A 2-Categorical Approach to Bisimulation Congruences

FMM: Formal Methods for Mobility
Marseille 7.09.03

Vladimiro Sassone

with Pawel Sobocinski

University of Sussex, MYTHS

≪ ⋘ ⋙ ≫

Introduction

- Process calculi are often presented as:
  1. Syntax
  2. Structural congruence
  3. Reactions
  4. Labelled transition system(s) (LTS)

Introduction

- Process calculi are often presented as:
  1. Syntax
  2. Structural congruence
  3. Reactions
  4. Labelled transition system(s) (LTS)

- This talk is about categorical machinery which allows the derivation of a LTS from reactions.

- Bisimulation on such an LTS is a congruence, provided a general condition is met.

A Reduction System for CCS

Syntax:

$$p ::= \sum_i \alpha_i.p_i \mid p \mid p \mid (\nu a)p \mid A\langle \vec{u} \rangle$$

with $\vec{A}(\vec{x}) = \vec{p}$ a set of parametric, mutually recursive definitions.

Structural Congruence:

summands in $\sum_i$ can be rearranged arbitrarily

$|$ is a monoid with $0 \triangleq \sum_{\varnothing}$ for unit

$(\nu a)p \equiv (\nu b)p\{a := b\}$ (b not in p)

$(\nu a)(p \mid q) \equiv (\nu a)p \mid q$ (a not in q)

$A\langle \vec{u} \rangle \equiv p\{\vec{x} := \vec{u}\}$ (if $A(\vec{x}) = p$ is a def)

Reduction Rules: $(a.p + \sum_i \alpha_i.p_i) \mid (\bar{a}.q + \sum_j \beta_j.q_j) \searrow p \mid q$

Reactive Contexts: $\mathscr{E} ::= (\_) \mid \mathscr{E} \mid p \mid p \mid \mathscr{E} \mid (\nu a)\mathscr{E}$

Labelled Transitions: . . .

Coinduction Principles for Reductions

The reactions in a process calculus often give the computational intuitions of the calculus (eg. CCS, pi-calculus, ambient-calculus, . . .)

How can we give a congruent equivalence using the reaction rules?

≪ ⋘ ⋙ ≫

Coinduction Principles for Reductions

The reactions in a process calculus often give the computational intuitions of the calculus (eg. CCS, pi-calculus, ambient-calculus, . . .)

How can we give a congruent equivalence using the reaction rules?

- Barbed bisimulation $\cong$
  - (Milner, Sangiorgi 92)

≪ ⋘ ⋙ ≫

Coinduction Principles for Reductions

The reactions in a process calculus often give the computational intuitions of the calculus (eg. CCS, pi-calculus, ambient-calculus, . . .)

How can we give a congruent equivalence using the reaction rules?

- Barbed bisimulation $\cong$
  - (Milner, Sangiorgi 92)
- Equating insensitive terms
  - (Honda, Yoshida 95)

Coinduction Principles for Reductions

The reactions in a process calculus often give the computational intuitions of the calculus (eg. CCS, pi-calculus, ambient-calculus, . . .)

How can we give a congruent equivalence using the reaction rules?

- Barbed bisimulation $\cong$
  - (Milner, Sangiorgi 92)
- Equating insensitive terms
  - (Honda, Yoshida 95)
- Deriving an LTS
  - (Sewell 98, Leifer and Milner 00, this talk)

Coinduction Principles for Reductions

The reactions in a process calculus often give the computational intuitions of the calculus (eg. CCS, pi-calculus, ambient-calculus, . . .)

How can we give a congruent equivalence using the reaction rules?

- Barbed bisimulation $\cong$
  - (Milner, Sangiorgi 92)
- Equating insensitive terms
  - (Honda, Yoshida 95)
- Deriving an LTS
  - (Sewell 98, Leifer and Milner 00, this talk)

LTSs Desiderata:

- Operational Correspondence: $p \searrow q$ iff $p \xrightarrow{\tau} q$ (up to $\equiv$)

Contexts as Labels

The intuition:

$$a \xrightarrow{\mathscr{C}} b \text{ iff } \mathscr{C}[a] \searrow b$$

≪ ⋘ ⋙ ≫

UNIVERSITY OF SUSSEX

Contexts as Labels

The intuition:

$$a \xrightarrow{\mathscr{C}} b \text{ iff } \mathscr{C}[a] \searrow b$$

For instance:

$$a \xrightarrow{-|\bar{a}} \mathbf{0} \qquad M \xrightarrow{(\lambda x.-)N} M\{N/x\} \qquad \mathsf{K}M \xrightarrow{-N} M$$

Contexts as Labels

The intuition:

$$a \xrightarrow{\mathscr{C}} b \text{ iff } \mathscr{C}[a] \searrow b$$

For instance:

$$a \xrightarrow{-|\bar{a}} \mathbf{0} \qquad M \xrightarrow{(\lambda x.-)N} M\{N/x\} \qquad \mathbf{K}M \xrightarrow{-N} M$$

Yep, but not quite:

- Too many labels not desirable:
  - Useless combinatorial explosion: $\lambda x.xx \xrightarrow{-MN} MMN$
  - Messes up the bisimulation (too coarse): $l \xrightarrow{\mathscr{D}} \mathscr{D}[r]$ for all rules $l \searrow r$.

UNIVERSITY OF SUSSEX

Contexts as Labels

The intuition:

$$a \xrightarrow{\mathscr{C}} b \text{ iff } \mathscr{C}[a] \searrow b$$

For instance:

$$a \xrightarrow{-|\bar{a}} \mathbf{0} \qquad M \xrightarrow{(\lambda x.-)N} M\{N/x\} \qquad \mathbf{K}M \xrightarrow{-N} M$$

Yep, but not quite:

- Too many labels not desirable:
  - Useless combinatorial explosion: $\lambda x.xx \xrightarrow{-MN} MMN$
  - Messes up the bisimulation (too coarse): $l \xrightarrow{\mathscr{D}} \mathscr{D}[r]$ for all rules $l \searrow r$.

Choose only 'minimal' redex-enabling contexts

- Case analysis of basic situations: Sewell. Abstract approach: Leifer-Milner

$$\boxed{\boxed{\boxed{a} \mid \bar{a}} \mid a}$$

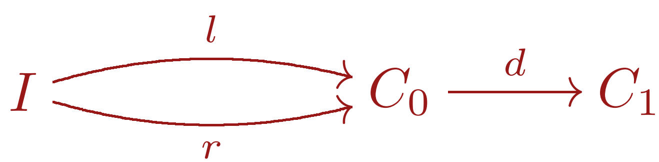
Reactive Systems

A generalisation of ground term rewriting systems.

- A category $\mathbb{C}$ with distinguished object I.
- A set of reaction rules $\mathcal{R} \subseteq \bigcup_{C \in \mathbb{C}} \mathbb{C}(I, C) \times \mathbb{C}(I, C)$.
- A set $\mathbb{D}$ of arrows of $\mathbb{C}$ called the reactive contexts.
  Assume that $d_0 \,.\, d_1 \in \mathbb{D}$ implies d_0 and $d_1 \in \mathbb{D}$.

The reaction relation is defined as

$$a \longrightarrow b \quad \text{iff} \quad a = d \,.\, l,\ b = d \,.\, r,\ d \in \mathbb{D} \text{ and } \langle l, r \rangle \in \mathcal{R}.$$

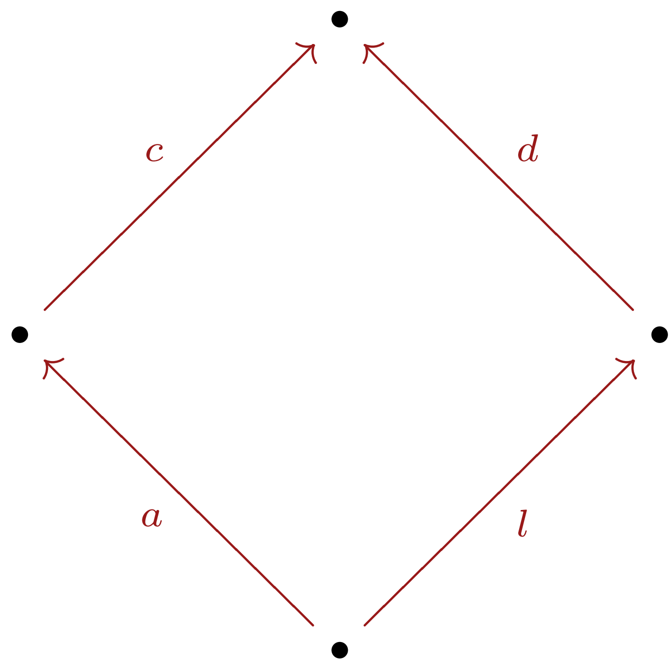


Well-known Reactive Systems

- Term rewriting systems
- Graph rewriting systems
  - via cospans
- Simple process calculi
  - with terms up to structural congruence

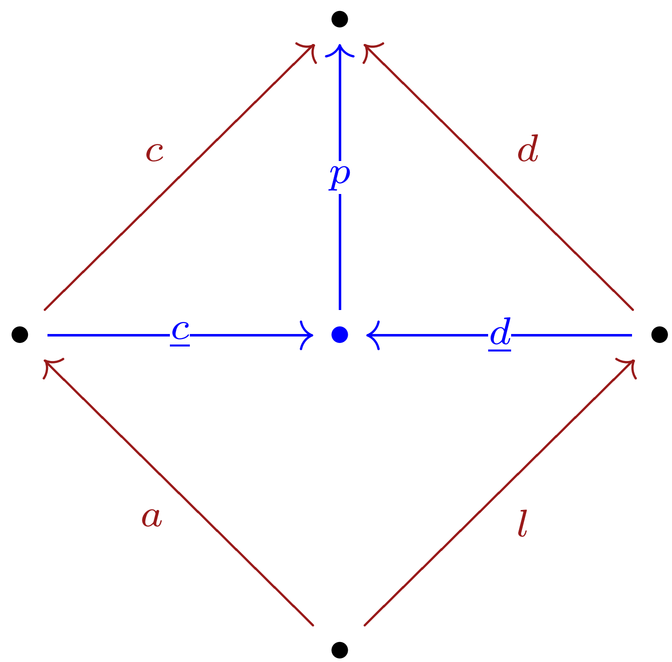
RPOs

Suppose that $\mathbb{C}$ is a category and consider a redex square



RPOs

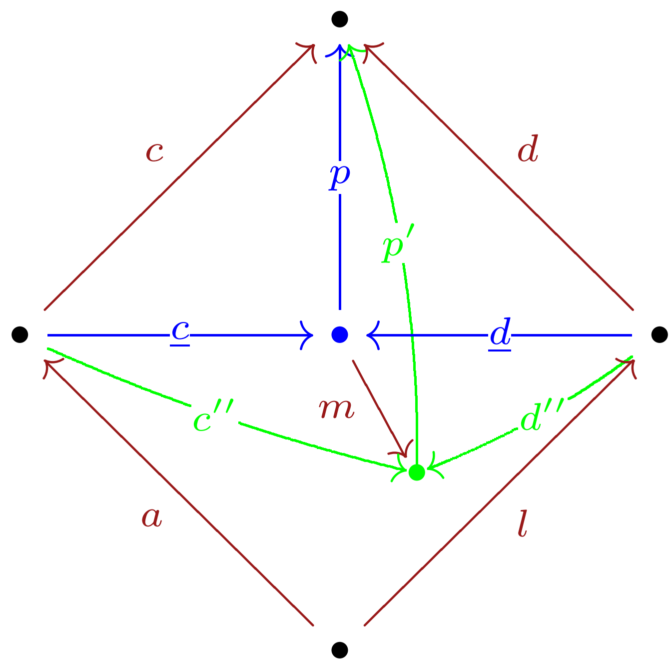
Suppose that $\mathbb{C}$ is a category and consider a redex square



- a relative pushout (RPO) is a tuple $\langle \underline{c}, \underline{d}, p \rangle$ which satisfies the universal property that:

RPOs

Suppose that $\mathbb{C}$ is a category and consider a redex square

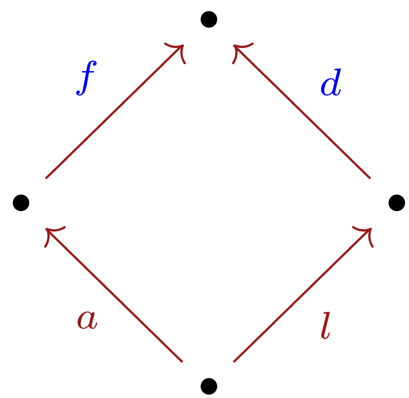


- a relative pushout (RPO) is a tuple $\langle \underline{c}, \underline{d}, p \rangle$ which satisfies the universal property that:
- for any other such $\langle c', d', p' \rangle$ there exists a unique mediating morphism m.

Deriving LTS

The LTS derived from the reactive system has:

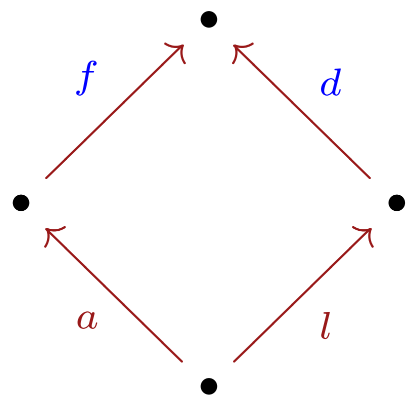
- Nodes: $a : I \to N$
- Transitions: $a \xrightarrow{f} dr$ iff for $\langle l, r \rangle \in \mathcal{R}$ and $d \in \mathbb{D}$, $\langle f, d, \mathrm{id} \rangle$ is a relative pushout (idem pushout or IPO) of the square



Deriving LTS

The LTS derived from the reactive system has:

- Nodes: $a : I \rightarrow N$
- Transitions: $a \xrightarrow{f} dr$ iff for $\langle l, r\rangle \in \mathcal{R}$ and $d \in \mathbb{D}$, $\langle f, d, \mathrm{id}\rangle$ is a relative pushout (idem pushout or IPO) of the square



- Thm. If all redex squares like the above have IPOs then the bisimulation on the derived LTS is a congruence [Leifer-Milner 00]

Applying RPOs

- When applied to term rewriting, RPOs yield the same LTS as Sewell's.
- Leifer (2000) found RPOs in a restricted class of action graph contexts.
- Milner (2001) worked out RPOs for a graphical formalism called bigraphs.
- Jensen and Milner (2003) derived (essentially) the usual π labelled bisimulation on asynchronous π using RPOs.

Applying RPOs

- When applied to term rewriting, RPOs yield the same LTS as Sewell's.
- Leifer (2000) found RPOs in a restricted class of action graph contexts.
- Milner (2001) worked out RPOs for a graphical formalism called bigraphs.
- Jensen and Milner (2003) derived (essentially) the usual π labelled bisimulation on asynchronous π using RPOs.

What about even very simple process calculi?

The technique doesn't actually scale up!

A Simple Calculus

Syntax:

$$p \ ::= \ \mathbf{0} \ | \ a \ | \ \overline{a} \ | \ p \,|\, p \qquad \text{where } a \in N.$$

Structural Congruence:

'|' associative, commutative with identity $\mathbf{0}$

Reactions:

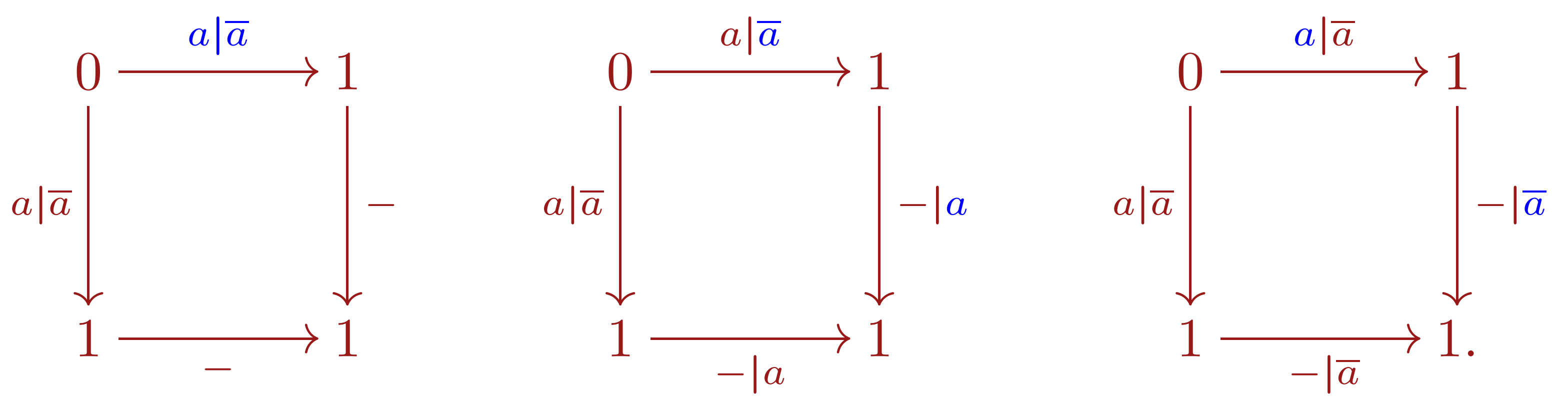
$$a \,|\, \overline{a} \longrightarrow \mathbf{0}$$

The Standard Labelled Transition System:

$$a \xrightarrow{a} \mathbf{0} \qquad \overline{a} \xrightarrow{\overline{a}} \mathbf{0} \qquad a \,|\, \overline{a} \xrightarrow{\tau} \mathbf{0}$$

$$\frac{p \xrightarrow{x} p'}{q \,|\, p \xrightarrow{x} q \,|\, p'} \qquad \frac{p \equiv p' \quad p' \xrightarrow{x} q' \quad q' \equiv q}{p \xrightarrow{x} q}$$

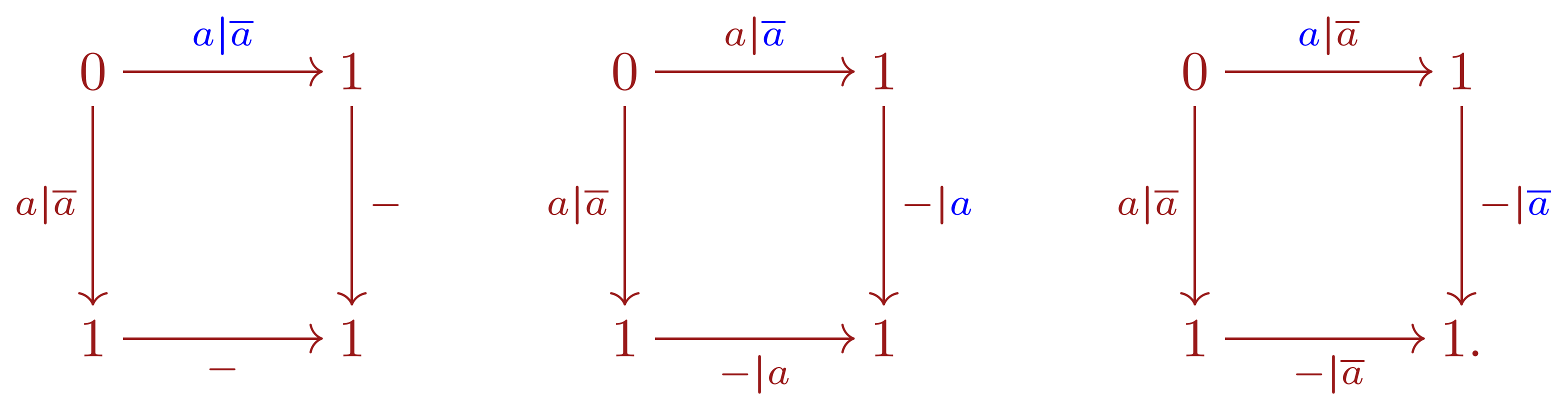
Simple Calculus (ctd.)



Only the left one could possibly be an IPO!

Yet, because of the structural congruence, the redex could partially come from the context. The derived LTS cannot account for this: only a $a \mid \overline{a} \xrightarrow{-} \mathbf{0}$ transition. And that is bad!

Simple Calculus (ctd.)



Only the left one could possibly be an IPO!

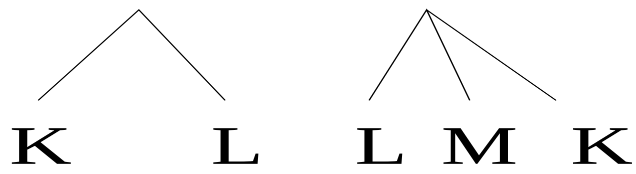
Yet, because of the structural congruence, the redex could partially come from the context. The derived LTS cannot account for this: only a $a \mid \overline{a} \xrightarrow{-} \mathbf{0}$ transition. And that is bad!

We need to keep track of structural congruence to locate the reaction.

Bunches and Wiring

Studied by Leifer and Milner (2000). A 'minimalistic' approach to connections and wiring.

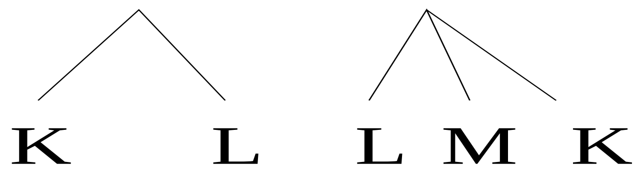
- A bunch



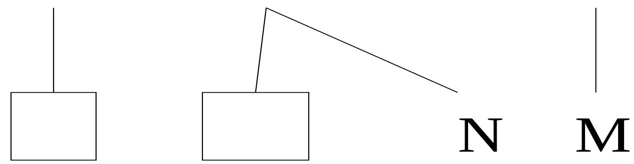
Bunches and Wiring

Studied by Leifer and Milner (2000). A 'minimalistic' approach to connections and wiring.

- A bunch



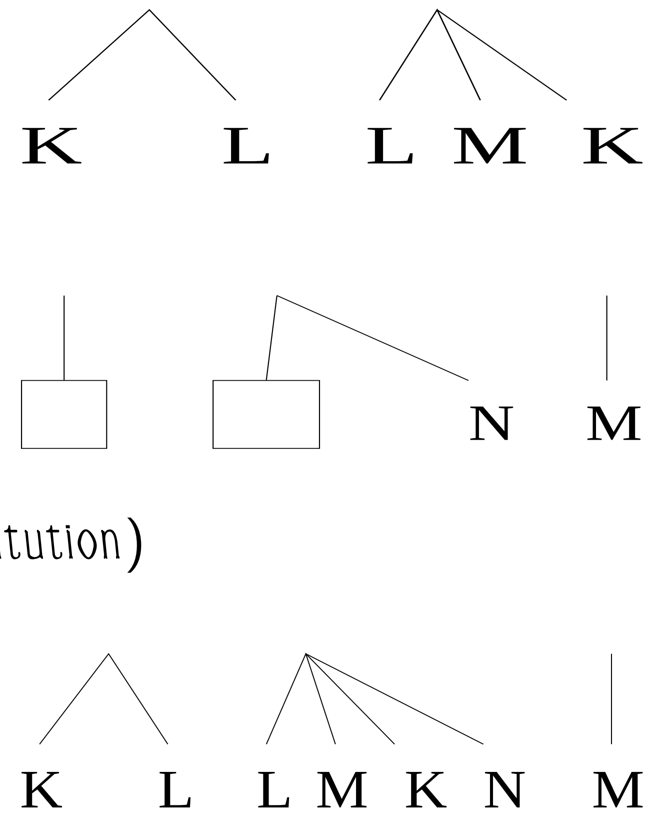
- A bunch context



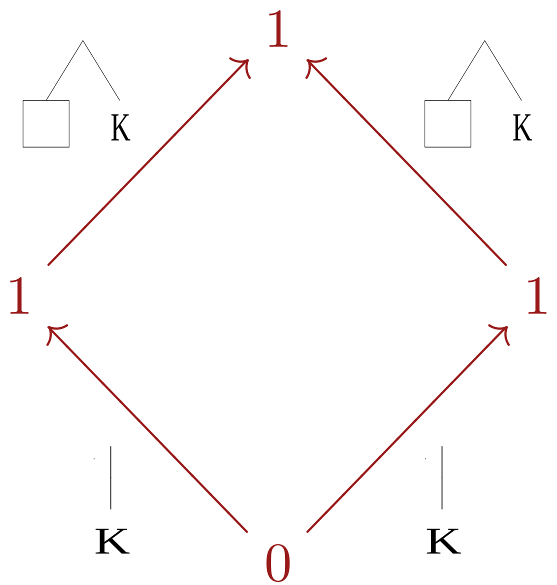
Bunches and Wiring

Studied by Leifer and Milner (2000). A 'minimalistic' approach to connections and wiring.

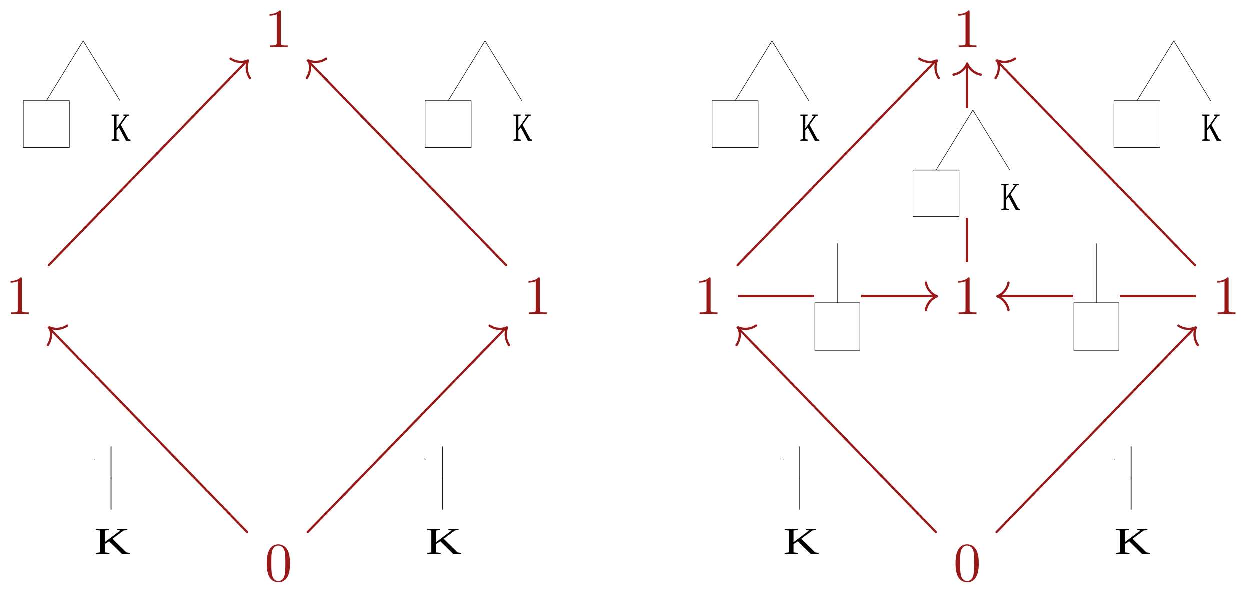
- A bunch
- A bunch context
- The composition (substitution)



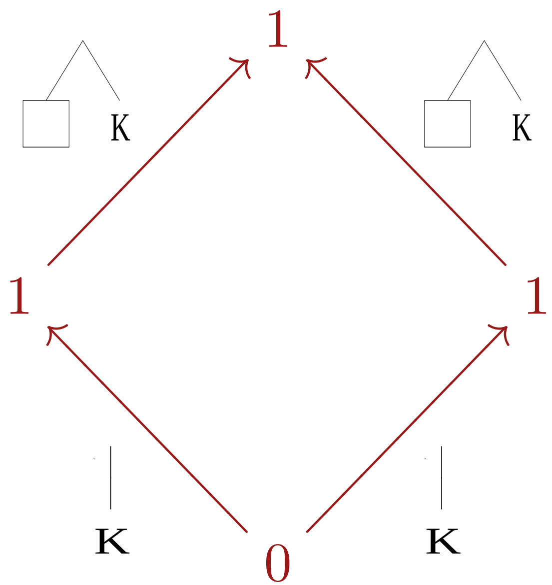
A commutative diamond

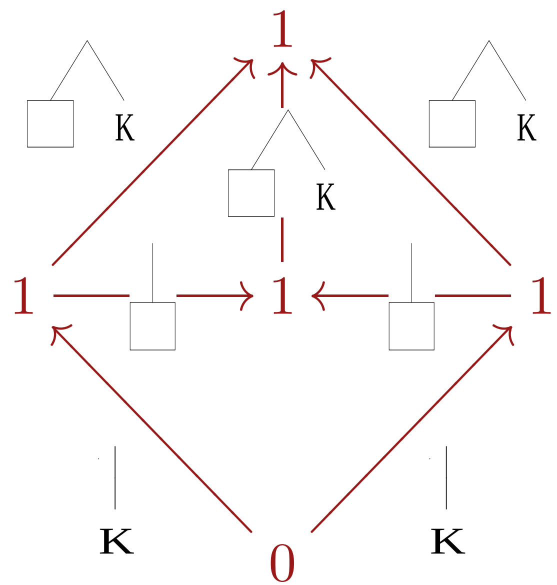


A commutative diamond

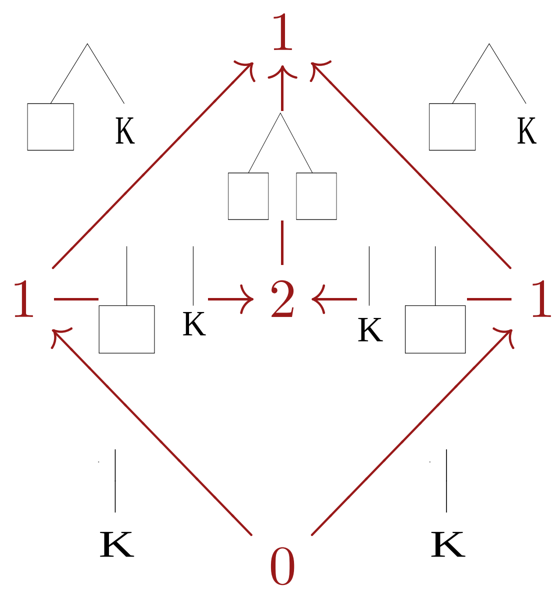


A commutative diamond

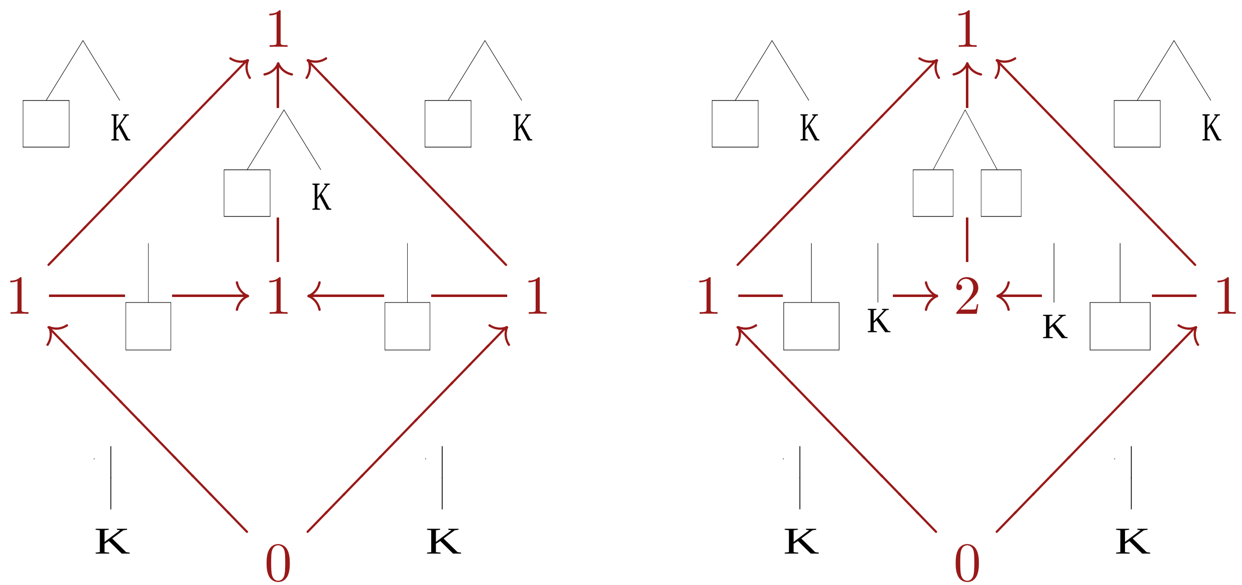




Another Canditate



Same problem, again!



They are incomparable!

2-categories

In other words, we need to keep track of how regions in the diagrams commute!

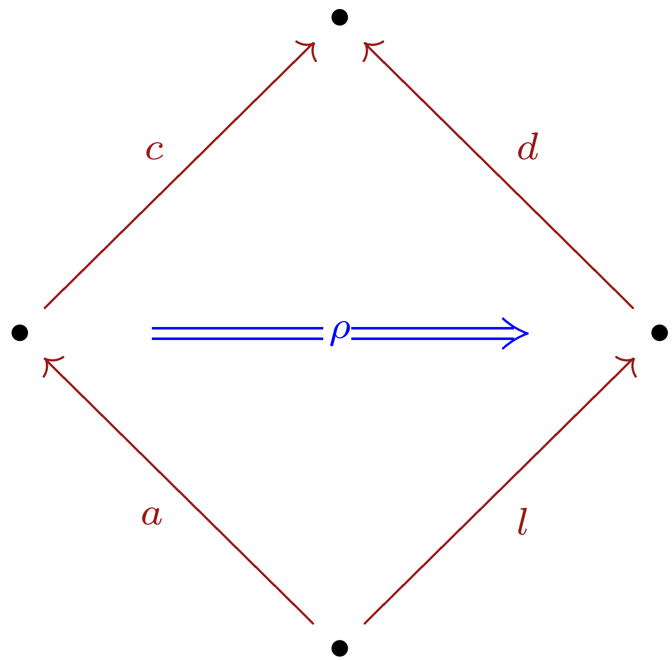
2-categories:

- Objects and arrows like in categories
- 2-cells: morphisms between arrows

$$A \underset{b}{\overset{a}{\rightrightarrows}} B \quad (\Downarrow)$$

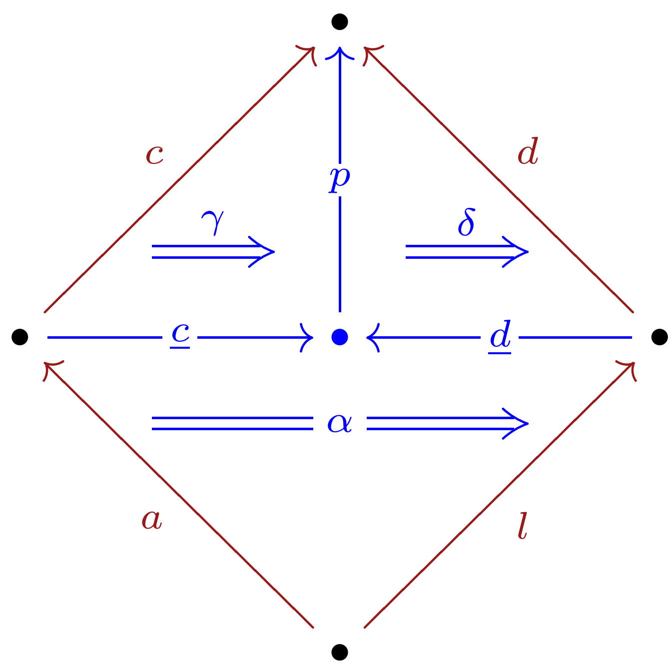
GRPOs

Suppose that $\mathbb{C}$ is a 2-category with all 2-cells isomorphisms



GRPOs

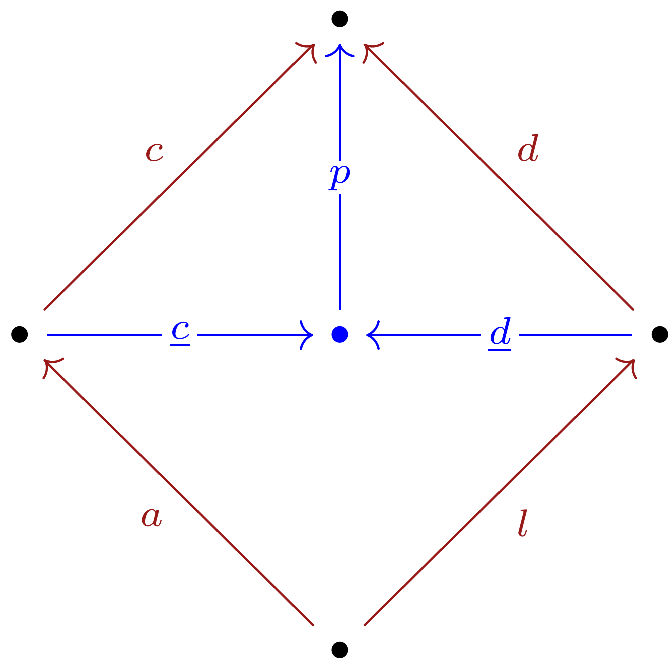
Suppose that $\mathbb{C}$ is a 2-category with all 2-cells isomorphisms



- a G-relative pushout (GRPO) is a tuple $\langle \underline{c}, \underline{d}, p, \alpha, \gamma, \delta \rangle$ which satisfies $\delta l \bullet p\alpha \bullet \gamma a = \rho$, and the universal property that:
- for any other such $\langle c', d', p', \alpha', \gamma', \delta' \rangle$ there exists an essentially unique mediating morphism m.

GRPOs

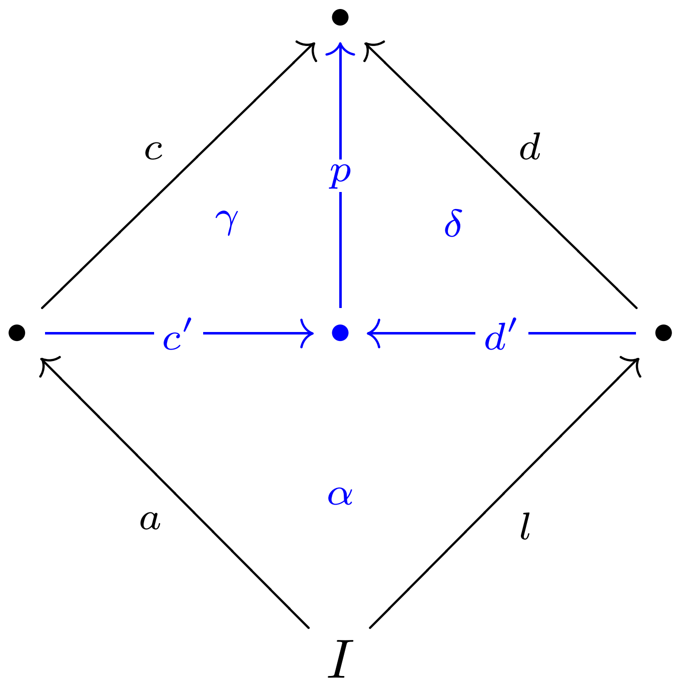
Suppose that $\mathbb{C}$ is a category



- a relative pushout (RPO) is a tuple $\langle \underline{c}, \underline{d}, p \rangle$ which satisfies $p\underline{c} = c$ and $p\underline{d} = d$, and the universal property that:
- for any other such $\langle c', d', p' \rangle$ there exists a unique mediating morphism m.

Essential Uniqueness (I)

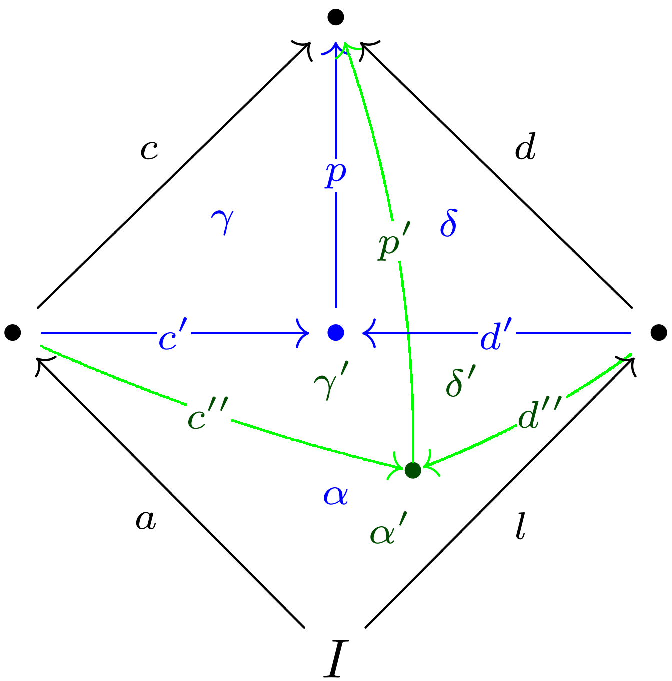
Suppose that $\mathbb{C}$ is a 2-category with all 2-cells isomorphisms



➤ $\delta l \bullet p\alpha \bullet \gamma a = \rho$

Essential Uniqueness (I)

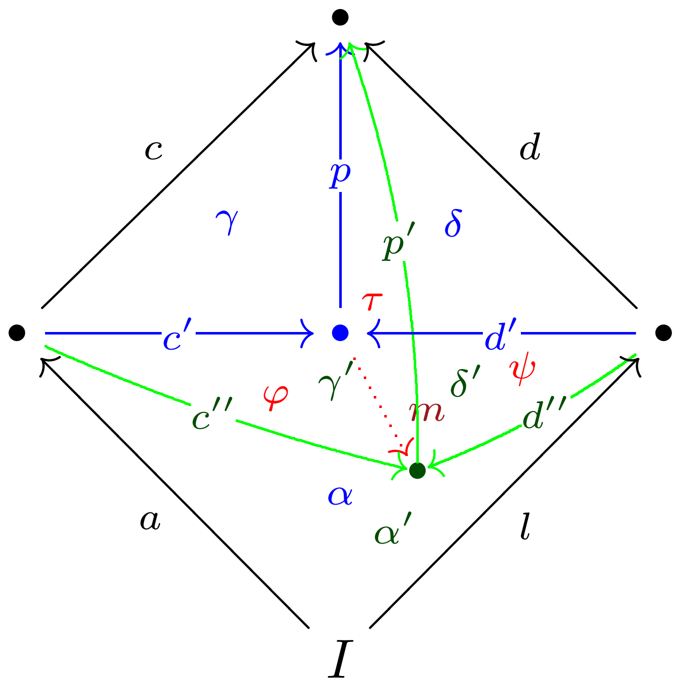
Suppose that $\mathbb{C}$ is a 2-category with all 2-cells isomorphisms



➤ $\delta l \bullet p\alpha \bullet \gamma a = \rho$ and $\delta' l \bullet p\alpha' \bullet \gamma' a = \rho$

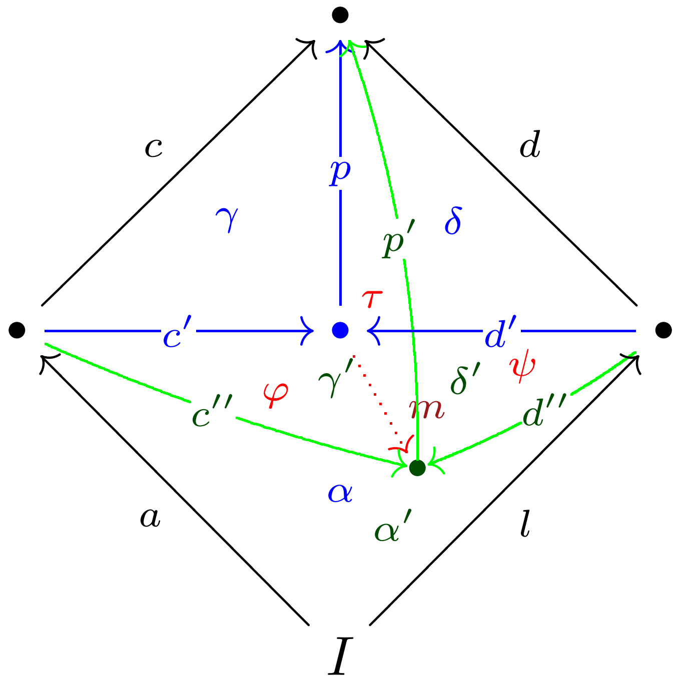
Essential Uniqueness (I)

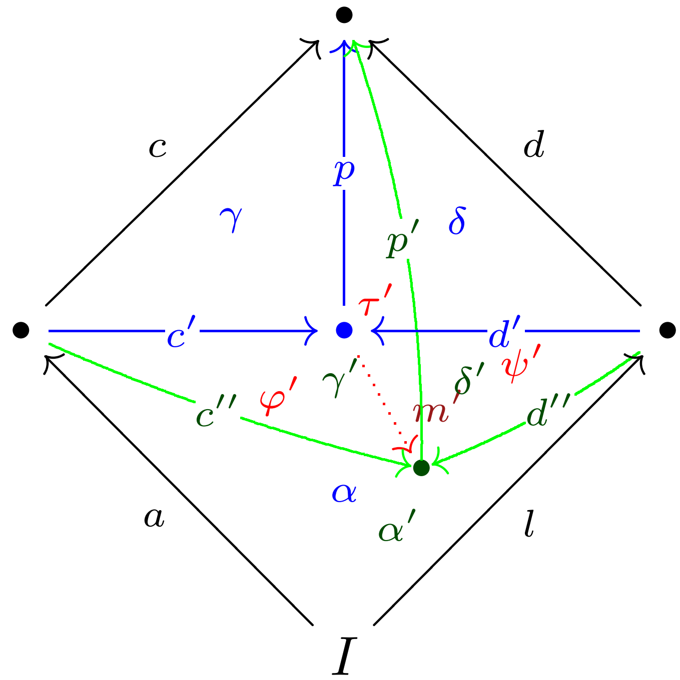
Suppose that $\mathbb{C}$ is a 2-category with all 2-cells isomorphisms



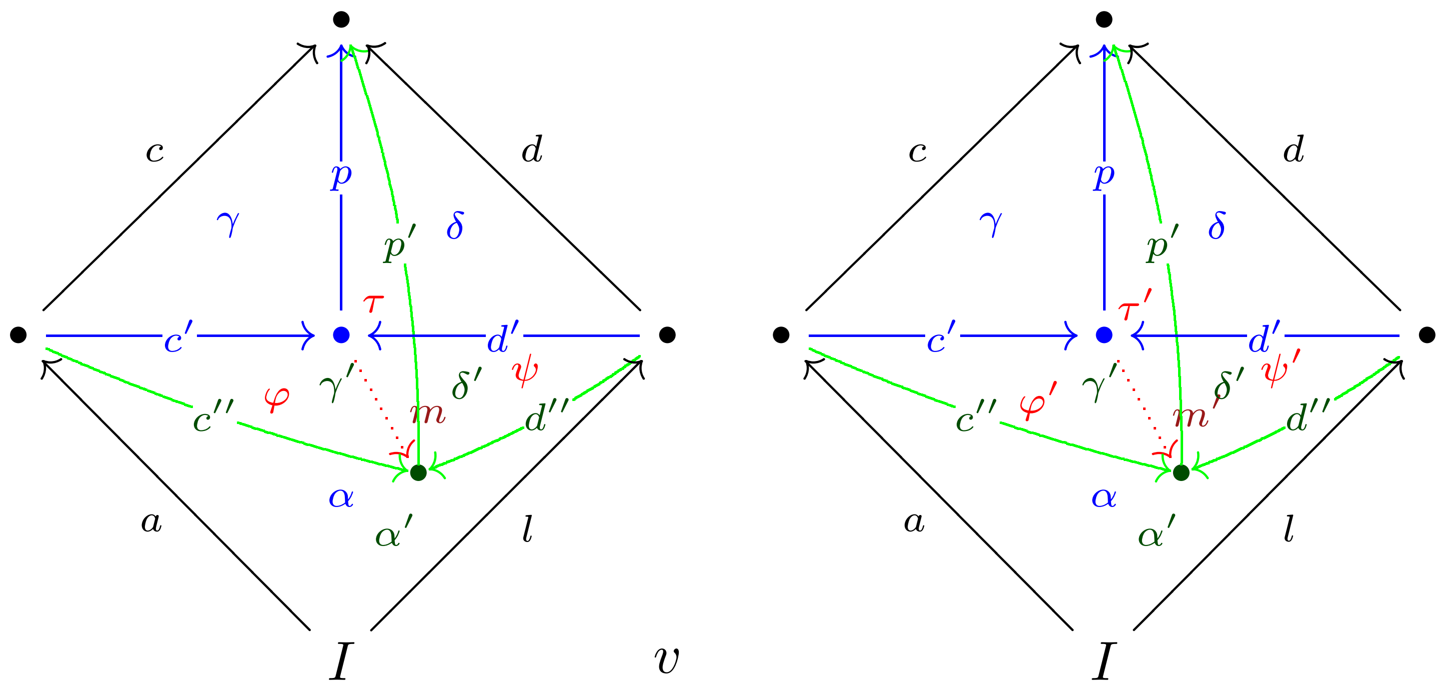
- $\delta l \bullet p\alpha \bullet \gamma a = \rho$ and $\delta' l \bullet p\alpha' \bullet \gamma' a = \rho$
- $\tau . c' \bullet p' . \varphi \bullet \gamma' = \gamma$, $\delta' \bullet p' . \psi \bullet \tau^{-1} . f = \delta$, $\psi . l \bullet m . \alpha \bullet \varphi . a = \alpha'$

Essential Uniqueness (II)





Essential Uniqueness (II)

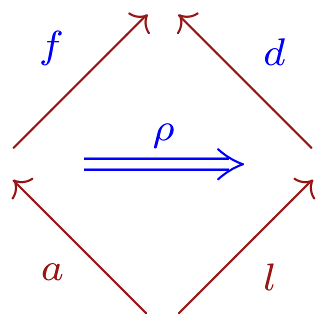


- There exists unique $\xi : m \Longrightarrow m'$ such that
  - $\xi \,.\, c' \bullet \varphi = \varphi'$;
  - $\psi \bullet \xi^{-1} \,.\, f = \psi'$;
  - $\tau' \bullet p' \,.\, \xi = \tau$.

Deriving LTS

The LTS derived from the reactive system with structural congruence has:

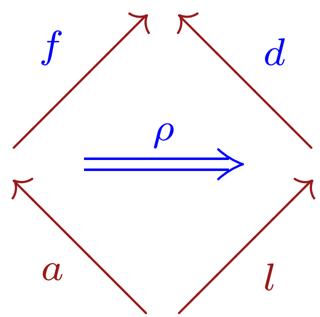
- Nodes: $[a] : I \to N$
- Transitions: $[a] \xrightarrow{[f]} [dr]$ iff there exists a 2-cell α such that for $\langle l, r \rangle \in \mathcal{R}$ and $d \in \mathbb{D}$, $\langle f, d, \mathrm{id}, \rho, \mathbf{1}, \mathbf{1} \rangle$ is a G-relative pushout (G-idem pushout or GIPO) of the square



Deriving LTS

The LTS derived from the reactive system with structural congruence has:

- Nodes: $[a] : I \to N$
- Transitions: $[a] \xrightarrow{[f]} [dr]$ iff there exists a 2-cell α such that for $\langle l, r\rangle \in \mathcal{R}$ and $d \in \mathbb{D}$, $\langle f, d, \mathrm{id}, \rho, \mathbf{1}, \mathbf{1}\rangle$ is a G-relative pushout (G-idem pushout or GIPO) of the square



Thm. If every square in $\mathbb{C}$ such as the one above has a GRPO, then the LTS bisimulation on the synthesised LTS is a congruence.

the theory is a generalisation of the theory of RPOs.

Simple Process Calculus (ctd.)

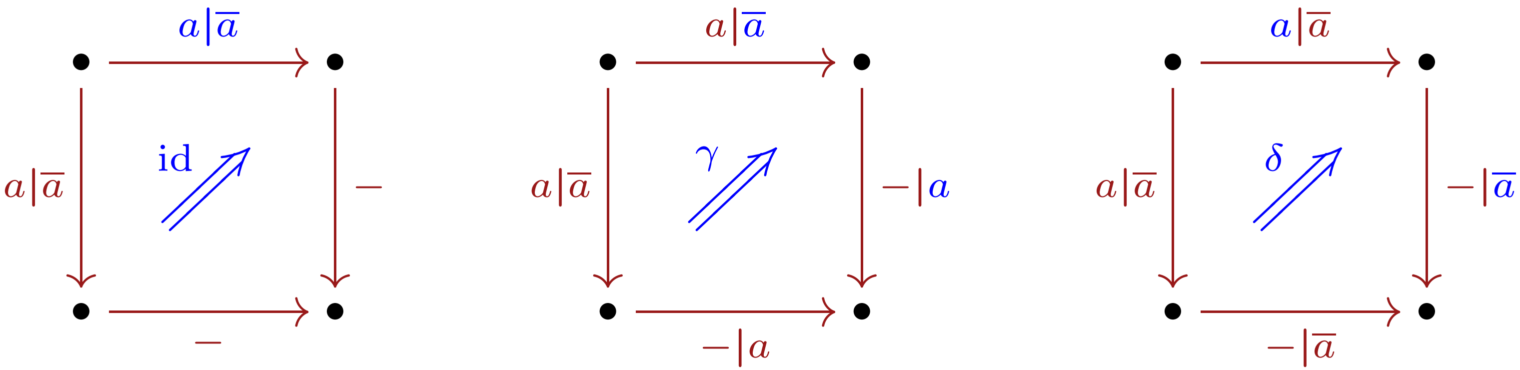
Let $\mathbb{C}$ be the 2-category with

- A single Object
- Arrows: strings $a_1 \mid a_2 \mid \ldots \mid a_n$
  - composition by concatenation
- 2-cells: permutations $a_1 \mid a_2 \mid \ldots \mid a_n \Rightarrow a_{\sigma(1)} \mid a_{\sigma(2)} \mid \ldots \mid a_{\sigma(n)}$

Then GRPOs exist and give the expected LTS.

In particular, $a \mid \overline{a}$ has transitions $a \mid \overline{a} \xrightarrow{-\mid a} a$ and $a \mid \overline{a} \xrightarrow{-\mid \overline{a}} \overline{a}$, witnesses of its potential interations with the environment.

Simple Calculus (ctd.)



Each of these squares is a GRPO!

The 2-cells trace the structural congruence and place the reaction.

Note: γ and δ swap the 1st/3rd and 2nd/3rd element, so as to put in evidence the intended redex.

≪ ⋘ ⋙ ≫

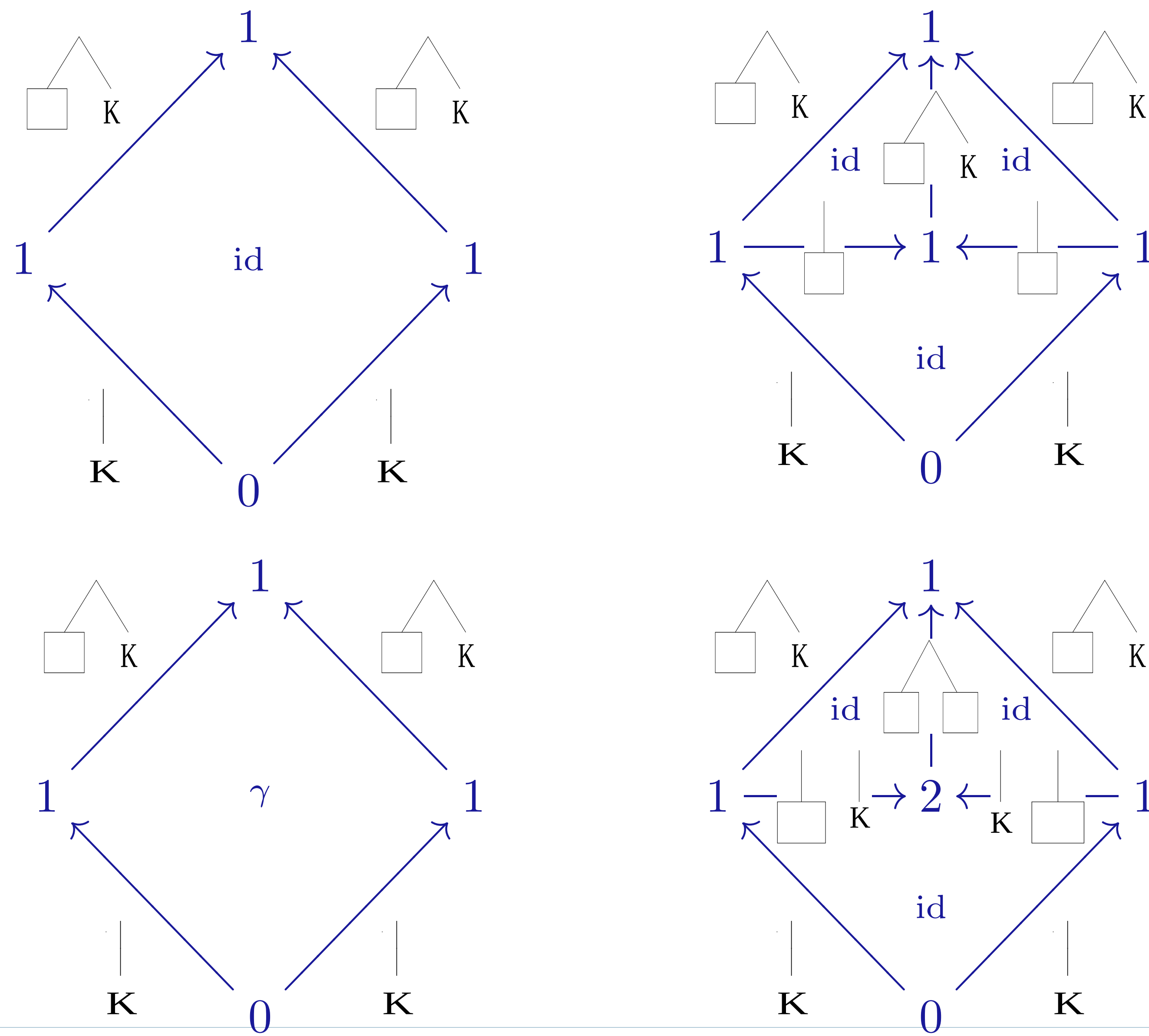
UNIVERSITY OF SUSSEX

Bunches (ctd.)

Thm. The G-category of bunches and bunch isomorphisms as 2-cells has all GRPOs.

Bunches (ctd.)

Thm. The G-category of bunches and bunch isomorphisms as 2-cells has all GRPOs.



Conclusion

- GRPOs have been applied successfully to simple, yet significative examples such as Leifer-Milner's category of 'bunches and wires' and simple process algebras.
- They are a more standard categorical alternative to previous theory
  - Milner's 'precategories;'
  - Leifer's 'functorial reactive systems.'
- There is an easy encoding of the these two theories into the G-world.

  So far, the price of the initial 2-categorical investment seems worth paying...

Future Work

- Extend to more complicated process calculi (e.g. ambients), with complex structural congruences (e.g. replication).
- Apply the theory to graph rewriting to obtain interesting new semantics.