Notes on Linear Logic
The Chambery-Torino school of Theoretical Computer Science
Aussois, January 29th to February 1st, 2003

Laurent Regnier

CNRS - Institut de Mathématique de Luminy

regnier@iml.univ-mrs.fr

Introduction

Linear Logic is:

- a Curry-Howard logic;
- a decomposition of intuitionistic logic;
- a symetrization of intuitionistic logic;
- a constructivization of classical logic;
- a new syntax for proofs;

To summarize: a useful tool for studying programming languages, particularly (extensions of) lambda-calculus.

Coherent semantics

Coherent spaces

Coherent space X =

- a set $|X|$ called the *web* of X;
- a reflexive, symetric relation $\underset{X}{\frown}$ on $|X|$ called the *coherence*.

A *clique* of X is a set $x \subset |X|$ such that

$$\text{for all } a, b \text{ in } x,\ a \underset{X}{\frown} b$$

$\mathcal{C}(X)$ is the set of cliques of X.

An equivalent definition

Let $|X|$ be a set and $x, y \subset |X|$; we define

$$x \perp y \text{ iff } |x \cap y| \leq 1$$

If $C \subset \mathcal{P}(|X|)$ then

$$C^{\perp} = \{y \text{ s.t. for all } x \in C, x \perp y\}$$

A coherent space is a pair $(|X|, \mathcal{C}(X))$ where $|X|$ is a set and $\mathcal{C}(X) \subset \mathcal{P}(X)$ is such that:

$$\mathcal{C}(X)^{\perp\perp} = \mathcal{C}(X)$$

Stable functions

A function $F : \mathcal{C}(X) \to \mathcal{C}(Y)$ is *stable* if:

- it is continuous (increasing + commutation to directed unions);
- it commutes to compatible intersections:

  for all x, y in $\mathcal{C}(X)$, if $x \cup y \in \mathcal{C}(X)$ then $F(x \cap y) = F(x) \cap F(y)$

Lemma 1 (Fundamental lemma of stable functions) *F is stable iff*

- *it is increasing;*
- *for any $x \in \mathcal{C}(X)$ and any $b \in F(x)$ there is $x_0 \subset_{\text{fin}} x$ such that:*
  - *$b \in F(x_0)$;*
  - *for all $y \subset x$, if $b \in F(y)$ then $x_0 \subset y$.*

The space $X \to Y$

The *trace* of $F : X \to Y$ is

$$\operatorname{Tr} F = \{(x_0, b) \text{ s.t. } x_0 \in \mathcal{C}_{\text{fin}}(X) \text{ is minimal so that } b \in F(x_0)\}$$

The space $X \to Y$ is defined by:

- $|X \to Y| = \mathcal{C}_{\text{fin}}(|X|) \times |Y|$;
- $(x_0, b) \frown_{X \to Y} (y_0, c)$ iff $x_0 \cup y_0 \notin \mathcal{C}(X)$ or $b \frown_X c$
  iff $x_0 \cup y_0 \in \mathcal{C}(X) \Rightarrow b \frown_X c$

$X \to Y$ is the coherent space of (traces of) stable functions. In particular given $f \in \mathcal{C}(X \to Y)$ then $\operatorname{Fun} f$ is the stable function defined by:

$$\operatorname{Fun} f(x) = \{a \text{ s.t. there is an } x_0 \subset_{\text{fin}} x,\ (x_0, a) \in f\}$$

Linear maps

A stable function is *linear* if it commutes to any unions and not only directed ones.

Lemma 2 *$F : X \to Y$ is linear iff it is stable and for any $(x_0, b) \in \mathrm{Tr}\, F$, x_0 is a singleton*

The *linear trace* of a linear map F is

$$\mathrm{Tr}_l\, F = \{(a, b) \in |X| \times |Y|, b \in F(\{a\})\}$$

$X \multimap Y$ is the space of linear traces defined by:

- $|X \multimap Y| = |X| \times |Y|$;
- $(a, b) \frown_{X \multimap Y} (a', b')$ iff $a \smile_X a'$ or $b \frown_Y b'$.

The decomposition of the intuitionistic arrow

Let $!X$ be the coherent space defined by:

- $|!X| = \mathcal{C}_{\text{fin}}(X)$;
- $x_0 \underset{!X}{\frown} y_0$ iff $x_0 \cup y_0 \in \mathcal{C}(X)$.

Then we have:

$$X \rightarrow Y = !X \multimap Y$$

Linear connectives: multiplicatives

Linear negation: $|X^\perp| = |X|$, $a \frown\!\!\!\!\smile_{X^\perp} b$ iff $a = b$ or $\neg a \frown\!\!\!\!\smile_X b$;

tensor: $|X \otimes Y| = |X| \times |Y|$;

$(a, b) \frown\!\!\!\!\smile_{X \otimes Y} (a', b')$ iff $a \frown\!\!\!\!\smile_X a'$ and $b \frown\!\!\!\!\smile_Y b'$;

par: $|X \parr Y| = |X| \times |Y|$;

$(a, b) \frown_{X \parr Y} (a', b')$ iff $a \frown_X a'$ or $b \frown_Y b'$;

linear implication: $X \multimap Y = X^\perp \parr Y$.

one and bottom: $|1| = |\perp| = \{*\}$;

Linear connectives: additives

with: $|X_1 \& X_2| = |X_1| + |X_2|$;

$a \frown_{X_1 \& X_2} b$ iff $a, b \in |X_i|$, $a \frown_{X_i} b$ for $i = 1$ or 2

or $a \in |X_i|$, $b \in |X_j|, i \neq j$.

$X \& Y$ is the cartesian product of X_1 and X_2 (w.r.t stable and linear functions).

plus: $|X_1 \oplus X_2| = |X_1| + |X_2|$;

$a \frown_{X_1 \oplus X_2} b$ iff $a, b \in |X_i|$, $a \frown_{X_i} b$ for $i = 1$ or 2

$X \oplus X_2$ is the direct sum of X and Y w.r.t. linear functions;

top and zero: $|\top| = |0| = \emptyset$.

Exponential connectives

Of course: $|!X| = \mathcal{C}_{\mathrm{fin}}(X)$;

$$x_0 \underset{!X}{\frown} y_0 \text{ iff } x_0 \cup y_0 \in \mathcal{C}(X);$$

Why not: $?X = (!X^\perp)^\perp$, in particular

$|?X| = \{$finite anticliques of $X\}$.

Isomorphisms

commutativity and associativity isos.

Neutrals

$$X \otimes 1 \sim X, \quad X \parr \perp \sim X, \quad X \& \top \sim X, \quad X \oplus 0 \sim X$$

de Morgan laws

$$(X^\perp)^\perp = X$$

$$(X \otimes Y)^\perp = X^\perp \parr Y^\perp, \quad (X \& Y)^\perp = X^\perp \oplus Y^\perp, \quad (!X)^\perp = ?X^\perp$$

$$Y^\perp \multimap X^\perp \sim X \multimap Y$$

Distributivity: $X \otimes (Y \oplus Z) \sim (X \otimes Y) \oplus (X \otimes Z) \qquad X \otimes 0 \sim 0$

The exponential iso: $!(X \& Y) \sim !X \otimes !Y \qquad !\top = 1$

Important maps

projections: $\pi_i : X_1 \& X_2 \multimap X_i$;

pairing: $\langle F_1, F_2 \rangle : X \multimap Y_1 \& Y_2$ given $F_i : X \multimap Y_i$;

tensoring: $F_1 \otimes F_2 : X_1 \otimes X_2 \multimap Y_1 \otimes Y_2$ given $F_i : X_i \multimap Y_i$;

Exponential maps

functorial promotion: given $F : X \multimap Y$, we define $!F : !X \multimap !Y$ by:

$$\mathrm{Tr_l}\, !F = \{(\{a_1, \ldots, a_n\}, \{b_1, \ldots, b_n\}), \text{ s.t. } \{a_1, \ldots, a_n\} \in \mathcal{C}(X) \text{ and } (a_i, b_i) \in \mathrm{Tr_l}\, F \text{ for all } i\}$$

weakening: $w_X : !X \multimap 1$: $\mathrm{Tr_l}\, w_X = \{(\emptyset, *)\}$;

dereliction: $d_X : !X \multimap X$: $\mathrm{Tr_l}\, d_X = \{(\{a\}, a),\ a \in |X|\}$;

contraction: $c_X : !X \multimap !X \otimes !X$:

$$\mathrm{Tr_l}\, c_X = \{(x_1 \cup x_2, (x_1, x_2)), \text{ s.t. } x_1, x_2 \in \mathcal{C}_{\mathrm{fin}}(X)\};$$

digging: $\mathrm{dig}_X : !X \multimap !!X$:

$$\mathrm{Tr_l}\, \mathrm{dig}_X = \{(x, \{x_1, \ldots, x_n\}), \text{ s.t. } x = x_1 \cup \cdots \cup x_n \text{ and } x, x_1, \ldots, x_n \in \mathcal{C}_{\mathrm{fin}}(X)\}.$$

Syntax of LL

Sequent calculus

Formulas are considered up to de Morgan laws. Sequents are sequence of (occurrences of) formulas considerer *up to permutation*: $\vdash A_1, \ldots, A_n$.

Id:

$$\frac{}{\vdash A^\perp, A} \qquad \frac{\vdash A^\perp, \Gamma \qquad \vdash A, \Delta}{\vdash \Gamma, \Delta}\ \text{cut}$$

Mult:

$$\frac{\vdash A, B, \Gamma}{\vdash A \parr B, \Gamma} \qquad \frac{\vdash \Gamma}{\vdash, \perp, \Gamma} \qquad \frac{\vdash A, \Gamma \qquad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta} \qquad \frac{}{\vdash 1}$$

Add:

$$\frac{\vdash A, \Gamma \qquad \vdash B, \Gamma}{\vdash A \& B, \Gamma} \qquad \frac{}{\vdash \top, \Gamma} \qquad \frac{\vdash A_i, \Gamma}{\vdash A_1 \oplus A_2, \Gamma}\ i = 1, 2$$

Exp:

$$\frac{\vdash A, \Gamma}{\vdash ?A, \Gamma}\ \text{der}$$

$$\frac{\vdash \Gamma}{\vdash ?A, \Gamma}\ \text{weak} \qquad \frac{\vdash ?A, ?A\Gamma}{?A, \Gamma}\ \text{cont} \qquad \frac{\vdash A, ?\Gamma}{\vdash !A, ?\Gamma}\ \text{prom}$$

Coherent semantics

Use the distinguished maps and isos to interpret LL proofs.

The category of coherent space and linear maps is a *linear category*:

- *-autonomous;
- monoidal comonad !, with in particular a natural transformation $\mathsf{m}_{A,B} : !A \otimes !B \multimap !(A \otimes B)$.

The CCC of stable maps is the co-Kleisli of the category of linear maps.

Important note: other comonads satisfying these conditions may be defined, typically the *multiset*-! (free comonoid).

LL subsystems and fragments

ILL: Intuitionistic Linear Logic; obtained by reformulating with two sided sequents and constraining sequents to contain at most one formula on the right;

MLL: Multiplicative Linear Logic: formulas are constrained to contain only multiplicative connectives;

MALL: Multiplicative-Additive LL, the "linear" fragment of Linear Logic;

MELL: Multiplicative-Exponential LL;

the intuitionistic variants: IMLL, IMALL, IMELL;

NL: Non commutative Linear Logic;

ELL, LLL: Elementary and Light linear logic;

$\mathbf{LL}_{\mathrm{pol}}$, LLP: Polarized linear logic.

Proof-nets: the natural deduction for MLL

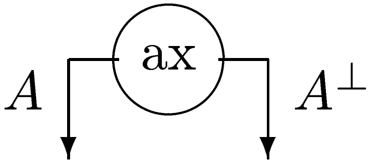
An MLL *proof-structure* is a dag; nodes (also called *links*) are labelled by one of: ax, cut, concl, $\otimes$, $\parr$; edges are labelled (or *typed*) by MLL formulas; the following *typing conditions* are further required:

- each concl node has exactly one *premisse* (entering edge) and no *conclusion* (exiting edge); the types of the premisses of all concl links are the conclusions of the overall proof-structure. In drawings concl links are often omitted, leaving their premisses as pending edges.
- each ax link has no premisse and two conclusions typed by dual formulas;
- each cut link has two premisses typed by dual formulas and no conclusions;
- each $\otimes$ (resp. $\parr$) link has two premisses typed by A and B and one conclusion typed by $A \otimes B$ (resp. $A \parr B$).

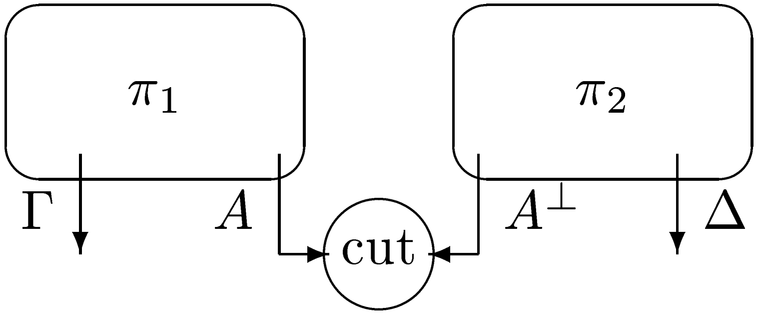
Translation of sequent calculus

A proof of an MLL sequent $\vdash A_1, \ldots, A_n$ is translated as a proof-structure with conclusions $A_1, \ldots, A_n$ as follows:

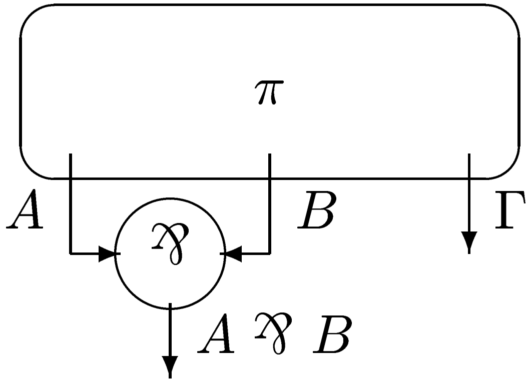
$$\frac{}{\vdash A, A^\perp} \quad \rightsquigarrow$$



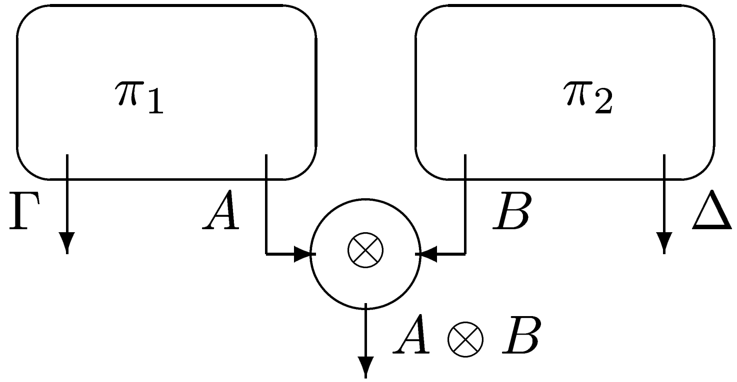
$$\frac{\begin{array}{c}\pi_1\\ \vdots \\ \vdash \Gamma, A\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots \\ \vdash A^\perp, \Delta\end{array}}{\vdash \Gamma, \Delta} \quad \rightsquigarrow$$



$$\dfrac{\begin{array}{c}\pi\\ \vdots\\ \vdash A, B, \Gamma\end{array}}{\vdash A \parr B, \Gamma} \quad \rightsquigarrow$$



$$\dfrac{\begin{array}{c}\pi_1\\ \vdots\\ \vdash \Gamma, A\end{array} \qquad \begin{array}{c}\pi_2\\ \vdots\\ \vdash B, \Delta\end{array}}{\vdash \Gamma, A \otimes B, \Delta} \quad \rightsquigarrow$$

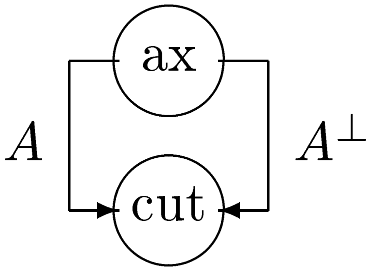


Proof-net

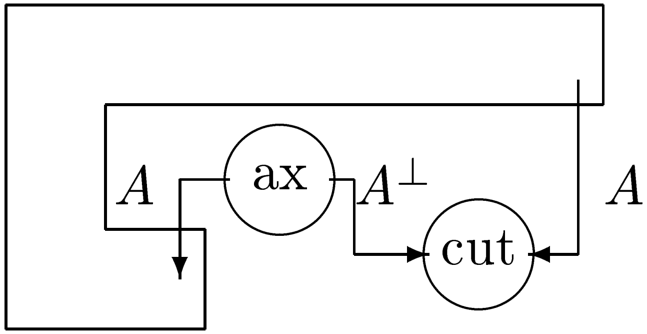
A *proof-net* is a proof-structure obtained by translating a sequent calculus proof.

There is an intrinsic caracterization of proof-nets called the *correctness condition.*

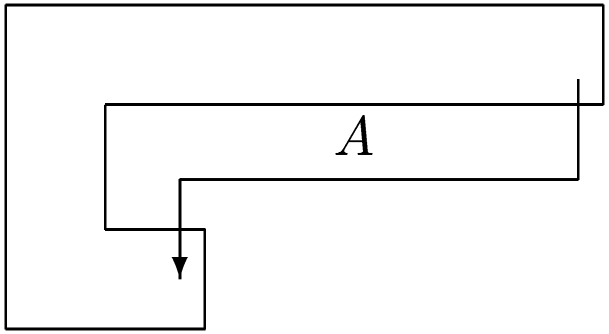
An incorrect proof-structure:

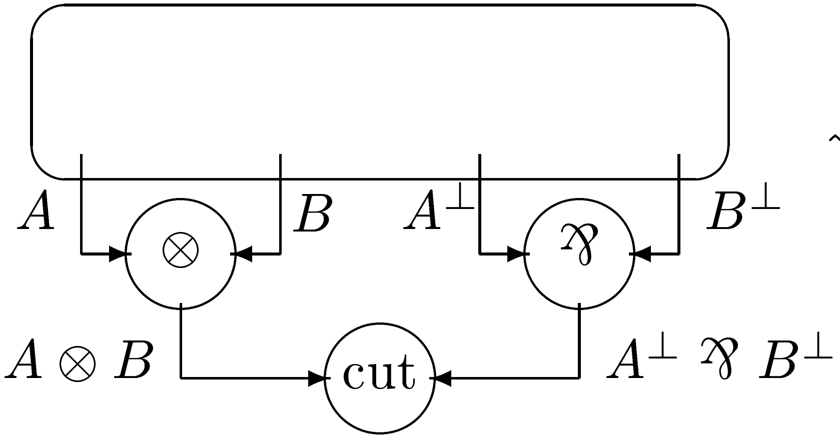


Cut elimination

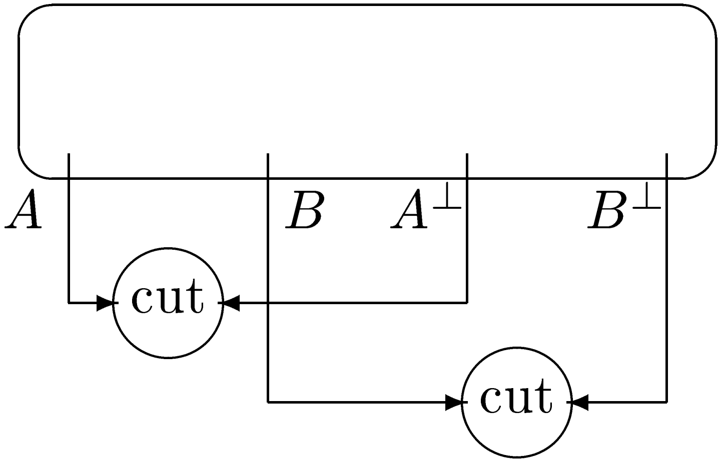


$\rightsquigarrow$





$\rightsquigarrow$



Exponential links

Add three new types of node:

dereliction: labelled with d: one premisse typed by A, one conclusion typed by $?A$;

weakening: labelled with w: no premisse, one conclusion labelled with $?A$;

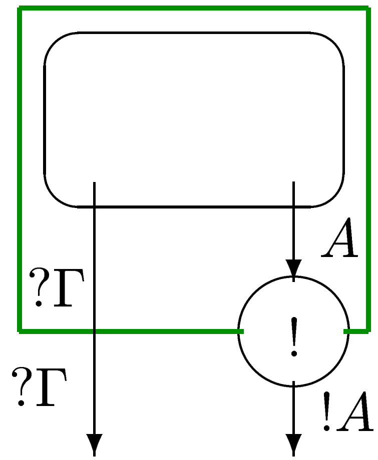
contraction: labelled with c: two premisses both typed by $?A$, one conclusion typed by $?A$;

promotion: also called *box*, no premisse, one conclusion with type $!A$ and additionnaly, any number of conclusions with type of the form $?B$.

MELL proof-structures

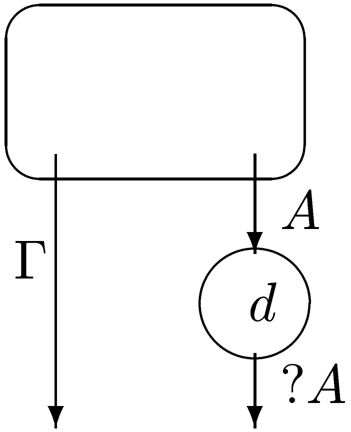
A MELL proof-structure is a proof-structure containing the nodes described so far together with: for each box link of conclusions $?\Gamma, !A$, an associated proof-structure with conclusions $?\Gamma, A$.

Boxes are drawn by putting their associated proof-structure inside a frame:

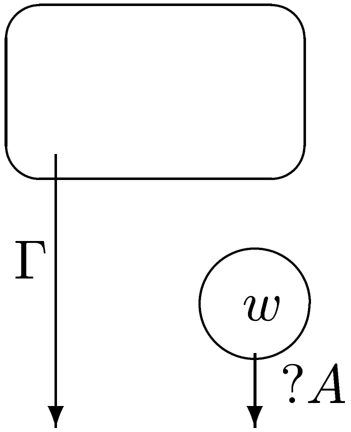


Translation of exponential rules

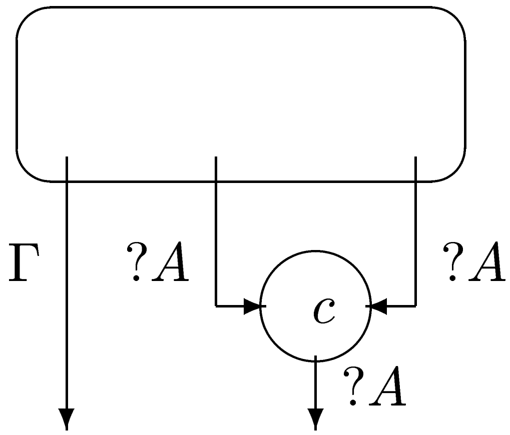
$$\frac{\begin{array}{c}\vdots\\ \vdash \Gamma, A\end{array}}{\vdash \Gamma, ?A} \quad \rightsquigarrow$$



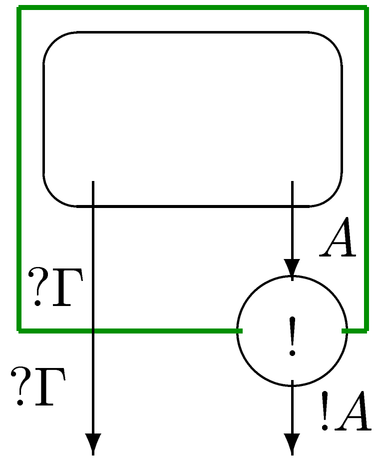
$$\frac{\begin{array}{c}\vdots\\ \vdash \Gamma\end{array}}{\vdash \Gamma, ?A} \quad \rightsquigarrow$$



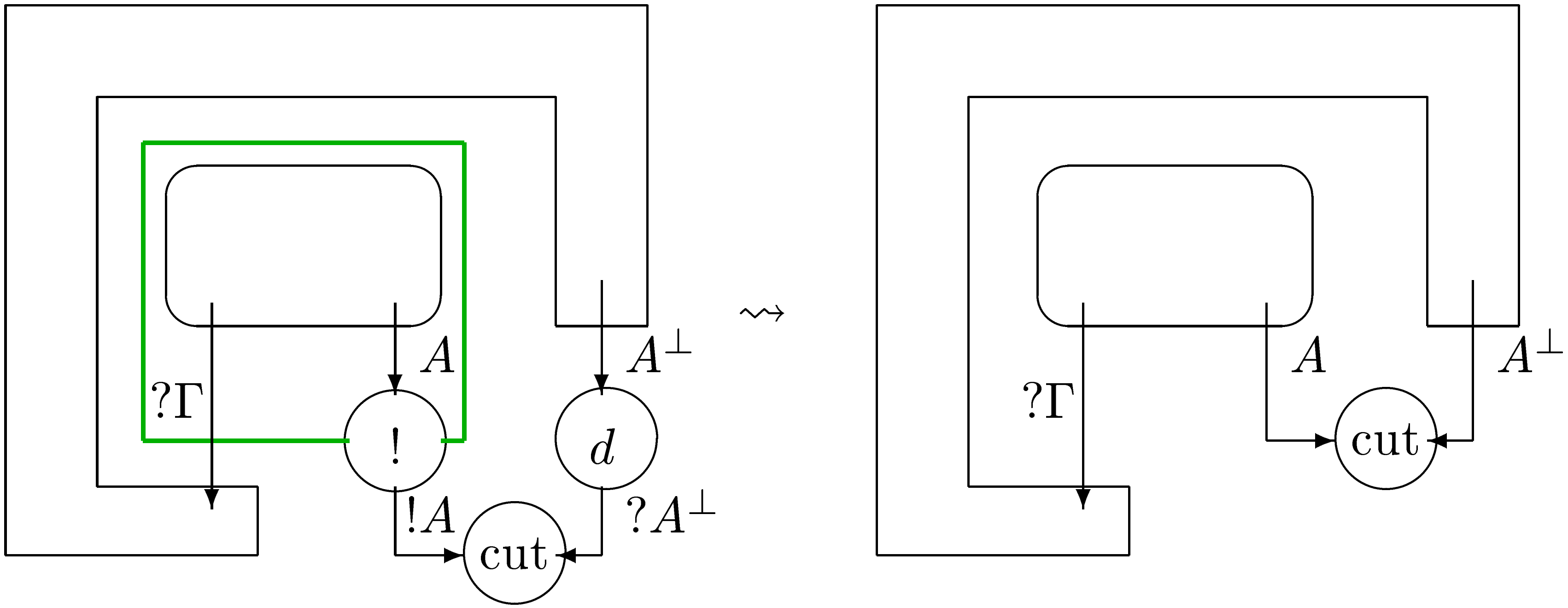
$$\dfrac{\begin{array}{c}\vdots\\ \vdash \Gamma, ?A, ?A\end{array}}{\vdash \Gamma, ?A} \rightsquigarrow$$

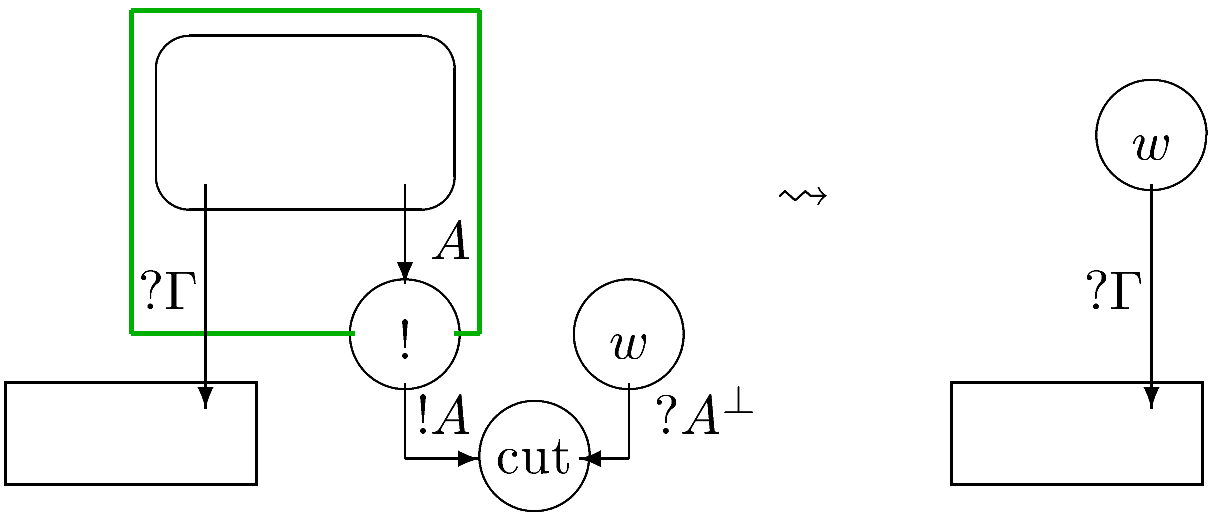


$$\dfrac{\begin{array}{c}\vdots\\ \vdash ?\Gamma, A\end{array}}{\vdash ?\Gamma, !A} \rightsquigarrow$$

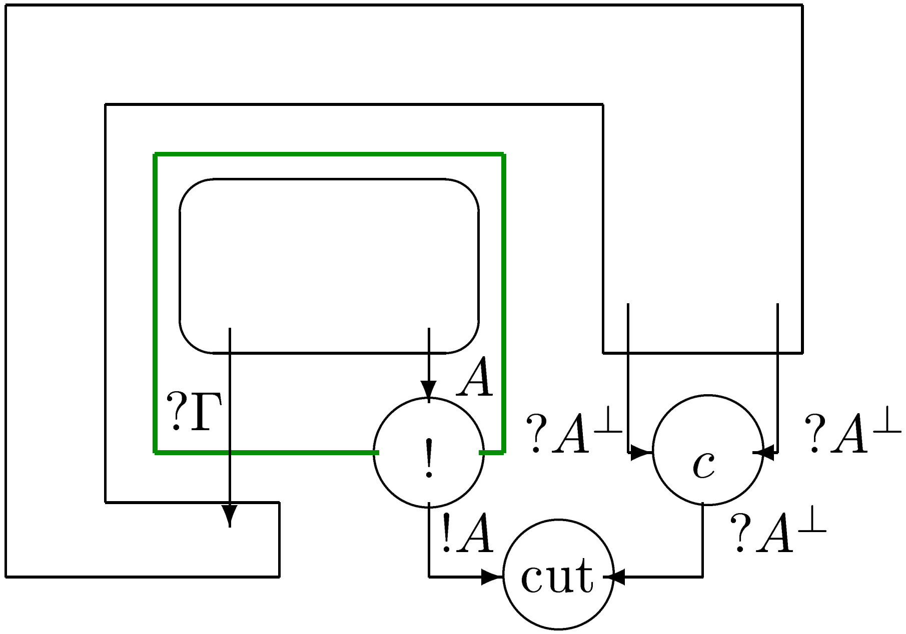


Exponential cut elimination

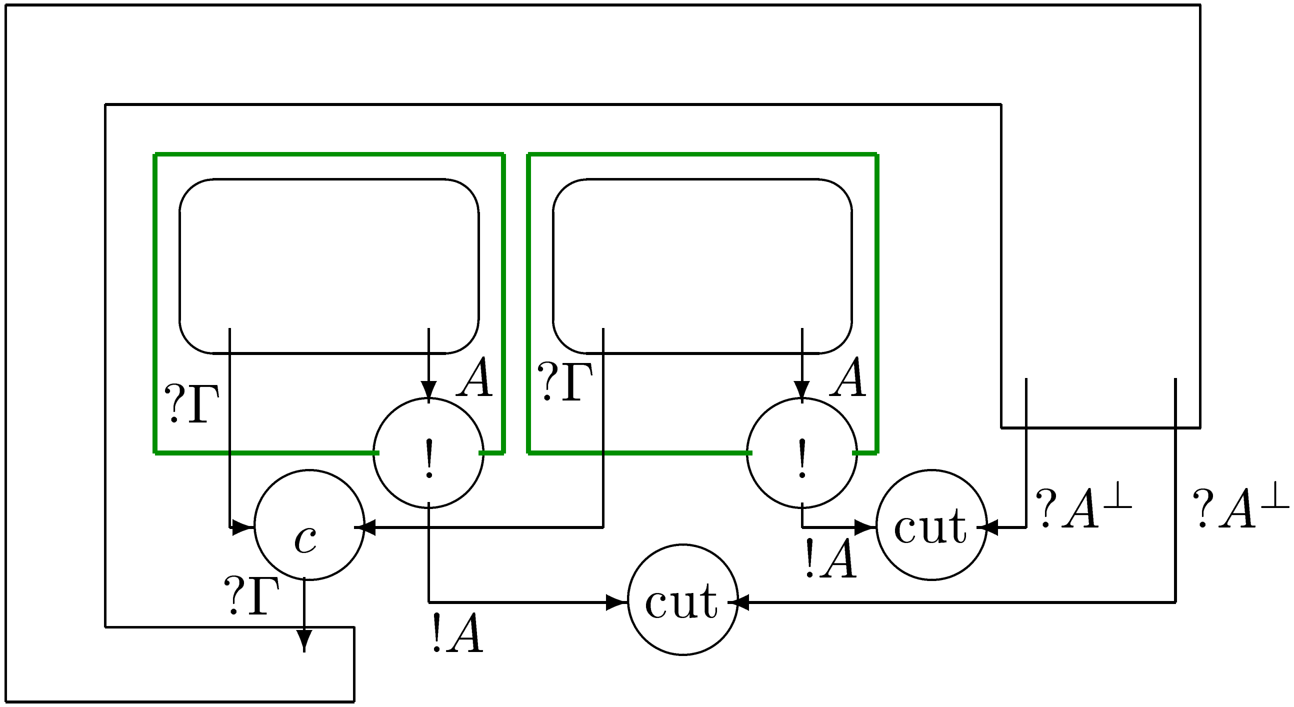


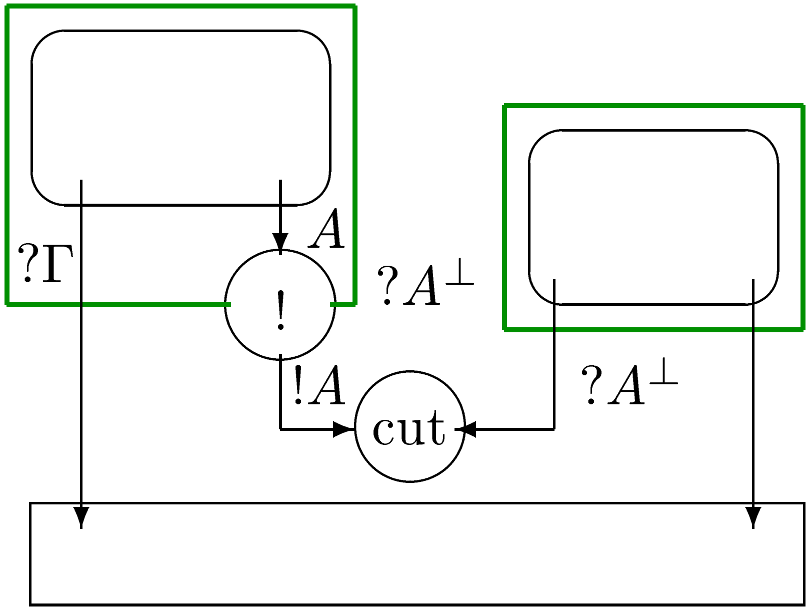


A
$?\Gamma$
!
w
$!A$
$?A^{\perp}$
cut
$\rightsquigarrow$
w
$?\Gamma$

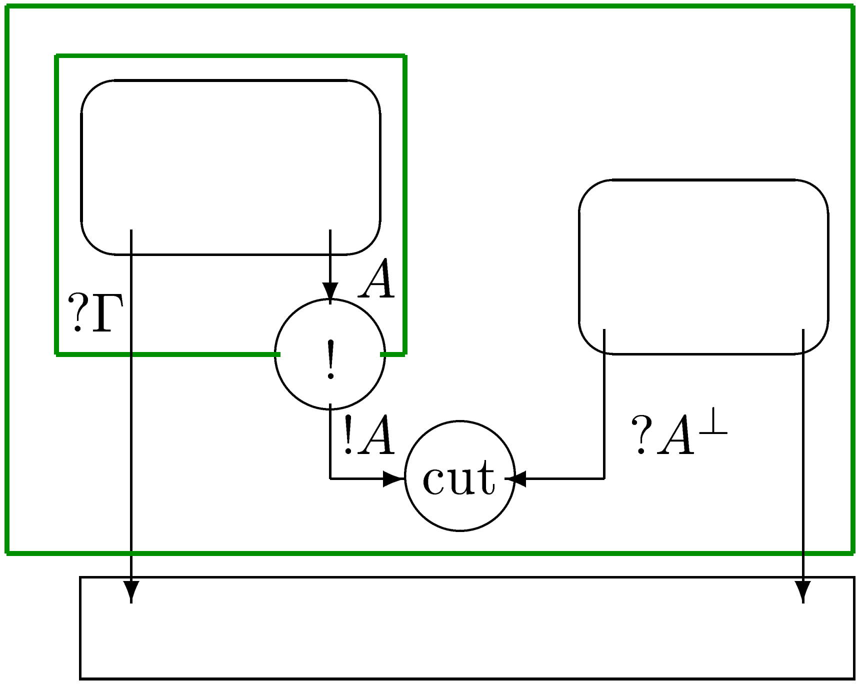


$\rightsquigarrow$





$\rightsquigarrow$



Translation of typed lambda-calculus: Girard's translation (CBN)

- Formula translation: $X^* = X \qquad (A \to B)^* = !A^* \multimap B^*$
- Term translation:

$$x_1 : A_1, \ldots x_n : A_n \vdash t : A \quad \rightsquigarrow \quad \begin{array}{c} \vdots\, t^* \\ \vdash ?(A_1^*)^\perp, \ldots, ?(A_n^*)^\perp, A^* \end{array}$$

$$x : A \vdash x : A \quad \rightsquigarrow \quad \dfrac{\overline{\vdash {A^*}^\perp, A^*}}{\vdash ?{A^*}^\perp, A^*}$$

$$\Gamma \vdash \lambda x^{A} \cdot u^{B} : A \to B \qquad \rightsquigarrow \qquad \dfrac{\begin{array}{c} \vdots\; u^{*} \\ \vdash ?{\Gamma^{*}}^{\perp}, ?{A^{*}}^{\perp}, B^{*} \end{array}}{\vdash ?{\Gamma^{*}}^{\perp}, ?{A^{*}}^{\perp} \parr B^{*}}$$

$$\Gamma \vdash u^{A\to B} v^A : B$$

$$\leadsto$$

$$\dfrac{\dfrac{}{\vdash ?\Gamma^{*\perp}, ?A^{*\perp} \parr B^*}\vdots u^* \qquad \dfrac{\dfrac{\dfrac{\vdots v^*}{\vdash ?\Gamma^{*\perp}, A^*}}{\vdash ?\Gamma^{*\perp}, !A^*} \qquad \dfrac{}{\vdash B^{*\perp}, B^*}}{\vdash ?\Gamma^{*\perp}, !A^* \otimes B^{*\perp}, B^*}}{\dfrac{\vdash ?\Gamma^{*\perp}, ?\Gamma^{*\perp}, B^*}{\vdash ?\Gamma^{*\perp}, B^*}\ \text{contr}}\ \text{cut}$$

The CBV translation

Formula translation:

$$X^* = !X$$

$$(A \to B)^* = !(A^* \multimap B^*)$$

Term translation:

$$\Gamma \vdash t : A \qquad \rightsquigarrow \qquad \begin{array}{c} \vdots \\ \vdash {\Gamma^*}^\perp, A^* \end{array}$$

Pure lambda-calculus

Let D be a recursive type defined by: $D = D \to D$; then any term is typable with type D (all free variables typed with D). Add to MELL a corresponding type O (for Output):

$$O = !O \multimap O = ?I \parr O$$

where $I = O^{\perp}$, thus satisfies:

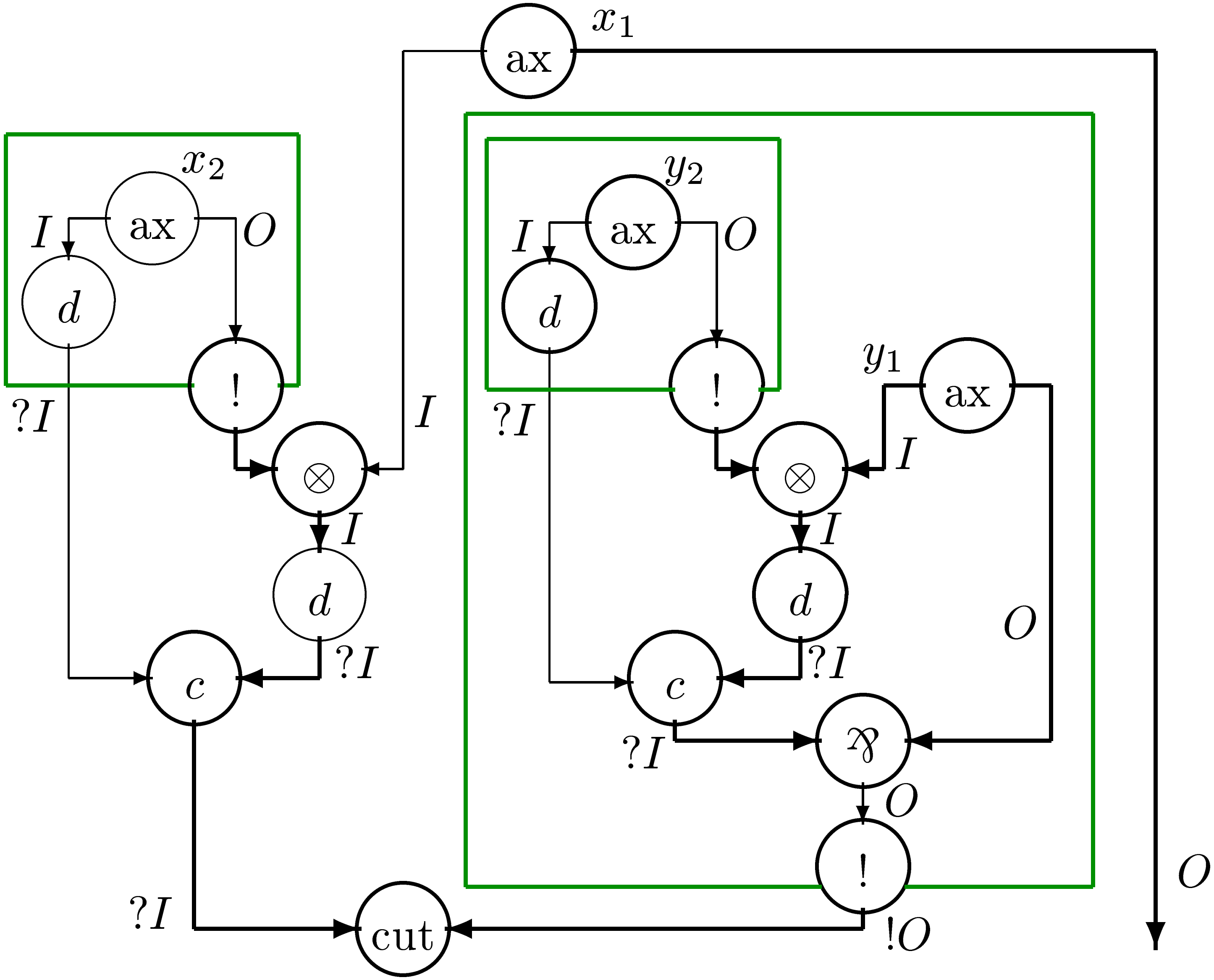
$$I = !O \otimes I$$

Then use the translation of typed lambda-calculus, adding $D^* = 0$.

The same kind of trick may be used to translate CBV lambda-calculus.

An example: delta delta

$(\lambda x.x_1 x_2)\lambda y.y_1 y_2 \rightsquigarrow$



Polarized linear logic

Back to coherent semantics

A *positive correlation space* is a coherent space P together with:

- an anticlique 1_P;
- a linear map $c_P : P \multimap P \otimes P$;

satisfying the following conditions (we denote $a \rightarrow_P b + c$ for $(a, (b, c)) \in c_P$) :

neutral: $a \rightarrow_P a + n$ for any $a \in |P|$ and $n \in 1_P$;

commutativity: $a \rightarrow_P b + c$ iff $a \rightarrow c + b$;

associativity: if $a \rightarrow_P b + c$ and $c \rightarrow_P d + e$ then there is an f such that $a \rightarrow_P f + e$ and $f \rightarrow_P b + d$

Categorically speaking

Positive correlation spaces are *commutative $\otimes$-comonoids* in the category of coherent spaces and linear maps, that is, objects P together with maps $1_P : P \multimap 1$ and $c_P : P \multimap P \otimes P$ satisfying:

$$P \xrightarrow{c_P} P \otimes P \xrightarrow{1_P} P \otimes 1 \xrightarrow{\sim} P = \mathrm{Id}_P$$

$$P \xrightarrow{c_P} P \otimes P \xrightarrow{\text{twist}} P \otimes P = P \xrightarrow{c_P} P \otimes P$$

$$\begin{tikzcd} P \arrow[r, "c_P"] \arrow[d, "c_P"'] & P \otimes P \arrow[d, "c_P \otimes P"] \\ P \otimes P \arrow[r, "P \otimes c_P"] & P \otimes (P \otimes P) \sim (P \otimes P) \otimes P \end{tikzcd}$$

Constructions of PCS

- 0, 1 are PCS;
- if X is any coherent space then $!X$ is a PCS (but not the free one, take the multiset oc for that);
- if P, Q are PCSs then $P \oplus Q$ and $P \otimes Q$ are PCSs;

Any PCS P is a !-coalgebra: there is a map: $\mathrm{coalg}_P : P \multimap !P$ satisfying appropriate conditions w.r.t. the comonad structure of !.

A *negative correlation space* is the dual of a positive correlation space. Categorically speaking, a NCS is a commutative $\parr$-monoid, also a $?$-algebra. Logically speaking, NCSs correspond to *reversible connectives*.

Polarized formulas

$$
\begin{array}{lllllllllllllll}
N & ::= & X & \mid & \bot & \mid & N \parr N & \mid & \top & \mid & N \& N & \mid & ?P & [\quad \mid & ?N \quad] \\
P & ::= & X^\perp & \mid & 1 & \mid & P \otimes P & \mid & 0 & \mid & P \oplus P & \mid & !N & [\quad \mid & !P \quad]
\end{array}
$$

Remark. The image of the CBN translation of intuitionistic logic is polarized. In particular intuitionistic formulas are associated with *negative* formulas.

Polarized systems

$\mathbf{LL}_{\mathrm{pol}}$**:** LL restricted to (strongly) polarized formulas;

LLP: $\mathrm{LL}_{\mathrm{pol}}$ + relaxed structural rules::

$$\frac{\vdash \Gamma}{\vdash N, \Gamma}\,\text{weak} \qquad \frac{\vdash N, N, \Gamma}{\vdash N, \Gamma}\,\text{cont} \qquad \frac{\vdash N, \mathcal{N}}{\vdash\, !N, \mathcal{N}}\,\text{prom}$$

where N is any negative formula (not necessarily a $?$-formula) and $\mathcal{N}$ is a context made only of negative formulas.

Remark: at most one positive formula may appear in a provable polarized sequent (assuming the appropriate slight modification of the $\top$-rule).

Translation of $\lambda\mu$-calculus (CBN)

- Same formula translation as in the intuitionistic case!
- Term translation:

$$x_1 : A_1, \ldots x_n : A_n \vdash t : C, \alpha_1 : B_1, \ldots, \alpha_m : B_m$$

$$\rightsquigarrow$$

$$\begin{array}{c} \vdots \, t^* \\ \vdash ?(A_1^*)^\perp, \ldots, ?(A_n^*)^\perp, C^*, B_1^*, \ldots, B_m^* \end{array}$$

Remark: only negative formulas appear in the conclusion sequent.

Variable, application and abstraction are translated as in the intuitionistic case, μ-abstraction and naming do nothing!

Example: Pierce law (`call/cc`)

$$\lambda f^{(A\to B)\to A}\mu\alpha^{A}[\alpha](f\lambda x^{A}\mu\beta^{B}[\alpha]x) : ((A\to B)\to A)\to A$$

$$\rightsquigarrow$$

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$\vdash A^{*\perp}, A^{*}$}
\UnaryInfC{$\vdash ?A^{*\perp}, A^{*}$}
\RightLabel{weak}
\UnaryInfC{$\vdash ?A^{*\perp}, B^{*}, A^{*}$}
\UnaryInfC{$\vdash ?A^{*\perp} \parr B^{*}, A^{*}$}
\RightLabel{prom}
\UnaryInfC{$\vdash !(?A^{*\perp} \parr B^{*}), A^{*}$}
\AxiomC{}
\UnaryInfC{$\vdash A^{*\perp}, A^{*}$}
\BinaryInfC{$\vdash !(?A^{*\perp} \parr B^{*}) \otimes A^{*\perp}, A^{*}, A^{*}$}
\UnaryInfC{$\vdash ?(!(?A^{*\perp} \parr B^{*}) \otimes A^{*\perp}), A^{*}, A^{*}$}
\RightLabel{contr}
\UnaryInfC{$\vdash !(?A^{*\perp} \parr B^{*}) \otimes A^{*\perp}, A^{*}$}
\UnaryInfC{$\vdash ?(!(?A^{*\perp} \parr B^{*}) \otimes A^{*\perp}) \parr A^{*}$}
\end{prooftree}
$$

That's all folks

But there are lots we haven't seen:

- Linear Logic programming language and automatic proving issues (e.g. provability in MLL is NP-complete);
- Semantics: hypercoherences, finiteness spaces, ...
- Geometry of Interaction, sharing reductions, abstract machines;
- Games semantics, dynamic models;
- Non commutative logic;
- ...

References

- Starting points on the web:
  - search "linear logic" on google;
  - these slides: `http://iml.univ-mrs.fr/~regnier/aussois/`
  - `http://iml.univ-mrs.fr:~lafont/linear/`
- General references about Linear Logic:
  - Girard's seminal paper "Linear Logic", in TCS(50), North Holland, 1987;
  - The proceedings of the 1994 Tokyo meeting: "Advances in Linear Logic", Girard, Lafont, Regnier editors, London Mathematical Society Lecture Notes Series, Cambridge University Press, 1995