A Story of λ-Calculus and Approximation

Axel Kerinec

Supervised by **Giulio Manzonetto**
University Sorbonne Paris-Nord

A Program

```
def _push(self, left, right, epel=False):
    mid = (self._words[left][0] + self._words[right][0]) / 2.0
    intervalSize = self._words[right][0] - self._words[left][0]
    if True: # True to use random interval
        eps = 0.0
        if epel == False:
            eps = self._EPSILON_2
        else:
            eps = self._EPSILON_1
        randomization = random.uniform(-1, 1) * eps
        frequency = mid + intervalSize * randomization
        cursor = self._findWordIndexFromFrequency(frequency)
    else: # to activate highest frequency in interval
        print('\n\nFREQ FREQ\n\n')
        boundLeft = self._findWordIndexFromFrequency(mid - intervalSize * self._EPSILON2/2)
        boundRight = self._findWordIndexFromFrequency(mid + intervalSize * self._EPSILON2/2)
        print(boundLeft,boundRight)
        print(self._words[boundLeft], self._words[boundRight])
        mobileCursor = boundLeft
        cursor = mobileCursor
        bestFreq = 0
        while mobileCursor <= boundRight:
            tmpFreq = self._words[mobileCursor][0]-self._words[mobileCursor-1][0]
            if tmpFreq > bestFreq:
                bestFreq = tmpFreq
                cursor = mobileCursor
            mobileCursor+=1
    self._stack.append(_State(left, cursor, right))
```

The λ-Calculus [Church 32]

$$(\Lambda) \qquad M, N ::= x \mid \lambda x.M \mid (MN)$$

$$(\Lambda) \qquad M, N ::= x \mid \lambda x.M \mid (MN)$$

$$(\lambda x.M)N \mapsto_\beta M\{N/x\}$$

The λ-Calculus [Church 32]

$$(\Lambda) \qquad M, N ::= x \mid \lambda x.M \mid (MN)$$

$$(\lambda x.M)N \mapsto_\beta M\{N/x\}$$

$$\begin{array}{ccccccccc} f(x) & = & x^2 & - & 3x & + & 42 \\ f(4) & = & 4^2 & - & 3 \times 4 & + & 42 \end{array}$$

$$(\Lambda) \qquad M, N ::= x \mid \lambda x.M \mid (MN)$$

$$(\lambda x.M)N \mapsto_\beta M\{N/x\}$$

$$\begin{array}{rcccccc} f(x) & = & x^2 & - & 3x & + & 42 \\ f(4) & = & 4^2 & - & 3 \times 4 & + & 42 \end{array}$$

$$(\lambda x.(x^2 - 3x + 42))4 \to_\beta 4^2 - 3 \times 4 + 42$$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \to_\beta x$$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \to_\beta x$$

$$\Delta = \lambda x.xx \qquad \mathbf{I}(\Delta(xx)) \to_\beta \Delta(xx) \to_\beta (xx)(xx)$$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \to_\beta x$$

$$\Delta = \lambda x.xx \qquad \mathbf{I}(\Delta(xx)) \to_\beta \Delta(xx) \to_\beta (xx)(xx)$$

$$\Omega = (\lambda x.xx)(\lambda x.xx) \qquad \Omega \to_\beta \Omega \to_\beta \cdots \to_\beta \Omega \to_\beta \cdots$$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \to_\beta x$$

$$\Delta = \lambda x.xx \qquad \mathbf{I}(\Delta(xx)) \to_\beta \Delta(xx) \to_\beta (xx)(xx)$$

$$\Omega = (\lambda x.xx)(\lambda x.xx) \qquad \Omega \to_\beta \Omega \to_\beta \cdots \to_\beta \Omega \to_\beta \cdots$$

$$\begin{aligned} \mathbf{Y} = \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \qquad \mathbf{Y} &\to_\beta \lambda f.(f((\lambda x.f(xx))(\lambda x.f(xx)))) \\ &\to_\beta \lambda f.(f(f((\lambda x.f(xx))(\lambda x.f(xx))))) \\ &\twoheadrightarrow_\beta \lambda f.(f(f(f(\cdots)))) \end{aligned}$$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \to_\beta x$$

$$\Delta = \lambda x.xx \qquad \mathbf{I}(\Delta(xx)) \to_\beta \Delta(xx) \to_\beta (xx)(xx)$$

$$\Omega = (\lambda x.xx)(\lambda x.xx) \qquad \Omega \to_\beta \Omega \to_\beta \cdots \to_\beta \Omega \to_\beta \cdots$$

$$\begin{aligned}\mathbf{Y} = \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \qquad \mathbf{Y} &\to_\beta \lambda f.(f((\lambda x.f(xx))(\lambda x.f(xx))))\\ &\to_\beta \lambda f.(f(f((\lambda x.f(xx))(\lambda x.f(xx)))))\\ &\twoheadrightarrow_\beta \lambda f.(f(f(f(\cdots))))\end{aligned}$$

M is solvable: $\exists x_1, \ldots x_n, M_1, \ldots, M_k$ *s.t.* $(\lambda x_1 \ldots x_n.M)M_1 \cdots M_k \twoheadrightarrow_\beta \mathbf{I}$

Normal Forms

$$\mathbf{I} = \lambda x.x \qquad \mathbf{I}x \rightarrow_\beta x$$

$$\Delta = \lambda x.xx \qquad \mathbf{I}(\Delta(xx)) \rightarrow_\beta \Delta(xx) \rightarrow_\beta (xx)(xx)$$

$$\Omega = (\lambda x.xx)(\lambda x.xx) \qquad \Omega \rightarrow_\beta \Omega \rightarrow_\beta \cdots \rightarrow_\beta \Omega \rightarrow_\beta \cdots$$

$$\begin{aligned}\mathbf{Y} = \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \qquad \mathbf{Y} &\rightarrow_\beta \lambda f.(f((\lambda x.f(xx))(\lambda x.f(xx))))\\ &\rightarrow_\beta \lambda f.(f(f((\lambda x.f(xx))(\lambda x.f(xx)))))\\ &\twoheadrightarrow_\beta \lambda f.(f(f(f(\cdots))))\end{aligned}$$

M is solvable: $\exists x_1, \ldots x_n, M_1, \ldots, M_k$ *s.t.* $(\lambda x_1 \ldots x_n.M)M_1 \cdots M_k \twoheadrightarrow_\beta \mathbf{I}$

Head Normal Form Theorem [Wadsworth 76]

M solvable iff $M \twoheadrightarrow_\beta \lambda x_1 \ldots x_n.yM_1 \cdots M_k$

Program Approximation [Barendregt 77]

Böhm tree of M

- If $M \twoheadrightarrow_\beta \lambda x_1 \ldots x_n . y\, M_1 \cdots M_k$ then

$$\mathrm{BT}_\beta(M) = \lambda x_1 \ldots x_n . y$$

$$\mathrm{BT}_\beta(M_1) \quad \cdots \quad \mathrm{BT}_\beta(M_k)$$

- Otherwise

$$\mathrm{BT}_\beta(M) = \bot$$

$$\begin{aligned} \mathbf{Y} &= \lambda f.(\lambda x.f(xx))(\lambda x.f(xx)) \\ &\twoheadrightarrow_\beta \lambda f.(f(f(\cdots))) \end{aligned}$$

$$\mathrm{BT}_\beta(\mathbf{Y})$$

$$\shortparallel$$

$$\lambda f$$

$$|$$

$$f$$

$$|$$

$$f$$

$$\vdots$$

Taylor Expansion

Function	
$f(x) = \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(a)(x-a)^n$	

Taylor Expansion

Function	*λ-calculus*
$f(x) = \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(a)(x-a)^n$	$(\lambda x.M)N = \sum_{n=0}^{\infty} \frac{1}{n!} (\lambda x.M)[\underbrace{N, \ldots, N}_{n}]$

A differential λ-calculus:

- resource-sensitive: in $(\lambda x.M)N$ can only replace one occurrence of x
- strongly normalising: each resource term has a normal form

Taylor Expansion

Function	*λ-calculus*
$f(x) = \sum_{n=0}^{\infty} \frac{1}{n!} f^{(n)}(a)(x-a)^n$	

A differential λ-calculus:

- resource-sensitive: in $(\lambda x.M)N$ can only replace one occurrence of x
- strongly normalising: each resource term has a normal form

Link with Böhm Tree [Ehrhard and Regnier 08]

$$\mathrm{NF}(\mathsf{T}(M)) = \mathsf{T}(\mathrm{BT}_\beta(M))$$

Denotational Semantics

Scott Continuous Semantics

A Program $\Gamma \vdash M : a$ is a continuous map $[\![M]\!] : [\![\Gamma]\!] \to [\![a]\!]$.

Denotational Semantics

Scott Continuous Semantics

A Program $\Gamma \vdash M : a$ is a continuous map $[\![M]\!] : [\![\Gamma]\!] \to [\![a]\!]$.

Relational Semantics

Quantitative informations:

- Number of steps to termination,
- Amount of resources used during the computation,

$\vdots$

Intersection Types [Coppo and Dezani 80]

$$a, b ::= o \mid a \multimap b$$

Denotational Semantics

Intersection Types [Coppo and Dezani 80]

$$a, b ::= o \mid a \multimap b \mid (a_1 \cap \cdots \cap a_k)$$

$$\frac{\Gamma \vdash M : a \qquad \Delta \vdash M : b}{\Gamma + \Delta \vdash M : a \cap b}$$

Denotational Semantics

Intersection Types [Coppo and Dezani 80]

$$a, b ::= o \mid a \multimap b \mid (a_1 \cap \cdots \cap a_k)$$

$$\frac{\Gamma \vdash M : a \qquad \Delta \vdash M : b}{\Gamma + \Delta \vdash M : a \cap b}$$

The intersection operator $\cap$ may be idempotent or not.

Denotational Semantics

Intersection Types [Coppo and Dezani 80]

$$a, b ::= o \mid a \multimap b \mid (a_1 \cap \cdots \cap a_k)$$

$$\frac{\Gamma \vdash M : a \qquad \Delta \vdash M : b}{\Gamma + \Delta \vdash M : a \cap b}$$

The intersection operator $\cap$ may be idempotent or not.

Interpretation

$$[\![M]\!]_{\vec{x}} = \{(\Gamma, a) \mid \ \Gamma \vdash M : a\}$$

Denotational Semantics

Intersection Types [Coppo and Dezani 80]

$$a, b ::= o \mid a \multimap b \mid (a_1 \cap \cdots \cap a_k)$$

$$\frac{\Gamma \vdash M : a \qquad \Delta \vdash M : b}{\Gamma + \Delta \vdash M : a \cap b}$$

The intersection operator $\cap$ may be idempotent or not.

Interpretation

$$[\![M]\!]_{\vec{x}} = \{(\Gamma, a) \mid \Gamma \vdash M : a\}$$

Looking for an Approximation Theorem:

$$[\![M]\!]_{\vec{x}} = \bigsqcup_{A \in \mathcal{A}_\beta(M)} [\![A]\!]_{\vec{x}}$$

Categorification

Set-Theoretic	Category-Theoretic
sets	small categories
functions	functors
equations	isomorphisms

Categorification

Set-Theoretic	Category-Theoretic
sets	small categories
functions	functors
equations	isomorphisms

Relations *become* *Distributors*

$r : A \times B \to \{0, 1\}$ $R : A^{op} \times B \to \mathrm{Set}$

Categorification

Set-Theoretic	Category-Theoretic
sets	small categories
functions	functors
equations	isomorphisms

Relations *become* *Distributors*

$r : A \times B \to \{0, 1\}$ $\quad$ $R : A^{op} \times B \to \mathrm{Set}$

Bicategory of Symmetric Categorical Sequences

Kleisli bicategory of the pseudocomonad ! on the bicategory of distributors

Categorified Graph Model

Relational Graph Model [Manzonetto and Ruoppolo 14]

A is a set U with an injection $\iota : \mathsf{M_f}(U) \times U \hookrightarrow U$.

Arrow type:

$$(a_1 \cap \cdots \cap a_k) \multimap a := \iota([a_1, \ldots, a_k], a)$$

Categorified Graph Model

Relational Graph Model [Manzonetto and Ruoppolo 14]

A is a set U with an injection $\iota : \mathsf{M_f}(U) \times U \hookrightarrow U$.

Arrow type:

$$(a_1 \cap \cdots \cap a_k) \multimap a := \iota([a_1, \ldots, a_k], a)$$

Theory of Böhm Trees [Breuvart, Manzonetto and Ruoppolo 18]

$$\mathrm{BT}_\beta(M) = \mathrm{BT}_\beta(N) \iff [\![M]\!]^U = [\![N]\!]^U, \text{ for some } U$$

Categorified Graph Model

Relational Graph Model [Manzonetto and Ruoppolo 14]

A is a set U with an injection $\iota : \mathrm{M_f}(U) \times U \hookrightarrow U$.

Arrow type:

$$(a_1 \cap \cdots \cap a_k) \multimap a := \iota([a_1, \ldots, a_k], a)$$

Categorified Graph Model (Def. 9.1.1)

A is a small category D with an embedding $\iota : !D^{\mathrm{op}} \times D \hookrightarrow D$.

Arrow type:

$$\langle a_1, \ldots, a_k \rangle \multimap a := \iota(\langle a_1, \ldots, a_k \rangle, a)$$

System $R_{\rightarrow}$ (Def. 9.2.1)

$$\frac{f : a' \to a}{x_1 : \langle\rangle, \ldots, x_i : \langle a' \rangle, \ldots, x_n : \langle\rangle \vdash x_i : a}$$

$$\frac{\Gamma, x : \vec{a} \vdash M : a \qquad f : (\vec{a} \multimap a) \to b}{\Gamma \vdash \lambda x.M : b}$$

$$\frac{\Gamma_0 \vdash M : \langle a_1, \ldots, a_k \rangle \multimap a \qquad (\Gamma_i \vdash N : a_i)_{i \in [k]}}{\Delta \vdash MN : a}$$

where $\eta : \Delta \to \sum_{j=0}^{k} \Gamma_j$

Interpretation

$$[\![M]\!]_{\vec{x}}(\Gamma, a) = \begin{cases} 1 & \text{if } \Gamma \vdash M : a \\ 0 & \text{otherwise} \end{cases}$$

Interpretation

$$[\![M]\!]_{\vec{x}} : (\overbrace{!D \times \cdots \times !D}^{\text{len}(\vec{x})\text{ times}})^{\mathrm{op}} \times D \to \mathrm{Set}$$

$$[\![M]\!]_{\vec{x}}(\Gamma, a) = \left\{ \begin{array}{c} \tilde{\pi} \\ \vdots \\ \vec{x} : \Gamma \vdash M : a \end{array} \right\}$$

Interpretation

$$[\![M]\!]_{\vec{x}} : (\overbrace{!D \times \cdots \times !D}^{\text{len}(\vec{x}) \text{ times}})^{\mathrm{op}} \times D \to \mathrm{Set}$$

$$[\![M]\!]_{\vec{x}}(\Gamma, a) = \left\{ \begin{array}{c} \tilde{\pi} \\ \vdots \\ \vec{x} : \Gamma \vdash M : a \end{array} \right\}$$

Soundness Theorem [Olimpieri 21]

If $M \to_\beta N$ then $[\![M \to_\beta N]\!]_{\vec{x}} : [\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}}$

Typed Reductions

Only some redexes are typed in derivations

$$\pi = \cfrac{\cfrac{f : a \to a'}{x : \langle\langle\rangle \multimap a\rangle \vdash x : \langle\rangle \multimap a'}}{x : \langle\langle\rangle \multimap a\rangle \vdash x(\mathbf{II})}$$

The redex $\mathbf{II} = (\lambda x.x)(\lambda x.x)$ is not typed in π.

Typed Reductions

Only some redexes are typed in derivations

$$\pi = \dfrac{\dfrac{f : a \to a'}{x : \langle\langle\rangle \multimap a\rangle \vdash x : \langle\rangle \multimap a'}}{x : \langle\langle\rangle \multimap a\rangle \vdash x(\mathbf{II})}$$

The redex $\mathbf{II} = (\lambda x.x)(\lambda x.x)$ is not typed in π.

- $\pi \in [\![M]\!]_{\vec{x}}(\Delta, a)$
- $M \to_\beta N$ contracting a redex of M typed in π
- $[\![M \to_\beta N]\!]_{\vec{x}}(\Delta, a)\pi = \pi' \in [\![N]\!]_{\vec{x}}(\Delta, a)$

Typed Reductions

Only some redexes are typed in derivations

$$\pi = \cfrac{\cfrac{f : a \to a'}{x : \langle\langle\rangle \multimap a\rangle \vdash x : \langle\rangle \multimap a'}}{x : \langle\langle\rangle \multimap a\rangle \vdash x(\mathbf{II})}$$

The redex $\mathbf{II} = (\lambda x.x)(\lambda x.x)$ is not typed in π.

- $\pi \in [\![M]\!]_{\vec{x}}(\Delta, a)$
- $M \to_\beta N$ contracting a redex of M typed in π
- $[\![M \to_\beta N]\!]_{\vec{x}}(\Delta, a)\pi = \pi' \in [\![N]\!]_{\vec{x}}(\Delta, a)$

The reduction on derivations is strongly normalising and confluent.

Typed Reductions

Only some redexes are typed in derivations

$$\pi = \cfrac{\cfrac{f : a \to a'}{x : \langle\langle\rangle \multimap a\rangle \vdash x : \langle\rangle \multimap a'}}{x : \langle\langle\rangle \multimap a\rangle \vdash x(\mathbf{II})}$$

The redex $\mathbf{II} = (\lambda x.x)(\lambda x.x)$ is not typed in π.

- $\pi \in [\![M]\!]_{\vec{x}}(\Delta, a)$
- $M \to_\beta N$ contracting a redex of M typed in π
- $[\![M \to_\beta N]\!]_{\vec{x}}(\Delta, a)\pi = \pi' \in [\![N]\!]_{\vec{x}}(\Delta, a)$

The reduction on derivations is strongly normalising and confluent.

Normalisation Theorem (Thm. 10.1.10)

$$\mathrm{Norm}_{\vec{x}}(M) : [\![M]\!]_{\vec{x}} \cong \mathrm{NF}([\![M]\!]_{\vec{x}})$$

Approximation Theorem

Minimal $\Lambda_\perp$-term for a Derivation

$$\dfrac{\dfrac{f : a' \to a}{x : \langle\langle\rangle \multimap a'\rangle \vdash x : \langle\rangle \multimap a}}{x : \langle\langle\rangle \multimap a'\rangle \vdash x\Omega : a}$$

$$A_\pi^x = x\bot$$

Approximation Theorem

Minimal $\Lambda_\perp$-term for a Derivation

$$\dfrac{\dfrac{f : a' \to a}{x : \langle\langle\rangle \multimap a'\rangle \vdash x : \langle\rangle \multimap a}}{x : \langle\langle\rangle \multimap a'\rangle \vdash x\Omega : a}$$

$$A^x_\pi = x\perp$$

$$\pi \in [\![M]\!]_{\vec{x}} \Rightarrow A_\pi \in \mathcal{A}_\beta(M)$$

Approximation Theorem

Minimal $\Lambda_\perp$-term for a Derivation

$$\dfrac{\dfrac{f : a' \to a}{x : \langle\langle\rangle \multimap a'\rangle \vdash x : \langle\rangle \multimap a}}{x : \langle\langle\rangle \multimap a'\rangle \vdash x\Omega : a}$$

$$A_\pi^x = x\perp$$

$$\pi \in [\![M]\!]_{\vec{x}} \Rightarrow A_\pi \in \mathcal{A}_\beta(M)$$

Commutation Theorem (Thm. 10.2.6)

$$\mathrm{NF}([\![M]\!]_{\vec{x}}) = [\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}}$$

Approximation Theorem

Minimal $\Lambda_\perp$-term for a Derivation

$$\dfrac{\dfrac{f : a' \to a}{x : \langle\langle\rangle \multimap a'\rangle \vdash x : \langle\rangle \multimap a}}{x : \langle\langle\rangle \multimap a'\rangle \vdash x\Omega : a}$$

$$A_\pi^x = x\perp$$

$$\pi \in [\![M]\!]_{\vec{x}} \Rightarrow A_\pi \in \mathcal{A}_\beta(M)$$

Commutation Theorem (Thm. 10.2.6)

$$\mathrm{NF}([\![M]\!]_{\vec{x}}) = [\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}}$$

Approximation Theorem (Thm. 10.2.7)

$$\mathsf{appr}_{\vec{x}}(M) : [\![M]\!]_{\vec{x}} \cong [\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}}$$

Approximation Theorem

Minimal $\Lambda_\perp$-term for a Derivation

$$\dfrac{\dfrac{f : a' \to a}{x : \langle\langle\rangle \multimap a'\rangle \vdash x : \langle\rangle \multimap a}}{x : \langle\langle\rangle \multimap a'\rangle \vdash x\Omega : a}$$

$$A_\pi^x = x\perp$$

$$\pi \in [\![M]\!]_{\vec{x}} \Rightarrow A_\pi \in \mathcal{A}_\beta(M)$$

Commutation Theorem (Thm. 10.2.6)

$$\mathrm{NF}([\![M]\!]_{\vec{x}}) = [\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}}$$

Approximation Theorem (Thm. 10.2.7)

$$\mathrm{appr}_{\vec{x}}(M) : [\![M]\!]_{\vec{x}} \cong [\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}}$$

Corollary: The model is sensible.

Theory of a Bicategorical Model

$$\mathrm{Th}(\mathcal{D}) = \{(M, N) \mid [\![M]\!]_{\vec{x}} = [\![N]\!]_{\vec{x}}\}$$

Theory of a Bicategorical Model

$$\mathrm{Th}(\mathcal{D}) = \{(M, N) \mid \theta : [\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}} \textit{ and } \theta \textit{ coherent wrt } \beta\textit{-normalization}\}$$

Theory of a Bicategorical Model

$\mathrm{Th}(\mathcal{D}) = \{(M, N) \mid \theta : [\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}} \textit{ and } \theta \textit{ coherent wrt } \beta\textit{-normalization}\}$

$$[\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}} \textit{ iff } \mathrm{BT}_\beta(M) = \mathrm{BT}_\beta(N)$$

(Thm. 11.2.4)

Proof.

($\Leftarrow$) By Approximation Theorem.

($\Rightarrow$) Assume $[\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}}$ and $\mathrm{BT}_\beta(M) \neq \mathrm{BT}_\beta(N)$:

- there is some $A \in \mathcal{A}_\beta(M) \setminus \mathcal{A}_\beta(N)$,
- so there is $\pi \in \mathrm{NF}([\![M]\!]_{\vec{x}}) = \mathrm{NF}([\![N]\!]_{\vec{x}})$ such that $A_\pi^{\vec{x}} = A$,
- and by definition $\pi \in [\![N']\!]_{\vec{x}}$ for some N' such that $N \twoheadrightarrow_\beta N'$.
- We obtain $A_\pi^{\vec{x}} = A \leq_\perp N'$, so $A \in \mathcal{A}_\beta(N)$. Contradiction. □

Reduction Strategies

In Call-by-Name λ-calculus:

$$(\lambda x.M)((\lambda y.N)L) \to_\beta ((\lambda y.N)L)[M/x] \to_\beta (L[M/x])[N[M/x]/y]$$

We want:

$$(\lambda x.M)((\lambda y.N)L) \to (\lambda x.M)(L[N/y]) \to (L[N/y])[M/x]$$

Call-by-Value λ-calculus [Plotkin 75]

$$
\begin{array}{lrcll}
(\Lambda^v) & V, U & ::= & x & \mid \lambda x.M \\
(\Lambda) & M, N & ::= & V & \mid (MN)
\end{array}
$$

$$(\lambda x.M)V \mapsto_{\beta_v} M\{V/x\}$$

Call-by-Value λ-calculus [Plotkin 75]

$$
\begin{array}{lrcll}
(\Lambda^v) & V, U & ::= & x & \mid \lambda x.M \\
(\Lambda) & M, N & ::= & V & \mid (MN)
\end{array}
$$

$$(\lambda x.M)V \mapsto_{\beta_v} M\{V/x\}$$

$L = (\lambda y.\Delta)(xx)\Delta = (\lambda y.(\lambda x.xx))(xx)(\lambda x.xx)$ in NF

Call-by-Value λ-calculus [Plotkin 75]

$$
\begin{array}{lrcl}
(\Lambda^v) & V, U & ::= & x \mid \lambda x.M \\
(\Lambda) & M, N & ::= & V \mid (MN)
\end{array}
$$

$$(\lambda x.M)V \mapsto_{\beta_v} M\{V/x\}$$

$L = (\lambda y.\Delta)(xx)\Delta = (\lambda y.(\lambda x.xx))(xx)(\lambda x.xx)$ in NF

Permutation Rules [Carraro and Guerrieri 14]

$$
\begin{array}{lcl}
(\lambda x.M)NN' & \mapsto_{\sigma_1} & (\lambda x.MN')N \\
V((\lambda x.M)N) & \mapsto_{\sigma_3} & (\lambda x.VM)N
\end{array}
$$

$L \rightarrow_{\sigma_3} (\lambda y.\Delta\Delta)(xx) \rightarrow_{\beta_v} (\lambda y.\Delta\Delta)(xx)$

A Call-by-Value Resource Calculus [Ehrhard 12]

Resource Expressions

$(r\Lambda^v)\quad u, v ::= x \mid \lambda x.t \qquad (r\Lambda^t)\quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

A Call-by-Value Resource Calculus [Ehrhard 12]

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \dots, v_l]$

$$[\lambda x.t][v_1, \dots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \dots, v_{f(k)}/x^k\} \text{ } if \ |m|_x = k \\ \emptyset \text{ } otherwise \end{cases}$$

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

$$[\lambda x.t][v_1, \ldots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \ldots, v_{f(k)}/x^k\} \ \textit{if} \ |m|_x = k \\ \emptyset \ \textit{otherwise} \end{cases}$$

$[\lambda x.x(xt_1)(\lambda y.t_2 t_3 x)][v_1, v_2, v_3]$

A Call-by-Value Resource Calculus [Ehrhard 12]

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad\qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

$$[\lambda x.t][v_1, \ldots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \ldots, v_{f(k)}/x^k\} \text{ if } |m|_x = k \\ \emptyset \text{ otherwise} \end{cases}$$

$$[\lambda x.x(x t_1)(\lambda y.t_2 t_3 x)][v_1, v_2, v_3] \quad \to \quad v_1(v_2 t_1)(\lambda y.t_2 t_3 v_3)$$

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

$$[\lambda x.t][v_1, \ldots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \ldots, v_{f(k)}/x^k\} \; \textit{if } |m|_x = k \\ \emptyset \; \textit{otherwise} \end{cases}$$

$$\begin{aligned} [\lambda x.x(xt_1)(\lambda y.t_2t_3x)][v_1, v_2, v_3] &\to v_1(v_2t_1)(\lambda y.t_2t_3v_3) \\ &\to v_2(v_3t_1)(\lambda y.t_2t_3v_1) \end{aligned}$$

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

$$[\lambda x.t][v_1, \ldots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \ldots, v_{f(k)}/x^k\} \ \mathit{if}\ |m|_x = k \\ \emptyset \ \mathit{otherwise} \end{cases}$$

$$[\lambda x.x(xt_1)(\lambda y.t_2t_3x)][v_1, v_2, v_3] \quad \to \quad \{v_2(v_3t_1)(\lambda y.t_2t_3v_1), \ldots, v_1(v_2t_1)(\lambda y.t_2t_3v_3)\}$$

A Call-by-Value Resource Calculus [Ehrhard 12]

Resource Expressions

$(r\Lambda^v) \quad u, v ::= x \mid \lambda x.t \qquad\qquad (r\Lambda^t) \quad t, s ::= (ts) \mid [v_1, \ldots, v_l]$

$$[\lambda x.t][v_1, \ldots, v_k] \mapsto_{\beta_v} \begin{cases} \sum_{f \in \mathfrak{S}_k} t\{v_{f(1)}/x^1, \ldots, v_{f(k)}/x^k\} \ \textit{if}\ |m|_x = k \\ \emptyset \ \textit{otherwise} \end{cases}$$

$$\begin{array}{lll} [v_1, \ldots v_k]t & \mapsto_0 & \emptyset \qquad \textit{if}\ k \neq 1 \\ [\lambda x.t]ss' & \mapsto_{\sigma_1} & [\lambda x.ts']s \\ [v]([\lambda x.t]s) & \mapsto_{\sigma_3} & [\lambda x.[v]t]s \end{array}$$

Taylor Expansion

Taylor Expansion

$$
\begin{array}{lcl}
\mathsf{T}(x) & = & \{[x^k] \mid k \geq 0\} \\
\mathsf{T}(\lambda x.M) & = & \{[\lambda x.m_1, \ldots, \lambda x.m_k] \mid k \geq 0, m_1, \ldots, m_k \in \mathsf{T}(M)\} \\
\mathsf{T}(M_1 M_2) & = & \{m_1 m_2 \mid m_1 \in \mathsf{T}(M_1), m_2 \in \mathsf{T}(M_2)\}
\end{array}
$$

Taylor Expansion

Taylor Expansion

$$
\begin{array}{lcl}
\mathsf{T}(x) & = & \{[x^k] \mid k \geq 0\} \\
\mathsf{T}(\lambda x.M) & = & \{[\lambda x.m_1, \ldots, \lambda x.m_k] \mid k \geq 0, m_1, \ldots, m_k \in \mathsf{T}(M)\} \\
\mathsf{T}(M_1 M_2) & = & \{m_1 m_2 \mid m_1 \in \mathsf{T}(M_1), m_2 \in \mathsf{T}(M_2)\}
\end{array}
$$

$$
\mathsf{T}(\Delta) = \{[\lambda x.[x^{n_1}][x^{m_1}], \ldots, \lambda x.[x^{n_k}][x^{m_k}]] \mid k \geq 0, \forall i \leq k, m_i, n_i \geq 0\}
$$

Taylor Expansion

Taylor Expansion

$$\begin{array}{lcl} \mathsf{T}(x) & = & \{[x^k] \mid k \geq 0\} \\ \mathsf{T}(\lambda x.M) & = & \{[\lambda x.m_1, \ldots, \lambda x.m_k] \mid k \geq 0, m_1, \ldots, m_k \in \mathsf{T}(M)\} \\ \mathsf{T}(M_1 M_2) & = & \{m_1 m_2 \mid m_1 \in \mathsf{T}(M_1), m_2 \in \mathsf{T}(M_2)\} \end{array}$$

$$NF(\mathsf{T}(M)) = \bigcup_{t \in \mathsf{T}(M)} NF_r(t).$$

Normalisation Theorem

If $M =_{\beta_v} N$ *then* $NF(\mathsf{T}(M)) = NF(\mathsf{T}(N))$

Taylor Expansion

Taylor Expansion

$$\begin{array}{lcl} \mathsf{T}(x) & = & \{[x^k] \mid k \geq 0\} \\ \mathsf{T}(\lambda x.M) & = & \{[\lambda x.m_1, \ldots, \lambda x.m_k] \mid k \geq 0, m_1, \ldots, m_k \in \mathsf{T}(M)\} \\ \mathsf{T}(M_1 M_2) & = & \{m_1 m_2 \mid m_1 \in \mathsf{T}(M_1), m_2 \in \mathsf{T}(M_2)\} \end{array}$$

$$NF(\mathsf{T}(M)) = \bigcup_{t \in \mathsf{T}(M)} NF_r(t).$$

Normalisation Theorem

If $M =_{\beta_v} N$ *then* $NF(\mathsf{T}(M)) = NF(\mathsf{T}(N))$

Context Lemma

If $\mathrm{NF}(\mathsf{T}(M)) = \mathrm{NF}(\mathsf{T}(N))$ then $\forall C(\!|-|\!), \mathrm{NF}(\mathsf{T}(C(\!|M|\!))) = \mathrm{NF}(\mathsf{T}(C(\!|N|\!)))$

$\perp$ represents an undefined value

$$
\begin{array}{llcl}
(\Lambda_\perp) & M, N & ::= & V \mid MN \\
(\Lambda^v_\perp) & V, U & ::= & \perp \mid x \mid \lambda x.M
\end{array}
$$

$\bot$ represents an undefined value

$$\begin{array}{llcl} (\Lambda_\bot) & M, N & ::= & V \mid MN \\ (\Lambda_\bot^v) & V, U & ::= & \bot \mid x \mid \lambda x.M \end{array}$$

$$\forall V \in \Lambda^V, \bot \sqsubseteq_\bot V$$

$\perp$ represents an undefined value

$$
\begin{array}{lrcl}
(\Lambda_\perp) & M, N & ::= & V \mid MN \\
(\Lambda^v_\perp) & V, U & ::= & \perp \mid x \mid \lambda x.M
\end{array}
$$

$$\forall V \in \Lambda^V, \perp \sqsubseteq_\perp V$$

Approximants :

$$
\begin{array}{lrcl}
(\mathcal{A}) & A & ::= & H \mid R \\
 & H & ::= & \perp \mid x \mid \lambda x.A \mid xHA_1 \cdots A_n \\
 & R & ::= & (\lambda x.A)(yHA_1 \cdots A_n)
\end{array}
$$

$\perp$ represents an undefined value

$$\begin{array}{llcl} (\Lambda_\perp) & M, N & ::= & V \mid MN \\ (\Lambda_\perp^v) & V, U & ::= & \perp \mid x \mid \lambda x.M \end{array}$$

$$\forall V \in \Lambda^V, \perp \sqsubseteq_\perp V$$

Approximants :

$$\begin{array}{llcl} (\mathcal{A}) & A & ::= & H \mid R \\ & H & ::= & \perp \mid x \mid \lambda x.A \mid xHA_1 \cdots A_n \\ & R & ::= & (\lambda x.A)(yHA_1 \cdots A_n) \end{array}$$

$$\mathcal{A}(M) = \{A \in \mathcal{A} \mid \exists N \in \Lambda \,.\, M \twoheadrightarrow_v N \text{ and } A \sqsubseteq_\perp N\}$$

Call-by-Value Böhm Tree (Sec. 3.2.1)

$\bot$ represents an undefined value

$$
\begin{array}{llcl}
(\Lambda_\bot) & M, N & ::= & V \mid MN \\
(\Lambda_\bot^v) & V, U & ::= & \bot \mid x \mid \lambda x.M
\end{array}
$$

$$\forall V \in \Lambda^V, \bot \sqsubseteq_\bot V$$

Approximants :

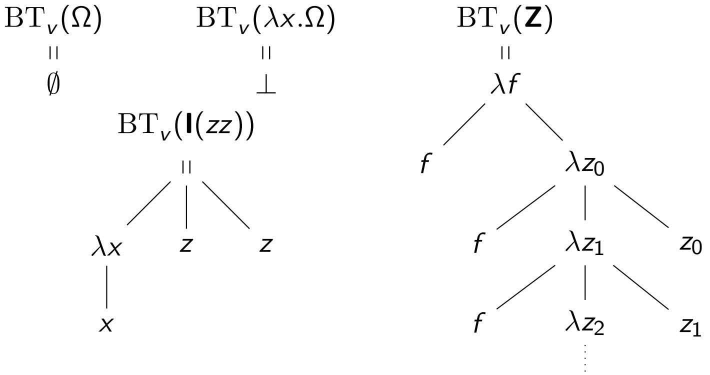
$$
\begin{array}{llcl}
(\mathcal{A}) & A & ::= & H \mid R \\
 & H & ::= & \bot \mid x \mid \lambda x.A \mid xHA_1 \cdots A_n \\
 & R & ::= & (\lambda x.A)(yHA_1 \cdots A_n)
\end{array}
$$

$$\mathcal{A}(M) \;=\; \{A \in \mathcal{A} \mid \exists N \in \Lambda \,.\, M \twoheadrightarrow_v N \text{ and } A \sqsubseteq_\bot N\}$$

$$\mathcal{BT}(M) \;=\; \bigsqcup \mathcal{A}(M)$$

$$\Delta = \lambda x.xx \qquad \Omega = \Delta\Delta \qquad \mathbf{L} = \lambda y.f(\lambda z.yyz)$$

$$\mathbf{Z} = \lambda f.\mathbf{LL} \to_v \lambda f.(f(\lambda z.\mathbf{LL}z)) \to_v \lambda f.(f(\lambda z.(f(\lambda z.\mathbf{LL}z)z))) \to_v \cdots$$



Taylor Expansion and Böhm Trees

$$\mathsf{T}^{\circ}(\mathrm{BT}_{v}(M)) = NF(\mathsf{T}(M))$$

(Thm. 5.1.8)

Characterisations of Valuability and Potential Valuability

M is:

	valuable	potentially valuable
Def	if $\exists V$ such that $M \twoheadrightarrow_v V$	if $\exists x_1, \dots x_n, \exists V_1, \dots V_l$ s.t. $(\lambda x_1 \dots x_n.M)V_1 \cdots V_l$ valuable
Thm	iff $\bot \in \mathcal{A}(M)$	iff $\mathcal{A}(M) \neq \emptyset$ (Thm. 5.2.4)

More Precise Approximants

M is solvable: $\exists x_1, \ldots x_n, V_1, \ldots, V_k$ s.t. $(\lambda x_1 \ldots x_n.M)V_1 \cdots V_k \twoheadrightarrow_{\beta_v} I$

More Precise Approximants

M is solvable: $\exists x_1, \ldots x_n, V_1, \ldots, V_k$ s.t. $(\lambda x_1 \ldots x_n.M) V_1 \cdots V_k \twoheadrightarrow_{\beta_v} I$

$$
\begin{array}{lcl}
S & ::= & H' \mid R' \\
H' & ::= & x \mid \lambda x.S \mid xHA_1 \cdots A_n \\
R' & ::= & (\lambda x.S)(yHA_1 \cdots A_n)
\end{array}
\qquad
\begin{array}{lcl}
U & ::= & \bot \mid \lambda x.U \\
 & \mid & (\lambda x.U)(yHA_1 \cdots A_n)
\end{array}
$$

More Precise Approximants

M is solvable: $\exists x_1, \ldots x_n, V_1, \ldots, V_k$ s.t. $(\lambda x_1 \ldots x_n.M)V_1 \cdots V_k \twoheadrightarrow_{\beta_v} I$

$$
\begin{array}{lcl}
S & ::= & H' \mid R' \\
H' & ::= & x \mid \lambda x.S \mid xHA_1 \cdots A_n \\
R' & ::= & (\lambda x.S)(yHA_1 \cdots A_n)
\end{array}
\qquad
\begin{array}{lcl}
U & ::= & \bot \mid \lambda x.U \\
 & \mid & (\lambda x.U)(yHA_1 \cdots A_n)
\end{array}
$$

Characterisation of Solvability (Thm. 5.3.6)

M is solvable iff $\exists A$ such that $A \in \mathcal{A}(M) \cap \mathcal{S}$

Corollary: M is unsolvable iff $\mathcal{A}(M) \subseteq \mathcal{U}$.

Adequacy

Observational equivalence: $M \equiv N$ *iff* $\forall C(\!|-|\!)$,

$$\exists V \in \Lambda^V,\ C(\!|M|\!) \twoheadrightarrow_{\beta_v} V \iff \exists U \in \Lambda^V,\ C(\!|N|\!) \twoheadrightarrow_{\beta_v} U$$

Adequacy

Observational equivalence: $M \equiv N$ *iff* $\forall C(\!|-|\!),$

$$\exists V \in \Lambda^V,\ C(\!|M|\!) \twoheadrightarrow_{\beta_\mathsf{v}} V \iff \exists U \in \Lambda^V,\ C(\!|N|\!) \twoheadrightarrow_{\beta_\mathsf{v}} U$$

Adequacy Theorem (Thm. 5.2.6)

If $\mathrm{BT}_v(M) = \mathrm{BT}_v(N)$ then $M \equiv N$

Adequacy

Observational equivalence: $M \equiv N$ *iff* $\forall C(\!|-|\!)$,
$\exists V \in \Lambda^V,\ C(\!|M|\!) \twoheadrightarrow_{\beta_v} V \iff \exists U \in \Lambda^V,\ C(\!|N|\!) \twoheadrightarrow_{\beta_v} U$

Adequacy Theorem (Thm. 5.2.6)

If $\mathrm{BT}_v(M) = \mathrm{BT}_v(N)$ then $M \equiv N$

Not fully abstract: $\Delta(yy)$ and $yy(yy)$ equivalent observationally but $\mathrm{BT}_v(\Delta(yy)) = \Delta(yy)$ and $\mathrm{BT}_v(yy(yy)) = yy(yy)$

Conclusion

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory

$$\mathrm{Th}(\mathcal{D}) = \{(M, N) \mid \theta : [\![M]\!]_{\vec{x}} \cong [\![N]\!]_{\vec{x}} \textit{ and } \theta \textit{ coherent wrt } \beta\textit{-normalization}\}$$

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion
  - Characterisation of Potential Valuability

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion
  - Characterisation of Potential Valuability
  - Adequacy Theorem

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion
  - Characterisation of Potential Valuability
  - Adequacy Theorem
- Call-by-Value Again!

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion
  - Characterisation of Potential Valuability
  - Adequacy Theorem
- Call-by-Value Again!
  - Characterisation of CbV Solvability

Conclusion

- Why Are Proofs Relevant in Proof-Relevant Models?
  - Categorified Graph Models
  - Approximation Theorem
  - Characterisation of the Theory
- Revisiting CbV Böhm Trees in the Light of their Taylor Expansion
  - CbV Böhm Trees
  - Link with Taylor Expansion
  - Characterisation of Potential Valuability
  - Adequacy Theorem
- Call-by-Value Again!
  - Characterisation of CbV Solvability

Not fully abstract: $\Delta(yy)$ and $yy(yy)$ equivalent observationally but

$$\mathrm{BT}_v(\Delta(yy)) = \Delta(yy) \textit{ and } \mathrm{BT}_v(yy(yy)) = yy(yy)$$

The End

Merci beaucoup pour votre attention!

Extend Notions to Böhm Trees

$$[\![\bot]\!]_{\vec{x}} = \bot_{D^{\&n},D}$$

Lemma: If $L \leq_\bot P$ then $[\![L]\!]_{\vec{x}} \subseteq [\![P]\!]_{\vec{x}}$

Consider $\langle \mathcal{A}_\beta(M), \leq_\bot \rangle$,

$$\begin{aligned} [\![-]\!]_{\vec{x}} : \mathcal{A}_\beta(M) &\to \mathrm{Dist}(!(D^{\&n}), D) \\ A &\mapsto [\![A]\!]_{\vec{x}}, \\ A \leq_\bot A' &\mapsto [\![A]\!]_{\vec{x}} \subseteq [\![A']\!]_{\vec{x}}. \end{aligned}$$

Interpretation of the Böhm Tree

$$[\![\mathrm{BT}_\beta(M)]\!]_{\vec{x}} = lim_{A \in \mathcal{A}_\beta(M)} [\![A]\!]_{\vec{x}}$$

Congruence

$$\dfrac{\begin{matrix}\pi_0\\ \vdots\\ \Gamma_0 \vdash \vec{b} \multimap a\end{matrix} \qquad \left(\begin{matrix}[f_i]\pi_{\sigma^{-1}(i)}\\ \vdots\\ \Gamma_{\sigma^{-1}(i)} \vdash b_i\end{matrix}\right)_{i=1}^{k} \qquad (1 \otimes (\sigma^{-1})^{\star}) \circ \eta}{\Delta \vdash a} \sim \dfrac{\begin{matrix}[\langle \sigma, \vec{f}\rangle \multimap 1]\pi_0\\ \vdots\\ \Gamma_0 \vdash \vec{a} \multimap a\end{matrix} \qquad \left(\begin{matrix}\pi_i\\ \vdots\\ \Gamma_i \vdash a_i\end{matrix}\right)_{i=1}^{k} \qquad \eta}{\Delta \vdash a}$$

where $\langle \sigma, f_1, \ldots, f_k\rangle : \vec{a} = \langle a_1, \ldots, a_k\rangle \to \vec{b} = \langle b_1, \ldots, b_k\rangle$, $\langle \sigma, \vec{g}\rangle : \vec{a}' \to \vec{a}$, $g : a \to a'$ and $\theta_i : \Gamma_i \to \Gamma_i'$

Example

Let $k \in \mathbb{N}$, $\sigma \in \mathfrak{S}_k$ and $\pi = \dfrac{\begin{matrix}\pi_0\\ \vdots\\ \Gamma_0 \vdash \langle a_1, \ldots, a_k\rangle \multimap a\end{matrix} \qquad \left(\begin{matrix}\pi_i\\ \vdots\\ \Gamma_i \vdash a_i\end{matrix}\right)_{i=1}^{k} \qquad \eta}{\Delta \vdash a}$

Let $\pi' = \dfrac{\begin{matrix}\pi_0[\sigma \multimap a]\\ \vdots\\ \Gamma_0 \vdash \langle a_{\sigma(1)}, \ldots, a_{\sigma(k)}\rangle \multimap a\end{matrix} \qquad \left(\begin{matrix}\pi_{\sigma(i)}\\ \vdots\\ \Gamma_{\sigma(i)} \vdash a_{\sigma(i)}\end{matrix}\right)_{i=1}^{k} \qquad \eta'}{\Delta \vdash a}$

and $\eta' = (1 \otimes (\sigma)^{\star}) \circ \eta$

Decategorification

Polr category of preorders and monotonic relations

Decategorification of Dist to Polr

1. small category A: $|\mathsf{Dec}(A)| = \mathrm{ob}(A)$ and $a \leq_{\mathsf{Dec}A} b$ whenever $A(a, b) \neq \emptyset$.
2. small categories A and B, $F : A \nrightarrow B$

$$\mathsf{Dec}_{A,B}(F) = \{\langle a, b\rangle \in |\mathsf{Dec}(A)^{op} \times \mathsf{Dec}(B)| \mid F(a, b) \neq \emptyset\}.$$

$$\mathsf{Dec}(\mathsf{T}_{\vec{x}}(M)) = [\![M]\!]_{\vec{x}}^{\mathrm{MPolr}}$$

Approximation Theorem: $[\![M]\!]_{\vec{x}}^{\mathrm{MPolr}} = [\![\mathrm{BT}_{\beta}(M)]\!]_{\vec{x}}^{\mathrm{MPolr}}$

$$\mathcal{B} = \mathrm{Th}(D_A) \subseteq \mathrm{Th}(U_{\mathsf{Dec}(A)})$$

Characterisation of $\mathrm{T}(M)$

Coherence relation $\sim$

on resource values:

- $x \sim x$
- $\lambda x.v_1 \sim \lambda x.v_2$ if $v_1 \sim v_2$

on resource λ-terms:

- $[m_1, \ldots, m_n] \sim [m_{n+1}, \ldots, m_k]$ if $\forall i, j \leq k, m_i \sim m_j$.
- $m_1 n_1 \sim m_2 n_2$ if $m_1 \sim m_2$ and $n_1 \sim n_2$

Characterisation of $\mathrm{T}(M)$

$A \in P(r\Lambda)$ maximal clique with finite height $\Leftrightarrow \exists M \in \Lambda$, $A = \mathrm{T}(M)$.

Characterisation of $\mathrm{BT}_v(M)$

Characterisation of $\mathrm{BT}_v(M)$

Let $\mathcal{X} \subseteq \mathcal{A}_v$ be a set of approximants. There exists $M \in \Lambda$ such that $\mathcal{A}_v(M) = \mathcal{X}$ if and only if the following three conditions hold:

1. $\mathcal{X}$ is directed and downward closed w.r.t. $\sqsubseteq_v$;
2. $\mathcal{X}$ is recursively enumerable;
3. $\mathrm{FV}(\mathcal{X})$ is finite.