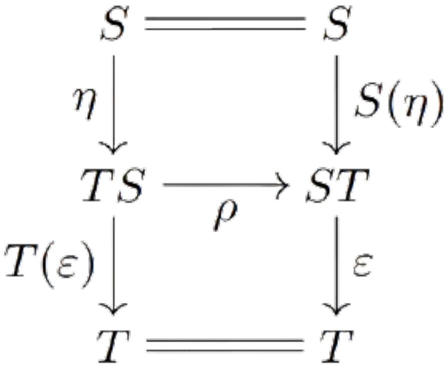
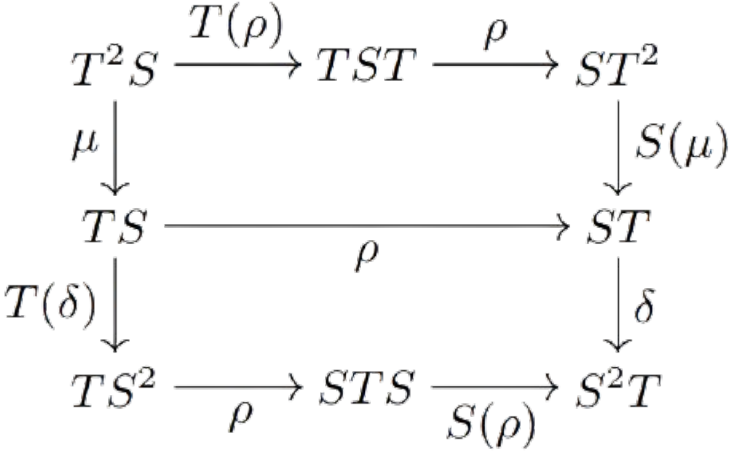
Stateful Structural Operational Semantics

*Sergey Goncharov, Stefan Milius, Lutz Schröder, **Stelios Tsampas**, Henning Urbat*

Let endofunctors Σ and B, modelling syntax and behavior

- GSOS laws are nat. transformations $\rho_X : \Sigma(X \times BX) \Rightarrow B\Sigma^\star X$
- More generally, any distributive law of a monad T over a comonad S, although these tend to be (co)freely generated





Any and all transitions of a composite term are determined by the transitions of its subterms

Σ-algebra

B-coalgebra

$$
\begin{array}{ccccc}
\Sigma X & \xrightarrow{\quad g \quad} & X & \xrightarrow{\quad h \quad} & BX \\
\downarrow{\scriptstyle \Sigma\langle 1, h\rangle} & & & & \uparrow{\scriptstyle Bg^{\#}} \\
\Sigma(X \times BX) & & \xrightarrow{\qquad \rho \qquad} & & B\Sigma^{*} X
\end{array}
$$

- 📜 There is the category ***ρ-bialg*** of ρ-bialgebras:
  - ☐ Objects of ***ρ-bialg*** are ρ-bialgebras
  - ☐ Morphisms are maps that are both algebra and coalgebra homomorphisms at the same time
- 📝 The initial Σ-algebra (A,α) extends to the initial ρ-bialgebra
- 📝 The final B-coalgebra (Z,z) extends to the final ρ-bialgebra

Corollary. *There is a unique bialgebra map beh : A → Z, mapping terms to their behaviors. Hence,* ***bisimilarity is a congruence*****.*

An example

$$\frac{P \xrightarrow{\delta} P'}{P + Q \xrightarrow{\delta} P'} \qquad \frac{Q \xrightarrow{\delta} Q'}{P + Q \xrightarrow{\delta} Q'}$$

$$BX = \mathcal{P}_f(\Delta_\tau \times X), \Sigma X = X \times X$$

$$\begin{array}{rcl} \rho \, : \, \Sigma(\mathrm{Id} \times B) & \Longrightarrow & B\Sigma^* \\ (x, X) + (y, Y) & \mapsto & X \cup Y \end{array}$$

A *While* language

$$\texttt{skip}\ \frac{}{s, \texttt{skip} \downarrow s}$$

$$\texttt{asn}\ \frac{}{s, (x := e) \downarrow s_{[x \leftarrow [e]_s]}}$$

$$\texttt{while1}\ \frac{[e]_s = 0}{s, \texttt{while}\ e\ p \downarrow s}$$

$$\texttt{while2}\ \frac{[e]_s \neq 0 \quad q = \texttt{while}\ e\ p}{s, q \rightarrow s, (p;\ q)}$$

$$\texttt{seq1}\ \frac{s, p \downarrow s'}{s, (p;\ q) \rightarrow s', q}$$

$$\texttt{seq2}\ \frac{s, p \rightarrow s', p'}{s, (p;\ q) \rightarrow s', (p';\ q)}$$

- Originally, Turi considered functor $(S \times (X+1))^S$ to model the behavior of imperative languages
- For instance, sequential composition had a component of the type $S \times (X \times (S \times (X+1))^S)^2 \to S \times (\Sigma^\star X + 1)$

$$s, (x, f), (y, g) \mapsto \begin{cases} (s', \ (x' \ ; y)) & \text{if } f(s) = (s', \ x') \\ (s', \ y) & \text{if } f(s) = (s', \checkmark). \end{cases}$$

- Originally, Turi considered functor $(S \times (X + 1))^S$ to model the behavior of imper… …es
- For instance, sequenti… …mponent of the type $S \times (X$ … $(\Sigma^\star X + 1)$

OVERKILL

$$s, (x, f), (y, \quad \ldots \quad (x' ; y)) \quad \text{if } f(s) = (s', x')$$
$$(s', y) \quad \text{if } f(s) = (s', \checkmark).$$

- A behavior of $(S \times (X+1))^S$ implies contexts are far more capable than what they really are. For a comparison:

$$\texttt{seq1}\ \frac{s, p \downarrow s'}{s, (p;\ q) \to s', q} \qquad \texttt{seq2}\ \frac{s, p \to s', p'}{s, (p;\ q) \to s', (p';\ q)}$$

$$\texttt{ob}\ \frac{s_1, p \to s_1', p_1' \qquad s_2, p \to s_2', p_2' \qquad s_3, p \to s_3', p_3'}{s, \lceil p \rceil \to s_4, p_3'}$$

Something simpler

$$\texttt{seq1}\ \frac{s, p \downarrow s'}{s, (p;\ q) \to s', q} \qquad \texttt{seq2}\ \frac{s, p \to s', p'}{s, (p;\ q) \to s', (p';\ q)}$$

$$S \times (X \times S \times (X+1))^2 \to (S \times \Sigma^\star X + 1)$$

$$(s, (x, s', *), (y, \_, \_)) \mapsto (s', y)$$

$$(s, (x, s', x'), (y, \_, \_)) \mapsto (s', (x';\ y))$$

Definition 3.7. A *stateful SOS law* is a natural transformation

$$\delta_X \colon S \times \Sigma(X \times S \times (X + 1)) \to S \times (\Sigma^\star X + 1) \qquad (X \in \mathbf{Set}).$$

Stateful SOS laws are in a bijective correspondence with *stateful SOS specifications*, i.e. systems whose rules look like (for $W \subseteq \{1, \ldots, n\}$)

$$\frac{(s, x_j \to s'_j, y_j)_{j \in W} \qquad (s, x_j \downarrow s'_j)_{j \in \{1,\ldots,n\} \smallsetminus W}}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t}$$

Compositionality in imperative languages

- Thing is, one can still have wonky, “concurrency” rules like

$$\frac{s, p \to s', p'}{s, p \triangleleft q \to s', q \triangleleft p'} \qquad \frac{s, p \downarrow s'}{s, p \triangleleft q \to s', q}$$

- Let $BX = S \times (X + 1)$ and $TX = (BX)^S$
- Stateful SOS specification exhibit compositionality in domain νT (the final coalgebra of T), which is typically too fine-grained

Compositionality in imperative languages

- Let $BX = S \times (X + 1)$ and $TX = (BX)^S$
- Standard choices for semantic domain are *trace semantics* $(\nu B)^S$ and *termination semantics* $(S + 1)^S$
- When is trace semantics compositional?
- When is termination semantics compositional?
- These questions are outright undecidable and compositionality is challenging to prove in a per-case basis

How about restricting the rule format for compositionality?

Streamlined stateful SOS

- Compositional trace semantics $(\nu B)^S$
- Receiving rules have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t} \quad \text{where } t = \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j] \text{ or } t = y_j;$$

- Compositional trace semantics $(\nu B)^S$
- Receiving rules have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t} \qquad \text{where } t = \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j] \text{ or } t = y_j;$$

$$\frac{s, p \to s', p'}{s, p \lhd q \to s', q \lhd p'} \qquad \frac{s, p \downarrow s'}{s, p \lhd q \to s', q}$$

Streamlined stateful SOS

- Compositional trace semantics $(\nu B)^S$
- Receiving rules have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t} \qquad \text{where } t = \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j] \text{ or } t = y_j;$$

$$\frac{s, p \to s', p'}{s, p \lhd q \to s', q \lhd p'} \qquad \frac{s, p \downarrow s'}{s, p \lhd q \to s', q}$$

Streamlined stateful SOS

- Compositional trace semantics $(\nu B)^S$
- Receiving rules have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t} \qquad \text{where } t = \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j] \text{ or } t = y_j;$$

$$\frac{s, p \to s', p' \qquad [\mathtt{i}]_s \neq 0 \qquad \neg P(s')}{s, (p;\ q) \to s', (p';\ q)}$$

Streamlined stateful SOS

- Compositional trace semantics $(\nu B)^S$
- Receiving rules have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', t} \qquad \text{where } t = \mathsf{f}(x_1, \ldots, x_n)[y_j / x_j] \text{ or } t = y_j;$$

$$\frac{s, p \to s', p' \qquad [\mathtt{i}]_s \neq 0 \qquad \neg P(s')}{s, (p;\ q) \to s', (p';\ q)}$$

- Compositional termination semantics $(S+1)^S$
- All rules with a progressing premiss have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j]}$$

- Compositional termination semantics $(S+1)^S$
- All rules with a progressing premiss have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j]}$$

$$\frac{s, p \to s', p'}{s, p \lhd q \to s', q \lhd p'} \qquad \frac{s, p \downarrow s'}{s, p \lhd q \to s', q}$$

Cool stateful SOS

- Compositional termination semantics $(S+1)^S$
- All rules with a progressing premiss have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j]}$$

$$\frac{s, p \to s', p'}{s, p \lhd q \to s', q \lhd p'} \qquad \frac{s, p \downarrow s'}{s, p \lhd q \to s', q}$$

Cool stateful SOS

- Compositional termination semantics $(S+1)^S$
- All rules with a progressing premiss have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j]}$$

$$\frac{s, p \to s', p' \qquad [\mathtt{i}]_s \neq 0 \qquad P(s')}{s, (p;\ q) \to s', q}$$

Cool stateful SOS

- Compositional termination semantics $(S+1)^S$
- All rules with a progressing premiss have to be as follows

$$\frac{s, x_j \to s', y_j}{s, \mathsf{f}(x_1, \ldots, x_n) \to s', \mathsf{f}(x_1, \ldots, x_n)[y_j/x_j]}$$

$$\frac{s, p \to s', p' \qquad [\mathtt{i}]_s \neq 0 \qquad P(s')}{s, (p;\ q) \to s', q}$$

Thank you

Full paper at https://arxiv.org/pdf/2202.10866.pdf