An intensionally fully-abstract sheaf model for $\pi$

Clovis Eberhart       Tom Hirschowitz       Thomas Seiller
CNRS and Université Savoie Mont Blanc  CNRS and Université Paris 7
Issues raised by standard operational semantics

Standard operational semantics

Execution traces = paths in labelled transition systems (LTSs).

As Castellan, Clairambault, and Winskel '15 argue:
Different interleaving of independent actions $\leadsto$ different paths.
- State explosion problem in verification.
- Loss of causality information $\leadsto$ difficult error diagnostics.
Causal models

Intended to restore causality information.

Petri nets ('81), CCS ('82)
Nielsen, Plotkin, Winskel

duplication & uniformity

output of private channels

Linear logic ('03 - '07)
Melliès

π-calculus ('12)
Grafa et al.

- Castellan, Clairambault, Winskel ('15): as Melliès + concurrent strategies.
- All three extensions: very hard!
A different approach to causal models

- First main result published at Calco '13: intensional full abstraction for CCS.
- Here, extended to the π-calculus.

**Construction of model**
- Same pattern as for CCS.
- Difficulty: need to restrict traces to subconfigurations.
- Dealt with using factorisation systems.

**Proof of intensional full abstraction**
- New proof method required.
- Actually simpler than for CCS.
An important architectural difference

Standard denotational semantics:
- a large `ambient' category: event structures, concurrent games;
- interpretation of terms/programs in this ambient category.

Here:
- For each considered calculus, a playground \(\approx\) a notion of trace.
- Intuition: a playground gives the `rules of the game'.
- Denotations are then innocent presheaves on traces.

Hopefully: paves the way for studying relations between calculi.
Traces

Very intensional notion of trace

- Configurations \( X, Y, \ldots \approx \) network topologies:
  - Agents.
  - Communication channels between them.
- Traces \( Y \rightarrow X \) describe each agent's actions leading from \( X \) to \( Y \).

(Where bits of \( Y \) come from in \( X \))
# Naive strategies

**Naive strategies: presheaves on traces**

- Each trace \(\mapsto\) possibly empty set of ways of accepting them.
- Cf. presheaf models (Joyal, Nielsen, Winskel '93).
- Deals at once with:
  - prefix-closedness,
  - permutation of independent actions,
  - channel renaming (cf. nominal sets).

**Problem:** too general

Agents may `communicate' without using the network.
To rectify the deficiency, restrict to

<table>
<thead>
<tr>
<th>Innocent strategies: sheaves on traces</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Accepting a trace should be `local'.</td>
</tr>
<tr>
<td>- I.e., determined only by each agent's `view' of the trace.</td>
</tr>
</tbody>
</table>

Each trace covered by its collection of views | Grothendieck topology
---|---
Ways of accepting trace $u$ | Sheaf condition
$\cong$ collections of ways of accepting $u$'s views
Configurations

- $\bullet$ $\approx$ agent.
- $\circ$ $\approx$ communication channel.
- Edges: agent knows channel.
- Now, traces:
  - Actions are not a mere binary relation (initial, final configuration).
  - Indeed, want to represent how one moves from initial to final configuration.
  - We use cospans: initial $\rightarrow$ stuff $\leftarrow$ final.
  - What stuff? A kind of higher-dimensional graph.
  - Formally: presheaves on a countable category $\mathbb{C}$, see paper.
Generators for actions: particular presheaves on $\mathbb{C}$

Output

Input

Synchronisation
Generators for actions: particular presheaves on $\mathbb{C}$

These presheaves vaguely look like actions.

How to
- add temporal information (initial/final),
- put generators in context,
- compose them to get traces with more than one action?
Temporal (initial/final) information through cospans

Cospan for the input action:

final configuration

initial configuration

drawn for conciseness as:
Inclusion into larger configurations

Definition
Interface of the cospan for a generator: channels shared between initial and final.

Intuition: glue $Z$ and initial configuration (resp. action, final) along $I$. 
Sequential composition of traces

- By composition in $\text{Cospan}(\widehat{C})$!
- Retains causality, not syntactic ordering.
- $\leadsto$ a category $\mathcal{T}_X$ of traces over $X$.

- Naive strategies over $X$: $\widehat{\mathcal{T}}_X = [\mathcal{T}_X^{\text{op}}, \text{sets}]$. 
Views and innocence

Strategies on a configuration $X = \text{sheaves on } \mathcal{I}_X \cong \text{presheaves on } \mathcal{V}_X$. 
The problem

- Everything works as in previous work on CCS.
- Except:

  Needed for the machinery to work

  A way of restricting traces over $X$ to any subconfiguration $Y \hookrightarrow X$. 
The basic idea

Given any cospan \((s, v)\) as on the right

we compute its restriction along \(Y \xrightarrow{h} X\) by:

1. factorising \(v \circ h\) as \(h' \circ v'\), where
   \(v'\) does as many actions as it can;
2. then taking the pullback of \(s\) and \(h'\).

What does it mean to `do as many actions as one can'? 

Factorisation system!
Generating cofibrations

Factorisation system generated from a set of so-called cofibrations.

Consider the set $\mathcal{Z}_0$ of inclusions $X \to A$
- of the initial configuration of a generator
- into the generator itself ($\in \widehat{C}$).
Horizontal maps

- Consider now maps $g$ right-orthogonal to $\mathcal{V}_0$, i.e., for all commuting squares

$$
\begin{array}{c}
\text{X} \\
\downarrow t \\
\text{A}
\end{array}
\xrightarrow{u}
\begin{array}{c}
\text{C} \\
\downarrow h \\
\text{D}
\end{array}
\xleftarrow{g}
\begin{array}{c}
\text{t} \\
\downarrow h \\
\text{D}
\end{array}
\xleftarrow{g}
\begin{array}{c}
\text{D}
\end{array}
$$

with $t \in \mathcal{V}_0$, there exists a unique filler $h$ making both triangles commute.

- Idea: $g$ may not add new actions from $C$.
- Indeed: any added action was already in $C$.

**Notation**

$t \perp g$, $\mathcal{V}_0 \perp g$, or $g \in \mathcal{V}_0^\perp$. 
A factorisation system

**Theorem (Bousfield)**

Any morphism $A \to B$ factors as

$$A \xrightarrow{v} C \xrightarrow{h} D$$

with $v \in \perp (\mathcal{Z}_0 \perp)$ and $h \in \mathcal{Z}_0 \perp$.

Not quite there yet: need to prove the obtained $(v', s')$ is again a trace!

**Theorem**

Traces are stable under restriction.
Main result

- We define a translation $\llbracket \cdot \rrbracket : \Pi \rightarrow$ Strategies.
- Compositional $\sim$ easy to define semantic counterparts to testing equivalences.
- Idea: $P$ passes the test $T$ iff $P \mid T$ satisfies some property.

  (e.g., eventually `ticks')

- Notation: $P \mid T \in \bot$.
- $P \sim Q$ iff $\forall T, (P \mid T \in \bot) \iff (Q \mid T \in \bot)$.

For any testing equivalence (with mild hypotheses):

Theorem (intensional full abstraction)

The translation induces a bijection on quotients:

$$\Pi /\sim \cong \text{Strategies} /\sim.$$
Conclusion

- Notably left out of this talk:
  - Proper definition of \( \mathbb{C} \).
  - Proof that traces are stable under restriction.
  - New approach to proving intensional full abstraction.

- Future work:
  - more complex calculi (functional, then functional & concurrent);
  - applying notion of trace (see EI talk);
  - study morphisms between calculi.
Universal property of restriction
From presheaves on views to sheaves on traces

Use right Kan extension: for any configuration $X$, consider

$$\mathcal{V}_X^{\text{op}} \xrightarrow{i^{\text{op}}} \mathcal{F}_X^{\text{op}}$$

Explicit formula

- General: $S'(p) = \int_{v \in \mathcal{V}_X} S(v) \mathcal{F}_X(v, p)$.
- Boolean case; $p$ accepted iff all its views are:

$$S'(p) = \bigwedge_{\{v \xrightarrow{a} p \in \mathcal{F}_X\}} S(v).$$