Call-by-Value, Again!

A. Kerinec, G. Manzonetto and S. Ronchi Della Rocca

LIPN, Université Sorbonne Paris-Nord, France
LIPN, Université Sorbonne Paris-Nord, France
Dipartimento di Informatica, Università di Torino, Italy

Programming Language Theory

λ-calculus

$$\begin{array}{lrcl} \text{terms}: & \Lambda : M, N & ::= & x \mid \lambda x.M \mid (MN) \\ \beta\text{-reduction}: & (\lambda x.M)N & \mapsto_\beta & M\{N/x\} \end{array}$$

$$(x(\lambda x.yx)(\lambda y.xy)yx)\{N/x\} = N(\lambda x.yx)(\lambda y.Ny)yN$$

Approximation Theory

Böhm Trees

- Approximants

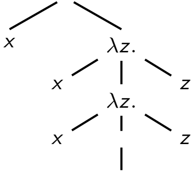
$$A, A_i ::= \bot \mid \lambda\vec{x}.yA_1 \cdots A_n$$

- Approximants of a λ-term M

$$\mathcal{A}(M) = \{A \mid M \to^* N, A \sqsubseteq_\bot N\}$$

- Böhm tree of M

$\mathcal{BT}(Yx)$ with $Yx =_\beta x(\lambda z.Yxz)$



Denotational Models

- Filter Models

≃ Intersection type systems

$$\alpha, \beta ::= a \mid \alpha \wedge \beta \mid \alpha \to \beta$$

- Interpretation of a λ-term M

$$[\![M]\!] = \{\alpha \mid \exists\Gamma, \Gamma \vdash M : \alpha\}$$

- Approximation Theorem

$$\Gamma \vdash M : \alpha \iff \exists A \in \mathcal{A}(M) . \Gamma \vdash A : \alpha$$

Operational Properties of Programs

A term M is:

- Normalizing: if $M \to_\beta^* V$ for some V in NF.
- Head normalizing: if $M \to_\beta^* \lambda x_1 \ldots x_n . x M_1 \cdots M_l$.
- Looping: if M is not head-normalizing.
  Exemple : $\omega_3 = \lambda x.xxx$
  $\Omega_3 = \omega_3\omega_3 \to_\beta \Omega_3\omega_3 \to_\beta \Omega_3\omega_3\omega_3 \ldots$

Solvability:

M is solvable if $\exists x_1, \ldots x_n, \exists M_1, \ldots, M_k$ such that

$$(\lambda x_1 \ldots x_n . M) M_1 \cdots M_k \to_\beta^* I$$

(I = the identity, a completely defined result)

Characterizations of Solvability

M is solvable exactly when...

Characterizations

- Logical: In a suitable intersection type assignment system

$$\exists \Gamma, \alpha \,.\, \Gamma \vdash M : \alpha, \text{ with } \alpha \text{ ``proper''}$$

- Semantical:

$$\mathcal{BT}(M) \neq \bot$$

- Operational :

$$M \text{ is head normalizing}$$

Λ_{CBV}: Call-by-Value λ-calculus

Λ_{CBV}: Call-by-Value λ-calculus

$$\lambda\text{-terms}: \quad \begin{array}{lrcl} \text{values}: & \mathrm{Val} : V, U & ::= & x \mid \lambda x.M \\ \text{terms}: & \Lambda : M, N & ::= & (MN) \mid V \end{array}$$

Λ_{CBV}: Call-by-Value λ-calculus

$$\lambda\text{-terms}: \quad \begin{array}{llcl} \text{values}: & \mathrm{Val} : V, U & ::= & x \mid \lambda x.M \\ \text{terms}: & \Lambda : M, N & ::= & (MN) \mid V \end{array}$$

β_v-reduction:

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

Λ_{CBV}: Call-by-Value λ-calculus

$$\lambda\text{-terms}: \quad \begin{array}{lrcl} \text{values}: & \mathrm{Val} : V, U & ::= & x \mid \lambda x.M \\ \text{terms}: & \Lambda : M, N & ::= & (MN) \mid V \end{array}$$

β_v-reduction:

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

And σ-rules:

$$\begin{array}{llll} (\lambda x.M)NN' & \mapsto_{\sigma_1} & (\lambda x.MN')N & \textit{with } x \notin \mathrm{fv}(N') \\ V((\lambda x.M)N) & \mapsto_{\sigma_3} & (\lambda x.VM)N & \textit{with } x \notin \mathrm{fv}(V) \end{array}$$

CbV Approximants

$\bot$ represents an undefined value.

$$\frac{V \in \mathrm{Val}}{\bot \sqsubseteq_{\bot} V}$$

CbV Approximants

$\bot$ represents an undefined value.

$$\frac{V \in \mathrm{Val}}{\bot \sqsubseteq_{\bot} V}$$

Approximants

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

Approximants of a Term

$$\mathcal{A}(M) = \{A \in \mathcal{A} \text{ s.t. } \exists N \in \Lambda \,.\, M \to_v^* N \text{ and } A \sqsubseteq_\perp N\}$$

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

Böhm Trees: Examples

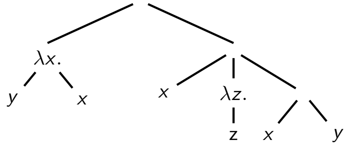


Figure: Böhm tree of $(\lambda x.yx)(x(\lambda z.z(xy)))$

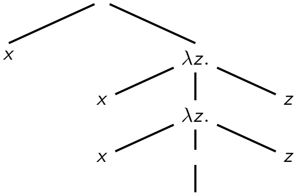


Figure: Böhm tree of $Yx =_{\beta_v} x(\lambda z.Yxz)$

Figure: Böhm tree of $\Omega = (\lambda x.xx)(\lambda x.xx)$

$\perp$

Figure: Böhm tree of $\lambda x.\Omega$

Approximation Theorem

Approximation Theorem :

Let $M \in \Lambda$, $\alpha \in \mathit{Types}$ and Γ be an environment:

$$\Gamma \vdash M : \alpha \iff \exists A \in \mathcal{A}(M) \,.\, \Gamma \vdash A : \alpha$$

Relational Model

Observational Equivalence:

$$M =_{op} N \quad \Longleftrightarrow \quad \forall C : \exists V . \left[\, C[M] \to^{*} V \Leftrightarrow \exists U, C[N] \to^{*} U \,\right]$$

Definition :

A model, with an interpretation $[\![\cdot]\!]$ is:

- adequate if $[\![M]\!] = [\![N]\!] \Rightarrow M =_{op} N$
- fully abstract if $[\![M]\!] = [\![N]\!] \Leftrightarrow M =_{op} N$

Type Assignment System

Countable set $A = \{a, b, c, \ldots\}$ of *atomic types*.

$$\begin{array}{llll} (\mathit{Types}) & \alpha, \beta & ::= & a \mid [\,] \mid \sigma \to \alpha \\ (\mathit{Multi-Types}) & \sigma, \tau, \rho & ::= & [\alpha_1, \ldots, \alpha_n] \quad \text{with } \alpha_i \neq [\,] \end{array}$$

Inference rules:

$$\frac{}{x : [\alpha] \vdash x : \alpha} \qquad \frac{\Gamma, x : \sigma \vdash M : \alpha}{\Gamma \vdash \lambda x.M : \sigma \to \alpha} \qquad \frac{\Gamma_0 \vdash M : \sigma \to \alpha \quad \Gamma_1 \vdash N : \sigma}{\Gamma_0 + \Gamma_1 \vdash MN : \alpha}$$

$$\frac{V \in \mathrm{Val}}{\vdash V : [\,]} \qquad \frac{\Gamma_1 \vdash M : \alpha_1 \quad \cdots \quad \Gamma_n \vdash M : \alpha_n \quad n > 0}{\sum_{i=1}^{n} \Gamma_i \vdash M : [\alpha_1, \ldots, \alpha_n]}$$

In the abstraction rule: $x \notin \mathit{dom}(\Gamma)$.

Type Assignment System

Countable set $A = \{a, b, c, \dots\}$ of *atomic types*.

$$
\begin{array}{llll}
(\mathit{Types}) & \alpha, \beta & ::= & a \mid [\,] \mid \sigma \to \alpha \\
(\mathit{Multi-Types}) & \sigma, \tau, \rho & ::= & [\alpha_1, \dots, \alpha_n] \quad \text{with } \alpha_i \neq [\,]
\end{array}
$$

Inference rules:

$$
\frac{}{x : [\alpha] \vdash x : \alpha} \qquad \frac{\Gamma, x : \sigma \vdash M : \alpha}{\Gamma \vdash \lambda x.M : \sigma \to \alpha} \qquad \frac{\Gamma_0 \vdash M : \sigma \to \alpha \quad \Gamma_1 \vdash N : \sigma}{\Gamma_0 + \Gamma_1 \vdash MN : \alpha}
$$

$$
\frac{V \in \mathrm{Val}}{\vdash V : [\,]} \qquad \frac{\Gamma_1 \vdash M : \alpha_1 \quad \cdots \quad \Gamma_n \vdash M : \alpha_n \quad n > 0}{\sum_{i=1}^{n} \Gamma_i \vdash M : [\alpha_1, \dots, \alpha_n]}
$$

In the abstraction rule: $x \notin dom(\Gamma)$.

Type Assignment System

Countable set $A = \{a, b, c, \dots\}$ of *atomic types*.

$$
\begin{array}{llll}
(\mathit{Types}) & \alpha, \beta & ::= & a \mid [\,] \mid \sigma \to \alpha \\
(\mathit{Multi-Types}) & \sigma, \tau, \rho & ::= & [\alpha_1, \dots, \alpha_n] \quad \text{with } \alpha_i \neq [\,]
\end{array}
$$

Inference rules:

$$
\frac{}{x : [\alpha] \vdash x : \alpha} \qquad \frac{\Gamma, x : \sigma \vdash M : \alpha}{\Gamma \vdash \lambda x.M : \sigma \to \alpha} \qquad \frac{\Gamma_0 \vdash M : \sigma \to \alpha \quad \Gamma_1 \vdash N : \sigma}{\Gamma_0 + \Gamma_1 \vdash MN : \alpha}
$$

$$
\frac{V \in \mathrm{Val}}{\vdash V : [\,]} \qquad \frac{\Gamma_1 \vdash M : \alpha_1 \quad \cdots \quad \Gamma_n \vdash M : \alpha_n \quad n > 0}{\sum_{i=1}^{n} \Gamma_i \vdash M : [\alpha_1, \dots, \alpha_n]}
$$

In the abstraction rule: $x \notin dom(\Gamma)$.

General Properties

Not fully abstract:

Recursion operator $Zx = x(\lambda z.Zxz)$

for example $\lambda x.(\lambda y.x(\lambda z.yyz))(\lambda y.x(\lambda z.yyz))$

Composition operator $B = \lambda xyz.x(yz)$

$$I =_{op} ZB$$

$$but\ \mathcal{A}(I) = \{\lambda x.\bot, \lambda x.x\}\ and\ \mathcal{A}(ZB) = \{\lambda x_1 \ldots x_n.\bot | n \geq 1\}$$

$\vdash I : [[] \to []] \to [] \to []$ and $\nvdash ZB : [[] \to []] \to [] \to []$

Closer to full abstraction than Ehrhard's model

Definitions of Valuability and Potential Valuability

Definition: M is

- valuable if $\exists V$ such that $M \to_v^* V$
- potentially valuable if $\exists x_1, \ldots x_n, \exists M_1, \ldots M_l$ such that $(\lambda x_1 \ldots x_n.M)M_1 \cdots M_l$ is valuable

Example

- $I, \Delta, \Delta(II)$ are (potentially) valuable and solvable.
- $Proj_1 x(\lambda x.\Omega), xy(I\Delta)$ and $(\Delta)(xy)$ are not valuable, but potentially valuable and solvable.
- $\lambda x.\Omega$ is valuable, but unsolvable.
- $\Omega, \Omega(xy), (\lambda y.\Delta)(xI)\Delta, I\Omega$ are not potentially valuable nor unsolvable. The same holds for YM, where Y is a fixed point operator and M is a λ-term.

Characterizations of Valuability and Potential Valuability

Theorem :

Let $M \in \Lambda$, then:

- M is valuable $\iff \exists \Gamma, \Gamma \vdash M : [\,] \iff \bot \in \mathcal{A}(M)$.
- M is potentially valuable $\iff \exists \Gamma, \alpha \,.\, \Gamma \vdash M : \alpha \iff \mathcal{A}(M) \neq \varnothing$.

More Precise Approximants

Refined Approximants

Subsets $\mathcal{S}, \mathcal{U} \subseteq \mathcal{A}$:

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

Some Examples

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

Example

- $x, I, I(zz), \Delta(zz), (\lambda x.(I(yz)))(zy\bot) \in \mathcal{S}$.
- $\bot, \lambda x_0 \ldots x_n.\bot, (\lambda x.\bot)(zz), (\lambda x.(\lambda y.\bot)(wz))(zw) \in \mathcal{U}$.
- Finally, notice that $\mathcal{A}(\Omega), \mathcal{A}(ZI), \mathcal{A}(\lambda x.\Omega) \subseteq \mathcal{U}$.

Characterizations of Solvability

Trivial type: $\sigma_1 \to \sigma_2 \to \cdots \to \sigma_n \to [\,]$ with $n \geq 0$.
A type not trivial is proper.

Theorem : Characterizations of Solvability

For $M \in \Lambda$, the following are equivalent:

- M is solvable
- $\exists \alpha$ proper, $\exists \Gamma$ such that $\Gamma \vdash M : \alpha$
- $\exists A$ such that $A \in \mathcal{A}(M) \cap \mathcal{S}$

Characterizations of Solvability

Trivial type: $\sigma_1 \to \sigma_2 \to \cdots \to \sigma_n \to [\,]$ with $n \geq 0$.
A type not trivial is proper.

Theorem : Characterizations of Solvability

For $M \in \Lambda$, the following are equivalent:

- M is solvable
- $\exists\alpha$ proper, $\exists\Gamma$ such that $\Gamma \vdash M : \alpha$
- $\exists A$ such that $A \in \mathcal{A}(M) \cap \mathcal{S}$

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

Semi-Sensible Model

The type assignment system induces a relational model $\mathcal{M}$.

Corollary :

The model $\mathcal{M}$ is not sensible, but semi-sensible.

Decidability of the Inhabitation Problem

The inhabitation problem for system $\mathcal{M}$

For every environment Γ and type α is there a λ-term M satisfying $\Gamma \vdash M : \alpha$?

Inhabitation algorithm

$$\frac{}{\perp \in \mathsf{IM}(\varnothing;\ [\,])} \qquad \frac{}{\perp \in \mathsf{IT}(\varnothing;\ [\,])} \qquad \frac{}{x \in \mathsf{IT}(x : [\alpha];\ \alpha)}$$

$$\frac{A \in \mathsf{IT}(\Gamma, x : \sigma;\ \alpha)}{\lambda x.A \in \mathsf{IT}(\Gamma;\ \sigma \to \alpha)} \qquad \frac{A_i \in \mathsf{IT}(\Gamma_i;\ \alpha_i) \quad \uparrow\{A_i\}_{i\in I} \quad A = \bigsqcup_{i\in I} A_i}{A \in \mathsf{IM}(\Sigma_{i\in I}\Gamma_i;\ [\alpha_i]_{i\in I})}$$

$$\frac{A_j \in \mathsf{IM}(\Gamma_j;\ \sigma_j) \quad 0 \le j \le n \quad A_0 \in \mathcal{H}}{xA_0 \cdots A_n \in \mathsf{IT}(\sum_{j=0}^{n} \Gamma_j + x : [\sigma_0 \to \cdots \to \sigma_n \to \alpha];\ \alpha)}$$

$$\frac{A_j \in \mathsf{IM}(\Gamma_j;\ \sum_{i=0}^{m} \tau_j^i) \quad 0 \le j \le n \quad A_0 \in \mathcal{H} \quad A \in \mathsf{IT}(\Gamma_{n+1}, x : [\alpha_i]_{0\le i\le m};\ \alpha)}{(\lambda x.A)(yA_0 \cdots A_n) \in \mathsf{IT}(\sum_{j=0}^{n+1} \Gamma_j + y : [\tau_0^i \to \cdots \to \tau_n^i \to \alpha_i]_{0\le i\le m};\ \alpha)}$$

Figure: Inhabitation algorithm for system $\mathcal{M}$. In the last rule $x \notin \mathit{free\text{-}var}(yA_0 \cdots A_n)$.

Algorithm Properties

The inhabitation algorithm terminates.

Theorem : Soundness and Completeness

- If $A \in \mathsf{IT}(\Gamma; \alpha)$ then, $\forall M \in \Lambda$ such that $A \sqsubseteq_{\perp} M$, we have $\Gamma \vdash M : \alpha$.
- If $\Gamma \vdash M : \alpha$ then $\exists A \in \mathsf{IT}(\Gamma; \alpha)$ such that $A \in \mathcal{A}(M)$.

Conclusion

Future work:

- extend results to other models of the class
- study of categorical construction

PROBLEM!

$$w((\lambda x.w')(zy)) \to_{\sigma_3} (\lambda x.ww')(zy)$$

$\Gamma = w : [\sigma \to \alpha], z : [b_1 \to [\,], b_2 \to [\,]], y : [b_1, b_2], w' : [a_1, a_2]$

$\Gamma \vdash w((\lambda x.w')(zy)) : \alpha$ but $\Gamma \nvdash (\lambda x.ww')(zy)$

because with:

$z : [b_1 \to [\,]], y : [b_1] \vdash zy : [\,]$
$z : [b_2 \to [\,]], y : [b_2] \vdash zy : [\,]$

we do not obtain $z : [b_1 \to [\,], b_2 \to [\,]], y : [b_1, b_2] \vdash zy : [\,]$

New Type Assignment System

Inference rules:

$$\frac{}{x : [\alpha] \vdash x : \alpha} \qquad \frac{\Gamma, x : \sigma \vdash M : \alpha}{\Gamma \vdash \lambda x.M : \sigma \to \alpha} \qquad \frac{\Gamma_0 \vdash M : \sigma \to \alpha \quad \Gamma_1 \vdash N : \sigma}{\Gamma_0 + \Gamma_1 \vdash MN : \alpha}$$

$$\frac{V \in \mathrm{Val}}{\vdash V : []} \qquad \frac{\Gamma_1 \vdash M : \alpha_1 \quad \cdots \quad \Gamma_n \vdash M : \alpha_n \quad n > 0}{\sum_{i=1}^{n} \Gamma_i \vdash M : [\alpha_1, \ldots, \alpha_n]}$$

$$\frac{\Gamma_1 \vdash M : [] \quad \cdots \quad \Gamma_n \vdash M : [] \quad \Gamma_{n+1} \vdash M : \alpha \quad n > 0}{\sum_{i=1}^{n+1} \Gamma_i \vdash M : [\alpha]}$$

In the abstraction rule: $x \notin dom(\Gamma)$.