Basic operational preorders for algebraic effects in general, and for combined probability and nondeterminism in particular

Computer Science Logic 2018

Aliaume Lopez
Alex Simpson
September 7, 2018
Context
Three approaches to semantics

**Operational** describe evaluation steps

**Denotational** compositional mathematical model

**Axiomatics** axiomatise behaviour

Contextual preorder

1. Tied to operational semantics
2. $P_1 \sqsubseteq_{ctxt} P_2$ iff in any context $C$, the behaviour of $C[P_1]$ approximates the behaviour of $C[P_2]$. 
[Johann et al., 2010a]

**Why?** Operational semantics works *great* but needs to be adapted in each case.
[Johann et al., 2010a]

**Why?** Operational semantics works *great* but needs to be adapted in each case

**Objective?** Give a *generic* operational semantics for a large class of languages
[Johann et al., 2010a]

Why? Operational semantics works *great* but needs to be adapted in each case.

Objective? Give a *generic* operational semantics for a large class of languages.

How?
1. Parametrize with a signature of effect operations $\Sigma$
2. Reduce a program to an *effect tree*
3. Define a $\preccurlyeq$ preorder on $\text{Trees}_{\text{Nat}}$
[Johann et al., 2010a]

Why? Operational semantics works *great* but needs to be adapted in each case.

Objective? Give a *generic* operational semantics for a large class of languages.

How?
1. Parametrize with a signature of effect operations $\Sigma$
2. Reduce a program to an effect tree
3. Define a $\preceq$ preorder on $\text{Trees}_{\text{Nat}}$

Result? Generic operational definition of contextual preorder.
Contextual preorders

Morris-style

Input: A preorder $\preceq$ for type Nat

Output:

\[ P_1 \sqsubseteq_{\text{ctxt}} P_2 \iff \forall C[-] \text{ context}, \|C[P_1]\| \preceq \|C[P_2]\| \quad (1) \]
Contextual preorders

Morris-style

**Input:** A preorder $\preceq$ for type Nat

**Output:**

\[ P_1 \sqsubseteq_{ctxt} P_2 \iff \forall C[-] context, |C[P_1]| \preceq |C[P_2]| \quad (1) \]

GOM

**Input:** A preorder $\preceq$ for type Nat

**Output:** A logical relation (!) on programs that characterises contextual preorder (Morris-Style)
Example of trees

Let $\Sigma = \{\text{pr}\}$ be a signature containing one binary effect construction.

Properties

Trees$_{\text{Nat}}$ is a DCPO and a continuous $\Sigma$-algebra
What are the conditions on $\preceq$ in GOM?

**Admissible** If $t_i \preceq t_i'$ and $(t_i)_i, (t_i')_i$ are an ascending chains then

$$\bigvee_i t_i \preceq \bigvee_i t_i'$$

(2)

Compatible with least upper bounds

**Compositional** If $t \preceq t'$ and $\rho \preceq \rho'$ (pointwise) then $t\rho \preceq t'\rho'$

Compositional reasoning is possible
**Contributions**

**General** Identify three different ways to produce well-behaved preorders
Contributions

**General** Identify three different ways to produce well-behaved preorders

**Specific** Examine how they apply to a specific signature

\[ \Sigma_{pr/nd} = \{\text{pr, or}\} \] (3)
Contributions

**General** Identify three different ways to produce well-behaved preorders

**Specific** Examine how they apply to a specific signature

\[ \Sigma_{pr/nd} = \{pr, or\} \] (3)

**Coincidence** Prove that the three ways of defining \( \preceq_{pr/nd} \) lead to the same contextual preorder
Well-behaved preorders
Methods for defining preorders

Following three common approaches to semantics

• From some operational construction \( \preceq_{\text{op}} \)
• From a denotation \( \langle \cdot \rangle \) \( \preceq_{\text{den}} \)
• From axiomatic definitions \( \preceq_{\text{ax}} \)
Randomised Algorithms with Scheduler

\[ \sum \text{ coin "pr", demon "or" } \]
\[ \preceq \text{ capture the behaviour ... and satisfies the requirements } \]

Example of program

\((1 \text{ pr } 2) \text{ or } 3\)
Operationally defined preorders
Compare Markov Decision Processes pointwise, where a point is a goal set $X \subseteq \text{Nat}$:

$$t \preceq_{\text{badOp}} t' \iff \forall X \subseteq \text{Nat}, \quad \inf_{\pi} \mathbb{E}^{\pi}(t \in X) \leq \inf_{\pi} \mathbb{E}^{\pi}(t' \in X)$$
The issue

1. The following trees are equated

   \[ \text{or} \quad \text{pr} \quad 3 \quad \sim_{\text{badOp}} \quad \text{or} \quad \text{pr} \quad \text{or} \quad \text{or} \quad 3 \]

2. If compositionality holds for \( \leq_{\text{badOp}} \) then

   \[ x \text{ or} (y \text{ pr} z) = (x \text{ or} y) \text{ pr} (x \text{ or} z) \]  \hspace{1cm} (4)

3. Which is does not hold for \( \simeq_{\text{badOp}} \) (easy substitution)

4. And should never hold [Mislove et al., 2004]
Compare Markov Decision Processes pointwise, where a point is a payoff function $h : \text{Nat} \rightarrow \overline{\mathbb{R}}_+$:

$$t \preceq_{\text{op}} t' \iff \forall h : \text{Nat} \rightarrow \overline{\mathbb{R}}_+, \quad \inf_{\pi} \mathbb{E}^\pi(h(t)) \leq \inf_{\pi} \mathbb{E}^\pi(h(t'))$$

**Proposition**
The preorder $\preceq_{\text{op}}$ is admissible and compositional

**Remark**
The proof requires some topological arguments...
Denotationally defined preorders
The idea

**Input**
1. Continuous $\Sigma$-algebra $D$
2. $[\cdot] : \mathbb{N}_\bot \rightarrow D$ continuous $\Sigma$-algebra homomorphism

**Output** The preorder $\preceq_{\text{den}}$

$$
\begin{align*}
\mathbb{N} & \xrightarrow{j} D \\
i & \downarrow i \\
\text{Trees}(\mathbb{N}) & \xrightarrow{[\cdot]} D
\end{align*}
$$

$$
t \preceq_{\text{den}} t' \iff [t] \leq_D [t'] \quad (5)
$$
Properties of \( \preceq_{\text{den}} \)

1. Automatically *admissible* (continuity)
2. Automatically *compatible* (\( \Sigma \)-algebra)
3. Not always *compositional*!
### Factorisation

The map \( j : \mathbb{N} \rightarrow D \) is said to have the *factorisation property* if, for every function \( f : \mathbb{N} \rightarrow D \), there exists a continuous homomorphism \( h_f : D \rightarrow D \) such that \( f = h_f \circ j \).

\[
\begin{array}{ccc}
\mathbb{N} & \xrightarrow{j} & D & \xrightarrow{h_f} & D \\
\downarrow{f} \quad & & & & \quad \downarrow{f}
\end{array}
\]

### Idea

We then have \([t\sigma] = h_\sigma([t])\) which is continuous in \( t \) with a fixed \( \sigma \).
Well behaved denotational preorder

**Proposition**
If $j: \mathbb{N} \to D$ has the factorisation property then the relation $\preceq_D$ is substitutive, hence it is an admissible compositional precongruence.

**In practice [Proposition 16]**
It is usually not necessary to prove the factorisation property directly. Instead it holds as a consequence of the continuous algebra $D$ and map $j : \text{Nat} \to D$ being derived from a suitable monad.
Applying to the running example

Using Kegel's results [Keimel and Plotkin, 2017]

\[ \mathcal{V}_{\leq 1} X \text{ } \omega \text{CPO of (discrete) subprobability distributions over } X. \]
\[ S\mathcal{V}_{\leq 1} X \text{ } \omega \text{CPO of nonempty Scott-compact convex upper-closed subsets of } \mathcal{V}_{\leq 1} X \text{ ordered by reverse inclusion } \supseteq. \]

\[
\text{or}(A, B) = \text{Conv}(A \cup B) \quad (6)
\]
\[
\text{pr}(A, B) = \left\{ \frac{1}{2} a + \frac{1}{2} b \mid a \in A, b \in B \right\} \quad (7)
\]
Axiomatically defined preorders
Theories

**Equation**  $e \leq e'$ with $e, e' \in \text{Trees}(\text{Vars})$

**Clause**  (Infinitary) Horn-Clause of equations

**Theory**  Set of Horn-Clauses
Theories

Equation  \( e \leq e' \) with \( e, e' \in \text{Trees}(\text{Vars}) \)

Clause  (Infinitary) Horn-Clause of equations

Theory  Set of Horn-Clauses

Axiomatically defined preorder

Definition  There exists a smallest admissible preorder \( \preceq_{\text{ax}} \) that models \( T \)

Property  \( \preceq_{\text{ax}} \) is compositional
Axioms for Pr and Nd

**Bot:** $\bot \leq x$
Axioms for Pr and Nd

Bot: \( \perp \leq x \)

Prob: \( x \text{ pr } x = x, \ x \text{ pr } y = y \text{ pr } x, \)
\( (x \text{ pr } y) \text{ pr } (z \text{ pr } w) = (x \text{ pr } z) \text{ pr } (y \text{ pr } w) \)

Appr: \( x \text{ pr } y \leq y \implies x \leq y \) (!)
Axioms for Pr and Nd

**Bot:** $\bot \leq x$

**Prob:**
- $x \text{pr} x = x$
- $x \text{pr} y = y \text{pr} x$,
- $(x \text{pr} y) \text{pr} (z \text{pr} w) = (x \text{pr} z) \text{pr} (y \text{pr} w)$

**Appr:** $x \text{pr} y \leq y \implies x \leq y$

**Nondet:**
- $x \text{or} x = x$
- $x \text{or} y = y \text{or} x$
- $x \text{or} (y \text{or} z) = (x \text{or} y) \text{or} z$

**Dem:** $x \text{or} y \geq x$
Axioms for Pr and Nd

Bot: \( \bot \leq x \)

Prob: \( x \text{pr} x = x, \ x \text{pr} y = y \text{pr} x, \)
\( (x \text{pr} y) \text{pr} (z \text{pr} w) = (x \text{pr} z) \text{pr} (y \text{pr} w) \)

Appr: \( x \text{pr} y \leq y \implies x \leq y \) (!)

Nondet: \( x \text{or} x = x, \ x \text{or} y = y \text{or} x, \ x \text{or} (y \text{or} z) = (x \text{or} y) \text{or} z \)

Dem: \( x \text{or} y \geq x \)

Dist: \( x \text{pr} (y \text{or} z) = (x \text{pr} y) \text{or} (x \text{pr} z) \) (!)
The coincidence theorem
Coincidence

For probability and non-determinism

\[ \preceq_{op} = \preceq_{den} = \preceq_{ax} \]

Proof sketch

1. Equality on trees without or nodes
2. Equality for trees with finite number of or nodes (!)
3. General equality using finite approximations and admissibility
Summary and limitations
Summary and limitations

What has been done

- Denotational and Axiomatic definitions of preorders
- Applied to a specific signature \( \Sigma = \{ \text{pr}, \text{or} \} \)

Limitations

- Some effects are not algebraic
- The preorder for *countable* non-determinism is not admissible

Thank You!
*ArXiv e-prints.*

*Mathematical Structures in Computer Science.*  
To appear.


