Types avec intersection dans une extension de $\lambda\mu\widetilde{\mu}$

Pierre Lescanne, LIP, ENS de Lyon

travail fait en collaboration avec Silvia Ghilezan et Daniel Dougherty

The implicative sequent calculus

Propositions are made only

- of propositional variables
- and of the implication operators.

The implicative sequent calculus (the rules)

$$\frac{}{\Gamma, A \vdash \Delta, A}\,(ax)$$

The implicative sequent calculus (the rules)

$$\frac{}{\Gamma, A \vdash \Delta, A}\,(ax)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \to B \vdash \Delta}\,(\to L) \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta}\,(\to R)$$

The implicative sequent calculus (the rules)

$$\frac{}{\Gamma, A \vdash \Delta, A}\,(ax)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \to B \vdash \Delta}\,(\to L) \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta}\,(\to R)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}\,(cut)$$

A proof of the Peirce law

$$\vdash ((A \to B) \to A) \to A$$

A proof of the Peirce law

$$\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\ (\to R)$$

A proof of the Peirce law

$$\dfrac{\dfrac{\vdash A \to B, A \qquad A \vdash A}{(A \to B) \to A \vdash A}\,(\to L)}{\vdash ((A \to B) \to A) \to A}\,(\to R)$$

A proof of the Peirce law

Easy

$$\dfrac{\dfrac{\vdash A \to B, A \qquad A \vdash A}{(A \to B) \to A \vdash A}\,(\to L)}{\vdash ((A \to B) \to A) \to A}\,(\to R)$$

A proof of the Peirce law

$$\dfrac{\vdash A \to B, A \qquad \boxed{\dfrac{}{A \vdash A}\,(ax)}}{\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\,(\to R)}\,(\to L)$$

A proof of the Peirce law

$$\dfrac{\boxed{\dfrac{\overline{A \vdash B, A}\;(ax)}{\vdash A \to B, A}\;(\to R)} \qquad \overline{A \vdash A}\;(ax)}{\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\;(\to R)}\;(\to L)$$

A proof of the Peirce law

$$\dfrac{\dfrac{\dfrac{}{A \vdash B, A}\,(ax)}{\vdash A \to B, A}\,(\to R) \qquad \dfrac{}{A \vdash A}\,(ax)}{\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\,(\to R)}\,(\to L)$$

The active formula

The active formula is the formula on the lower part of a rule

which is “split” by the rule.

For instance in

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \rightarrow B \vdash \Delta} (\rightarrow L)$$

the active formula is $A \rightarrow B$.

The active formula

It makes sense to track the active formulae and to suppose that A and B become the new active formulae:

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \to B \vdash \Delta} (\to L)$$

Similarly

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta} (\to R)$$

We have to prove B using the proposition A and to split B if necessary.

Active formula

But our proof of the Peirce law does not fulfill this statement on active formulae.

$$\dfrac{\dfrac{\dfrac{}{A \vdash B, A}\,(ax)}{\vdash A \to B, A}\,(\to R) \qquad \dfrac{}{A \vdash A}\,(ax)}{\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\,(\to R)}\,(\to L)$$

Active formula

But our proof of the Peirce law does not fulfill this statement on active formulae.

$$\dfrac{\dfrac{\dfrac{}{A \vdash B, A}\,(ax)}{\vdash A \to B, A}\,(\to R) \qquad \dfrac{}{A \vdash A}\,(ax)}{\dfrac{(A \to B) \to A \vdash A}{\vdash ((A \to B) \to A) \to A}\,(\to R)}\,(\to L)$$

The rules of the implicative sequent calculus with active formulae

$$\frac{}{\Gamma, A \vdash \Delta, A}(L-ax) \qquad \frac{}{\Gamma, A \vdash \Delta, A}(R-ax)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \to B \vdash \Delta}(\to L) \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta}(\to R)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}(cut)$$

The rules of the implicative sequent calculus with active formulae

Four requirements:

- One needs to introduce two axioms according to the side of the active formula.
- In (cut) the new introduced proposition becomes the active formula.
- The lower sequent of (cut) has no active formula.
- One needs to introduce a new rule that activates a formula and enables a (cut) above that rule.

The rules of the implicative sequent calculus with active formulae

$$\frac{}{\Gamma, A \vdash \Delta, A}(L-ax) \qquad \frac{}{\Gamma, A \vdash \Delta, A}(R-ax)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \rightarrow B \vdash \Delta}(\rightarrow L) \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \rightarrow B, \Delta}(\rightarrow R)$$

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}(cut)$$

$$\frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A, \Delta}(\mu) \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma, A \vdash \Delta}(\widetilde{\mu})$$

A new proof of the Peirce law

$$
\dfrac{(A \to B) \to A \vdash (A \to B) \to A, A \qquad \dfrac{\mathcal{A}_1 \qquad \mathcal{A}_2}{(A \to B) \to A, (A \to B) \to A \vdash A}}{\dfrac{\dfrac{(A \to B) \to A \vdash A}{(A \to B) \to A \vdash A}\ (\mu)}{\vdash ((A \to B) \to A) \to A}\ (\to R)}\ (cut)
$$

where

$$\frac{\mathcal{A}_1 \qquad \mathcal{A}_2}{(A \to B) \to A, (A \to B) \to A \vdash A}$$

$$=$$

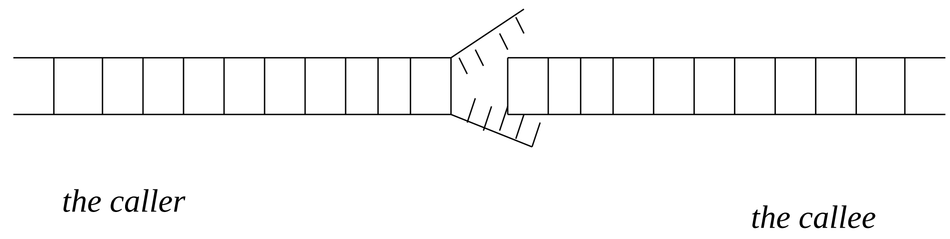
$$\dfrac{\dfrac{\dfrac{(A \to B) \to A, A \vdash A, B, A \qquad (A \to B) \to A, A, A \vdash A, B}{(A \to B) \to A, A \vdash B, A}\,(cut)}{(A \to B) \to A, A \vdash B, A}\,(\mu)}{(A \to B) \to A \vdash A \to B, A}\,(\to R) \qquad (A \to B) \to A, A \vdash A$$

$$\overline{(A \to B) \to A, (A \to B) \to A \vdash A}\;(\to L)$$

The model of computation: Herbelin's calculus

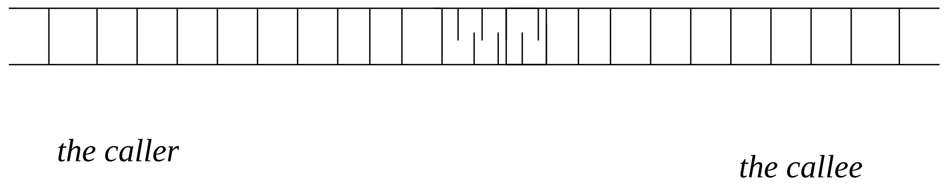
A model of computation

The model of computation relies on capsules



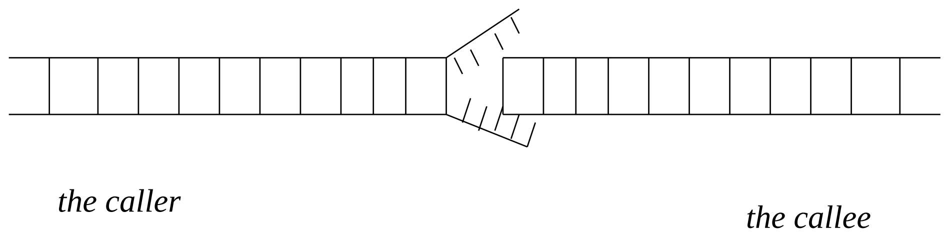
A model of computation

The model of computation relies on capsules



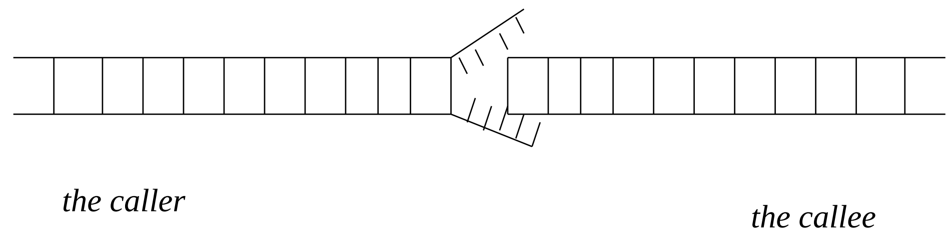
A model of computation

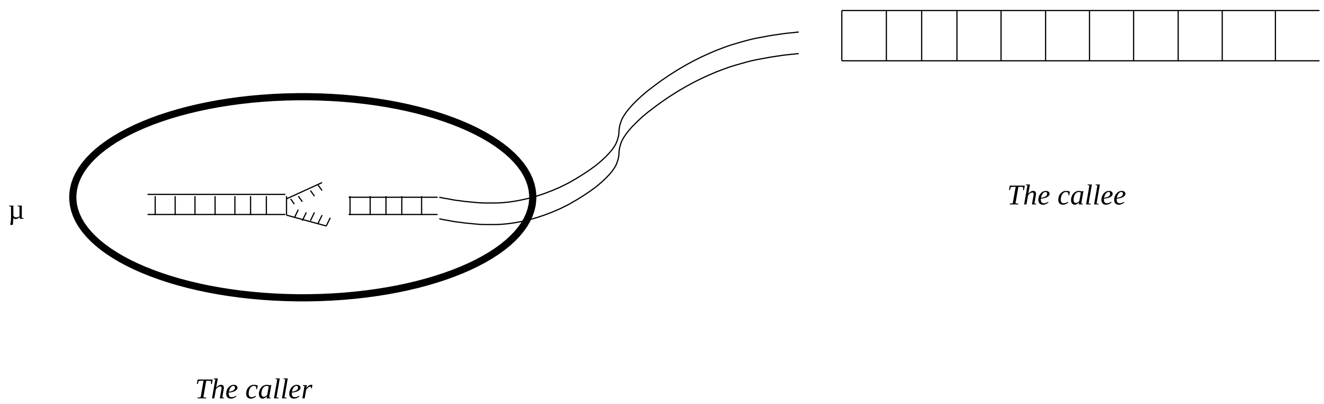
The model of computation relies on capsules



A model of computation

The model of computation relies on capsules





A model of computation : GEMINI

The model of computation relies on capsules $\langle r \parallel e \rangle$
that contain two constituents:

- a caller r
- and a callee e .

with the syntax

$$\begin{aligned} c &::= \langle r \parallel e \rangle \\ r &::= x \mid \lambda x.r \mid \mu\alpha.c \\ e &::= \alpha \mid r \bullet e \mid \tilde{\mu}x.c \end{aligned}$$

Callers

A caller is

- either a variable x,
- or a λ-abstraction $\lambda x.r$ which expects a value to take the place of x in r,
- or a μ-abstraction $\mu\alpha.c$ which expects a callee to take the place of α in c producing a new capsule.

Note: values and callers are the same.

Callees

A callee is basically a list of values, more precisely it is

- either a variable α,
- or a pair $r \bullet e$ of a value (caller) r and a callee e,
- or an $\widetilde{\mu}$-abstraction $\widetilde{\mu}x.c$

The reductions

$$
\begin{array}{llcl}
(\lambda) & \langle \lambda x \cdot r \parallel r' \bullet e \rangle & \longrightarrow & \langle r[x \leftarrow r'] \parallel e \rangle \\
(\mu) & \langle \mu\alpha \cdot c \parallel e \rangle & \longrightarrow & c[\alpha \leftarrow e] \\
(\widetilde{\mu}) & \langle r \parallel \widetilde{\mu} x \cdot c \rangle & \longrightarrow & c[x \leftarrow r]
\end{array}
$$

The reductions

$$
\begin{array}{llcl}
(\lambda) & \langle \lambda x \cdot r \parallel r' \bullet e \rangle & \longrightarrow & \langle r[x \leftarrow r'] \parallel e \rangle \\
(\mu) & \langle \mu\alpha \cdot c \parallel e \rangle & \longrightarrow & c[\alpha \leftarrow e] \\
(\widetilde{\mu}) & \langle r \parallel \widetilde{\mu}x \cdot c \rangle & \longrightarrow & c[x \leftarrow r]
\end{array}
$$

The system is ambiguous !

$\langle \mu\alpha.c \parallel \widetilde{\mu}x.c' \rangle$ has two possible reductions at the top.

$c[\alpha \leftarrow \widetilde{\mu}x.c'], \quad c[x \leftarrow \mu\alpha \cdot c]$ is a critical pair.

The reductions

$$\begin{array}{llcl} (\lambda) & \langle \lambda x \cdot r \,\|\, r' \bullet e \rangle & \longrightarrow & \langle r[x \leftarrow r'] \,\|\, e \rangle \\ (\mu) & \langle \mu\alpha \cdot c \,\|\, e \rangle & \longrightarrow & c[\alpha \leftarrow e] \\ (\widetilde{\mu}) & \langle r \,\|\, \widetilde{\mu}x \cdot c \rangle & \longrightarrow & c[x \leftarrow r] \end{array}$$

Can we type capsules, callers and callees?

- to prove that nothing wrong can happen, i.e., capsules reduces always to capsules,
- to prove termination, *i.e.*, a typed capsule always reduces to a normal form whatever strategy we adopt.

The link between

the sequent calculus

and

Herbelin’s calculus

The type judgments

Thanks to colors, I will consider three types of judgments

They can be seen as annotations of sequent calculus judgments;

Judgments for capsules

$$c : x_1 : A_1, \ldots, x_p : A_p \vdash \alpha_1 : B_1, \ldots, \alpha_q : B_q$$

or in short $c : \Gamma \vdash \Delta$,

Judgments for capsules

In $c : x_1 : A_1, \ldots, x_p : A_p \vdash \alpha_1 : B_1, \ldots, \alpha_q : B_q$,

one says that

- c takes the x_i as arguments with type A_i
- c returns α_j as results with type B_j.

Judgments for callers

$$x_1 : A_1, \ldots, x_p : A_p \vdash r : A, \alpha_1 : B_1, \ldots, \alpha_q : B_q$$

or in short $\Gamma \vdash r : A, \Delta$,

or $\Gamma \vdash \boxed{r : A}, \Delta$, when one does not have color.

Judgments for callees

$$x_1 : A_1, \ldots, x_p : A_p, e : A \vdash \alpha_1 : B_1, \ldots, \alpha_q : B_q$$

or in short $\Gamma, e : A \vdash \Delta$.

or $\Gamma, \boxed{e : A} \vdash \Delta$, when one does not have color.

The type system $G^{\rightarrow}$

$$\frac{}{\Gamma, \alpha : A \vdash \alpha : A, \Delta}\,(L-ax) \qquad \frac{}{\Gamma, x : A \vdash x : A, \Delta}\,(R-ax)$$

$$\frac{\Gamma \vdash r : A, \Delta \qquad \Gamma, e : B \vdash \Delta}{\Gamma, r \bullet e : A \rightarrow B \vdash \Delta}\,(\rightarrow L) \qquad \frac{\Gamma, x : A \vdash r : B, \Delta}{\Gamma \vdash \lambda x.r : A \rightarrow B, \Delta}\,(\rightarrow R)$$

$$\frac{\Gamma \vdash r : A, \Delta \qquad \Gamma, e : A \vdash \Delta}{\langle r \parallel e \rangle : (\Gamma \vdash \Delta)}\,(cut)$$

$$\frac{c : (\Gamma \vdash \beta : B, \Delta)}{\Gamma \vdash \mu\beta.c : B, \Delta}\,(\mu) \qquad \frac{c : (\Gamma, x : A \vdash \Delta)}{\Gamma, \widetilde{\mu}x.c : A \vdash \Delta}\,(\widetilde{\mu})$$

$$\frac{}{\Gamma, A \vdash \Delta, A}\,(L - ax) \qquad \frac{}{\Gamma, A \vdash \Delta, A}\,(R - ax)$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \to B \vdash \Delta}\,(\to L) \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \to B, \Delta}\,(\to R)$$

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}\,(cut) \qquad \frac{\Gamma \vdash B, \Delta}{\Gamma \vdash B, \Delta}\,(\mu)$$

$$\frac{}{\Gamma, \alpha : A \vdash \alpha : A, \Delta}\,(L - ax) \qquad \frac{}{\Gamma, x : A \vdash x : A, \Delta}\,(R - ax)$$

$$\frac{\Gamma \vdash r : A, \Delta \quad \Gamma, e : B \vdash \Delta}{\Gamma, r \bullet e : A \to B \vdash \Delta}\,(\to L) \qquad \frac{\Gamma, x : A \vdash r : B, \Delta}{\Gamma \vdash \lambda x.r : A \to B, \Delta}\,(\to R)$$

$$\frac{\Gamma \vdash r : A, \Delta \quad \Gamma, e : A \vdash \Delta}{\langle r \parallel e \rangle : (\Gamma \vdash \Delta)}\,(cut) \qquad \frac{c : (\Gamma \vdash \beta : B, \Delta)}{\Gamma \vdash \mu\beta.c : B, \Delta}\,(\mu)$$

Curry-Howard correspondence

One gets a Curry-Howard correspondence, namely

- terms are proofs,
- types are propositions,
- term reductions are proof simplifications (normalization).

Peirce law again

Let T be $(A \to B) \to A$.

$$\dfrac{x : T, y : A \vdash y : A, \beta : B, \alpha : A \qquad x : T, y : A, \alpha : A \vdash \alpha : A, \beta : B}{\langle y \parallel \alpha \rangle : (x : T, y : A \vdash \beta : B, \alpha : A)}\,(cut)$$

$$\dfrac{\langle y \parallel \alpha \rangle : (x : T, y : A \vdash \beta : B, \alpha : A)}{x : T, y : A \vdash \mu\beta.\langle y \parallel \alpha \rangle : B, \alpha : A}\,(\mu)$$

$$\dfrac{x : T, y : A \vdash \mu\beta.\langle y \parallel \alpha \rangle : B, \alpha : A}{x : T \vdash \lambda y.\mu\beta.\langle y \parallel \alpha \rangle : A \to B, \alpha : A}\,(\to R)$$

$$\dfrac{x : T \vdash \lambda y.\mu\beta.\langle y \parallel \alpha \rangle : A \to B, \alpha : A \qquad x : T, \alpha : A \vdash \alpha : A}{x : T, (\lambda y.\mu\beta.\langle y \parallel \alpha \rangle) \bullet \alpha : T \vdash \alpha : A}\,(\to L)$$

Peirce law again

Let T be $(A \to B) \to A$.

$$\dfrac{x : T, y : A \vdash y : A, \beta : B, \alpha : A \qquad x : T, y : A, \alpha : A \vdash \alpha : A, \beta : B}{\langle y \parallel \alpha \rangle : (x : T, y : A \vdash \beta : B, \alpha : A)} \, (cut)$$

$$\dfrac{\langle y \parallel \alpha \rangle : (x : T, y : A \vdash \beta : B, \alpha : A)}{x : T, y : A \vdash \mu\beta.\langle y \parallel \alpha \rangle : B, \alpha : A} \, (\mu)$$

$$\dfrac{x : T, y : A \vdash \mu\beta.\langle y \parallel \alpha \rangle : B, \alpha : A}{x : T \vdash \lambda y.\mu\beta.\langle y \parallel \alpha \rangle : A \to B, \alpha : A} \, (\to R)$$

$$\dfrac{x : T \vdash \lambda y.\mu\beta.\langle y \parallel \alpha \rangle : A \to B, \alpha : A \qquad x : T, \alpha : A \vdash \alpha : A}{x : T, (\lambda y.\mu\beta.\langle y \parallel \alpha \rangle) \bullet \alpha : T \vdash \alpha : A} \, (\to$$

is called $\mathcal{A}$ in the following screens.

The tree for typing Peirce law is

$$\dfrac{\dfrac{\dfrac{x : T \vdash x : T, \alpha : A \quad \mathcal{A}}{\langle x \mid (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : (x : T \vdash \alpha : A)}\,(cut)}{x : T \vdash \mu\alpha.\langle x \parallel (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : A,}\,(\mu)}{\vdash \lambda x.\mu\alpha.\langle x \parallel (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : ((A \to B) \to A) \to A,}\,(\to L)$$

The tree for typing Peirce law is

$$\dfrac{\dfrac{\dfrac{x : T \vdash x : T, \alpha : A \quad \mathcal{A}}{\langle x \mid (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : (x : T \vdash \alpha : A)}\,(cut)}{x : T \vdash \mu\alpha.\langle x \parallel (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : A,}\,(\mu)}{\vdash \lambda x.\mu\alpha.\langle x \parallel (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle : ((A \to B) \to A) \to A,}\,(\to L)$$

The term with type the Pierce law is

$$\lambda x.\mu\alpha.\langle x \parallel (\lambda y.\mu\beta.\langle y \parallel \alpha\rangle) \bullet \alpha\rangle.$$

Reductions as simplifications of proofs

Reductions are simplifications (normalizations) of proofs

Let us look at

$$(\lambda) \quad \langle \lambda x \cdot r \parallel r' \bullet e \rangle \quad \longrightarrow \quad \langle r[x \leftarrow r'] \parallel e \rangle$$

It corresponds to

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{\Gamma, x : A \vdash r : B, \Delta}}{\Gamma \vdash \lambda x.r : A \to B, \Delta}(\to R) \quad \dfrac{\Gamma \vdash r' : A, \Delta \quad \Gamma, e : B \vdash \Delta}{\Gamma, r' \bullet e : A \to B \vdash \Delta}(\to L)}{\langle \lambda x \cdot r \parallel r' \bullet e \rangle : \Gamma \vdash \Delta}(cut)$$

and

$$\dfrac{\dfrac{\mathcal{D}[x \leftarrow r']}{\Gamma, \vdash r[x \leftarrow r'] : B, \Delta} \quad \Gamma, e : B \vdash \Delta}{\langle r[x \leftarrow r'] \parallel e \rangle : \Gamma \vdash \Delta}(cut)$$

It corresponds to

$$\dfrac{\dfrac{\dfrac{\mathcal{D}}{\Gamma, x : A \vdash r : B, \Delta}}{\Gamma \vdash \lambda x.r : A \to B, \Delta}(\to R) \quad \dfrac{\Gamma \vdash r' : A, \Delta \quad \Gamma, e : B \vdash \Delta}{\Gamma, r' \bullet e : A \to B \vdash \Delta}(\to L)}{\langle \lambda x \cdot r \parallel r' \bullet e \rangle : \Gamma \vdash \Delta}(cut)$$

and

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash r' : A, \Delta}{} \quad \dfrac{\Gamma \vdash r' : A, \Delta}{} \quad \mathcal{D}}{\Gamma, \vdash r[x \leftarrow r'] : B, \Delta} \quad \Gamma, e : B \vdash \Delta}{\langle r[x \leftarrow r'] \parallel e \rangle : \Gamma \vdash \Delta}(cut)$$

Termination or strong normalization

If c is typable in $G^{\rightarrow}$, then c does not start a non terminating reduction.

Characterizing strongly normalizable terms

Characterization of strong normalization

We want to characterize strong normalization by a type system.

Hence a system with intersection.

The intersection part of the system $G^{\cap}$

$$\frac{\Gamma, x : A, e : C \vdash \Delta}{\Gamma, x : A \cap B, e : C \vdash \Delta} \qquad \frac{\Gamma, x : A \vdash v : C, \Delta}{\Gamma, x : A \cap B \vdash v : C, \Delta} (\cap L_v)$$

$$\frac{\Gamma \vdash v : A, \Delta \qquad \Gamma \vdash v : B, \Delta}{\Gamma \vdash v : A \cap B, \Delta} (\cap R_v)$$

$$\frac{\Gamma, e : A \vdash \Delta \qquad \Gamma, e : B \vdash \Delta}{\Gamma, e : A \cap B \vdash \Delta} (\cap L_e)$$

$$\frac{\Gamma, e : C \vdash \Delta, \alpha : A}{\Gamma, e : C \vdash \Delta, \alpha : A \cap B} \qquad \frac{\Gamma \vdash v : C, \Delta, \alpha : A}{\Gamma \vdash v : C, \Delta, \alpha : A \cap B} (\cap R_e)$$

We have only $\cap$-introduction rules.

We have no $\cap$-introduction rules for variables.

Characterization of strongly normalizing terms

Theorem: c is typable if and only if c is strongly normalizing.

Typability of strongly normalizing terms

The proof is based on a perpetual strategy.

$$
\begin{array}{lcl}
\text{perpc}\ \langle \mu\alpha.c \parallel \widetilde{\mu}x.c' \rangle & = & \textbf{if}\ c[\alpha \leftarrow \widetilde{\mu}x.c'] \notin SN_c\ \textbf{then}\ c[\alpha \leftarrow \widetilde{\mu}x.c'] \\
 & & \textbf{else}\ c'[x \leftarrow \mu\alpha.c] \\
\text{perpc}\ \langle \mu\alpha.c \parallel e \rangle & = & (\textbf{assume}\ e \neq \widetilde{\mu}x.c') \\
 & & \quad \textbf{if}\ \alpha \in \mathtt{Fv_e}(c)\ \textbf{or}\ \mathsf{nf}(e)\ \textbf{then}\ c[\alpha \leftarrow e] \\
 & & \quad \textbf{else}\ \textbf{then}\ \langle \mu\alpha.c \parallel \text{perpe}\ e \rangle \\
\text{perpc}\ \langle r \parallel \widetilde{\mu}x.c \rangle & = & (\textbf{assume}\ r \neq \mu\alpha.c) \\
 & & \quad \textbf{if}\ x \in \mathtt{Fv_r}(c)\ \textbf{or}\ \mathsf{nf}(r)\ \textbf{then}\ c[x \leftarrow r] \\
 & & \quad \textbf{else}\ \langle \text{perpr}\ r \parallel \widetilde{\mu}x.c \rangle \\
\text{perpc}\ \langle \lambda x.r \parallel r' \bullet e' \rangle & = & \textbf{if}\ x \in \mathtt{Fv_r}(r)\ \textbf{or}\ \mathsf{nf}(r')\ \textbf{then}\ \langle r[x \leftarrow r'] \parallel e' \rangle \\
 & & \textbf{else}\ \langle \lambda x.r \parallel (\text{perpr}\ r') \bullet e' \rangle \\
\text{perpc}\ \langle y \parallel e \rangle & = & (\textbf{assume}\ e \neq \widetilde{\mu}x.c') \\
 & & \quad \textbf{if}\ \mathsf{nf}(e)\ \textbf{then}\ unit \\
 & & \quad \textbf{else}\ \langle y \parallel \text{perpe}\ e \rangle \\
\text{perpc}\ \langle r \parallel \beta \rangle & = & (\textbf{assume}\ r \neq \mu\alpha.c) \\
 & & \quad \textbf{if}\ \mathsf{nf}(r)\ \textbf{then}\ unit \\
 & & \quad \textbf{else}\ \langle \text{perpr}\ r \parallel \beta \rangle
\end{array}
$$

$$
\begin{array}{rcll}
\text{perpr } \lambda x.r & = & \textbf{if } \mathsf{nf}(r) \textbf{ then } unit & :: \quad \nu\varphi : \text{Unit} \\
 & & \textbf{else } \lambda x.(\text{perpr } r) & :: \quad \rho : \text{CalleR} \\
\text{perpr } \mu\alpha.c & = & \textbf{if } \mathsf{nf}(c) \textbf{ then } unit & :: \quad \nu\varphi : \text{Unit} \\
 & & \textbf{else } \mu\alpha.(\text{perpc } c) & :: \quad \rho : \text{CalleR} \\
\text{perpr } x & = & unit & :: \quad \nu\varphi : \text{Unit}
\end{array}
$$

$$
\begin{array}{rcll}
\text{perpe } r \bullet e & = & \textbf{if } \mathsf{nf}(r) \textbf{ and } \mathsf{nf}(e) \textbf{ then } unit & :: \quad \nu\varphi : \text{Unit} \\
 & & \textbf{if } \mathsf{nf}(r) \textbf{ and } \neg\mathsf{nf}(e) \textbf{ then } r \bullet (\text{perpe } e) & :: \quad \epsilon : \text{CalleE} \\
 & & \textbf{if } \neg\mathsf{nf}(r) \textbf{ and } \neg\mathsf{nf}(e) \textbf{ then } (\text{perpr } r) \bullet e & :: \quad \epsilon : \text{CalleE} \\
\text{perpe } \widetilde{\mu}x.c & = & \textbf{if } \mathsf{nf}(c) \textbf{ then } unit & :: \quad \nu\varphi : \text{Unit} \\
 & & \textbf{else } \widetilde{\mu}x.(\text{perpc } c) & :: \quad \epsilon : \text{CalleE} \\
\text{perpe } \alpha & = & unit & :: \quad \nu\varphi : \text{Unit}
\end{array}
$$

Typability of strongly normalizing terms

If

- $c \in SN_c$
- and for all c' such that

  $h(c')\langle h(c)$ (where $h(c)$ is the length of the longest reduction at c),

  c' is typable,

then c is typable.

One considers the typability of those such that $h(c') =$ perpc c

except in one case.

Strong normalization of typable terms

We use a new kind of reducibles sets

based on maximal stable pairs.

Stable pairs

Two sets $\emptyset \subset X \subseteq$ CalleR and $\emptyset \subset Y \subseteq$ CalleE

form a non trivial pair (X, Y).

The pair (X, Y) is stable if

for $r \in X$ and $e \in Y$ the capsule $\langle r \parallel e \rangle$ is SN.

Stable pairs

Two sets $\subset X \subseteq$ CalleR and $\subset Y \subseteq$ CalleE

form a non trivial pair.

The pair (X, Y) is stable if

for $r \in X$ and $e \in Y$ the capsule $\langle r \parallel e \rangle$ is SN.

(Var_r, Var_e) is stable.

Maximal stable

We define a partial order on pairs.

$$\mathcal{P} \sqsubseteq \mathcal{Q} \quad \text{iff} \quad \mathcal{P}_r \subseteq \mathcal{Q}_r \land \mathcal{P}_e \subseteq \mathcal{Q}_e$$

Lemma: Let $\mathcal{P}$ be a maximal stable.

1. For each $r \in \Lambda_r$: $\quad r \in \mathcal{P}_r$ if for all $e \in \mathcal{P}_e, \langle r \parallel e \rangle$ is SN.
2. For each $e \in \Lambda_e$: $\quad e \in \mathcal{P}_e$ if for all $r \in \mathcal{P}_r, \langle r \parallel e \rangle$ is SN.

Operations on stables

- $\mathcal{P} \bigsqcap \mathcal{Q}$ is given by
  - $(\mathcal{P} \bigsqcap \mathcal{Q})_r = \mathcal{P}_r \bigcap \mathcal{Q}_r$
  - $(\mathcal{P} \bigsqcap \mathcal{Q})_e = \mathcal{P}_e \bigcap \mathcal{Q}_e$
- $\mathcal{P} \hookrightarrow \mathcal{Q}$ is given by
  - $(\mathcal{P} \hookrightarrow \mathcal{Q})_r = \{\lambda x.r \mid \forall r' \in \mathcal{P}_r, r[x \leftarrow r'] \in \mathcal{Q}_r\}$
  - $(\mathcal{P} \hookrightarrow \mathcal{Q})_e = \{r \bullet e \mid r \in \mathcal{P}_r \text{ and } e \in \mathcal{Q}_e\}$

Operations on stables

- $\mathcal{P} \prod \mathcal{Q}$ is given by
  - $(\mathcal{P} \prod \mathcal{Q})_r = \mathcal{P}_r \bigcap \mathcal{Q}_r$
  - $(\mathcal{P} \prod \mathcal{Q})_e = \mathcal{P}_e \bigcap \mathcal{Q}_e$
- $\mathcal{P} \hookrightarrow \mathcal{Q}$ is given by
  - $(\mathcal{P} \hookrightarrow \mathcal{Q})_r = \{\lambda x.r \mid \forall r' \in \mathcal{P}_r, r[x \leftarrow r'] \in \mathcal{Q}_r$
  - $(\mathcal{P} \hookrightarrow \mathcal{Q})_e = \{r \bullet e \mid r \in \mathcal{P}_r \text{ and } e \in \mathcal{Q}_e\}$

Lemma If $\mathcal{P}$ and $\mathcal{Q}$ are non trivial stable
then $\mathcal{P} \prod \mathcal{P}$ and $\mathcal{P} \hookrightarrow \mathcal{P}$ are stable.

Reducible sets $\mathcal{S}^A$

The set $\mathcal{S} = \{\mathcal{S}^T \mid T \text{ a type}\}$ is defined as follows:

- When T is a basic type, $\mathcal{S}^T$ is any maximal stable pair.
- $\mathcal{S}^{A \to B}$ is any maximal stable extension of $\mathcal{S}^A \hookrightarrow \mathcal{S}^B$.
- $\mathcal{S}^{A \cap B}$ is any maximal stable extension of $\mathcal{S}^A \sqcap \mathcal{S}^B$.

Proposition: If t is typable of type A then $t \in \mathcal{S}^A$.

Strong normalization of typable

Theorem: Terms typable in $G^{\cap}$ are SN.

Corollary: Terms typable in $G^{\rightarrow}$ are SN.

Conclusion

We have a proof of the strong normalization of typable terms.

We know how to deal with an ambiguous (*i.e.,*non confluent) system.

We can characterize strongly normalizing terms.