and only if they are n-nested bisimilar for each $n \in I\!N$ [8].

Next, we define a bisimulation-checker operator $B(\_, \_)$ (for an arbitrary set of operators and an arbitrary image finite transition relation) in the OTYFT format. This precisely characterizes the finest trace congruence definable by operators in the OTYFT format since firstly, it shows that if two processes are non-bisimilar then there is a context (namely our bisimulation-checker operator) that can distinguish them up to trace equivalence (thus, trace congruence implies bisimilarity) and secondly, by Theorem 4.10, if they are bisimilar, then they are bisimilar under all contexts and hence, they are trace congruent.

Example 6.5 (A Bisimulation Checker in OTYFT) The following OTSS defines a class of binary operators $B^n(\_, \_)$ and $Q^n_a(\_, \_)$ (for all $n \in I\!N$ and $a \in Act$) where B_n checks whether its two arguments are n-nested bisimilar and Q^n_a is an auxiliary operator (for defining B_n) which checks whether the second argument can make an a-step to something $(n-1)$-nested bisimilar to the first argument.

$\downarrow$	$\dfrac{B^n(x,y) \stackrel{no}{\rightarrow} z}{B^n(x,y) \stackrel{no}{\rightarrow} z}$ $\quad$ $\dfrac{B^n(y,x) \stackrel{no}{\rightarrow} z}{B^n(y,x) \stackrel{no}{\rightarrow} z}$	$\dfrac{Q^n_a(x',y) \stackrel{yes}{\rightarrow} z}{Q^n_a(x',y) \stackrel{yes}{\rightarrow} z}$	$\dfrac{y \stackrel{a}{\rightarrow} y' \quad B^{n-1}(x,y') \stackrel{yes}{\rightarrow} z}{Q^n_a(x,y) \stackrel{yes}{\rightarrow} 0}$	$\dfrac{}{B^0(x,y) \stackrel{yes}{\rightarrow} 0}$
	$\dfrac{}{B^n(x,y) \stackrel{yes}{\rightarrow} 0}$	$\dfrac{x \stackrel{a}{\rightarrow} x'}{B^n(x,y) \stackrel{no}{\rightarrow} 0}$		

The above rules are obtained by applying the translation from pure NTYFT to OTYFT from Definition 5.1 to the bisimulation-checker from [10] (which is in the pure NTYFT format).

The rules are self-explanatory. The auxiliary operator Q^n_a assumes that the first argument has already made an a-transition and checks whether the second argument can make an a-transition such that the target is $(n-1)$-bisimilar to the first argument. If such a transition is possible, it will make a transition with label *yes*. The n-nested bisimulation-checker operator checks whether one argument can make a transition that cannot be mimicked by the other argument (to something $(n-1)$-nested bisimilar) and if it finds such a transition the bisimulation checker makes a *no* transition. Otherwise, if making a transition with label *no* is not possible, then it makes a *yes*-transition.

Using the definition of n-nested bisimulation checker, one can define the bisimulation-checker operator $B(\_,\_)$ as follows (other rules presented before with their ordering should be added to the following OTSS; the two partitions of rules remain unrelated as far as ordering is concerned).

$\downarrow$	$\dfrac{B^1(x,y) \stackrel{no}{\rightarrow} y'}{B(x,y) \stackrel{no}{\rightarrow} 0}$ $\quad$ $\dfrac{B^2(x,y) \stackrel{no}{\rightarrow} y'}{B(x,y) \stackrel{no}{\rightarrow} 0}$ $\quad \ldots$
	$\dfrac{}{B(x,y) \stackrel{yes}{\rightarrow} 0}$

Following [8], for image-finite processes, two processes are bisimilar, when they are n-nested bisimilar for all $n \in \mathbb{N}$. The above rules thus define a bisimulation checker for image-finite processes.

By adding the above specification to any OTSS, one can check bisimilarity of two processes p and p' only by checking the *yes*/*no* trace (of length one) generated by the process $B(p, p')$. Hence, bisimilarity can be traced using the operators definable in the OTYFT format, while it cannot be traced by the operators definable in the OSOS format of [22].

The above example is indicative of the extra expressive power gained by the extension from OSOS to OTYFT which is demonstrated by the extra distinguishing power of definable operators. In other words, the operators definable in the OTYFT format can distinguish processes up to strong bisimilarity while those definable in the OSOS format do not go further than ready simulation equivalence.

7 Related Work

In this section we discuss the related notions of semantics for SOS, the expressive power of rule formats, and the rôle of orderings in some other fields of computer science.

In the paper we have commented on and adapted to our setting several notions of semantics proposed by van Glabbeek [9]. These include both the model-theoretic and proof-theoretic notions in the settings of two-valued and three-valued solutions, respectively. We have strived to give the same names to our notions as those of the corresponding notions in [9] in order to ease comparison. Figure 4 lists the main solutions that we discussed against the original solutions given by van Glabbeek. Additionally, we considered the notion of stratification as suggested in the context of logic programming by Przymusinski in [19] and used for TSS's in the NTYFT format by Groote [10]. This corresponds to Solution 10 in [9]. Our favorite notion of semantics is the least three-valued stable model but, in order to guarantee congruence meta-results, we restrict ourselves to OTSS's with complete semantics.

TSS's in the GSOS format employ unique supported models as semantics [5] (our Semantics 4). This solution coincides with all acceptable notions of semantics that we have discussed here for TSS's in the GSOS format (and indeed in the OSOS format, since there are semantics-preserving translations in each direction) and in simpler formats including the De Simone format [20].

The expressive power of rule formats increases with the generality of rules and with additions such as orderings on rules. A good analysis of the expressive power of other formats can be found in [1]. Here we only recall the main results in an informal manner. We have

Semantics		
in this paper		in [9]
2	Least model	2
3	Least supported model	3
4	Unique supported model	4
$5 = 6$	Least well-supported model	5
$7 = 9$	Least three-valued stable model	I
$8 = 10$	Least three-valued supported model	II
11	Complete	7
12	Irrefutable	9

Fig. 4. Comparison of our semantics with van Glabbeek's solutions in [9].

recalled in Section 6 that completed trace congruence can be used to distinguish processes up to ready simulation for TSS's in the GSOS format or the OSOS format. If one disallows negative information in rules, either in the form of negative premises or rule orderings, but permits arbitrary literals in the premises and conclusions of rules, as in the TYFT/TYXT format, then completed trace congruence can distinguish processes up to 2-nested simulation. It requires rules with both complex literals and negative information to be able to test bisimilarity with completed trace congruence as, for example, in the NTYFT format and OTYFT format (see Section 6). An interesting question arises: to what extent can OTYFT be simplified and still retain enough distinguishing power to test bisimulation? It is known that the NTREE format should suffice [7], where NTREE rules are TREE rules with addition of arbitrary negative literals. Our initial investigation indicates that the TREE format [7] equipped with orderings and extended with a single additional XYXT rule might do the job.

The orderings on SOS rules are an instance of a more general phenomenon of "priorities" in computer science. Priority, according to the Oxford Paperback Dictionary, means "being earlier or more important" and indicates that an object has "precedence in rank" when compared with other objects. In the context of this work, priorities specify the order of application of operational rules. In term rewriting, priorities are used to fix the order of application of ambiguous rewrite rules; and in operating systems, priorities, as a part of the preemption mechanism, set the order of execution of scheduled tasks.

In the remainder of this section we consider more carefully priorities in term rewriting. *Priority Rewrite Systems*, PRS for short, are term rewriting systems where rewrite rules are equipped with a priority ordering [4,14]. Consider a PRS inspired by Example 4 in [4]. The signature Σ contains booleans $\mathtt{t}$ and $\mathtt{f}$, a constant b, and a binary operator eq that tests equality. The rewrite rules with a priority ordering are given below.

$\downarrow$	$eq(x,x) \rightarrow \mathsf{t}$	$b \rightarrow \mathsf{t}$
	$eq(x,y) \rightarrow \mathsf{f}$	

The rewrite $eq(\mathsf{t},\mathsf{t}) \rightarrow \mathsf{f}$ by the lower rewrite rule for eq is disabled because the higher rule is enabled; hence $eq(\mathsf{t},\mathsf{t}) \rightarrow \mathsf{t}$. Also, the application of the lower rule to get $eq(eq(b,\mathsf{t}),\mathsf{t}) \rightarrow \mathsf{f}$ is disallowed because, although $eq(b,\mathsf{t})$ and t are not syntactically equal, $eq(b,\mathsf{t})$ can be rewritten eventually to t: $eq(b,\mathsf{t}) \rightarrow eq(\mathsf{t},\mathsf{t}) \rightarrow \mathsf{t}$. Hence, $eq(eq(b,\mathsf{t}),\mathsf{t}) \rightarrow \mathsf{t}$ by the higher rule for eq. In general, a rewrite rule r' with a lower priority than r can be applied to rewrite term t in favor of r, if no *internal* reduction (reduction sequence leaving the head operator unaffected) of t can produce an r-redex.

An operational semantics of PRS's in terms of TSS's is given by van de Pol in [18]. He translates a PRS to a TSS and shows that the sound and complete (as in term rewriting) rewrite set for the PRS coincides with the least well-supported model of the TSS as in [9]. However, crucial SOS rules used in the translation do not fit into the NTYFT/NTYXT format: some rules use universal quantification in the premises, while others may rely on the syntactic equality of arguments in the source of the conclusion. It would be interesting to investigate if such a translation can be expressed in terms of general ordered SOS rules. In the meanwhile we give a TSS for the above PRS which satisfies the property that $s \rightarrow a$ iff $s \xrightarrow{a} \mathbf{0}$ for all closed terms s over Σ where a is either t or f. The TSS has an additional constant $\mathbf{0}$ and the following ordered rules:

$\downarrow$	$\dfrac{x \xrightarrow{a} x' \quad y \xrightarrow{a} y'}{eq(x,y) \xrightarrow{\mathsf{t}} \mathbf{0}}$	$\dfrac{x \rightarrow x'}{eq(x,y) \rightarrow eq(x',y)}$	$\dfrac{y \rightarrow y'}{eq(x,y) \rightarrow eq(x,y')}$
	$\dfrac{}{eq(x,y) \xrightarrow{\mathsf{f}} \mathbf{0}}$		

$\downarrow$	$\dfrac{x \rightarrow y \quad y \xrightarrow{a} z}{x \xrightarrow{a} z}$	$\dfrac{}{b \rightarrow \mathsf{t}}$	$\dfrac{}{\mathsf{t} \xrightarrow{\mathsf{t}} \mathbf{0}}$	$\dfrac{}{\mathsf{f} \xrightarrow{\mathsf{f}} \mathbf{0}}$

Note that $eq(eq(b,\mathsf{t}),\mathsf{t}) \rightarrow eq(eq(\mathsf{t},\mathsf{t}),\mathsf{t})$ since $eq(b,\mathsf{t}) \rightarrow eq(\mathsf{t},\mathsf{t})$, and then $eq(eq(\mathsf{t},\mathsf{t}),\mathsf{t}) \xrightarrow{\mathsf{t}} \mathbf{0}$ by the higher rule for eq. Hence, $eq(eq(b,\mathsf{t}),\mathsf{t}) \xrightarrow{\mathsf{t}} \mathbf{0}$ by the XYXT rule directly above.

Consider $eq(eq(b,\mathsf{f}),\mathsf{t})$. We get $eq(b,\mathsf{f}) \rightarrow eq(\mathsf{t},\mathsf{f})$. Since $\mathsf{f} \nrightarrow$ and $\mathsf{f} \stackrel{\mathsf{t}}{\nrightarrow}$, we deduce $eq(b,\mathsf{f}) \xrightarrow{\mathsf{f}}$ by the lower rule. Also, $s \stackrel{\mathsf{t}}{\nrightarrow}$ for all s such that $eq(b,\mathsf{f}) \rightarrow^{*} s$. So, $eq(b,\mathsf{f})$ cannot do a t transition immediately, nor after any number of unlabelled transitions. Therefore $eq(b,\mathsf{f}) \stackrel{\mathsf{t}}{\nrightarrow}$, which implies $eq(eq(b,\mathsf{f}),\mathsf{t}) \xrightarrow{\mathsf{f}} \mathbf{0}$.

8 Conclusions and Future Work

In this paper, we presented several ways of giving a meaning to ordered SOS specifications. Furthermore, we gave semantics-preserving translations (w.r.t. our chosen notion of semantics) between general ordered SOS and (unordered) SOS rule formats, namely the OTYFT and the NTYFT rule formats respectively. The paper is concluded by studying the relative expressive power of the existing OSOS and our novel OTYFT rule formats for ordered SOS. Our results show that the OTYFT rule format is strictly more expressive than the OSOS rule format. This means that there exist transition systems that can be specified by the OTYFT rule format but not by the OSOS rule format.

As pointed out throughout the paper, there are several issues concerning ordered SOS which remain to be studied in the future. The following is a inconclusive list of some ongoing and future research directions.

(1) Universal Quantification in SOS: Both in practical applications [2,18] and in our translation from OTYFT to NTYFT (if one drops the second condition in item 2 of Definition 4.8), one notices the possibility of universally quantifying variables appearing in the target of premises. Inspired by this, in [16], we defined a a generalization of the NTYFT format, called the UNIVERSAL NTYFT format, which allows for such quantifications while preserving the congruence property. The link between the UNIVERSAL NTYFT format and (the generalization of) the OTYFT format need to be further studied.

(2) Semantics of PRS vs. SOS: As already mentioned in Section 7 further investigation is needed as to the relationship between the meaning of PRS's and TSS's. Resolving

the above-mentioned item (i.e., universal quantification in SOS) can help us relate the notions of semantics in [18] and the notions studied in this paper.

(3) Ordered Tree Rules: The issue of expressive power of ordered Tree rules versus rules in the OTYFT format is another subject for our future research.

Acknowledgements

We wish to thank the FSTTCS 2006 referees for helpful comments and suggestions. Anonymous referees of the SOS special issue of Information and Computation provided insightful and useful comments which are gratefully acknowledged. The work of the first author has been partially supported by the projects 'Unifying Framework for Operational Semantics" (nr. 070030041) and "The Equational Logic of Parallel Processes" (nr. 060013021) of The Icelandic Research Fund. The last author would like to thank the University of Leicester for granting study leave, and acknowledge gratefully support from EPSRC, grant EP/D001307/1 entitled "Priorities in Operational Semantics and Term Rewriting", and from Nagoya University during a research visit.

References

[1] Luca Aceto, Wan J. Fokkink, and Chris Verhoef. Structured operational semantics. In Jan A. Bergstra, Alban Ponse, and Scott A. Smolka, editors, *Handbook of Process Algebra*, pages 197–292. Elsevier Science, 2001.

[2] Luca Aceto and Matthew Hennessy. Termination, deadlock, and divergence. *Journal of the ACM*, 39:147–187, 1992.

[3] J.C.M. (Jos) Baeten, Jan A. Bergstra, Jan Willem Klop. Syntax and defining equations for an interrupt mechanism in process algebra. *Fundamenta Informaticae*, XI(2):127–168, 1986.

[4] J.C.M. (Jos) Baeten, Jan A. Bergstra, Jan Willem Klop, and W. Peter Weijland. Term-rewriting systems with rule priorities. *Theoretical Computer Science*, 67(2&3):283–301, 1989.

[5] Bard Bloom, Sorin Istrail and Albert R. Meyer. Bisimulation Can't Be Traced. *Journal of the ACM*, 42(1):232–268, 1995.

[6] Roland Bol and Jan Friso Groote. The meaning of negative premises in transition system specifications. *Journal of the ACM*, 43(5):863–914, September 1996.

[7] Wan J. Fokkink and Rob J. van Glabbeek. Ntyft/ntyxt rules reduce to ntree rules. *Information and Computation*, 126(1):1–10, 1996.

[8] Rob J. van Glabbeek. Bounded Nondeterminism and the Approximation Induction Principle in Process Algebra. In Franz-Josef Brandenburg, Guy Vidal-Naquet and Martin Wirsing, Eds., *Proceedings of the 4th Annual Symposium on Theoretical Aspects of Computer Science (STACS'87)*, volume 247 of *Lecture Notes in Computer Science*, pages 336–347, Springer-Verlag, Berlin, Germany, 1987.

[9] Rob J. van Glabbeek. The meaning of negative premises in transition system specifications II. *Journal of Logic and Algebraic Programming*, 60-61:229–258, 2004.

[10] Jan Friso Groote. Transition system specifications with negative premises. *Theoretical Computer Science*, 118(2):263–299, 1993.

[11] Jan Friso Groote and Frits W. Vaandrager. Structured Operational Semantics and Bisimulation as a Congruence. *Information and Computation*, 100(2):202–260, 1992.

[12] Matthew C.B. Hennessy and Tim Regan. A process algebra for timed systems. *Information and Computation*, 117(2):221–239, 1995.

[13] Ruggero Lanotte and Simone Tini. Probabilistic congruence for semistochastic generative processes. In Vladimiro Sassone, editor, *Proceedings of the 8th International Conference on Foundations of Software Science and Computational Structures (FOSSACS'05)*, volume 3441 of *Lecture Notes in Computer Science*, pages 63–78. Springer-Verlag, Berlin, Germany, 2005.

[14] Chilukuri K. Mohan. Priority rewriting: Semantics, confluence, and conditional. In Nachum Dershowitz, editor, *Proceedings of the 3rd International Conference on Rewriting Techniques and Applications (RTA'89)*, volume 355 of *Lecture Notes in Computer Science*, pages 278–291. Springer-Verlag, Berlin, Germany, 1989.

[15] MohammadReza Mousavi, Iain C.C. Phillips, Michel A. Reniers and Irek Ulidowski. The Meaning of Ordered SOS. In S. Arun-Kumar and and Naveen Garg, editors, *Proceedings of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)*, volume 4337 of Lecture Notes in Computer Science, pages 334–345, Springer-Verlag, Berlin, Germany, 2006.

[16] MohammadReza Mousavi and Michel Reniers. A Congruence Rule Format with Universal Quantification. *Proceedings of the 4th Workshop on Structural Operational Semantics (SOS'07)*, Electronic Notes in Theoretical Computer Science, Elsevier, 2007. To appear.

[17] G. D. Plotkin. A structural approach to operational semantics. *JLAP*, 60-61:17–139, 2004.

[18] Jaco van de Pol. Operational semantics of rewriting with priorities. *Theoretical Computer Science*, 200(1-2):289–312, 1998.

[19] Teodor C. Przymusinski. The Well-Founded Semantics Coincides with the Three-Valued Stable Semantics. *Fundamenta Informaticae*, 13(4):445–463, 1990.

[20] Robert de Simone. Higher-Level Synchronizing Devices in MEIJE-SCCS. *Theoretical Computer Science* 37:245–267, 1985.

[21] Irek Ulidowski and Iain C.C. Phillips. Formats of ordered SOS rules with silent actions. In Michel Bidoit and Max Dauchet, editors, Proceedings of the 7th International Joint Conference on Theory and Practice of Software Development (TAPSOFT'97), volume 1214 of *Lecture Notes in Computer Science*, pages 297–308. Springer-Verlag, Berlin, Germany, 1997.

[22] Irek Ulidowski and Iain C.C. Phillips. Ordered SOS Rules and Process Languages for Branching and Eager Bisimulations. *Information and Computation*, 178(1):180–213, 2002.