Congruences for Contextual Graph-Rewriting

Vladimiro Sassone, Pawel Sobocinski
Dagstuhl, June 7, 2004
Plan of the talk

1. Deriving bisimulation congruences
2. Cospans as generalised contexts
3. Bisimulation for graph rewriting
Deriving Congruences

Many syntactic formalisms for concurrency and mobility

Unification efforts:

1. Milner et al ‘90s-now: action calculi, bigraphs


3. Sewell, Leifer, Milner, Sassone and Sobocinski: meta theory of process calculi
Labels in LTS

Slogan: Labels should be smallest contexts which allow reaction/interaction

eg. simple CCS-style calculus \[ a.P + P' \xrightarrow{-\{\bar{a}\}} P \]

Sewell (1998): Detailed syntactic analysis of simplified process calculi


Sassone and Sobocinski (2002): 2-categorical generalisation to allow handling of structural congruences.
A reactive system

- objects = typed "holes"
- arrows = contexts
- 2-cells = "structural congruence"

\[ a \rightarrow b \]

if there exists
\[ \langle l, r \rangle, d \in D \]

and
\[ \rho : dl \Rightarrow a, \rho' : b \Rightarrow dr \]
GRPOs

Given $\alpha: ca \Rightarrow db$

$\langle I_5, e, f, g, \beta, \gamma, \delta \rangle$

$\delta b \cdot g \beta \cdot \gamma a = \alpha$
LTS

- Nodes: $[a]: 0 \rightarrow I_1$
- Labels: $[a] \xrightarrow{[f]} [a']$

$\exists \langle l, r \rangle \in R \quad \exists f \in C \quad \exists d \in D \quad \exists \alpha : f a \Rightarrow dl \quad \exists \alpha' : d r \Rightarrow a'$

and $I_2$ is a GRPO wrt itself
Properties of LTS

- *Bisimulation* is a congruence
- *Trace equivalence* is a congruence
- *Failures equivalence* is a congruence
What’s the point?

Why am I telling you all this??
Cospan Bicategories

Given $\mathbf{C}$, $\text{Cospan}(\mathbf{C})$ has

- Objects: those of $\mathbf{C}$
- Arrows: cospans $I_1 \xrightarrow{f} \mathbf{C} \xleftarrow{g} I_2$
- 2-cells: cospan “homorphisms”
- Composition by pushout along common interfaces.
- Intuitively: category of contexts over $\mathbf{C}$. 
Composition

Identities: \( I_1 \xrightarrow{id} I_1 \xleftarrow{id} I_1 \)

Composition by pushout

\[
\begin{array}{c}
C +_{I_2} D \\
\end{array}
\]

\[
\begin{array}{ccc}
I_1 & \xrightarrow{f} & C \\
& \xleftarrow{g} & I_2 \\
& \xrightarrow{f'} & D \\
& \xleftarrow{g'} & I_3 \\
\end{array}
\]

\[a : (C +_{I_2} D) +_{I_3} E \rightarrow C +_{I_2} (D +_{I_3} E)\]

\[e_l : (I_1 +_{I_1} C) \rightarrow C\]

\[e_r : (C +_{I_2} I_2) \rightarrow C\]

satisfying coherence
Cospans on Graphs

What is this when \( C \) is Graphs?
Desiderata

For a suitable, general class of categories $\mathbf{C}$, $\text{Cospan}(\mathbf{C})$ has redex-GRPOs.

Would allow to derive a coinduction principle for each “category of contexts” over a suitable $\mathbf{C}$. 
What is an adhesive category?
A category $\mathbf{C}$ is adhesive when

1. It has pushouts along monos
2. It has pullbacks
3. pushouts along monos are VK squares
Van Kampen Square

- Given a cube with back faces pullbacks:
- top face pushout iff front faces pullbacks
Graphs is Adhesive

You didn’t expect otherwise, did you??
Left-Linear Cospans

When $\mathbf{C}$ is adhesive $\mathbf{LLC}(\mathbf{C})$ is the bicategory

- objects as in $\mathbf{C}$
- arrows cospans $I_1 \xrightarrow{m} C \xleftarrow{g} I_2$
Theorem: Suppose that $\mathcal{C}$ is an adhesive category.

Then, $\text{LLC}(\mathcal{C})$ has redex-GRPOs.
Example 1

All morphisms mono

\[
\begin{array}{c}
\text{d} \\
\text{b} \\
\text{c} \\
\text{a} \\
\text{d} \\
\text{b} \\
\text{c} \\
\text{a} \\
\text{d} \\
\text{b} \rightarrow \text{c} \\
\text{a} \\
\end{array}
\]
Example 2

$o_A$ and $o_L$ not mono
Example 3

\[ o_A \text{ not mono} \]
GRPOs in LLC(C)

Given redex square...
GRPOs of Cospans

... find minimal factorisation
Construction

\[ Y = A \cup_X B \]
Graph Rewriting as Reactive System

For every span \( L \leftarrow \stackrel{l}{K} \rightarrow \stackrel{r}{R} \)

let \( \langle 0 \rightarrow L \leftarrow K, 0 \rightarrow R \leftarrow K \rangle \in \mathcal{R} \)

**Lemma:**

\[
\longrightarrow \quad \text{double-pushout rewrite}
\]

\[
\longrightarrow \quad \text{reaction relation in reactive system}
\]

\( C \longrightarrow D \quad \text{iff} \quad \overset{0}{C} \longrightarrow \overset{0}{D} \)
LTS for graph rewriting

The resulting LTS has:

- **Nodes**: graphs (up-to-iso) with output interface (possibly non-mono)
- **Labels**: smallest graph contexts (up-to-iso) which allow reaction

**Theorem**: Bisimulation, trace equivalence, failures equivalence are congruences
Advantages of LTS

- Transfer of concepts from process algebra to graph rewriting
- Labelled, compositional semantics
  - the class of adhesive categories covers many categories with “graph-like” objects
And what’s this for?

What’s missing here??
Special Cases

Rewriting with borrowed contexts [Ehrig and Koenig (2004)]

- LTS for graph rewriting, up-to-iso not taken into account, all interfaces mono

  **Theorem**: when restricting our approach to linear cospans we derive *the same* LTS

  **Corollary**: their congruence theorem

- Bigraphs...
The case of Bigraphs

- **Bigraphs** can be seen as LLC(dpl-grph).
- It follows from the theorem that **Bigraphs** has GRPOs.
- Main difference with Milner’s original bigraphs: input-linearity and name aliasing.

The case of **Trigraphs** ... as above

...
Conclusion

- Construction of labels for an interesting class of reactive systems
- Two applications so far, more in the future?
Minimality

\begin{align*}
\varphi: e' \Rightarrow he \\
\psi: hf \Rightarrow f'
\end{align*}

\begin{align*}
\tau e \cdot g' \varphi \cdot \gamma' &= \gamma \\
\delta' \cdot g' \psi \cdot \tau^{-1} f &= \delta \\
\psi b \cdot h \beta \cdot \varphi a &= \beta'
\end{align*}
Essential Uniqueness

\[ \exists! \xi : h \rightarrow h' \]

\[ \xi e \cdot \varphi = \varphi' \]
\[ \psi \cdot \xi^{-1} f = \psi' \]
\[ \tau' \cdot g' \xi = \tau \]