A Congruence Rule Format with Universal Quantification

MohammadReza Mousavi [a,b,1], Michel Reniers [a]

[a] Department of Computer Science, Eindhoven University of Technology, P.O. Box 513, NL-5600 MB Eindhoven, The Netherlands

[b] Department of Computer Science, Reykjavík University, Kringlan 1, IS-103 Reykjavík, Iceland

Abstract

We investigate the addition of universal quantification to the meta-theory of Structural Operational Semantics (SOS). We study the syntax and semantics of SOS rules extended with universal quantification and propose a congruence rule format for strong bisimilarity that supports this new feature.

Keywords: Structural Operational Semantics (SOS), Universal Quantification, SOS Rule Formats, Bisimulation, Congruence.

1 Introduction

Structural Operational Semantics (SOS) [14] has been widely used in a variety of forms. Transition System Specification (TSS) [7] is a formalization of SOS which defines a rigorous syntactic and semantic framework for SOS. The notion of TSS paved the way for building up meta-theories around SOS [1,10]; theories about congruence rule formats [7,4] are examples of such meta-theories.

The semantics of a TSS [4,5] comes with an implicit *existential quantification* of valuations of variables used in the rule: if *there exists a* substitution on variables (appearing in the rule) such that the premises of the rule are satisfied, then the conclusion (with the same substitution applied to it) follows. The following example illustrates this.

[1] Corresponding author: `m.r.mousavi@tue.nl`. The work of this 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.

Example 1.1 *Consider the following TSS.*

$$(\mathbf{aaa})\frac{}{a \xrightarrow{a} a} \qquad (\mathbf{aab})\frac{}{a \xrightarrow{a} b} \qquad (\mathbf{bbb})\frac{}{b \xrightarrow{b} b} \qquad (\mathbf{f})\frac{x \xrightarrow{a} y \wedge y \xrightarrow{b}\!\!\!\!\!/\;\,}{f(x) \xrightarrow{c} y}$$

In the aforementioned TSS, it is possible to derive $f(a) \xrightarrow{c} a$ *from* $(\mathbf{f})$ *using rule* $(\mathbf{aaa})$ *since* there exists *a substitution for* x *and* y*, namely,* $\sigma(x) = a$ *and* $\sigma(y) = a$ *such that* $\sigma(x) \xrightarrow{a} \sigma(y)$ *and* $\sigma(y) \xrightarrow{b}\!\!\!\!\!/\;\,$. *The semantics of TSS neglects the fact that there is another substitution* σ', *with* $\sigma'(x) = a$ *and* $\sigma'(y) = b$ *such that the premises of* $(\mathbf{f})$ *do not hold (which is justified due to the aforementioned existential quantification).*

From a purely theoretical viewpoint, there is no reason to only use existential quantification (implicitly) in the premises and it makes sense to study the meta-theory of an SOS framework in which universal quantification over (valuations of) variables is also allowed. Universal quantification in SOS rules appears in practice, too [2,3,11,15]. The following examples illustrate the use of universal quantification in practical instances of operational semantics.

Example 1.2 *In [2], the weak termination predicate is characterized as follows:* $p\surd\!\!\!\surd$ *iff*

(i) $p \xrightarrow{\tau}\!\!\!\!\!/\;\,$ *and* $p\surd$*, or*

(ii) $p \xrightarrow{\tau}$ *and for each* q*,* $p \xrightarrow{\tau} q$ *implies* $q\surd\!\!\!\surd$*.*

A straightforward formalization of this definition by means of deduction rules gives

$$\frac{x \xrightarrow{\tau}\!\!\!\!\!/\;\, \wedge x\surd}{x\surd\!\!\!\surd} \qquad \frac{x \xrightarrow{\tau} \wedge \forall_y (x \xrightarrow{\tau} y \Rightarrow y\surd\!\!\!\surd)}{x\surd\!\!\!\surd}$$

By rewriting implication (and negation), this predicate can be conveniently formulated in terms of deduction rules as follows:

$$\frac{x \xrightarrow{\tau}\!\!\!\!\!/\;\, \wedge x\surd}{x\surd\!\!\!\surd} \qquad \frac{x \xrightarrow{\tau} \wedge \forall_y (x \xrightarrow{\tau}\!\!\!\!\!/\;\, y \vee y\surd\!\!\!\surd)}{x\surd\!\!\!\surd}$$

Example 1.3 *Also in [2], semantical divergence* $p\Downarrow$ *is defined formally by*

$$\frac{x\downarrow \wedge \forall_y (x \xrightarrow{\tau} y \Rightarrow y\Downarrow)}{x\Downarrow} \qquad \text{or by} \qquad \frac{x\downarrow \wedge \forall_y (x \xrightarrow{\tau}\!\!\!\!\!/\;\, y \vee y\Downarrow)}{x\Downarrow}$$

One may argue that universal quantification in the above examples (and other similar ones such as [15, Definition 33]) can be resolved by a semantics-preserving transformation which replaces universally quantified variables / terms with all their possible instantiations. This is not always desired. In a general framework, universal quantification can indeed be seen as an acronym for a (usually infinite) number of existentially quantified premises and/or rules. However, once due to the restrictions in the meta-theory, one restricts the framework to a certain form of rules (i.e., a certain congruence rule format such as the NTyft format), this transformation may change the shape of the rules and take the specification beyond the restricted framework.

Hence, it is worth investigating a framework in which universal quantification is genuinely present and study how much of the meta-theory carries over to this setting. This has already been noted in [17, Section 2] where the author writes.

> ... Moreover, we think it would be a better idea to study a format that allows universal quantification.

This paper takes a step towards the addition of universal quantification to the SOS meta-theory. We slightly extend the syntax of SOS rules with one level of universal quantification. Inspired by examples such as those mentioned before, we also introduce the use of disjunction in the premises of a rule. We define the semantics of such SOS rules as expected. As the main contribution, we propose a congruence rule format for strong bisimilarity that supports these new features. To our knowledge no such format (supporting universal quantification) exists. Our meta-language for SOS rules is still restricted; one may consider a language in which an arbitrary first order predicate formula (thus, an arbitrary mix of existential and universal quantification) is allowed in the premises of deduction rules but this is an extremely complicated problem which we could not tackle in one go.

2 Universal Quantification in TSS

In this section, we first fix a syntax for TSS's with universal quantification and then proceed with defining their semantics. There is little novelty concerning the notions presented here; most of the notions can be traced back to those presented in [15,5].

Fix a *signature* Σ, i.e., a collection of function symbols f, g, ... with fixed arities (natural numbers), $ar(f)$, $ar(g)$, ..., and a countable set $X = \{x, y, z, x_0, \ldots\}$ of *variables*. Function symbols $a, b, c, \ldots$ with arity 0 are also called *constants*. *Open terms* $t, t', t_i \in \mathbb{T}$ are defined inductively using function symbols and variables (while respecting the arities of function symbols). We denote variables of a term t by $\mathcal{V}(t)$. Closed terms $p, q \ldots \in \mathbb{C}$ are terms containing no variable. A *substitution* maps variables to terms and it is closed if its range is a subset of $\mathbb{C}$.

A (transition) clause Φ is defined by the following grammar.

$$\Phi ::= t \xrightarrow{p} t' \mid t \stackrel{p}{\nrightarrow} t' \mid \bigwedge_{i \in I} \Phi_i \mid \bigvee_{i \in I} \Phi_i$$

We restricted the syntax of clauses as given above to facilitate better presentation; adding implication and negation to the above syntax is straightforward but causes a more complicated presentation, especially for our congruence results.

Clauses of the form $t \xrightarrow{p} t'$ and $t \stackrel{p}{\nrightarrow} t'$ are called *positive and negative (transition) formulae*, respectively. The aforementioned formulae are said to *deny* each other, denoted by $\neg t \xrightarrow{p} t' = t \stackrel{p}{\nrightarrow} t'$ and $\neg t \stackrel{p}{\nrightarrow} t' = t \xrightarrow{p} t'$. The formulae $\bigwedge_{i \in I} \Phi_i$ and $\bigvee_{i \in I} \Phi_i$ denote conjunction and disjunction, respectively, over formulae parameterised by an index variable i quantified over a possibly infinite index set I. To unclutter presentation we do not treat predicate formulae in our framework (e.g., formulae of the form $P(t)$ or $\neg P(t)$) but allow for their presence and consider them as transition formulae with dummy labels and targets. The rest of this paper can be re-phrased in the genuine presence of predicate formulae without any substantial change in the formal development of the paper.

We intend to add one level of universal quantification and this suffices for the applications we have encountered thus far in the literature. It seems reasonable to make the already existing and implicit existential quantification in rules explicit. This raises the question as to whether these existentially bound variables are bound outside or inside the universal quantification, i.e., whether the clause should be augmented as $\exists_{\widetilde{z_0}}\forall_{\widetilde{z_1}}\Phi$ or $\forall_{\widetilde{z_1}}\exists_{\widetilde{z_0}}\Phi$ where $\widetilde{z_0}$ represents the set of (formerly implicitly bound) existentially quantified variables. We decided to go for maximum generality in our setting and avoid the design decision altogether. In other words, we allow for disjoint sets of existentially quantified variables appearing before as well as after the universal quantification. Of course, the ultimate goal would be to have a general first order language and allow for all sorts of nested quantifiers.

Another decision we made is to write the quantifiers in front of the deduction rules since they may also apply to the occurrences of the quantified variables in the target of the conclusion.

A TSS is a set of deduction rules of the form

$$(\mathbf{r})\ \exists_{\widetilde{z_0}}\forall_{\widetilde{z_1}}\exists_{\widetilde{z_2}}\ \frac{\Phi}{\phi},$$

where $\widetilde{z_0}$, $\widetilde{z_1}$, and $\widetilde{z_2}$ stand for sets of variables, Φ is a transition clause and ϕ is a positive formula. Clause Φ is called the *premises* (each formula appearing in Φ is called a *premise*) and ϕ is a positive formula which is called the *conclusion* of deduction rule $(\mathbf{r})$. Assume that $\phi = t \xrightarrow{p} t'$; we call t the *source* of the above deduction rule. A deduction rule (TSS) without universal quantification and disjunction is called a *traditional* deduction rule (TSS). For such traditional deduction rules one can represent the conjunction of transition formulae as a set (as we do in Definition 2.3 below). To avoid any ambiguity we assume that $\mathcal{V}(r) \subseteq \mathcal{V}(t) \cup \widetilde{z_0} \cup \widetilde{z_1} \cup \widetilde{z_2}$ (where $\mathcal{V}(r)$ denotes all variables appearing in the premises and the conclusion of the deduction rule $(\mathbf{r})$) and assume that $\mathcal{V}(t)$, $\widetilde{z_0}$, $\widetilde{z_1}$ and $\widetilde{z_2}$ are all pairwise disjoint.

We decided to quantify on valuations of variables only since intuitively, quantification over valuations of (open) terms reduces to quantification over variables. In the remainder of this paper, for better presentation, we assume that premises are in the disjunctive normal form, i.e., of the form $\bigvee_{i\in I}\bigwedge_{j\in J}\phi_{ij}$ where I and J are index sets, and ϕ_{ij} is a positive or negative formula. Note that in the absence of universal quantification, a rule with a number of disjuncts as premises can be represented by a number of rules; one for each disjunct. However, in the presence of universal quantification, this cannot be done because $\forall_z(\phi \vee \psi)$ is not equivalent to $(\forall_z\phi) \vee (\forall_z\psi)$.

Example 2.1 *The weak termination operator of Example 1.2 is rephrased in our syntax as follows.*

$$\forall_y \frac{x \overset{\tau}{\nrightarrow} y \wedge x\surd}{x \surd\!\!\!\surd} \qquad \exists_{y'}\forall_y \frac{(x \xrightarrow{\tau} y' \wedge x \overset{\tau}{\nrightarrow} y) \vee (x \xrightarrow{\tau} y' \wedge y \surd\!\!\!\surd)}{x \surd\!\!\!\surd}$$

Example 2.2 *The semantical divergence of Example 1.3 can be rewritten into the following extended TSS.*

$$\forall_y \frac{(x\downarrow \wedge x \stackrel{\tau}{\nrightarrow} y) \vee (x\downarrow \wedge y \Downarrow)}{x \Downarrow}$$

2.1 Extended TSS: Semantics

Semantics of extended TSS's do not defer much from traditional TSS's. In [4,5], the following notion of three-valued stable model is defined for *closed traditional* TSS's, i.e., TSS's containing only traditional deduction rules which do not contain any variable.

Definition 2.3 (Proof) *A traditional deduction rule* $\frac{\Phi}{\phi}$ *is* provable *from a closed traditional TSS* R*, denoted by* $R \vdash \frac{\Phi}{\phi}$*, when there exists a well-founded upwardly branching tree with formulae as nodes and of which*

- *the root is labeled by* ϕ*;*
- *if a node is labeled by* ψ *and the nodes above it form the set* K *then one of the following two cases hold:*
  · $\psi \in \Phi$ *and* $K = \emptyset$*;*
  · ψ *is a positive transition formula and* $\frac{K}{\psi} \in R$*.*

When the TSS is known from the context, we omit it from the notation and just write $\vdash \frac{\Phi}{\phi}$.

Definition 2.4 (Truth) *A negative transition formula* $\phi = p \stackrel{l}{\nrightarrow} p'$ *is* true *for a set* PF *of positive formulae, denoted by* $PF \vDash \phi$ *when* $p \stackrel{l}{\rightarrow} p' \notin PF$*. A set* NF *of negative formulae is true for the set* PF*, denoted by* $PF \vDash NF$ *when for all* $\phi \in NF$*,* $PF \vDash \phi$*.*

Definition 2.5 (Three-Valued Stable Models) *A pair* (C, U) *of sets of* positive closed formulae *(where* C *stands for* Certain *and* U *for* Unknown*; the third value is determined by the formulae not in* U*) is called a three-valued stable model for a TSS when* $C \subseteq U$ *and*

- *for all* $\phi \in C$*,* $\vdash \frac{N}{\phi}$ *for a set* N *of negative closed transition formulae such that* $U \vDash N$*;*
- *for all* $\phi \in U$*,* $\vdash \frac{N}{\phi}$ *for a set* N *of negative closed transition formulae such that* $C \vDash N$*.*

In [16,5], it has been shown that every TSS admits a least three-valued stable model with respect to the information theoretic ordering (i.e., $(C, U) \leq (C', U')$ when $C \subseteq C'$ and $U' \subseteq U$). A TSS is called *complete* [5] (or *positive after reduction* [4]) if for its least three-valued stable model (C, U), $C = U$.

To define the semantics of traditional TSS's in general, one has to instantiate the deduction rules with all closing substitutions and then use the above definition on the resulting closed TSS. A similar approach, as suggested in [15], can be used to define a meaning for TSS's with universal quantifiers. First, each rule is replaced with a number of traditional rules, of which the premises contain all possible instantiations for the universally quantified variables and some instance of the existentially quantified ones (for each instance of the universally quantified variables, similar to the idea of Skolemization). Second, the remaining variables, i.e., the variables in the source of the conclusion, are instantiated with all possible substitutions resulting in a closed traditional TSS. The following definitions formalize this idea.

Definition 2.6 *For a traditional deduction rule $r = \frac{\Phi}{\phi}$, its closure, $cl(r)$ is the set of deduction rules $\{\frac{\sigma(\Phi)}{\sigma(\phi)} \mid \sigma : X \to \mathbb{C}\}$. Closure of a traditional TSS R, denoted by $cl(R)$ is defined by $\bigcup_{r \in R} cl(r)$. The semantics of R is defined by the semantics of $cl(R)$.*

For each deduction rule r of the following form,

$$(\mathbf{r}) \exists_{\widetilde{z_0}} \forall_{\widetilde{z_1}} \exists_{\widetilde{z_2}} \frac{\bigvee_{i \in I} \bigwedge_{j \in J} \phi_{ij}}{t \xrightarrow{l} t'}$$

$sk(r)$ is the set of all deduction rules $sk(r, \sigma_0, \sigma_{10}, \ldots, \sigma_{20}, \ldots, i_0, \ldots \mid i_j)$ for each substitution $\sigma_0 : \widetilde{z_0} \to \mathbb{C}$, series of substitutions $\sigma_{10}, \sigma_{11}, \ldots : \widetilde{z_1} \to \mathbb{C}$ such that for each variable $z \in \widetilde{z_1}$, $\{\sigma_{10}(z), \sigma_{11}(z), \ldots\} = \mathbb{C}$, series of substitutions $\sigma_{20}, \sigma_{21}, \ldots : \widetilde{z_2} \to \mathbb{C}$, series of indices $i_0, i_1, \ldots \in I$ and each $i_j \in \{i_0, i_1, \ldots\}$ which is defined as follows.

$$\frac{(\bigwedge_{j \in J} \sigma_0 \cdot \sigma_{10} \cdot \sigma_{20}(\phi_{i_0 j})) \wedge (\bigwedge_{j \in J} \sigma_0 \cdot \sigma_{11} \cdot \sigma_{21}(\phi_{i_1 j})) \wedge \ldots}{\sigma_0 \cdot \sigma_{1 i_j} \cdot \sigma_{2 i_j}(t \xrightarrow{l} t')}$$

In the above deduction rule $\cdot$ denotes function composition and it binds stronger than function application. If any of the sets $\widetilde{z_i}$ (for each $i \in \{0, 1, 2\}$) is empty then one should drop all σ_{ij} (σ_i, for $i = 0$) from the definition of $sk(r)$. Also in the case of $I = \emptyset$, all i_j components should be dropped from the definition of $sk(r)$.

Note that the above deduction rule is traditional and hence, sets of such deduction rules can be given a semantics using traditional ways of assigning meaning to TSS's. The meaning of a TSS R with universal quantification is defined as the meaning of $\bigcup_{r \in R} sk(r)$.

The following simple example illustrates the semantics of extended TSS's.

Example 2.7 *Consider the following TSS.*

$$(\mathbf{aaa})\frac{}{a \xrightarrow{a} a} \qquad (\mathbf{aab})\frac{}{a \xrightarrow{a} b} \qquad (\mathbf{bab})\frac{}{b \xrightarrow{a} b} \qquad (\mathbf{bbb})\frac{}{b \xrightarrow{b} b}$$

$$(\mathbf{f}) \forall_y \exists_z \frac{x \overset{a}{\nrightarrow} y \vee y \xrightarrow{b} z}{f(x) \xrightarrow{c} c}$$

Assume $A = \{(\mathbf{aaa}), (\mathbf{aab}), (\mathbf{bab}), (\mathbf{bbb})\}$; it holds that $A = \bigcup_{r \in A} sk(r)$, i.e., since deduction rules in A do not contain quantified variables, their Skolemization yields *the same deduction rules. However,* (**f**) *contains a universally quantified variable* y *and an existentially quantified variable* z*. (In terms of the notation used in Definition 2.6,* $\widetilde{z_0} = \emptyset$, $\widetilde{z_1} = \{y\}$ *and* $\widetilde{z_2} = \{z\}$*.) Let* $\phi_0 = x \overset{a}{\nrightarrow} y$ *and* $\phi_1 = y \overset{b}{\rightarrow} z$*;* $sk(f)$ *is defined as follows.*

$$\{(sk(f, \sigma_{10} = [y \mapsto p_{10}], \sigma_{20} = [z \mapsto p_{20}], \ldots, i_0, i_1, \ldots, i))\frac{\bigwedge_{j \in I\!N} \sigma_{1j} \cdot \sigma_{2j}(\phi_{i_j})}{f(x) \overset{c}{\rightarrow} c} \mid$$
$$\forall_{k \in \{0,1\}, l \in I\!N} \;\; p_{kl} \in \mathbb{C} \wedge \{p_{10}, p_{11}, \ldots\} = \mathbb{C} \wedge i_l \in \{0,1\} \wedge i \in \{0,1\}\}$$

For example, $sk(f, \sigma_{10}, \sigma_{20}, \ldots, i_0, \ldots, i)$ *where* $i_j = 0$*, for each* $j \in I\!N$ *(for arbitrary* i*,* σ_{1j} *and* σ_{2j}*) is the following deduction rule.*

$$\frac{\bigwedge_{p \in \mathbb{C}} x \overset{a}{\nrightarrow} p}{f(x) \overset{c}{\rightarrow} c}$$

The least three-valued stable model of the TSS is the pair (C, U) *where* $C = U = \{a \overset{a}{\rightarrow} a,\ a \overset{a}{\rightarrow} b,\ b \overset{a}{\rightarrow} b,\ b \overset{b}{\rightarrow} b,\ f(p) \overset{c}{\rightarrow} c \mid p \in \mathbb{C} \setminus \{a\}\}$*. Hence, the TSS is complete.*

3 Universal NTyft

3.1 Bisimilarity and Congruence

Strong bisimulation [12], as defined below, is a key notion of behavioral equivalence in concurrency theory.

Definition 3.1 (Bisimulation and Bisimilarity) *A symmetric relation* $R \subseteq \mathbb{C} \times \mathbb{C}$ *is a bisimulation relation when for all* $p, q \in \mathbb{C}$ *such that* $p\ R\ q$*,* $l \in \mathbb{C}$*, and* $p' \in \mathbb{C}$*, if* $p \overset{l}{\rightarrow} p'$ *then there exists a* q'*,* $q \overset{l}{\rightarrow} q'$ *and* $p'\ R\ q'$*.*

Two closed terms p *and* q *are* bisimilar*, denoted by* $p \leftrightarroweq q$*, when there exists a bisimulation relation* R *such that* $p\ R\ q$*.*

To treat bisimilarity compositionally and algebraically, it is essential to make sure that it is a congruence relation, i.e., one can replace equals by equals.

Definition 3.2 (Congruence) *An equivalence relation* R *is a* congruence *w.r.t. a function symbol* f *(with an arbitrary arity* n*), when for all* $\overrightarrow{p}$*,* $\overrightarrow{q}$*, if* $\overrightarrow{p}\ R\ \overrightarrow{q}$ *then* $f(\overrightarrow{p})\ R\ f(\overrightarrow{q})$*.* R *is a congruence w.r.t. a signature* Σ *when it is a congruence for all function symbols* $f \in \Sigma$*.*

3.2 Rule Format and Its Proof

In this section, the rule format is defined that should guarantee congruence of bisimilarity and this is proven.

Definition 3.3 (Variable Dependency Ordering) *For a deduction rule* r *of the* *form* $(\mathbf{r})\exists_{\widetilde{z_0}}\forall_{\widetilde{z_1}}\exists_{\widetilde{z_2}}\frac{\bigvee_{k\in K}(\bigwedge_{i\in I_k} t_i \xrightarrow{l_i} y_i \wedge \bigwedge_{j\in J_k} t'_j \overset{l'_j}{\nrightarrow} y'_j)}{t \xrightarrow{l} t'}$, *the variable dependency ordering* $<_r$ *is the smallest relation containing all pairs* (u, y_i) *and* (u', y'_j) *where* $u \in \mathcal{V}(t_i)$ *and* $u' \in \mathcal{V}(t'_j)$ *for each* $i \in I_k$*,* $j \in J_k$ *and* $k \in K$*.*

A deduction rule (TSS) is well-founded if the variable dependency ordering of the rule is (all its deduction rules are) well-founded.

Definition 3.4 (UNTyft/UNTyxt) *A deduction rule of the following form*

$$(\mathbf{r})\exists_{\widetilde{z_0}}\forall_{\widetilde{z_1}}\exists_{\widetilde{z_2}}\frac{\bigvee_{k\in K}(\bigwedge_{i\in I_k} t_i \xrightarrow{l_i} y_i \wedge \bigwedge_{j\in J_k} t'_j \overset{l'_j}{\nrightarrow} y'_j)}{t \xrightarrow{l} t'},$$

is in the UNTyft format when it satisfies the following conditions:

(i) *t is of the form $f(\overrightarrow{x})$;*

(ii) $\forall_{i,i'\in\bigcup_{k\in K} I_k} y_i \neq y_{i'} \wedge\ y_i \notin \mathcal{V}(t)$ *and* $\forall_{j,j'\in\bigcup_{k\in K} J_k} y'_j \neq y'_{j'} \wedge\ y'_j \notin \mathcal{V}(t)$ *(targets of positive and negative transition formulae are all distinct variables and are all different from variables in the source of the conclusion);*

(iii) $\widetilde{z_1} \cap \{y_i \mid i \in I_k, k \in K\} = \emptyset$ *(universally quantified variables cannot appear as targets of positive premises)*

(iv) $\{y'_j \mid j \in J_k, k \in K\} \subseteq \widetilde{z_1}$ *(all targets of negative premises should be universally quantified);*

(v) $\forall_{z\in\widetilde{z_0}}\forall_{k\in K}\forall_{i\in I_k}\ z = y_i \Rightarrow \forall_{u\in\widetilde{z_1}\cup\widetilde{z_2}}\neg(u <_r z)$ *(if an existentially quantified variable in $\widetilde{z_0}$ appears in the target of a premise, then it does not depend on variables among those in $\widetilde{z_1}$ or $\widetilde{z_2}$).*

A deduction rule of the above form is in the UNTyxt rule format when t is of the form x and it satisfies items (ii)-(v)*.*

A TSS is in the UNTyft(/UNTyxt) format when all its deduction rules are.

An immediate question that comes to mind is how the UNTyft format compares to the NTyft format. It is not hard to see that the UNTyft format extends the NTyft formats (by taking sets K to be a singleton, $\widetilde{z_0}$ to be $\emptyset$ and $\widetilde{z_1}$ to be $\{y_j \mid j \in J_k, k \in K\}$, one obtains the NTyft format). In terms of expressive power, i.e., the set of definable transition relations, the following example shows that the UNTyft format is strictly more expressive than the NTyft format. (The example is essentially taken from [9, Example 4.9].)

Example 3.5 (UNTyft vs. NTyft)

$$\frac{}{a \xrightarrow{a} d} \quad \frac{}{b \xrightarrow{a} d} \quad \frac{}{b \xrightarrow{a} c} \quad \frac{}{c \xrightarrow{b} d} \quad \exists_{y',y''}\ \forall_{y,z}\ \frac{(x \xrightarrow{a} y' \wedge x \overset{a}{\nrightarrow} y) \vee (x \xrightarrow{a} y'' \wedge y \overset{b}{\nrightarrow} z)}{f(x) \xrightarrow{c} d}$$

The above TSS is in the UNTyft format, it 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 proof for* $f(a) \xrightarrow{c} d \in C$ *(a minimal proof is a proof* *in which no formula appears more than once in a branch of the proof tree); using the same deduction rule leading to this proof and a new substitution, we prove that* $f(b) \xrightarrow{c} d$ *(contradiction).*

Assume that the proof for $f(a) \xrightarrow{c} d$ *is due the rule* $(\mathbf{r})\frac{\{t_i \xrightarrow{p_i} y_i | i \in I\} \quad \{t_j \stackrel{p_j}{\nrightarrow} | j \in J\}}{f(x) \xrightarrow{c} t}$ *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:*

(i) $x \xrightarrow{a} y_i$ *or* $a \xrightarrow{a} y_i$*, for some* $i \in I$*,*

(ii) $b \xrightarrow{a} y_i$*, for some* $i \in I$*,*

(iii) $t_i \xrightarrow{b} y_i$ *or* $c \xrightarrow{b} y_i$*, where* $\sigma(t_i) = c$ *and* $i \in I$*,*

(iv) $t_j \stackrel{a}{\nrightarrow}$ *where* t_j *can be an arbitrary term but* a*,* b *or* x*, and* $j \in J$*,*

(v) $t_j \stackrel{b}{\nrightarrow}$ *where* t_j *can be any term such that* $\sigma(t_j) \neq c$*, and* $j \in J$*,*

(vi) $t_j \stackrel{c}{\nrightarrow}$ *where* t_j *can be any term but* $f(a)$ *or* $f(x)$ *and* $j \in J$*, (these two cases are excluded since otherwise,* $f(a) \xrightarrow{c} d$ *cannot be included in* C*),*

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 proof of* $f(a) \xrightarrow{c} d$ *due to* $(\mathbf{r})$ *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 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 proof for* $\sigma'(f(x) \xrightarrow{c} t)$*, i.e.* $f(b) \xrightarrow{c} d$*.*

Theorem 3.6 *For a complete and well-founded TSS in the* UNTyft/UNTyxt *format bisimilarity is a congruence.*

Proof. We prove the theorem for a TSS in the UNTyft format. For deduction rules in the UNTyxt format essentially the same proof technique can be adopted. We use the following auxiliary definition for our inductive proof.

Definition 3.7 (Reduction Technique for SOS with Negative Premises) *For an ordinal* α*, define:*

$$C_\alpha \doteq \{\phi \Vdash \frac{N}{\phi} \wedge \exists_{\beta<\alpha} U_\beta \vDash N\}$$
$$U_\alpha \doteq \{\phi \Vdash \frac{N}{\phi} \wedge \forall_{\beta<\alpha} C_\beta \vDash N\}$$

It follows from the above two items that $C_0 = \emptyset$, $U_0 = \{\phi \Vdash \frac{N}{\phi}\}$.

It follows from Tarski's fixpoint theorem (observing that $C_\alpha \subseteq C_\beta$ and $U_\beta \subseteq U_\alpha$ for $\alpha \leq \beta$) that the above reduction procedure will reach a fixpoint at an ordinal, say λ and it follows from the above definition that (C_λ, U_λ) is indeed the least three-valued stable model.

Define R to be the smallest congruence containing the bisimilarity $\underline{\leftrightarrow}$ associated with the TSS in the UNTyft format. If we show that R is a bisimulation relation, then the theorem follows. Instead, by an induction on α, we simultaneously prove that the following two statements hold for each $(p,q) \in R$, for each $l, p' \in \mathbb{C}$, and for each α.

(i) $p \xrightarrow{l} p' \in C_\alpha \Rightarrow \exists_{q'} q \xrightarrow{l} q' \in C_\lambda \wedge (p',q') \in R$;

(ii) $p \xrightarrow{l} p' \in U_\lambda \Rightarrow \exists_{q'} q \xrightarrow{l} q' \in U_\alpha \wedge (p',q') \in R$;

Once we prove the above two statements, the transfer conditions for bisimulation (w.r.t. $C_\lambda = U_\lambda$) follow by taking α to be λ and from the fact that $C_\lambda = U_\lambda$ (due to completeness of the TSS under consideration).

Note that bisimilarity is an equivalence and so is R; thus, we assume the symmetric statements for q without having to prove them. The above statements hold trivially for all p and q such that $p \underline{\leftrightarrow} q$. Hence, we focus on terms of the form $p = f(\overrightarrow{p})$ and $q = f(\overrightarrow{q})$ where $\overrightarrow{p}\ R\ \overrightarrow{q}$.

(i) It follows from Definition 3.7 that $\vdash \dfrac{N}{p \xrightarrow{l} p'}$ for some N and some $\beta < \alpha$ such that $U_\beta \vDash N$, and from Definition 3.4 that there exists a deduction rule r of the following form

$$(\mathbf{r}) \exists_{\widetilde{z_0}} \forall_{\widetilde{z_1}} \exists_{\widetilde{z_2}} \frac{\bigvee_{k\in K}(\bigwedge_{i\in I_k} t_i \xrightarrow{l_i} y_i \wedge \bigwedge_{j\in J_k} t'_j \overset{l'_j}{\nrightarrow} y'_j)}{t \xrightarrow{l} t'},$$

and (according to Definition 2.6) there exist substitutions $\sigma_p : \mathcal{V}(\overrightarrow{x}) \to \mathbb{C}$ and $\sigma_0 : \widetilde{z_0} \to \mathbb{C}$ such that $\sigma_p(\overrightarrow{x}) = \overrightarrow{p}$ and for all substitutions $\sigma_1 : \widetilde{z_1} \to \mathbb{C}$, there exists a substitution $\sigma_{\sigma_1} : \widetilde{z_2} \to \mathbb{C}$ and an index $k \in K$, such that all positive formulae with indices i and j with $i \in I_k$ and $j \in J_k$ under $\sigma = \sigma_p \cdot \sigma_0 \cdot \sigma_1 \cdot \sigma_{\sigma_1}$ hold, i.e., $\vdash \dfrac{N_i}{\sigma(t_i \xrightarrow{l_i} y_i)}$ with a smaller proof structure and $N_i \subseteq N$ for each $i \in I_k$, and $\sigma(t'_j \xrightarrow{l_j} y'_j) \in N$ for each negative premise $j \in J_k$. We proceed with an induction on the proof structure for $\vdash \dfrac{N}{p \xrightarrow{l} p'}$.

In a traditional proof method for congruence rule formats (e.g., that of [7]), one aims at defining a new substitution σ' such that $\sigma'(\overrightarrow{x}) = \overrightarrow{q}$ and $\sigma(u)\ R\ \sigma'(u)$ for each variable $u \in X$; furthermore, while completing the definition of σ', one shows, using the induction hypothesis, that all the premises also hold under σ', thus, obtaining a proof for $q \xrightarrow{l} q'$, for some q' such that $\sigma(t')\ R\ q'$ and $q' = \sigma'(t')$. Our proof method is slightly more involved. Since we have a universal quantification over variables in $\widetilde{z_1}$, we are allowed to use the fact that under all substitutions $\sigma_1 : \widetilde{z_1} \to \mathbb{C}$ at least one disjunct among the premises holds by choosing an appropriate σ_{σ_1}. Thus, during the construction of σ', as explained below, we also change substitution σ into some $\sigma'' = \sigma_p \cdot \sigma_0 \cdot \sigma''_1 \cdot \sigma_{\sigma''_1}$ (by choosing a σ''_1 which is appropriate for our proof obligation), while preserving $\sigma''(u)\ R\ \sigma'(u)$. Note that σ and σ'' agree on the variables in $\widetilde{z_0}$. Furthermore, σ and σ' agree on the variables in $\widetilde{z_1}$.

Let $\sigma_q : \mathcal{V}(\overrightarrow{x}) \to \mathbb{C}$ be such that $\sigma_q(\overrightarrow{x}) = \overrightarrow{q}$. Given σ_0 and for each σ_1 as given above, we aim at constructing new substitutions σ'_0, σ''_1 and σ'_{σ_1} such that $\sigma''_1(z_1)\ R\ \sigma_1(z_1)$ and $\sigma''(u)\ R\ \sigma'(u)$ for each $z_1 \in \widetilde{z_1}$ and for each $u \in X$ where $\sigma'' = \sigma_p \cdot \sigma_0 \cdot \sigma''_1 \cdot \sigma_{\sigma''_1}$ and $\sigma' = \sigma_q \cdot \sigma'_0 \cdot \sigma_1 \cdot \sigma'_{\sigma_1}$. Note that $\sigma_{\sigma''_1}$ need not be re-defined; given σ''_1, it is determined by the deduction rule chosen to derive $p \xrightarrow{l} p'$ according to Definition 2.6, i.e., if $\sigma''_1 = \sigma_{1j}$, for some $j \in \mathbb{N}$, $\sigma_{\sigma''_1}$ is σ_{2j}.

To define σ' and σ'', we start with σ'_{11} where $\sigma'_{11}(u) = \sigma(u)$ for each variable in $u \in (\widetilde{z_0} \cup \widetilde{z_2}) \setminus \{y_i, y_j \mid i \in I_{k'}, j \in J_{k'}, k' \in K\}$ and undefined otherwise, and a substitution σ''_{11} such that $\sigma''_{1u} = \sigma(u)$ for each variable in $u \in \widetilde{z_1} \setminus \{y_j \mid j \in J_{k'}, k' \in K\}$.

Consider substitutions σ'_{1i} and σ''_{1i} and a variable u such that for each variable x preceding u in the variable dependency graph either $\sigma'_{1i}(x)$ or $\sigma''_{1i}(x)$ is defined. Furthermore, we assume that for all such variables x, $\sigma''_i(x)\ R\ \sigma'_i(x)$ where $\sigma'_i = \sigma_q \cdot \sigma_1 \cdot \sigma'_{1i}$, $\sigma''_i = \sigma_p \cdot \sigma_0 \cdot \rho_i \cdot \sigma_{\rho_i}$ and $\rho_i = \sigma_1 \uparrow \sigma''_{1i}$ where $(\sigma_1 \uparrow \sigma''_{1i})(x) = \sigma''_{1i}(x)$ if $\sigma''_{1i}(x)$ is defined and $\sigma_1(x)$ otherwise.

We define a procedure which takes any such variable u and substitutions σ'_{1i} and σ''_{1i} and defines the substitution σ'_{1i+1} which agrees with σ'_{1i} on the domain of σ'_{1i} and extends the domain of σ'_{1i} with u, if $u \in \widetilde{z_0} \cup \widetilde{z_2}$ in such a way that $\sigma''_i(u)\ R\ \sigma'_{i+1}(u)$. Furthermore, if $u \in \widetilde{z_1}$, we define a value for $(\sigma''_{1i+1})(u)$, in such a way that $\sigma''_{1i}(u)\ R\ \rho(\sigma_1)(u)$, thus in both cases maintaining $\sigma''_{i+1}\ R\ \sigma'_{i+1}(u)$. If $u \in \widetilde{z_1}$, then σ'_{1i+1} is the same as σ'_{1i}; if $u \notin \widetilde{z_1}$, then σ''_{1i+1} is the same as σ''_{1i}. Then, substitutions σ' and σ'' are defined as the greatest fixed point of the chain σ'_i's and σ''_i's (taking the above-mentioned procedure as a monotone function, with the subset relation on the union of the domains of the substitutions σ'_i and σ''_{1i} as the ordering).

We make a case distinction based on the status of variable u with respect to set $\widetilde{z_0}$, $\widetilde{z_1}$ and $\widetilde{z_2}$. (We shall still use an induction on α and inside that an induction on the structure of the proof in the following items.)

(a) Assume that $u \in \widetilde{z_0} \cup \widetilde{z_2}$; then, u can only be a variable $y_{i'}$ for some $i' \in I_k$ and $k \in K$ (i.e., the target of a positive premise). We distinguish the following two cases based on the status of $\sigma''_i(t_{i'} \xrightarrow{l_{i'}} y_{i'})$ with respect to C_α.

  Assume that $\sigma''_i(t_{i'} \xrightarrow{l_{i'}} y_{i'})$ is among the premises of the rule proving $\sigma(t \xrightarrow{l} t')$, i.e., $\sigma''_i(t_{i'} \xrightarrow{l_{i'}} y_{i'}) \in C_\alpha$ with a proof structure which is smaller than the proof of $\sigma(t \xrightarrow{l} t')$. Considering that $\sigma''_i(t_{i'})\ R\ \sigma'_i(t'_i)$, the induction hypothesis on the structure of the proof applies and we have that $\sigma'_i(t_{i'}) \xrightarrow{l_{i'}} q_{i'} \in C_\lambda$ for some q_i such that $\sigma(u)\ R\ q_{i'}$ (and thus, $\sigma''_i(u)\ R\ q_i$). Define $\sigma'_{1i+1}(u) = q_i$.

  Otherwise, assume that $\sigma''_i(t_{i'} \xrightarrow{l_{i'}} y_{i'})$ is not in the proof tree for $\sigma(t \xrightarrow{l} t')$. Take $\sigma'_{1i+1}(u) = \sigma(u)$.

  Note that since $u \notin \widetilde{z_1}$, in both cases $\sigma''_{i+1} = \sigma''_i$.

(b) Assume that $u \in \widetilde{z_1}$; then, $u = y_j$, for some $j \in J_k$ and $k \in K$. We distinguish the following two cases.

  Either $\exists_{\beta<\alpha}\ \forall_{p_j}\ \sigma(y_j)\ R\ p_j \Rightarrow \sigma''_i(t_j) \xrightarrow{l_j} p_j \notin U_\beta$; it follows from the induction hypothesis (on α; contraposition of item (ii)) that $\sigma'_i(t_j) \xrightarrow{l_j} p_j \notin$

U_λ. Define $\sigma''_{1i+1}(u) = \sigma(u)$.

Or $\forall_{\beta<\alpha}\ \exists_{p_j}\ \sigma(y_j)\ R\ p_j \wedge \sigma''_i(t_j) \xrightarrow{l_j} p_j \in U_\beta$. It follows from the fact that for all $\gamma \leq \gamma'$, $U'_\gamma \subseteq U_\gamma$ that $\exists_{p_j}\ \forall_{\beta<\alpha}\ \sigma(y_j)\ R\ p_j \wedge \sigma''_i(t_j) \xrightarrow{l_j} p_j \in U_\beta$. Define $\sigma''_{1i+1}(u) = p_j$.

This way, we have completed the definition of σ' and σ''. There is a $k \in K$ such that for all $i' \in I_k$, $\sigma''(t_{i'} \xrightarrow{l_{i'}} y_{i'}) \in C_\alpha$ and it follows from the construction of σ' that $\sigma'(t_i \xrightarrow{l_i} y_i) \in C_\lambda$. Furthermore, it holds for all $j \in J_k$ that $\sigma''(t_j \xrightarrow{l_j} y_j) \notin U_\beta$, for some $\beta \leq \alpha$. It again follows from the above construction of σ'' that $\sigma''(t_j \xrightarrow{l_j} p_j) \notin U_\beta$ for all p_j such that $\sigma(u_j) = \sigma'(u_j)\ R\ p_j$ and hence, $\sigma'(t_j \xrightarrow{l_j} p_j) \notin U_\lambda$. This completes the proof for $q \xrightarrow{l} \sigma'(t') \in C_\lambda$ and we have that $\sigma''(t)\ R\ \sigma'(t)$.

(ii) The case is dual to the above case. One just has to replace the sets C_α with U_λ and C_λ with U_α, simultaneously.

□

3.3 (Counter-)Examples

In this section, we give a few (counter-)examples witnessing the generality of our rule format. First, we show that our format is general enough to cover our motivating examples.

Example 3.8 *The deduction rules for weak termination and divergence as specified, respectively, in Examples 2.1 and 2.2 are in the* UNTyft/UNTyxt *format.*

Next, we show that the syntactic constraints concerning the UNTyft format cannot be simply dropped or the congruence meta-result will be ruined. The first condition in the UNTyft format concerns the source of the conclusion and it is among the conditions of the ordinary Tyft and NTyft formats. Thus, counter-examples given in [7,6] work in this case, as well. Constraint (ii) is about distinctness of variables appearing as targets of premises. Our addition to the traditional constraints of the NTyft format is that we prohibited the repetition of target variables among different disjuncts. The following counter-example shows that this additional constraint cannot be dropped.

Example 3.9 *The following specification conforms to all constraints of the* UNTyft *format but constraint (ii) in that variable y is repeated in the target of the premises of the left-most deduction rule. Moreover, it is complete and is well-founded.*

$$\forall_y \frac{x \overset{a}{\nrightarrow} y \vee x \overset{b}{\nrightarrow} y}{f(x) \xrightarrow{c} c} \qquad \frac{}{a \xrightarrow{a} a} \qquad \frac{}{a \xrightarrow{b} a} \qquad \frac{}{b \xrightarrow{a} a} \qquad \frac{}{b \xrightarrow{b} b}$$

For the above specification, it holds that $a \underline{\leftrightarrow} b$ but it does not hold that $f(a) \underline{\leftrightarrow} f(b)$ since $\forall_y\ b \overset{a}{\nrightarrow} y \vee b \overset{b}{\nrightarrow} y$ but it does not hold that $\forall_y\ a \overset{a}{\nrightarrow} y \vee a \overset{b}{\nrightarrow} y$, namely $a \xrightarrow{a} a$ and $a \xrightarrow{b} a$.

Constraint (iii) states that universally quantified variables cannot appear as targets of positive premises. The following counter-example shows the role of this constraint in establishing congruence.

Example 3.10 *The following TSS is complete and well-founded and satisfies all constraints of the* UNTyft *format but constraint (iii).*

$$\exists_x \frac{}{a \xrightarrow{a} x} \qquad \frac{}{b \xrightarrow{a} a} \qquad \frac{}{b \xrightarrow{a} c} \qquad \frac{}{b \xrightarrow{a} f(a)} \qquad \forall_y \frac{x \xrightarrow{a} y}{f(x) \xrightarrow{c} a}$$

It holds for the above TSS that $a \leftrightarrow b$ *but* $f(a) \xrightarrow{c} a$ *while* $f(b) \stackrel{c}{\not\rightarrow} a$ *(since, for example,* $b \stackrel{a}{\not\rightarrow} b$*).*

The fourth syntactic constraint states that targets of negative premises should be universally quantified. The following counter-example witnesses that this constraint cannot be dropped.

Example 3.11 *The following deduction rules satisfy the constraints of the* UNTyft *format apart from constraint (iv) in that the negative premise* $x \stackrel{a}{\not\rightarrow} y$ *has an existentially quantified variable* y *as its target.*

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

$$\exists_{y,y',y''} \frac{x \stackrel{a}{\not\rightarrow} y \quad y \xrightarrow{b} y' \quad x \xrightarrow{a} y''}{f(x) \xrightarrow{a} c}$$

The above TSS is complete and its associated transition relation is

$$\{a \xrightarrow{a} c, a \xrightarrow{a} c', b \xrightarrow{a} c, c \xrightarrow{b} c, c' \xrightarrow{b} c, f(b) \xrightarrow{a} c\}.$$

Hence, we observe that $a \leftrightarrow b$ *but it does not hold that* $f(a) \leftrightarrow f(b)$*. Thus, bisimilarity is not a congruence.*

The last constraint on the UNTyft format concerns the variables in $\widetilde{z_0}$ when they appear as a target of a premise. Namely, such variables should not depend on variables in $\widetilde{z_1}$ or $\widetilde{z_2}$ (in the sense of Definition 3.3). The first counter-example given below shows that a direct dependency on variables in $\widetilde{z_1}$ can be damaging. The second counter-example shows the same for a direct dependency on variables in $\widetilde{z_2}$. Both counter-examples can be easily adapted for indirect dependencies.

Example 3.12 *The following deduction rules satisfy the constraints of the* UNTyft *format apart from constraint (v) in that variable* z_0 *depends on a universally bound variable* z_1 *in deduction rule* (**g**)*.*

$$(\mathbf{a})\frac{}{a \xrightarrow{a} b} \qquad (\mathbf{b})\frac{}{b \xrightarrow{a} a} \qquad (\mathbf{g})\exists_{z_0}\forall_{z_1}\frac{f(z_1, x) \xrightarrow{b} z_0}{g(x) \xrightarrow{c} c}$$

$$(\mathbf{f0})\forall_{y_0}\exists_{y_1}\frac{x_0 \stackrel{a}{\not\rightarrow} y_0 \quad x_1 \xrightarrow{a} y_1}{f(x_0, x_1) \xrightarrow{b} y_1} \qquad (\mathbf{f1})\exists_{y_0,y_1}\frac{x_0 \xrightarrow{a} y_0 \quad x_1 \xrightarrow{a} y_1}{f(x_0, x_1) \xrightarrow{b} a}$$

First of all note that $a \leftrightarrow b$ because they only afford a-transitions to each other. Furthermore, from (**f0**)*, it follows that for all $p \notin \{a, b\}$, we have $f(p, a) \xrightarrow{b} b$ and $f(p, b) \xrightarrow{b} a$. Thirdly, from* (**f1**)*, we can deduce that $f(a, b) \xrightarrow{b} a$ and $f(b, b) \xrightarrow{b} a$. Thus, we conclude that for all $p \in \mathbb{C}$, $f(p, b) \xrightarrow{b} a$. Hence, we have that $g(b) \xrightarrow{c} c$.*

It does not hold that for all $p \in \mathbb{C}$, $f(p, a) \xrightarrow{b} p'$ for any $p' \in \mathbb{C}$; the only possible candidates for such p' are a and b both of which fail (above-mentioned transitions to a cannot be derived from (**f0**) *and transitions to b cannot be derived from* (**f1**)*). Hence, we have that $g(a) \stackrel{c}{\nrightarrow}$ which shows that the congruence result is ruined.*

Example 3.13 *The following TSS is a modified version of the one specified in Example 3.12. The deduction rules satisfy the constraints of the UNTyft/UNTyxt format apart from constraint (v) in that variable z_0 depends on an existentially bound variable z_2.*

$$(\mathbf{a})\frac{}{a \xrightarrow{a} b} \qquad (\mathbf{b})\frac{}{b \xrightarrow{a} a} \qquad (\mathbf{x})\frac{}{x \xrightarrow{d} x}$$

$$(\mathbf{f0})\forall_{y_0}\exists_{y_1}\frac{x_0 \stackrel{a}{\nrightarrow} y_0 \quad x_1 \xrightarrow{a} y_1}{f(x_0, x_1) \xrightarrow{b} y_1} \qquad (\mathbf{f1})\exists_{y_0,y_1}\frac{x_0 \xrightarrow{a} y_0 \quad x_1 \xrightarrow{a} y_1}{f(x_0, x_1) \xrightarrow{b} a}$$

$$(\mathbf{g})\exists_{z_0}\forall_{z_1}\exists_{z_2}\frac{z_1 \xrightarrow{d} z_2 \quad f(z_2, x) \xrightarrow{b} z_0}{g(x) \xrightarrow{c} c}$$

The transition relation induced by the above TSS is the same as the one in Example 3.12 except for that it also includes a d-self-loop on all closed terms. Thus, $a \leftrightarrow b$ and it does not hold that $g(a) \leftrightarrow g(b)$.

4 Conclusions

Results. We extended the syntax and the semantics of SOS specifications with one level of universal quantification, explicit notions of existential quantification (before and after the universal quantifier), conjunction and disjunction. We proposed a rule format with the above-mentioned features that guarantees the induced bisimilarity to be a congruence.

Future Work. From a theoretical viewpoint, a much more challenging goal is to introduce a framework supporting the full first-order logic. We plan to investigate the possibility of relaxing the well-foundedness assumption in our congruence meta-results. Expressiveness of the UNTyft rule format with respect to the NTyft and the Tyft format is another topic for our future research. Our main point of inspiration for the introduction of universal quantification originated from our study of ordered SOS [13,8]. There, we observed that in order to translate general Tyft rules with ordering into the NTyft format, we need some extra expressive power, possibly modeled by universal quantification over variables. Otherwise, the ordered version of the Tyft format is strictly more expressive that the NTyft format and a direct translation (involving no auxiliary operators) is shown to be impossible in [9]. It remains thus to show that the UNTyft format indeed gives us sufficient expressive power to remove ordering from ordered Tyft rules.

Acknowledgments. Constructive discussions with Iain Phillips and Irek Ulidowski led to research questions which form the subject matter of this paper. Comments of the anonymous reviewers of the SOS workshop helped us improve the paper and are thus acknowledged.

References

[1] Aceto, L., W. J. Fokkink and C. Verhoef, *Structural operational semantics*, in: J. A. Bergstra, A. Ponse and S. A. Smolka, editors, *Handbook of Process Algebra, Chapter 3*, pp. 197–292.

[2] Aceto, L. and M. Hennessy, *Termination, deadlock, and divergence*, Journal of the ACM (JACM) **39** (1992), pp. 147–187.

[3] Baeten, J. C. M. and J. A. Bergstra, *Processen en procesexpressies*, Informatie **30** (1988), pp. 177–248, (In Dutch).

[4] Bol, R. and J. F. Groote, *The meaning of negative premises in transition system specifications*, Journal of the ACM (JACM) **43** (1996), pp. 863–914.

[5] van Glabbeek, R. J., *The meaning of negative premises in transition system specifications II*, Journal of Logic and Algebraic Programming (JLAP) **60-61** (2004), pp. 229–258.

[6] Groote, J. F., *Transition system specifications with negative premises*, Theoretical Computer Science (TCS) **118** (1993), pp. 263–299.

[7] Groote, J. F. and F. W. Vaandrager, *Structured operational semantics and bisimulation as a congruence*, Information and Computation (I&C) **100** (1992), pp. 202–260.

[8] Mousavi, M., I. C. C. Phillips, M. A. Reniers and I. Ulidowski, *The meaning of ordered SOS*, in: *Proceedings of the 26th Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'06)*, Lecture Notes in Computer Science **4337** (2006), pp. 334–345.

[9] Mousavi, M., I. C. C. Phillips, M. A. Reniers and I. Ulidowski, *Semantics and expressiveness of ordered SOS*, Technical Report CSR-07-07, Department of Computer Science, Eindhoven University of Technology, Eindhoven, The Netherlands (2007).

[10] Mousavi, M., M. A. Reniers and J. F. Groote, *SOS formats and meta-theory: 20 years after*, Theoretical Computer Science (TCS) **373** (2007), pp. 238–272.

[11] Mousavi, M., M. Sirjani and F. Arbab, *Formal semantics and analysis of component connectors in Reo*, in: *Proceedings of the 4th International Workshop on the Foundations of Coordination Languages and Software Architectures (FOCLASA'05)*, Electronic Notes in Theoretical Computer Science **154**, 2006, pp. 83–99.

[12] Park, D. M., *Concurrency and automata on infinite sequences*, in: *Proceedings of the 5th GI Conference*, Lecture Notes in Computer Science **104**, pp. 167–183.

[13] Phillips, I. C. C. and I. Ulidowski, *Ordered SOS rules and process languages for branching and eager bisimulations*, Information and Computation (I&C) **178** (2002), pp. 180–213.

[14] Plotkin, G. D., *The origins of structural operational semantics*, Journal of Logic and Algebraic Programming (JLAP) **60** (2004), pp. 3–15.

[15] van de Pol, J., *Operational semantics of rewriting with priorities*, Theoretical Computer Science (TCS) **200** (1998), pp. 289–312.

[16] Przymusinski, T. C., *The well-founded semantics coincides with the three-valued stable semantics*, Fundamenta Informaticae **13** (1990), pp. 445–463.

[17] Verhoef, C., *A congruence theorem for structured operational semantics with predicates and negative premises*, Nordic Journal of Computing **2** (1995), pp. 274–302.