The $\lambda\bar{\lambda}$-calculus

A dual calculus for unconstrained strategies

Alexis Goyet

PPS

December 20, 2012

Introduction

Game semantics:

A semantical model in which the execution of a program is seen as a dialog between the program (the Player) and its environment (the Opponent). Different programs are represented by different *strategies*.

Introduced to give the first fully abstract model for PCF (a purely functional language).

Introduction

The original presentation of game semantics immediately introduced *constraints*:

- Innocence (on information)
- Innocence (on actions)
- Bracketing
- Determinism

Introduction

Lifting these constraints yields models for additional language features:

- Innocence (information): integer stores
- Innocence (actions): functional stores
- Bracketing: control operators
- Determinism

A functional language extended with all these features would correspond to the unconstrained model.

But can we find a more direct fit?

Introduction

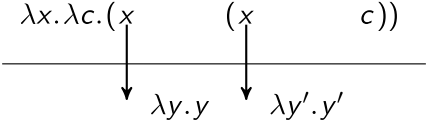
We want to give a language which corresponds to unconstrained game semantics as *directly* as possible.

This will allow to:

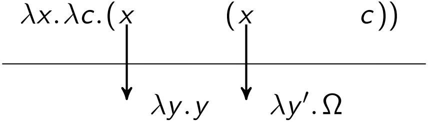
- Explain concepts from game semantics syntactically (eg. innocent expansion, duality).
- Give languages for classes of strategies defined in semantical works.

Interaction with and without references

$$(\lambda x.\lambda c.(x\ (x\ c)))\ (\lambda y.y)$$

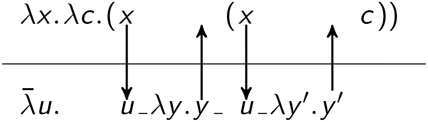


$\text{new } r = \text{true};$
$(\lambda x.\lambda c.(x\ (x\ c)))\ (\text{if } !r \text{ then } (r := \text{false};\ \lambda y.y) \text{ else } \lambda y'.\Omega)$

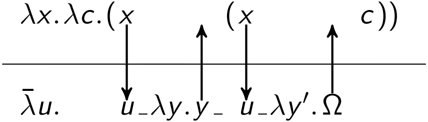


Interaction with and without references

$$(\lambda x.\lambda c.(x\ (x\ c)))\ (\lambda y.y)$$

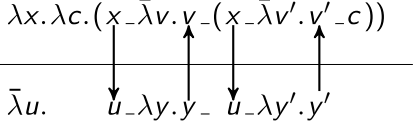


new r = true;
$(\lambda x.\lambda c.(x\ (x\ c)))$ (if $!r$ then (r := false; $\lambda y.y$) else $\lambda y'.\Omega$)

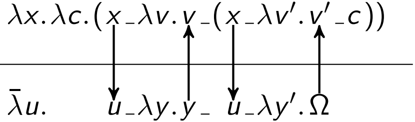


Interaction with and without references

$$(\lambda x.\lambda c.(x\ (x\ c)))\ (\lambda y.y)$$



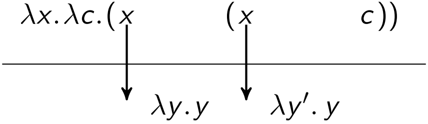
new r = true;
$(\lambda x.\lambda c.(x\ (x\ c)))$ (if $!r$ then ($r :=$ false; $\lambda y.y$) else $\lambda y'.\Omega$)



Interaction with functional references

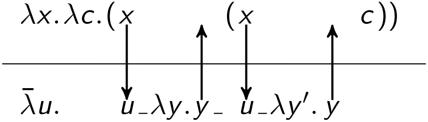
new r = true; new r_2;

$(\lambda x.\lambda c.(x\ (x\ c)))$ (if $!r$ then ($r :=$ false; $\lambda y.(r_2 := y;\ y)$) else $\lambda y'.!r_2$)



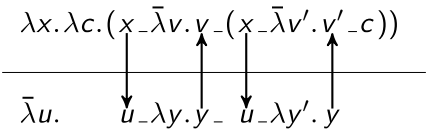
Interaction with functional references

new $r = $ true; new r_2;
$(\lambda x.\lambda c.(x\ (x\ c)))$ (if $!r$ then ($r :=$ false; $\lambda y.(r_2 := y;\ y)$) else $\lambda y'.!r_2$)



Interaction with functional references

new $r = $ true; new r_2;
$(\lambda x.\lambda c.(x\ (x\ c)))$ (if $!r$ then $(r := \text{false};\ \lambda y.(r_2 := y;\ y))$ else $\lambda y'.!r_2$)



The $\lambda\bar{\lambda}$-calculus syntax

$$\begin{array}{rl} & \lambda x.\lambda c.x\_\bar{\lambda} v.v\_x\_\bar{\lambda} v'.v'\_c\_\bar{\epsilon} \\ \triangleleft & \bar{\lambda} u. \quad u\_\lambda y.y\_u\_\lambda y'.y'\_\bar{\epsilon} \end{array}$$

$$\to^* \quad \lambda c.c\_\bar{\epsilon}$$

Syntax

$$\begin{array}{rcccccccccccc} \phi, \psi & := & \epsilon & \mid & \lambda x.\phi & \mid & x\_\sigma & \mid & \phi + \psi & \mid & \phi \triangleleft \tau & \mid & \overset{L}{\nabla} s.\phi \\ \sigma, \tau & := & \bar{\epsilon} & \mid & \bar{\lambda} u.\sigma & \mid & u\_\phi & \mid & \sigma + \tau & \mid & \sigma \triangleleft \tau & & \end{array}$$

Variables: x, y, z
Co-variables: u, v, w

A loose comparison to other languages

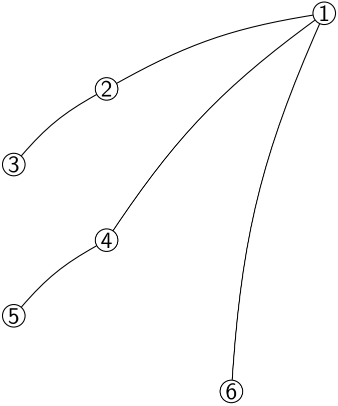
Syntax

$$\begin{array}{rcccccccccccccc} \phi, \psi & := & \epsilon & \mid & \lambda x.\phi & \mid & x\_\sigma & \mid & \phi + \psi & \mid & \phi \triangleleft \tau & \mid & \overset{L}{\nabla} S.\phi \\ \sigma, \tau & := & \bar{\epsilon} & \mid & \bar{\lambda} u.\sigma & \mid & u\_\phi & \mid & \sigma + \tau & \mid & \sigma \triangleleft \tau \end{array}$$

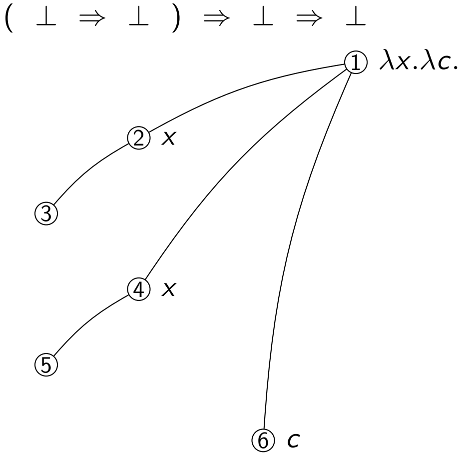
λ-calculus with references	$\lambda\bar{\lambda}$-calculus	CCS
λx	λx	
new u;	$\bar{\lambda} u$	
$x\ M$	x_σ	$x \cdot P$
$u := M$;	u_ϕ	$\bar{x} \cdot P$
	$\phi + \psi$	$P + Q$
$M\ N$	$\phi \triangleleft \sigma$	$P \mid Q$
skip	$\bar{\epsilon}, \epsilon$	0
	$\overset{(x,u)}{\nabla}.\phi$	$\nu x.P$

Strategies

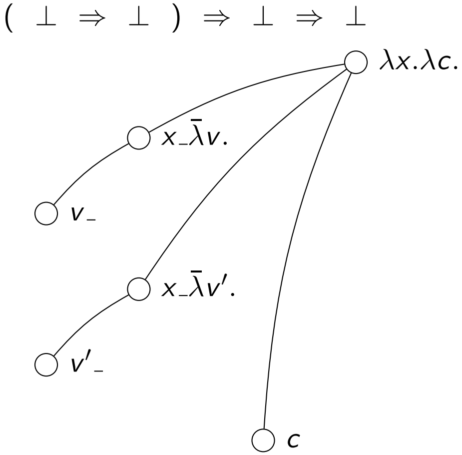
$$(\ \bot \Rightarrow \bot\ ) \Rightarrow \bot \Rightarrow \bot$$



Strategies



Strategies



$$\lambda x.\lambda c.x\_\bar{\lambda}v.v\_x\_\bar{\lambda}v'.v'\_c$$

Reduction rules (intuition)

$$\lambda x.\phi \ \triangleleft \ \bar{\lambda} u.\sigma \quad \rightarrow^* \quad \overset{(x,u)}{\nabla}.(\phi \triangleleft \sigma)$$

$$x\_\sigma \ \triangleleft \ u\_\phi \quad \rightarrow^* \quad x\_u\_(\sigma \triangleright \phi)$$

$$\overset{(x,u)}{\nabla}.x\_u\_\phi \quad \rightarrow^* \quad \overset{(x,u)}{\nabla}.\phi \qquad\qquad \overset{(x,u)}{\nabla}.x\_v\_\phi \quad \rightarrow^* \quad \bar{\epsilon}$$

$$\lambda x.x\_\sigma \ \triangleleft \ \bar{\lambda} u.u\_\phi \quad \rightarrow^* \quad \overset{(x,u)}{\nabla}.(\sigma \triangleright \phi)$$

Innocence as a syntactical expansion

Syntactical fixed-point

$$\nu\alpha.\phi \;\rightarrow\; \phi[\alpha\backslash\nu\alpha.\phi]$$

$$\begin{array}{ll} & tr(\lambda y.y) \\ = & \bar{\lambda} u.u\_\nu\alpha.\lambda y.y\_u\_\alpha \\ \rightarrow & \bar{\lambda} u.u\_\lambda y.y\_u\_\nu\alpha.\lambda y.y\_u\_\alpha \\ \rightarrow & \bar{\lambda} u.u\_\lambda y.y\_u\_\lambda y'.y'\_u\_\nu\alpha.\lambda y.y\_u\_\alpha \\ \ldots & \end{array}$$

Reduction (big steps)

$$\begin{array}{rl} & \lambda x.\lambda c.x\_\bar{\lambda} v.v\_x\_\bar{\lambda} v'.v'\_c \\ \lhd & \bar{\lambda} u. \quad u\_\lambda y.y\_u\_\lambda y'.y' \end{array}$$

$$\begin{array}{lrrr} & \lambda x.\lambda c.x\_\bar{\lambda} v.v\_x\_\bar{\lambda} v'.v'\_c\_\bar{\epsilon} \lhd \bar{\lambda} u.u\_\lambda y.y\_u\_\lambda y'.y'\_\bar{\epsilon} \\ \to^* & \lambda c. \overset{(x,u)}{\nabla} .(& \bar{\lambda} v.v\_x\_\bar{\lambda} v'.v'\_c\_\bar{\epsilon} \rhd & \lambda y.y\_u\_\lambda y'.y'\_\bar{\epsilon} \,) \\ \to^* & \lambda c. \overset{(y,v)(x,u)}{\nabla} .(& x\_\bar{\lambda} v'.v'\_c\_\bar{\epsilon} \lhd & u\_\lambda y'.y'\_\bar{\epsilon} \,) \\ \to^* & \lambda c. \overset{(y,v)(x,u)}{\nabla} .(& \bar{\lambda} v'.v'\_c\_\bar{\epsilon} \rhd & \lambda y'.y'\_\bar{\epsilon} \,) \\ \to^* & \lambda c. \overset{(y',v')(y,v)(x,u)}{\nabla} .(& c\_\bar{\epsilon} \lhd & \bar{\epsilon} \,) \\ \to^* & \lambda c.c\_\bar{\epsilon} & & \end{array}$$

Suppose we want to distinguish these two λ-terms:

$$\lambda f.\lambda g.\lambda c.(f\ (g\ c))$$
$$\lambda f.\lambda g.\lambda c.(g\ (f\ c))$$

The applicative context $C[]$ does not distinguish them, because the two sides (for f and g) do not interact:

$$C[] = ([]\ \bar{\lambda}z.z)\ \bar{\lambda}z'.z'$$

But consider this approximation of $C[]$ in the $\lambda\bar{\lambda}$-calculus:

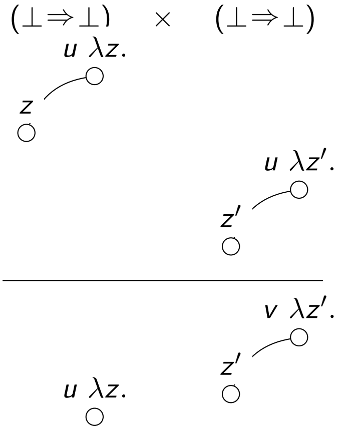
$$\begin{aligned} & ([] \triangleleft \bar{\lambda}u.u\_\lambda z.z\_\bar{\epsilon}) \triangleleft \bar{\lambda}v.v\_\lambda z'.z'\_\bar{\epsilon} \\ \approx \quad & [] \triangleleft \bar{\lambda}u.\bar{\lambda}v.(u\_\cdots + v\_\cdots) \end{aligned}$$

Now u and v are shared by the two sides.

A modified context:

$$[] \triangleleft \bar{\lambda} u.\bar{\lambda} v.(u\_\lambda z.z\_v\_\lambda z'.z'\_\bar{\epsilon} + v\_\lambda z'.z'\_u\_\lambda z.\epsilon)$$

And its strategy:



Typing

$$\frac{}{\Gamma \mid \epsilon : A \vdash \ \mid \Delta}$$

$$\frac{}{\Gamma \mid \ \vdash \bar{\epsilon} : 1 \mid \Delta}$$

$$\frac{\Gamma, x : A \mid \phi : B \vdash \ \mid \Delta}{\Gamma \mid \lambda x.\phi : A \times B \vdash \ \mid \Delta}$$

$$\frac{\Gamma \mid \ \vdash \sigma : B \mid u : A, \Delta}{\Gamma \mid \ \vdash \bar{\lambda} u.\sigma : A \times B \mid \Delta}$$

$$\frac{\Gamma, x : \neg(A \times B) \mid \ \vdash \sigma : A \mid \Delta}{\Gamma, x : \neg(A \times B) \mid x\_\sigma : B \vdash \ \mid \Delta}$$

$$\frac{\Gamma \mid \phi : A \vdash \ \mid u : \neg A, \Delta}{\Gamma \mid \ \vdash u\_\phi : 1 \mid u : \neg A, \Delta}$$

$$\frac{\Gamma \mid \phi : A \vdash \ \mid \Delta \qquad \Gamma \mid \psi : A \vdash \ \mid \Delta}{\Gamma \mid \phi + \psi : A \vdash \ \mid \Delta}$$

$$\frac{\Gamma \mid \ \vdash \sigma : A \mid \Delta \qquad \Gamma \mid \ \vdash \tau : A \mid \Delta}{\Gamma \mid \ \vdash \sigma + \tau : A \mid \Delta}$$

$$\frac{\Gamma \mid \phi : A \times B \vdash \ \mid \Delta \qquad \Gamma \mid \ \vdash \sigma : A \mid \Delta}{\Gamma \mid \phi \triangleleft \sigma : B \vdash \ \mid \Delta}$$

$$\frac{\Gamma \mid \ \vdash \sigma : A \mid \Delta \qquad \Gamma \mid \ \vdash \tau : B \mid \Delta}{\Gamma \mid \ \vdash \sigma \triangleleft \tau : A \times B \mid \Delta}$$

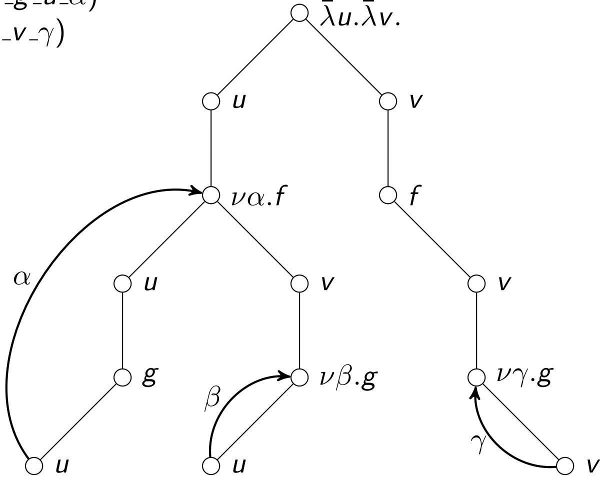
$$\frac{\Gamma, \vec{x} : L \mid \phi : S \times A \vdash \ \mid \ \vec{v} : S, \ \vec{u} : L, \Delta}{\Gamma \mid \overset{(x_i, u_i)_i}{\nabla} \vec{v}.\phi : A \vdash \ \mid \Delta}$$

Conclusion and future work

- Concepts of game semantics are given direct syntactic equivalent (duality, innocent expansion, Böhm-out...).
- Classes of strategies defined in theoretical works can be described as sub-languages.
- The language allows precise access control on the history of the interaction, which could be used to define constrained effects.

Arbitrary branches as views

$$\bar\lambda u.\bar\lambda v.(\quad u\_\nu\alpha.f\_(\quad u\_\nu\beta.g\_u\_\beta \\ + v\_g\_u\_\alpha) \\ + v\_\nu\gamma.f\_ v\_g\_v\_\gamma)$$



Typing example

$$\frac{}{\Gamma \mid \epsilon : A \vdash \ \mid \Delta} \qquad \frac{}{\Gamma \mid \ \vdash \bar{\epsilon} : 1 \mid \Delta}$$

$$\frac{\Gamma, x : A \mid \phi : B \vdash \ \mid \Delta}{\Gamma \mid \lambda x.\phi : A \times B \vdash \ \mid \Delta} \qquad \frac{\Gamma \mid \ \vdash \sigma : B \mid u : A, \Delta}{\Gamma \mid \ \vdash \bar{\lambda} u.\sigma : A \times B \mid \Delta}$$

$$\frac{\Gamma, x : \neg(A \times B) \mid \ \vdash \sigma : A \mid \Delta}{\Gamma, x : \neg(A \times B) \mid x\_\sigma : B \vdash \ \mid \Delta} \qquad \frac{\Gamma \mid \phi : A \vdash \ \mid u : \neg A, \Delta}{\Gamma \mid \ \vdash u\_\phi : 1 \mid u : \neg A, \Delta}$$

Example

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{y : \bot \mid \bar{\epsilon} : 1 \vdash \ \mid u : \bot \Rightarrow \bot}}{y : \bot \mid y\_\bar{\epsilon} : 1 \vdash \ \mid u : \bot \Rightarrow \bot}}{\mid \lambda y.y\_\bar{\epsilon} : \bot \vdash \ \mid u : \bot \Rightarrow \bot}}{\mid \ \vdash u\_\lambda y.y\_\bar{\epsilon} : 1 \mid u : \bot \Rightarrow \bot}}{\mid \ \vdash \bar{\lambda} u.u\_\lambda y.y\_\bar{\epsilon} : \bot \Rightarrow \bot \mid}$$

Reduction rules

$$
\begin{array}{ccc}
\phi \lhd \bar{\lambda}\vec{u}.\sigma_{\mathrm{g}} & \to & \nabla^{\vec{u}}.(\phi \lhd \sigma_{\mathrm{g}}) & u_i \notin \mathrm{FN}(\phi) \\
\lambda x.\phi \lhd \sigma_{\mathrm{g}} & \to & \lambda x.(\phi \lhd \sigma_{\mathrm{g}}) & x \notin \mathrm{FN}(\sigma_{\mathrm{g}}) \\
x\_\sigma \lhd \tau_{\mathrm{g}} & \to & x\_(\sigma \lhd \tau_{\mathrm{g}}) & \\
\\
\bar{\lambda}u.\sigma \lhd \tau & \to & \bar{\lambda}u.(\sigma \lhd \tau) & u \notin \mathrm{FN}(\tau) \\
\sigma_{\mathrm{g}} \lhd \bar{\lambda}v.\tau & \to & \bar{\lambda}v.(\sigma_{\mathrm{g}} \lhd \tau) & v \notin \mathrm{FN}(\sigma_{\mathrm{g}}) \\
u\_\phi \lhd v\_\psi & \to & u\_(\phi \lhd v\_\psi) + v\_(\psi \lhd u\_\phi) & \\
\\
\overset{L}{\nabla}^{u\cdot S}.\lambda x.\phi & \to & \overset{(x,u),L}{\nabla}^{S}.\phi & x \notin L, u \notin L \\
\overset{L}{\nabla}.\lambda x.\phi & \to & \lambda x.\overset{L}{\nabla}.\phi & x \notin L \\
\\
\overset{L}{\nabla}^{S}.x\_\bar{\lambda}\vec{v}.u\_\phi & \to & \overset{L}{\nabla}^{\vec{v}\cdot S}.\phi & (x,u) \in L \\
\overset{L}{\nabla}^{S}.x\_\bar{\lambda}\vec{v}.w\_\phi & \to & \epsilon & (x,u) \in L \\
\overset{L}{\nabla}^{S}.x\_\bar{\lambda}\vec{v}.\bar{\epsilon} & \to & \epsilon & (x,u) \in L \\
\\
\overset{L}{\nabla}^{\vec{u}}.x\_\bar{\lambda}\vec{v}.w\_\phi & \to & x\_\bar{\lambda}\vec{v}.\bar{\lambda}\vec{u}.w\_\overset{L}{\nabla}.\phi & x \notin L, w \notin L \\
\overset{L}{\nabla}^{\vec{u}}.x\_\bar{\lambda}\vec{v}.w\_\phi & \to & x\_\bar{\lambda}\vec{v}.\bar{\lambda}\vec{u}.\bar{\epsilon} & x \notin L, w \in L \\
\overset{L}{\nabla}^{\vec{u}}.x\_\bar{\lambda}\vec{v}.\bar{\epsilon} & \to & x\_\bar{\lambda}\vec{v}.\bar{\lambda}\vec{u}.\bar{\epsilon} & x \notin L
\end{array}
$$

fixed-point

$$\frac{\Gamma, \alpha^u : \neg A \mid \phi : A \vdash \ \mid u : \neg A, \Delta}{\Gamma \mid \ \vdash u\_\nu\alpha.\phi : 1 \mid u : \neg A, \Delta}$$

$$\frac{}{\Gamma, \alpha^u : \neg A \mid \ \vdash u\_\alpha : 1 \mid u : \neg A, \Delta}$$

$$\nu\alpha.\phi \ \rightarrow \ \phi[\alpha\backslash\nu\alpha.\phi]$$

Translation from the λ-calculus

$$\begin{array}{rcll}
\mathrm{Inn}(\phi) & = & \bar{\lambda} u.u_\nu\alpha.\mathrm{Inn}^{\alpha,u}(\phi) & u,\alpha \notin \mathrm{FN}(\phi) \\
\mathrm{Inn}^{\alpha,u}(\bar{\epsilon}) & = & u_\alpha & \\
\mathrm{Inn}^{\alpha,u}(v_\phi) & = & u_\alpha \triangleleft v_\mathrm{Inn}^{\alpha,u}(\phi) & \\
(\cdots) & & &
\end{array}$$

$$\begin{array}{rcl}
\mathrm{tr}(M) & = & \mathrm{Inn}([M]) \\
[MN] & = & [M] \triangleleft \mathrm{Inn}([N]) \\
[\lambda x.M] & = & \lambda x.[M] \\
[x] & = & x_\bar{\epsilon} \\
[\Omega] & = & \epsilon
\end{array}$$