

Structural Operational Semantics with First-Order Logic

Muck van Weerdenburg and Michel Reniers

Prose October 25, 2007

/department of mathematics and computer science



Outline

Introduction

Traditional Structural Operational Semantics Syntax and Semantics Well-Definedness Congruence Format

Structural Operational Semantics with First-Order Logic Syntax and Semantics Well-Definedness Congruence Format

Summary



Introduction - Structural Operational Semantics (SOS)

SOS allows for intuitive definition of operational semantics.

Operational semantics typically in terms of transition systems.

Popular for giving semantics to

- programming languages,
- process algebra,
- Petri nets,
- etc.



Introduction - Structural Operational Semantics (SOS)

Semantics is defined with rules.

"If P, then c": -

Set *P* of premises consists of positive and negative statements. $(x \xrightarrow{a} x' \text{ and } x \xrightarrow{a})$

Conclusion c is a (positive) statement.



Introduction - SOS formats

There exist (syntactic) formats.

- ntyft/ntyxt
- PANTH
- RBB-Cool
- etc.

These formats guarantee certain properties.

(Typically congruence of strong bisimilarity.)



Introduction - Example SOS

$$\frac{x \xrightarrow{a} x'}{a.x \xrightarrow{a} x} \qquad \frac{x \xrightarrow{a} x'}{x + y \xrightarrow{a} x'} \qquad \frac{x \xrightarrow{a} x'}{y + x \xrightarrow{a} x'} \qquad \frac{x \xrightarrow{a}}{x \sqrt{x}}$$

Rules are in ntyft/ntyxt format.

Strong bisimilarity is a congruence.



A predicate "is in a deadlock":

"We say that a process is in a deadlock [...] if it cannot do any action. That is, if p is such a process, we have that $p \xrightarrow{a} q$ and $p \xrightarrow{a} \sqrt{}$ for every a, q; [...]."

J.C.M. Baeten and J.A. Bergstra. Processen en procesexpressies. *Informatie*, 30(3):214–222, 1988.



A weak termination predicate $\sqrt{}$:

$$p \checkmark \Leftrightarrow \begin{cases} (i) & p \xrightarrow{\tau} \text{ and } p \checkmark, \text{ or} \\ (ii) & p \xrightarrow{\tau} \text{ and, for each } q, p \xrightarrow{\tau} q \text{ implies } q \checkmark \end{cases}$$

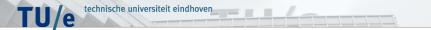
Luca Aceto and Matthew Hennessy. Termination, deadlock, and divergence. *Journal of the ACM*, 39(1):147–187, 1992.



A semantically divergence predicate \Downarrow :

 $p \downarrow$ and (for each $q, p \xrightarrow{\tau} q$ implies $q \Downarrow$) imply $p \Downarrow$

Luca Aceto and Matthew Hennessy. Termination, deadlock, and divergence. *Journal of the ACM*, 39(1):147–187, 1992.



M.R. Mousavi and M. Reniers gave a solution.

• Syntax is quite restricted:

$$\exists_{\overrightarrow{x}} \forall_{\overrightarrow{y}} \exists_{\overrightarrow{z}} \frac{\dots \wedge \dots \vee \dots}{t \stackrel{a}{\to} u}$$

- Unintuitive semantics (via translation to traditional rules).
- Certain other peculiarities.

M.R. Mousavi and M. Reniers. A Congruence Rule Format with Universal Quantification. *Proceedings of SOS'07*, (to appear).



Outline

Introduction

Traditional Structural Operational Semantics Syntax and Semantics Well-Definedness Congruence Format

Structural Operational Semantics with First-Order Logic Syntax and Semantics Well-Definedness Congruence Format

Summary



TU/e

technische universiteit eindhoven

Variables x, y, \ldots Functions $f, g, 0, 1, \delta, +, \ldots$

$$T ::= x \mid f(T, \ldots, T)$$

Predicates $P, Q, \stackrel{a}{\rightarrow}, \sqrt{, \ldots}$

atom ::= $P(T, ..., T) | \neg P(T) | \neg P(T, ..) |$?? Special negation $t \xrightarrow{a}$ ("there is no u such that $t \xrightarrow{a} u$ ")

Rule
$$\frac{S}{a}$$
 (where S is a set of atoms, a a positive atom



Derivability of an atom a given set of assumptions A.

$$\vdash -$$

a

if:

- $a \in A$, or
- there is are rule $\frac{S}{b}$ and substitution σ with • $a = b\sigma$ and • for all $c \in S \vdash \frac{A}{2\pi}$



$$\vdash \frac{\emptyset}{a.0 + 0 \stackrel{a}{\rightarrow} 0}$$

$$\Leftarrow \quad rule \frac{x \stackrel{a}{\rightarrow} x'}{x + y \stackrel{a}{\rightarrow} x'} \text{ and } \sigma = \{x \mapsto a.0, x' \mapsto 0, y \mapsto 0\}$$

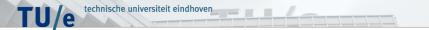
$$\vdash \frac{\emptyset}{a.0 \stackrel{a}{\rightarrow} 0}$$

$$\Leftarrow \quad rule \frac{x}{a.x \stackrel{a}{\rightarrow} x} \text{ and } \sigma = \{x \mapsto 0\}$$
true



 $\vdash \frac{\{0 \xrightarrow{a}\}}{0}$ rule $\frac{x \stackrel{a}{\leftrightarrow}}{x \sqrt{}}$ and $\sigma = \{x \mapsto 0\}$ \Leftarrow $\vdash \frac{\{0 \stackrel{a}{\not\rightarrow}\}}{0 \stackrel{a}{\not\rightarrow}}$ $\Leftarrow \qquad 0 \stackrel{a}{\not\rightarrow} \in \{0 \stackrel{a}{\not\rightarrow}\}$

true



Negative statements can cause problems:

 $\frac{\neg P}{P}$

We consider three-valued models $\langle C, U \rangle$.

C is "Certainly true", U is "Unknown". $((C \cup U)^{-1}$ is "certainly false.")

Above rule as three-valued model: $\langle \emptyset, \{P\} \rangle$



We write $A \vDash a$ if $a \in A$ and $A \vDash t \nleftrightarrow$ if there is no u with $t \to u \in A$. We also rewrite $A \vDash S$ if $A \vDash s$ for each $s \in S$.

 $\langle {\it C}, {\it U} \rangle$ is a three-valued stable model if

•
$$a \in C$$
 if, and only if, $\vdash \frac{N}{a}$ for some N with $C \cup U \models N$
• $a \in C \cup U$ if, and only if, $\vdash \frac{N}{a}$ for some N with $C \models N$
V is a set of negative atoms here.)

Interested in (information-)least three-valued stable model.

(|



Generally we are only interested in two-valued models.

That is, models $\langle C, U \rangle$ where $U = \emptyset$.

How to determine that a three-valued model is actually two-valued?



A stratification is a function f of positive atoms to a set S if

- S is well-ordered
 (i.e. there is no infinite sequence s > s' > ...)
- for all rules $\frac{S}{a}$ and substitutions σ we have that
 - $b \in S$ implies $f(b\sigma) \leq f(a\sigma)$
 - $t \stackrel{a}{\nrightarrow} \in S$ implies $f(t\sigma \stackrel{a}{\rightarrow} u) < f(a\sigma)$ for all u

If there is a stratification, then the model is two-valued.



$$\frac{x \xrightarrow{a} x'}{a.x \xrightarrow{a} x} \qquad \frac{x \xrightarrow{a} x'}{x + y \xrightarrow{a} x'} \qquad \frac{x \xrightarrow{a} x'}{y + x \xrightarrow{a} x'} \qquad \frac{x \xrightarrow{a}}{x \sqrt{x}}$$

$$f(t \xrightarrow{a} u) = 0 \qquad f(t \sqrt{}) = 1$$



 $\frac{\neg P}{P}$

f(P) < f(P)

/department of mathematics and computer science



$$- \frac{\neg P}{-}$$

P P

f(P) < f(P)

Least three-valued stable model is $\langle \{P\}, \emptyset \rangle$!

/department of mathematics and computer science



Why \leq for positives and < for negatives?

Essence is limiting the number of steps in three-valued model definition:

1.
$$a \in C$$
 if $\vdash \frac{N}{a}$ and $C \cup U \models N$.
2. for $\neg b \in N$, $b \in C \cup U$ if $\vdash \frac{N'}{b}$ and $C \models N'$
3. for $\neg c \in C$, $c \in C$ if ...
4. ...

The \leq for positives is to get f(b) < f(a) for $b \in N$ of $\vdash \frac{N}{a}$.



Traditional SOS - Congruence Format

If all rules are of the form:

$$\frac{\{t_i \xrightarrow{a} y_i : i \in I\} \cup \{t_j \xrightarrow{a} : j \in J\}}{f(x_1, \dots, x_n) \xrightarrow{a} t},$$

with all x_1, \ldots, x_n and y_i $(i \in I)$ distinct,

then strong bisimilarity is a congruence for all f. (i.e. if $p_i \leftrightarrow q_i$ for $1 \le i \le n$, then $f(p_1, \ldots, p_n) \leftrightarrow f(q_1, \ldots, q_n)$)



Outline

Introduction

Traditional Structural Operational Semantics Syntax and Semantics Well-Definedness Congruence Format

Structural Operational Semantics with First-Order Logic Syntax and Semantics Well-Definedness Congruence Format

Summary



Structural Operational Semantics with First-Order Logic

We want quantifications, implications etc.

That is, we want first-order logic formulae.

We also want it to be an extension of traditional SOS.

Finally, we want to lever traditional notions to this setting.



Infinitary First-Order Kleene Logic - Syntax

We want first-order logic in premises:

$$\varphi ::= a \mid \neg \varphi \mid \bigwedge \{\varphi, \ldots\} \mid \forall_x \varphi$$

Other operators are considered sugar:

true =
$$\bigwedge \emptyset$$
, $x \lor y = \neg \bigwedge \{\neg x, \neg y\}$, etc.

Literals atoms or negation of atoms. (I.e. no more $t \stackrel{a}{\nrightarrow}$.)



Set A makes φ true $(A \vDash \varphi)$:

TU/e

technische universiteit eindhoven

$$\begin{array}{lll} A \vDash a & \text{iff} & a \in A \\ A \vDash \neg \psi & \text{iff} & A \nvDash \psi \\ A \vDash \bigwedge \Psi & \text{iff} & \text{for all } \psi \in \Psi, A \vDash \psi \\ A \vDash \forall_{\mathsf{x}} \psi & \text{iff} & \text{for all term } t, A \vDash \psi[t/\mathsf{x}] \end{array}$$

Set A makes φ false $(A \not\models \varphi)$:

$$\begin{array}{lll} A \not\vDash a & \text{iff} & \neg a \in A \\ A \not\vDash \neg \psi & \text{iff} & A \vDash \psi \\ A \not\vDash \bigwedge \Psi & \text{iff} & \text{there is a } \psi \in \Psi \text{ with } A \not\vDash \psi \\ A \not\vDash \bigvee_x \psi & \text{iff} & \text{there is a } t \text{ with } A \not\vDash \psi[t/x] \end{array}$$



 φ -

/department of mathematics and computer science



A predicate "is in a deadlock":

"We say that a process is in a deadlock [...] if it cannot do any action. That is, if p is such a process, we have that $p \xrightarrow{a} q$ and $p \xrightarrow{a} \sqrt{}$ for every a, q; [...]."

$$\frac{\bigwedge_{a\in A}(\forall_y(\neg x \xrightarrow{a} y) \land \neg x \xrightarrow{a} \sqrt{)}}{\delta(x)}$$



A weak termination predicate $\sqrt{}$:

$$p \checkmark \Leftrightarrow \begin{cases} \text{(i)} & p \xrightarrow{\tau} \text{ and } p \checkmark, \text{ or} \\ \text{(ii)} & p \xrightarrow{\tau} \text{ and, for each } q, p \xrightarrow{\tau} q \text{ implies } q \checkmark \checkmark. \end{cases}$$
$$\frac{\forall_y (\neg x \xrightarrow{\tau} y) \land x \checkmark}{x \checkmark} \qquad \frac{\exists_y (x \xrightarrow{\tau} y) \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \checkmark)}{x \checkmark}$$



A semantically divergence predicate \Downarrow :

 $p \downarrow$ and (for each $q, p \xrightarrow{\tau} q$ implies $q \Downarrow$) imply $p \Downarrow$

$$\frac{x\downarrow\wedge\forall_y(x\stackrel{\tau}{\rightarrow}y\Rightarrow y\Downarrow)}{x\Downarrow}$$



Mousavi & Reniers have to write

$$\forall_{y} \frac{\bigwedge_{a \in \mathcal{A}} (x \xrightarrow{a} y \land x \xrightarrow{a} \sqrt{)}}{\delta(x)}$$

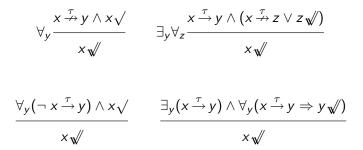
for

$$\frac{\bigwedge_{a\in\mathcal{A}}(\forall_y(\neg x \xrightarrow{a} y) \land \neg x \xrightarrow{a} \sqrt{)}}{\delta(x)}$$



for

Mousavi & Reniers have to write





Mousavi & Reniers have to write

$$\forall_{y} \frac{x \downarrow \land (x \xrightarrow{\tau} y \lor y \Downarrow)}{x \Downarrow}$$

for

$$\frac{x \downarrow \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \Downarrow)}{x \Downarrow}$$



FOL-SOS - Semantics

Derivability of an literal I given set of assumptions A.

 $A \vdash I$

if:

- $I \in A$, or
- there is are rule $\frac{\varphi}{2}$, set S and substitution σ with
 - $I = a\sigma$ and
 - $S \vDash \varphi \sigma$ and
 - for all $b \in S$, $A \vdash b$



 $\{\neg P\} \vdash Q$ $\Leftarrow \quad rule \ \frac{\neg P \land \exists_x R(x)}{Q} \text{ and } S = \{\neg P, R(c)\}$ $\{\neg P\} \vdash \neg P \text{ and } \{\neg P\} \vdash R(c)$ $\Leftarrow \quad \neg P \in \{\neg P\} \text{ and } rule \ \frac{\mathrm{true}}{R(c)} \text{ and } S = \emptyset$

true

/department of mathematics and computer science



We write rewrite $A \vDash L$ if $A \vDash I$ for each $I \in L$.

 $\langle {\it C}, {\it U} \rangle$ is a three-valued stable model if

- $a \in C$ if, and only if, $N \vdash a$ for some N with $C \cup U \vDash N$
- $a \in C \cup U$ if, and only if, $N \vdash a$ for some N with $C \vDash N$

(*N* is a set of negative literals here.)



Trivial translation of traditional to FOL-SOS:

S		$\bigwedge S'$
_	\mapsto	
а		а

S' is S with all $t \stackrel{a}{\nrightarrow}$ replaced by $\forall_x (\neg t \stackrel{a}{\rightarrow} x)$



From FOL-SOS to traditional SOS is possible for well-defined specifications.

Trivially:
$$\langle C, \emptyset \rangle$$
 gives $\{-: a \in C\}$.

When not well-defined:

$$\frac{}{a \to a} \quad \frac{}{b \to b} \quad \frac{}{a \to b} \quad \frac{}{b \to a}$$

technische universiteit eindhoven

FOL-SOS - Semantics

TU/e

Mousavi & Reniers:

For each deduction rule r of the following form,

$$\exists_{\widetilde{z_0}} \forall_{\widetilde{z_1}} \exists_{\widetilde{z_2}} \frac{\bigvee_{i \in I} \bigwedge_{j \in J} \phi_{ij}}{t \stackrel{I}{\to} t'}$$

$$sk(r) \text{ is } sk(r, \sigma_0, \sigma_{10}, \dots, \sigma_{20}, \dots, i_0, \dots \mid i_j) \text{ for each substitution}$$

$$\sigma_0 : \widetilde{z_0} \to \mathbb{C}, \text{ substitutions } \sigma_{10}, \sigma_{11}, \dots : \widetilde{z_1} \to \mathbb{C} \text{ such that for each}$$

$$z \in \widetilde{z_1}, \{\sigma_{10}(z), \sigma_{11}(z), \dots\} = \mathbb{C}, \text{ substitutions}$$

$$\sigma_{20}, \sigma_{21}, \dots : \widetilde{z_2} \to \mathbb{C}, \text{ indices } i_0, i_1, \dots \in I \text{ and each}$$

$$i_j \in \{i_0, i_1, \dots\} \text{ which is defined as follows.}$$

$$\frac{(\bigwedge_{j \in J} \sigma_0 \cdot \sigma_{10} \cdot \sigma_{20} \phi_{i_0j}) \land (\bigwedge_{j \in J} \sigma_0 \cdot \sigma_{11} \cdot \sigma_{21} \phi_{i_1j}) \land \dots}{\sigma_0 \cdot \sigma_{1i_i} \cdot \sigma_{2i_i} (t \stackrel{I}{\to} t')}$$



How to define stratifications w.r.t. formulae?

Looking at semantics: every *S* such that $S \vDash \varphi$.

Such sets S take role of premises in derivations.

But how to obtain such sets?



Easier: use set of literals that occur in φ :



A stratification is a function f of positive atoms to a set S if

- S is well-ordered
 (i.e. there is no infinite sequence s > s' > ...)
- for all rules $\frac{\varphi}{2}$ and substitutions σ we have that
 - $b \in \operatorname{Lit}(\varphi)$ implies $f(b\sigma) \leq f(a\sigma)$
 - $\neg b \in \operatorname{Lit}(\varphi)$ implies $f(b\sigma) < f(a\sigma)$

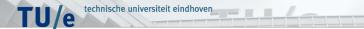
If there is a stratification, then the model is two-valued.



$$\frac{x \downarrow \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \Downarrow)}{x \Downarrow}$$

$$\operatorname{Lit}(x \downarrow \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \Downarrow)) = \{x \downarrow\} \cup \bigcup_t \{\neg x \xrightarrow{\tau} t, t \Downarrow\}$$

$$f(t\downarrow) = 0$$
 $f(t \stackrel{\tau}{\rightarrow} u) = 0$ $f(t\Downarrow) = 1$



$$\frac{\varphi}{f(x_1,\ldots,x_n)\overset{a}{\to}t},$$

- The right-hand sides of literals in φ are distinct variables different from x₁,..., x_n;
- 2. the right-hand sides of positive literals in φ are existentially bound;
- 3. the right-hand sides of negative literals in φ are universally bound;
- 4. the right-hand side variable of positive literals are bound inside the scope of the variables of the left-hand side of that literal.



$$\frac{\varphi}{f(x_1,\ldots,x_n)\overset{a}{\to}t},$$

The right-hand sides of literals in φ are distinct variables different from x_1, \ldots, x_n .

$$\begin{array}{l} x_1 \stackrel{a}{\rightarrow} y \wedge y \stackrel{b}{\rightarrow} z \\ x_1 \stackrel{a}{\rightarrow} x_2 \\ \exists_{x_3}(x_1 \stackrel{a}{\rightarrow} x_3) \vee \forall_{x_3}(x_2 \stackrel{a}{\rightarrow} x_3) \\ \exists_y(x_1 \stackrel{a}{\rightarrow} y \vee \forall_y(x_2 \stackrel{a}{\rightarrow} y)) \\ \exists_y(x_1 \stackrel{a}{\rightarrow} y \vee x_2 \stackrel{a}{\rightarrow} y) \end{array}$$



 $\operatorname{dv}(\varphi) \land \operatorname{ubrhs}(\varphi) \cap \{x_1, \ldots, x_n\} = \emptyset$



$$\frac{\varphi}{f(x_1,\ldots x_n) \xrightarrow{a} t},$$

The right-hand sides of positive literals in φ are existentially bound.

$$\begin{array}{l} x_1 \stackrel{a}{\rightarrow} y \\ \exists_y (x_1 \stackrel{a}{\rightarrow} y) \\ \forall_y (x_1 \stackrel{a}{\rightarrow} y) \\ \forall_y (x_2 \stackrel{a}{\rightarrow} y \Rightarrow x_1 \stackrel{a}{\rightarrow} z) \\ \forall_y (x_2 \stackrel{a}{\rightarrow} z \Rightarrow x_1 \stackrel{a}{\rightarrow} y) \end{array}$$



 $\operatorname{ext}_{\operatorname{FV}(\varphi)\setminus\{x_1,\ldots,x_n\}}(\varphi)$

$$\begin{array}{lll} \operatorname{ext}_{\mathcal{S}}(t \to u) &=& u \in \mathcal{S} \\ \operatorname{ext}_{\mathcal{S}}(\neg \psi) &=& \overline{\operatorname{ext}}_{\mathcal{S}}(\psi) \\ \operatorname{ext}_{\mathcal{S}}(\bigwedge \Psi) &=& \forall_{\psi \in \Psi} \operatorname{ext}_{\mathcal{S}}(\psi) \\ \operatorname{ext}_{\mathcal{S}}(\forall_{x}\psi) &=& \operatorname{ext}_{\mathcal{S} \setminus \{x\}}(\psi) \end{array}$$



$$\frac{\varphi}{f(x_1,\ldots,x_n)\overset{a}{\to}t},$$

The right-hand sides of negative literals in φ are universally bound.

$$\neg x_{1} \xrightarrow{a} y$$

$$\forall_{y} (\neg x_{1} \xrightarrow{a} y)$$

$$\forall_{y} (x_{2} \xrightarrow{a} y \Rightarrow x_{1} \xrightarrow{a} z)$$

$$\forall_{y} (x_{2} \xrightarrow{a} z \Rightarrow x_{1} \xrightarrow{a} y)$$





/department of mathematics and computer science



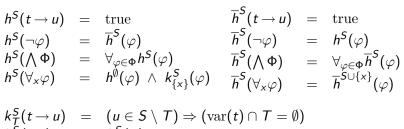
$$\frac{\varphi}{f(x_1,\ldots,x_n)\overset{a}{\to}t},$$

The right-hand side variable of positive literals are bound inside the scope of the variables of the left-hand side of that literal.

$$\exists_{z} (\forall_{y} (y \xrightarrow{a} z)) \forall_{y} (\exists_{z} (y \xrightarrow{a} z)) \exists_{y} (\exists_{z} (z \xrightarrow{a} y)) \exists_{y} (\forall_{w} (\exists_{z} (z \xrightarrow{a} y)))$$



 $h^{\mathrm{FV}(\varphi)\setminus\{x_1,\ldots,x_n\}}(\varphi)$



$$k_T^{S}(\neg \varphi) = k_T^{S}(\varphi)$$

$$k_T^{S}(\wedge \Phi) = \forall_{\varphi \in \Phi} k_T^{S}(\varphi)$$

$$k_T^{S}(\forall_x \varphi) = k_{T \cup \{x\}}^{S}(\varphi)$$



The examples all fit our congruence format.

$$\frac{\bigwedge_{a \in A} (\forall_y (\neg x \xrightarrow{a} y) \land \neg x \xrightarrow{a} \sqrt{)}}{\delta(x)}$$
$$\frac{\forall_y (\neg x \xrightarrow{\tau} y) \land x \sqrt{}}{x \sqrt{}} \qquad \frac{\exists_y (x \xrightarrow{\tau} y) \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \sqrt{})}{x \sqrt{}}$$
$$\frac{x \downarrow \land \forall_y (x \xrightarrow{\tau} y \Rightarrow y \Downarrow)}{x \Downarrow}$$



Traditional format is incorporated:

- 1. The right-hand sides of literals in φ are distinct variables different from x_1, \ldots, x_n ; \leftarrow traditional distinctness requirement
- 2. the right-hand sides of positive literals in φ are existentially bound; \leftarrow trivially
- the right-hand sides of negative literals in φ are universally bound; ⇐ only quantifications: ∀_x(¬ t → x)
- the right-hand side variable of positive literals are bound inside the scope of the variables of the left-hand side of that literal. ⇐ no (nested) quantifications



Mousavi & Reniers format (UNTyft/UnTyxt) is incorporated:

- 1. The right-hand sides of literals in φ are distinct variables different from x_1, \ldots, x_n ; \Leftarrow same
- 2. the right-hand sides of positive literals in φ are existentially bound; \Leftarrow same
- 3. the right-hand sides of negative literals in φ are universally bound; \Leftarrow same
- the right-hand side variable of positive literals are bound inside the scope of the variables of the left-hand side of that literal. ⇐ simpler variant due to ∃_x∀_y∃_z



Outline

Introduction

Traditional Structural Operational Semantics Syntax and Semantics Well-Definedness Congruence Format

Structural Operational Semantics with First-Order Logic Syntax and Semantics Well-Definedness Congruence Format

Summary



Summary

- Full first-order power in premises of rules
- Straightforward extension of semantics
- Conservative extension of traditional SOS (with sugar)
- Conservative extension of traditional congruence format
- Congruence format requirements easily calculable
- Suits known examples using quantifications



Thank you for your attention!