

# An abstract Howe theorem

Peio Borthelle   Tom Hirschowitz<sup>1</sup>   Ambroise Lafont<sup>2</sup>

<sup>1</sup>Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, LAMA  
Chambéry, France

<sup>2</sup>UNSW, Sydney

LIMD  
September 2020

# Motivation: generalisation of theorem statements

- Often, theorems are stated for **one** “typical” programming language.
- Goal: provide high-level tools for stating them **for all suitable languages and models**.

# State of the art

- **Formats**: Tyft/tyxt, GSOS, PATH,...
  - Do not cover denotational models (exclusively syntactic).
  - Low-level.
- **Bialgebraic semantics** (Turi and Plotkin '97).
  - Deeply developed.
  - Very good at quantitative semantics.
  - Functional languages only starting to be investigated (Peressotti '17).
- **Transition monads** (Hirschowitz et al. '20).
  - Focus on signatures (less primitive than ours).
  - No big metatheoretical theorem (yet).
- Previous work on **cellular monads** (POPL '19, SOS/EXPRESS '19).
  - Does not cover higher-order languages.
  - Virtually no notion of signature, models constructed by hand.

# Summary of contributions

1. General setting: **Howe context**.
2. Notion of **signature** for programming languages, in any Howe context.
3. Definition of **substitution-closed bisimilarity**.  
Particular case: open extension of Abramsky's **applicative** bisimilarity in cbn  $\lambda$ -calculus.
4. A **semantic format** for congruence of bisimilarity:

## Main theorem

If the signature **preserves functional bisimulations** (plus mild technical hypotheses), then substitution-closed bisimilarity is a congruence.

Proof: abstract analogue of Howe's method.

# This talk

Sketch main ideas on **one example**, big-step, cbn  $\lambda$ -calculus.

- ① Introduction
- ② Brief recap on applicative bisimilarity
- ③ Howe context for cbn  $\lambda$
- ④ Models of syntax
- ⑤ Models of transition rules
- ⑥ Substitution-closed bisimilarity
- ⑦ Main result
- ⑧ Conclusion

# Call-by-name $\lambda$ -calculus

Slightly non-standard presentation.

$$\frac{}{\lambda x.e \Downarrow e} \qquad \frac{e_1 \Downarrow e_3 \quad e_3[e_2] \Downarrow e_4}{e_1 \ e_2 \Downarrow e_4}$$

Typing:  $\Downarrow \subseteq$  closed terms  $\times$  terms with 1 free variable.

Example:

# Applicative bisimilarity

## Definition

A relation  $R$  between terms is

- *substitution-closed* iff

$$e_1 R e_2 \quad \text{then} \quad e_1[\sigma] R e_2[\sigma].$$

- a *bisimulation* iff

$$\begin{array}{ccc} e_1 & \text{--- } R \text{ ---} & e_2 \\ \Downarrow & & \Downarrow \\ e'_1 & \text{--- } R \text{ ---} & e'_2 \end{array} \quad \text{and} \quad \begin{array}{ccc} e_1 & \text{--- } R \text{ ---} & e_2 \\ \Downarrow & & \Downarrow \\ e'_1 & \text{--- } R \text{ ---} & e'_2 \end{array}$$

(for all / exists).

Standard applicative bisimilarity  $\coloneqq$  largest substitution-closed bisimulation.

Notation  $\sim^\otimes$ .

# Congruence theorem

## Theorem

*Applicative bisimilarity is a congruence, in particular*

- $e_1 \sim^\otimes e_2$  entails  $\lambda x.e_1 \sim^\otimes \lambda x.e_2$ ,
- $e_1 \sim^\otimes e_2$  and  $e_3 \sim^\otimes e_4$  entails  $e_1 e_3 \sim^\otimes e_2 e_4$ .



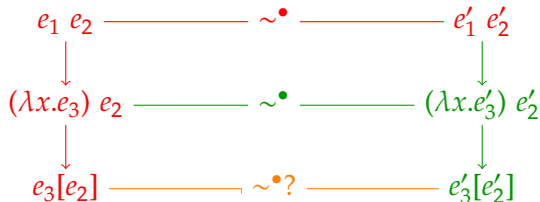
# Naive proof attempt

- Let  $\sim^\bullet$  denote the context closure of  $\sim^\otimes$ .
- Prove that it is a bisimulation.

Indeed, if so,

- $\sim^\otimes \subseteq \sim^\bullet \subseteq \sim^\otimes$ , hence
- $\sim^\otimes = \sim^\bullet$ ,
- but  $\sim^\bullet$  is context-closed.

But! Hard to prove the bisimulation property.



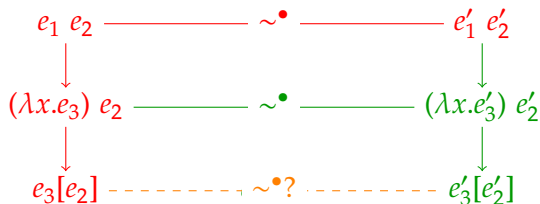
# Naive proof attempt

- Let  $\sim^\bullet$  denote the context closure of  $\sim^\otimes$ .
- Prove that it is a bisimulation.

Indeed, if so,

- $\sim^\otimes \subseteq \sim^\bullet \subseteq \sim^\otimes$ , hence
- $\sim^\otimes = \sim^\bullet$ ,
- but  $\sim^\bullet$  is context-closed.

But! Hard to prove the bisimulation property.



# Howe closure

Solution: context closure + 
$$\frac{e \sim^\bullet e' \sim e''}{e \sim^\bullet e''}$$

# What now?

- Categorical point of view on
  - syntax,
  - dynamics,
  - substitution-closed bisimulation,
  - Howe's method.
- Amenable to generalisation: see paper.

# Main question for us

What is a model of call-by-name  $\lambda$ -calculus?

# Models of syntax

The category  $\mathbb{F} := \mathbf{Set}_f^{op}$



Let  $X \in \widehat{\mathbb{F}}$ , **presheaf** on  $\mathbb{F}$ , i.e.,  $X: \mathbf{Set}_f \rightarrow \mathbf{Set}$ .

- $X(n)$ : “terms” with potential free variables in  $\{x_1, \dots, x_n\}$ .
- $X(n) \xrightarrow{X(f)} X(m)$ : “renaming”.

## Example

- $L(n)$  = actual terms over  $n$ .
- $$\begin{array}{ccc} L(2) & \xrightarrow{L(\text{swap})} & L(2) \\ \lambda x.(x_1 \ x_2 \ x) & \mapsto & \lambda x.(x_2 \ x_1 \ x). \end{array}$$

# Models of syntax

**Model of syntax:**  $X \in \widehat{\mathbb{F}}$  equipped with

**Operations**  $\lambda_n: X(n+1) \rightarrow X(n)$        $app_n: X(n)^2 \rightarrow X(n)$ .

**Substitution**

- Let  $(Y \otimes Z)(n) = \sum_p Y(p) \times Z(n)^p$  (modulo std eqs.).
- Elements  $y(\zeta)$  are like **formal substitutions**.
- Substitution:  $m_X: X \otimes X \rightarrow X$   
 $x(\chi) \mapsto x[\chi]$ .

**Variables**  $e_X: I \rightarrow X$ , where  $I(n) = n$  (notation for  $\{1, \dots, n\}$ ).

**Model of syntax =  $\Sigma_0$ -monoid :=**

$X \in \widehat{\mathbb{F}} + app, \lambda$ , substitution, variables + compatibility conditions.

# Free $\Sigma_0$ -transition monoids

## Definition

Category  $\Sigma_0$ -**mon** of  $\Sigma_0$ -monoids.

$$\widehat{\mathbb{F}} \begin{array}{c} \xrightarrow{\mathcal{L}_0} \\ \perp \\ \xleftarrow{\mathcal{U}_0} \end{array} \Sigma_0\text{-mon}$$

Syntax  $:= \mathcal{L}_0(\emptyset)$ .

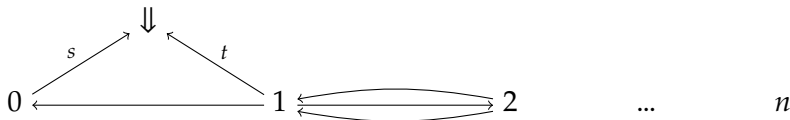
## Remark

“High-level” definition of syntax: no fuss about  $\alpha$ -equivalence.



# Transition systems

## The category $\mathbb{F}^{\Downarrow}$



Let  $X \in \widehat{\mathbb{F}^{\Downarrow}}$ , **presheaf** on  $\mathbb{F}^{\Downarrow}$ , i.e.,  $X: (\mathbb{F}^{\Downarrow})^{op} \rightarrow \mathbf{Set}$ .

- $X(n)$ ,  $X(f)$ : “terms” and “renaming” as before.
- $X(\Downarrow)$ : “evaluation witnesses”.
- $X(s): X(\Downarrow) \rightarrow X(0)$ : source/input.
- $X(t): X(\Downarrow) \rightarrow X(1)$ : body of value.

# Transition system with syntactic structure on states

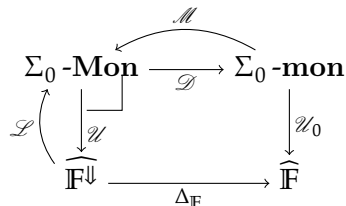
Consider the following pullback in **CAT**.

$$\begin{array}{ccc}
 \Sigma_0\text{-}\mathbf{Mon} & \xrightarrow{\mathcal{D}} & \Sigma_0\text{-}\mathbf{mon} \\
 \downarrow \mathcal{U} & & \downarrow \mathcal{U}_0 \\
 \widehat{\mathbb{F}}\Downarrow & \xrightarrow{\Delta_{\mathbb{F}}} & \widehat{\mathbb{F}}
 \end{array}$$

- Objects:  $X \in \widehat{\mathbb{F}}\Downarrow$  with  $\Sigma_0$ -monoid structure on the restriction  $\Delta_{\mathbb{F}}(X) \in \widehat{\mathbb{F}}$ .
- Name: **transition  $\Sigma_0$ -monoids**.

# Transition system with syntactic structure on states

Consider the following pullback in **CAT**.



- Objects:  $X \in \widehat{\mathbb{F}}^\downarrow$  with  $\Sigma_0$ -monoid structure on the restriction  $\Delta_{\mathbb{F}}(X) \in \widehat{\mathbb{F}}$ .
- Name: **transition  $\Sigma_0$ -monoids**.

Both projections have left adjoints!

# Models of transition rules

Remember our variant of cbn  $\lambda$ -calculus:

$$\frac{}{\lambda x.e \Downarrow e} \qquad \frac{e_1 \Downarrow e_3 \quad e_3[e_2] \Downarrow e_4}{e_1 \ e_2 \Downarrow e_4}$$

**Model of rules:** transition  $\Sigma_0$ -monoid  $X \in \widehat{\mathbb{F}}^\Downarrow$  equipped with

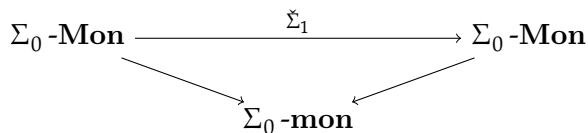
$$X(1) \rightarrow X(\Downarrow) \qquad \text{and} \qquad A_\beta(X) \rightarrow X(\Downarrow)$$

where  $A_\beta(X) = \{(r_1, e_2, r_2) \mid r_2 \cdot s = (r_1 \cdot t)[e_2]\}$ ,  
 + compatibility conditions for source and target.

# Models of rules as algebras

## Lemma

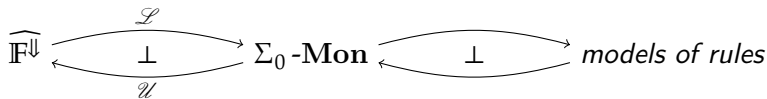
*Models of rules are **vertical** algebras for a suitable endofunctor*



Proof:  $\check{\Sigma}_1(X)(\Downarrow) = X(1) + A_\beta(X) \dots$

# Syntactic transition system

## Theorem



Syntactic transition system  $\mathbf{Z} :=$  initial model.

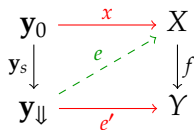
# Yoneda

Some presheaves in  $\widehat{\mathbb{F}^\Downarrow}$ :

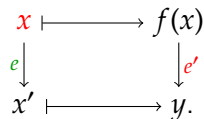
- $\mathbf{y}_0$ : just one element over 0.
- $\mathbf{y}_1$ : just one element over 1.
- $\mathbf{y}_\Downarrow$ : just one element over  $\Downarrow$  + its source and target.

# Functional bisimulation

A morphism  $f$  in  $\widehat{\mathbb{F}}^{\Downarrow}$  is a **functional bisimulation** iff



i.e., concretely



## Notation

$y_s \sqsupseteq f, f \in \{y_s\}^{\sqsupseteq}.$



# Bisimulation

## Definition

In  $\widehat{\mathbb{F}\Downarrow}$ , a span  $X \leftarrow R \rightarrow Y$  is a **bisimulation** iff both legs are functional bisimulations.

## Substitution-closed spans

For  $X \in \Sigma_0\text{-Mon}$ , a span  $R \rightarrow X \times X$  is **substitution-closed** iff (omitting  $\mathcal{D}$  for readability):

$$\begin{array}{ccccc}
 R \otimes X & \xrightarrow{\quad \text{dashed green arrow} \quad} & R & & \\
 \downarrow & & \downarrow & & \\
 X^2 \otimes X & \longrightarrow & (X \otimes X)^2 & \longrightarrow & X^2.
 \end{array}$$

Essentially:

$$x_1 R x_2 \quad \text{entails} \quad x_1[\sigma] R x_2[\sigma].$$

Substitution-closed bisimulation **relation** = applicative bisimulation.

# Substitution-closed bisimilarity

## Proposition

For any  $X \in \Sigma_0\text{-Mon}$ , there is a terminal substitution-closed bisimulation,  $\sim_X^\otimes$ , called **substitution-closed bisimilarity**.

Relevance:

- Recall  $\mathbf{Z}$ , the initial model.
- One can prove that  $\sim_{\mathbf{Z}}^\otimes$  coincides with the relation originally considered by Howe: **open extension of applicative bisimilarity**.

# Generalisation

cbn $\lambda$	general case
$\mathbb{F} \hookrightarrow \mathbb{F}^\Downarrow$	“two-level” category $\mathbb{C}_0 \hookrightarrow \mathbb{C}$
$\otimes$	monoidal structure on $\widehat{\mathbb{C}_0}$
$\mathbf{y}_0 \rightarrow \mathbf{y}_\Downarrow \leftarrow \mathbf{y}_1$	“border inclusions” from level 0
$\Sigma_0$	any “pointed strong” endofunctor
...	...

# Main result

## Theorem

For any *suitable* signature  $(\Sigma_0, \Sigma_1)$ , substitution-closed bisimilarity on the initial model  $(\sim_{\mathbf{Z}}^{\otimes})$  is a congruence.

$$\begin{array}{ccccc}
 \Sigma_0(\sim_{\mathbf{Z}}^{\otimes}) & \xrightarrow{\quad \quad \quad} & \sim_{\mathbf{Z}}^{\otimes} & & \\
 \downarrow & & \downarrow & & \\
 \Sigma_0(\mathbf{Z}^2) & \longrightarrow & \Sigma_0(\mathbf{Z})^2 & \longrightarrow & \mathbf{Z}^2
 \end{array}$$

Essentially:

$$e_1 \sim_{\mathbf{Z}}^{\otimes} e'_1, \dots, e_n \sim_{\mathbf{Z}}^{\otimes} e'_n \quad \text{entails} \quad op(e_1, \dots, e_n) \sim_{\mathbf{Z}}^{\otimes} op(e'_1, \dots, e'_n).$$

# What's suitable?

## Lemma

$\widehat{\mathbb{F}}_{\Downarrow}$  is isomorphic to the category of triples  $(X_0 \in \widehat{\mathbb{F}}, X_1 \in \mathbf{Set}, \partial_X)$ , where

$$\begin{array}{c} X_{\Downarrow} \\ \downarrow \partial_X \\ X_0(0) \times X_0(1). \end{array}$$

$\Sigma_0\text{-Mon}$ : same with  $\Sigma_0$ -monoid structure on  $X_0$ .

# What's suitable?

## Definition

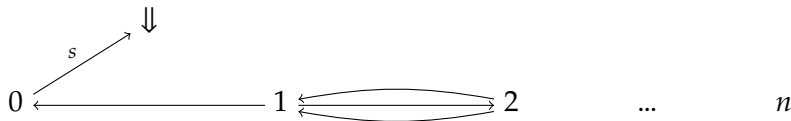
The functor  $\check{\Sigma}_1$  is **suitable** iff it may be decomposed as

$$\begin{array}{ccccc}
 X_{\Downarrow} & \xrightarrow{\Sigma_1} & \Sigma_1(X)_{\Downarrow} & \xrightarrow{\quad} & \Sigma_1(X)_{\Downarrow} \\
 \downarrow & & \downarrow & & \downarrow \\
 X_0(0) \times X_0(1) & & \Sigma_0(X_0)(0) \times X_0(1) & & \Sigma_0(X_0)(0) \times X_0(1) \\
 & & & & \downarrow \\
 & & & & X_0(0) \times X_0(1)
 \end{array}$$

such that  $\Sigma_1$  **preserves functional bisimulations**.

# Rigorous definition

## The category $\mathbb{F}_s$



Dynamic signatures  $\check{\Sigma}_1$  in fact defined to induce

$$\begin{array}{ccc}
 \Sigma_0\text{-}\mathbf{Mon} & \xrightarrow{\Sigma_1} & \widehat{\mathbb{F}}_s \\
 \downarrow & & \downarrow \\
 \widehat{\mathbb{F}} & \xrightarrow{\Sigma_0} & \widehat{\mathbb{F}}
 \end{array}$$

## Definition

Functional bisimulation in  $\widehat{\mathbb{F}}_s$ :  $\{s\}^\square$ .



# Why is cbn $\lambda$ suitable?

## Lemma

If  $\Sigma_1$  is *familial* then

$$\text{suitable} \iff \text{cellular}.$$

- **Cellular**  $\approx$  input arities of rules are in  $\mathbb{N}(\{s\}^{\mathbb{N}})$   
 $=$  input arities of rules are **functional cobisimulations**.
- Let us see why cbn  $\lambda$  input arities are cellular.

# Input arity

By example: 
$$\frac{e_1 \Downarrow e_3 \quad e_3[e_2] \Downarrow e_4}{e_1 \ e_2 \Downarrow e_4}$$

## Goal

Find  $E_\beta$  such that  $\Sigma_0\text{-Mon}(E_\beta, X) \cong A_\beta(X)$ , naturally in  $X$ .

$$\begin{array}{ccc}
 \mathcal{L}(\mathbf{y}_0) & \xrightarrow{\mathcal{L}(\mathbf{y}_s)} & \mathcal{L}(\mathbf{y}_\Downarrow) \\
 (k_\Downarrow \cdot t)[k_0] \downarrow & & \downarrow \\
 \mathcal{L}(\mathbf{y}_\Downarrow + \mathbf{y}_0) & \xrightarrow{\quad} & E_\beta
 \end{array}
 \qquad
 \begin{array}{ccc}
 A_\beta(X) & \xrightarrow{\quad} & X(\Downarrow) \\
 \downarrow & \lrcorner & \downarrow x_\Downarrow \mapsto x_\Downarrow \cdot s \\
 X(\Downarrow) \times X(0) & \xrightarrow{x_\Downarrow, x_0 \mapsto (x_\Downarrow \cdot t)[x_0]} & X(0)
 \end{array}$$

## Cellularity

The composite  $\mathcal{L}(\mathbf{y}_0 + \mathbf{y}_0) \rightarrow \mathcal{L}(\mathbf{y}_\Downarrow + \mathbf{y}_0) \rightarrow E_\beta$  is in  ${}^\square(\{s\}^\square)$ .

# Cellularity for cbn $\lambda$ -calculus

## Lemma

*Stability properties for functional cobisimulations.*

- *Contain  $s$  and all isomorphisms.*
- *Closed under (transfinite) composition.*
- *Closed under pushouts.*
- *Closed under retracts.*

$$\begin{array}{ccccc}
 \mathcal{L}(\mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow}) & & \\
 \downarrow & & \downarrow & & \\
 \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow} + \mathbf{y}_0) & \longrightarrow & E_{\beta}
 \end{array}$$

# Cellularity for cbn $\lambda$ -calculus

## Lemma

*Stability properties for functional cobisimulations.*

- Contain  $s$  and all isomorphisms.
- Closed under (transfinite) composition.
- Closed under pushouts.
- Closed under retracts.

$$\begin{array}{ccccc}
 \mathcal{L}(\mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow}) & & \\
 \downarrow & & \downarrow & \lrcorner & \\
 \mathcal{L}(\mathbf{y}_0 + \mathbf{y}_0) & \longrightarrow & \mathcal{L}(\mathbf{y}_{\Downarrow} + \mathbf{y}_0) & \longrightarrow & E_{\beta}
 \end{array}$$

# Summary

Semantic format for congruence of substitution-closed bisimilarity

Input arities should be functional cobisimulations.

- Shown here: example of  $\text{cbn } \lambda$ .
- In the paper:  $\text{cbv } \lambda$ .

# Perspectives

- More examples!
- Other kinds of bisimilarity: normal form, environmental, contextual,...  
(+ weak variants)
- Other kinds of results: type soundness, compiler correctness...