DEA–IA

VIII–PARIS–XIII

LOGIQUE LINÉAIRE

Morceaux choisis

Un tutoriel
par
Marcel Masseron

Institut Galilée
Université Paris XIII

2001

Conseils de lectures.

L'ouvrage suivant, issu d'un cours de DEA, constitue une excellente introduction et très abordable, à l'étude du calcul des séquents de Gentzen et de la logique linéaire, entre autres choses.

J.–Y. Girard, Y. Lafont, P. Taylor .— Proofs and types. Volume 7 of Cambridge U. Press, 1990.

Depuis la publication de l'article fondateur

J.–Y. Girard .— Linear logic. *Theoretical Computer Science*, 50 : 1–102, 1987.

la revue *Theoretical Computer Science* publie régulièrement des articles et des numéros spéciaux consacrés à la logique linéaire.

Citons encore

J.–Y. Girard, Y. Lafont, L. Regnier *(ed.)* .— Advances in linear logic. London Math. Soc. Lecture notes series 222, 1995.

L'article suivant

J.–Y. Girard .— Locus Solum. À paraître dans *Mathematical Structures in Computer Science.*

sera fondateur de la "ludique".

Enfin, les deux thèses assez récentes

R. Baudot .— Langages de programmation logique, non–commutativité et polarisation. Thèse de doctorat de l'université Paris 13, 2000.

P. Ruet .— Logique non–commutative et programmation concurrente par contrainte. Thèse de doctorat de l'université Paris 7, 1997.

contiennent une bonne introduction à la programmation en LL commutative et non–commutative.

La consultation des bibliographies des ouvrages cités ici sera indispensable pour poursuivre une étude sérieuse de la LL.

Ce texte a été composé en `Plain TEX` et les figures en `METAPOST`, dans les versions distribuées par l'association `GUTenberg`.
Les lettrines ont été composées en `METAFONT` par Yannis Haralambous.

Systèmes d'inférence.

Un système d'inférence est constitué d'un ensemble de **règles** (d'inférence), qui se présentent sous la forme

$$(\varrho)\,\frac{P_1 \quad \ldots \quad P_n}{C}$$

où

$P_1, \ldots, P_n$ est une suite de formules, appelées **premisses** de (ϱ)
C est une formule appelée **conclusion** of (ϱ).

Lorsque $n = 0$, une règle (ρ) s'appelle **un axiome** et est souvent représentée par la formule C elle–même.

Ces données sont présentées schématiquement, par exemple, $(A \to (B \to A))$ est un "l'axiome" posé pour toute formule A et toute formule B : on devrait les écrire avec des formules atomiques choisies et énoncer un principe de substitution.

Une **inférence**, que l'on appelera plus souvent une **preuve**, de la formule A à partir d'un ensemble Γ de formules (en symboles : "une preuve de $\Gamma \vdash A$") est un arbre dont :

- chaque feuille est, ou bien vide (c'est–à–dire un axiome), ou bien un élément de Γ,
- chaque nœud est une règle,
- la racine est A.

Une preuve de A est une preuve à partir de l'ensemble vide, c'est–à–dire une preuve de $\vdash A$.

Les plus connus et les plus anciens systèmes d'inférence, sur l'un desquels nous allons dire quelques mots, sont dus à Hilbert.

Un système à la Hilbert.

Axiomes :

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

$$((A \to (B \to C)) \to ((A \to B) \to (A \to C)))$$

$$(A \to (A \vee B))$$
$$(B \to (A \vee B))$$

$$((A \to C) \to ((B \to C) \to ((A \vee B) \to C)))$$

$$(A \to (B \to (A \wedge B)))$$

$$((A \wedge B) \to A)$$
$$((A \wedge B) \to B)$$

$$((\neg A \to \neg B) \to ((\neg A \to B) \to A))$$

Règle :

$$\frac{A \quad (A \to B)}{B}$$

Nous nous limitons délibérément au cas du calcul propositionnel. Ceci ne veut évidemment pas dire que le calcul des prédicats ne pose pas des problèmes intéressants, bien au contraire!

Les variables propositionnelles (qui sont les seules formules atomiques du calcul propositionnnel) seront désignées par des minuscules p, q, $\ldots$ et les formules en général, seront désignées par des majuscules A, B, $\ldots$

De Hilbert à Gentzen.

Le système de Hilbert comporte beaucoup d'axiomes et une seule règle.

Il est correct et complet mais souffre d'un grand défaut de modularité : chaque axiome s'exprime au moyen de plusieurs connecteurs, si bien qu'il n'est pas toujours facile de comprendre s'il est dédié plus particulièrement à l'un d'entre eux.
Pour utiliser ce système efficacement, il est nécessaire de commencer par la preuve de lemmes, isolant le rôle que peut jouer chaque connecteur.

Le plus connu est le **Théorème de déduction** :

Si $\{B\} \cup \Gamma \vdash A$ *est prouvable alors* $\Gamma \vdash (B \to A)$ *est prouvable.*

Grosso modo, le système à la Gentzen, que nous allons décrire maintenant, prend les lemmes comme règles : chaque connecteur y admet un "groupe de règles" qui lui est propre, et cette modularité est la propriété essentielle des systèmes à la Gentzen.

Dans un tel système, appelé **calcul des séquents**, les formules ne sont plus utilisées isolément, mais sont immergées dans un contexte, appelé **séquent** : un séquent est une paire (Γ, Δ) d'ensembles finis de formules, que l'on écrit

$$\Gamma \vdash \Delta$$

où les éléments de Γ and Δ sont simplement séparés par des virgules, par exemple, A, Γ est l'union de $\{A\}$ et de l'ensemble Γ.

Pour des raisons pratiques d'implantation, les ensembles qui composent un séquent sont représentés sous la forme de suites et des règles doivent prendre cette représentation en compte. Ces règles sont dites **structurelles** car elles ne concernent aucun connecteur en propre.

L'invention de la **logique linéaire** par Girard, reconsidère en profondeur le rôle joué par les règles structurelles, ce qui conduit en particulier à modifier la structure de Γ et Λ.

SYSTÈME **LK** (Logique classique).

Identité.

$$\frac{}{p \vdash p}$$

Coupure.

$$\frac{\Gamma \vdash \Delta, A \quad A, \Lambda \vdash \Pi}{\Gamma, \Lambda \vdash \Delta, \Pi}$$

Règles structurelles.

(E_d)	$\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \sigma(\Delta)}$ pour toute permutation σ	$\dfrac{\Gamma \vdash \Delta}{\sigma(\Gamma) \vdash \Delta}$ pour toute permutation σ	(E_g)
(C_d)	$\dfrac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A}$	$\dfrac{A, A, \Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}$	(C_g)
(A_d)	$\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A}$	$\dfrac{\Gamma \vdash \Delta}{A, \Gamma \vdash \Delta}$	(A_g)

Règles logiques.

$(\neg\_d)$	$\dfrac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$	$\dfrac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta}$	$(\neg\_g)$
$(\top\_d)$	$\dfrac{}{\vdash \top}$		
$(\wedge\_d)$	$\dfrac{\Gamma \vdash \Delta, A \quad \Lambda \vdash \Pi, B}{\Gamma, \Lambda \vdash \Delta, \Pi, (A \wedge B)}$	$\dfrac{A, B, \Gamma \vdash \Delta}{(A \wedge B), \Gamma \vdash \Delta}$	$(\wedge\_g)$
		$\dfrac{}{\bot \vdash}$	$(\bot\_g)$
$(\vee\_d)$	$\dfrac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, (A \vee B)}$	$\dfrac{A, \Gamma \vdash \Delta \quad B, \Lambda \vdash \Pi}{(A \vee B), \Gamma, \Lambda \vdash \Delta, \Pi}$	$(\vee\_g)$
$(\rightarrow\_d)$	$\dfrac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, (A \rightarrow B)}$	$\dfrac{\Gamma \vdash \Delta, A \quad B, \Lambda \vdash \Pi}{(A \rightarrow B), \Gamma, \Lambda \vdash \Delta, \Pi}$	$(\rightarrow\_g)$

Une preuve du séquent $\Gamma \vdash \Delta$ **est un arbre** dont :

> les feuilles sont vides (un axiome est sans prémisse),
> les nœuds sont des règles,
> la racine est le séquent $\Gamma \vdash \Delta$ lui–même.

Une preuve de la formule A **dans le calcul des séquents** est une preuve de $\vdash A$.

On peut définir **une théorie** en introduisant les séquents (et même les règles) adaptés, comme **axiomes (règles) propres** : sauf mention explicite du contraire, nous ne considérerons que la logique "pure", c'est–à–dire sans aucun axiome ni aucune règle propre à une théorie.

Voici quelques exemples de preuves dans LK, pour se faire la main!

$$\cfrac{\cfrac{\cfrac{\cfrac{}{A \vdash A} \quad \cfrac{}{A \vdash A}}{A, A \vdash (A \wedge A)}\ (\wedge\_d)}{A \vdash (A \wedge A)}\ (C\_g)}{\vdash (A \rightarrow (A \wedge A))}\ (\rightarrow\_d)$$

$$\cfrac{\cfrac{\cfrac{\cfrac{}{B \vdash B}}{A, B \vdash B}\ (A\_g)}{(A \wedge B) \vdash B}\ (\wedge\_g)}{\vdash ((A \wedge B) \rightarrow B)}\ (\rightarrow\_d)$$

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{B \vdash B} \quad \cfrac{}{A \vdash A}}{B, A \vdash (B \wedge A)}\ (\wedge\_d)}{A, B \vdash (B \wedge A)}\ (E\_g)}{(A \wedge B) \vdash (B \wedge A)}\ (\wedge\_g)}{\vdash ((A \wedge B) \rightarrow (B \wedge A))}\ (\rightarrow\_d)$$

Une mention (E) signale qu'un ou plusieurs échanges ont été appliqués à ce niveau.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A} \quad \dfrac{}{B \vdash B}}{A, (A \to B) \vdash B}\;\text{(E) }(\to\_g) \qquad \dfrac{}{C \vdash C}}{A, (A \to B), (B \to C) \vdash C}\;\text{(E) }(\to\_g)}{(A \to B), (B \to C) \vdash (A \to C)}\;(\to\_d)}{((A \to B) \wedge (B \to C)) \vdash (A \to C)}\;(\wedge\_g)}{\vdash (((A \to B) \wedge (B \to C)) \to (A \to C))}\;(\to\_d)
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A} \qquad \dfrac{\dfrac{}{B \vdash B} \quad \dfrac{}{C \vdash C}}{B, C \vdash (B \wedge C)}\;(\wedge\_d)}{A, B, C \vdash (A \wedge (B \wedge C))}\;(\wedge\_g)}{(A \wedge B), C \vdash (A \wedge (B \wedge C))}\;(\wedge\_g)}{((A \wedge B) \wedge C) \vdash (A \wedge (B \wedge C))}\;(\wedge\_g)}{\vdash (((A \wedge B) \wedge C) \to (A \wedge (B \wedge C)))}\;(\to\_d)
$$

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A} \quad \dfrac{}{C \vdash C}}{A, (A \to C) \vdash C}\;(\to\_g) \qquad \dfrac{\dfrac{}{B \vdash B} \quad \dfrac{}{C \vdash C}}{B, (B \to C) \vdash C}\;(\to\_g)}{A, B, ((A \to C) \vee (B \to C)) \vdash C, C}\;\text{(E) }(\vee\_d)}{A, B, ((A \to C) \vee (B \to C)) \vdash C}\;(C\_d)}{(A \wedge B), ((A \to C) \vee (B \to C)) \vdash C}\;(\wedge\_g)}{((A \to C) \vee (B \to C)) \vdash ((A \wedge B) \to C)}\;(\to\_d)}{\vdash (((A \to C) \vee (B \to C)) \to ((A \wedge B) \to C))}\;(\to\_d)
$$

A propos des axiomes d'identité.

Les axiomes $\overline{p \vdash p}$ sont posés pour les formules atomiques p (variables propositionnelles), mais en fait, le séquent $A \vdash A$ est prouvable pour toute formule A et il est courant de considérer $\overline{A \vdash A}$ comme un axiome, quelle que soit la formule A.
La construction des preuves est une simple induction sur les formules :

Le cas atomique est réglé par les axiomes eux–mêmes.

Le cas d'une unité ($\top$ ou $\bot$) s'obtient par l'application d'un affaiblissement convenable à son axiome.

Maintenant, si Ax_A et Ax_B sont des preuves respectivement de $A \vdash A$ et $B \vdash B$ alors

$$\dfrac{\dfrac{\begin{matrix} Ax_A \\ \text{☟} \\ A \vdash A \end{matrix}}{\neg A, A \vdash}\ (\neg_g)}{\neg A \vdash \neg A}\ \text{(E)}\ (\neg_d)
\qquad
\dfrac{\dfrac{\begin{matrix} Ax_A \\ \text{☟} \\ A \vdash A \end{matrix} \quad \begin{matrix} Ax_B \\ \text{☟} \\ B \vdash B \end{matrix}}{A, B \vdash (A \wedge B)}\ (\wedge_d)}{(A \wedge B) \vdash (A \wedge B)}\ (\wedge_g)$$

$$\dfrac{\dfrac{\begin{matrix} Ax_A \\ \text{☟} \\ A \vdash A \end{matrix} \quad \begin{matrix} Ax_B \\ \text{☟} \\ B \vdash B \end{matrix}}{(A \vee B) \vdash A, B}\ (\vee_g)}{(A \vee B) \vdash (A \vee B)}\ (\vee_d)
\qquad
\dfrac{\dfrac{\begin{matrix} Ax_A \\ \text{☟} \\ A \vdash A \end{matrix} \quad \begin{matrix} Ax_B \\ \text{☟} \\ B \vdash B \end{matrix}}{(A \to B), A \vdash B}\ (\to_g)}{(A \to B) \vdash (A \to B)}\ \text{(E)}\ (\to_d)$$

sont aussi des preuves.

Cette remarque est utile car elle entraîne que :
le résultat de la substitution, dans une preuve, de formules aux variables propositionnelles, est encore une preuve, à condition d'accepter les axiomes non–atomiques.

Inversion des règles logiques de LKP.

Le système LKP peut être modifié en un système équivalent dont les règles logiques sont inversibles (ceci ne s'étend pas aux quantificateurs).

Voyons le cas de $\wedge$ (les autres cas sont analogues ou même plus simples).

- $A, B, \Gamma \vdash \Delta$ est prouvable ssi $(A \wedge B), \Gamma \vdash \Delta$ l'est.

Dans un sens, l'implication est simplement la règle $\wedge\_g$, dans l'autre, nous avons :

$$\dfrac{\dfrac{\overline{A \vdash A} \quad \overline{B \vdash B}}{A, B \vdash (A \wedge B)}\,(\wedge\_d) \qquad (A \wedge B), \Gamma \vdash \Delta}{A, B, \Gamma \vdash \Delta}\ \text{(Coupure)}$$

- $\Gamma \vdash \Delta, A$ et $\Gamma \vdash \Delta, B$ sont prouvables ssi $\Gamma \vdash \Delta, (A \wedge B)$ l'est.

Dans un sens, nous avons :

$$\dfrac{\dfrac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma, \Gamma \vdash \Delta, \Delta, (A \wedge B)}\,(\wedge\_d)}{\Gamma \vdash \Delta, (A \wedge B)}\ \text{(E)}\ (C\_g,\, d)$$

et dans l'autre, par exemple :

$$\dfrac{\Gamma \vdash \Delta, (A \wedge B) \qquad \dfrac{\dfrac{\overline{A \vdash A}}{A, B \vdash A}\,\text{(E)}\ (A\_g)}{(A \wedge B) \vdash A}\,(\wedge\_g)}{\Gamma \vdash \Delta, A}\ \text{(Coupure)}$$

Ces observations conduisent à considérer le système suivant LKPI, équivalent à LKP, mais mieux adapté à la recherche de preuve.

Système propositionnel inversible LKPI (Logique classique).

Identité.

$$\overline{A, \Gamma \vdash \Delta, A}$$

Règles d'échange.

$$(E\_d) \quad \frac{\Gamma \vdash \Delta', A, B, \Delta''}{\Gamma \vdash \Delta', B, A, \Delta''} \qquad\qquad \frac{\Gamma', A, B, \Gamma'' \vdash \Delta}{\Gamma', B, A, \Gamma'' \vdash \Delta} \quad (E\_g)$$

Règles logiques.

$$(\neg\_d) \quad \frac{A, \Gamma \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \qquad\qquad \frac{\Gamma \vdash \Delta, A}{\neg A, \Gamma \vdash \Delta} \quad (\neg\_g)$$

$$(\wedge\_dI) \quad \frac{\Gamma \vdash \Delta, A \quad \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, (A \wedge B)} \qquad\qquad \frac{A, B, \Gamma \vdash \Delta}{(A \wedge B), \Gamma \vdash \Delta} \quad (\wedge\_g)$$

$$(\vee\_d) \quad \frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, (A \vee B)} \qquad\qquad \frac{A, \Gamma \vdash \Delta \quad B, \Gamma \vdash \Delta}{(A \vee B), \Gamma \vdash \Delta} \quad (\vee\_gI)$$

$$(\rightarrow\_d) \quad \frac{A, \Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, (A \rightarrow B)} \qquad\qquad \frac{\Gamma \vdash \Delta, A \quad B, \Gamma \vdash \Delta}{(A \rightarrow B), \Gamma \vdash \Delta} \quad (\rightarrow\_gI)$$

Ce système n'a pas vraiment d'existence légale . . . et peut être considéré comme plutôt *ad hoc*, dans le cadre de la logique classique. Il a cependant assez de propriétés pour être présenté ici : outre le fait que toutes ses règles logiques sont inversibles, ces règles sont aussi asynchrones, au sens suivant.

Asynchronisme.

Le choix de la formule d'un séquent à laquelle on rétro–applique une règle logique peut avoir une importance stratégique, mais n'a aucune importance théorique. En effet, les règles logiques de LKPI sont **asynchrones** :

toute preuve d'un séquent qui est la conclusion d'une règle logique ρ de LKPI peut se transformer en une preuve du même séquent, qui se termine par l'application de ρ.

Le moment est venu de répondre à la question :

QU'EST–CE QU'UN SEQUENT?

Clairement, $\vdash A$ n'est autre chose que la formule A elle–même.

Sinon, un séquent $\Gamma \vdash \Delta$ est une façon "négligée" de représenter n'importe quelle formule A telle que $\vdash A$ puisse s'obtenir à partir de $\Gamma \vdash \Delta$ par application de règles unaires de LKP : ce qui précède montre que réciproquement, il est alors possible de démontrer le séquent initial à partir de $\vdash A$.

Par exemple, pour $m \neq 0$ et $n \neq 0$:

$$A_1, \ldots, A_m \vdash B_1, \ldots, B_n$$

représente

$$(A_1 \wedge \cdots \wedge A_m) \rightarrow (B_1 \vee \cdots \vee B_n)$$

mais aussi

$$\neg A_1 \vee \cdots \vee \neg A_m \vee B_1 \vee \cdots \vee B_n$$

(où l'on doit encore ajouter des parenthèses pour obtenir de vraies formules!).

Plus généralement, si l'on tient absolument aux formules, on peut se souvenir de ce que :

"," à gauche, joue le rôle de "$\wedge$"

"," à droite, joue le rôle de "$\vee$"

"$\neg$" signifie que l'on vient de l'autre côté

"$\vdash$" joue le rôle de "$\rightarrow$"

Le séquent très spécial $\vdash$ (correspondant au cas particulier $m = 0$ et $n = 0$) appelé **séquent vide** (à ne pas confondre avec une absence de séquent!) signifie l'*absurde* : en effet, on peut en déduire n'importe quel autre séquent, par affaiblissement . . .

À propos du système LKPI.

Ce système est adapté à *la recherche de preuves* dont le propos est évident : *construire une preuve d'une formule donnée A* ou, plus généralement d'un séquent donné $\Gamma \vdash \Delta$. Une telle recherche se fait naturellement en procédant depuis la racine de l'arbre de l'éventuelle preuve; on est donc conduit à utiliser des règles “à l'envers”, c'est–à–dire du bas vers le haut : la réversibilité des règles logiques de LKPI fait donc merveille!
Voici un algorithme de recherche d'une preuve d'un séquent :
(l'état courant est un ensemble de séquents, initialement réduit au séquent donné)
Tant que c'est possible, appliquer à l'envers une des règles de LKPI à une formule de l'un des séquents de l'état courant.

Si l'on utilise les règles d'échange avec parcimonie (voir ci–dessous), ce processus est fini : en effet, chaque rétro–application d'une règle logique fait disparaître un opérateur! Il se présente alors deux cas :

- si le résultat est l'ensemble vide : on vient effectivement de construire une preuve du séquent initial;
- sinon : le séquent n'est pas prouvable dans LKPI (pas plus que dans LK).

Même dans ce dernier cas, le résultat est tout de même intéressant car, appliqué au séquent $\vdash A$, il fournit une forme normale conjonctive de A : la conjonction des formules représentées par les séquents de l'état final (qui sont constitués de formules atomiques).

La page suivante présente des schémas graphiques de la rétro–application des règles de LKPI, où les formules sont, dans la mesure du possible, traitées sur place, évitant ainsi tout échange. L'asynchronisme permet de choisir de façon arbitraire la formule que l'on désire décomposer. On pourra y ajouter les règles suivantes

$$\Gamma \vdash \Delta', A, \Delta'', A, \Delta''' \qquad\qquad \Gamma', A, \Gamma'', A, \Gamma''' \vdash \Delta$$

$$\Gamma \vdash \Delta', \Delta'', \Delta''', A \qquad\qquad A, \Gamma', \Gamma'', \Gamma''' \vdash \Delta$$

figurant la rétro–aplication d'une forme restreinte d'affaiblissement (l'affaiblissement en général est pris en charge par les axiomes).

Un système de décomposition pour **LKP** (logique classique).

$$\Gamma', A, \Gamma'' \vdash \Delta', A, \Delta''$$

$$\begin{tikzcd}[column sep=tiny]
\Gamma \vdash \Delta', \neg A, \Delta'' \arrow[d] \\
A, \Gamma \vdash \Delta', \Delta''
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
& \Gamma \vdash \Delta', (A \wedge B), \Delta'' \arrow[dl] \arrow[dr] & \\
\Gamma \vdash \Delta', A, \Delta'' & & \Gamma \vdash \Delta', B, \Delta''
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
\Gamma \vdash \Delta', (A \vee B), \Delta'' \arrow[d] \\
\Gamma \vdash \Delta', A, B, \Delta''
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
\Gamma \vdash \Delta', (A \rightarrow B), \Delta'' \arrow[d] \\
A, \Gamma \vdash \Delta', B, \Delta''
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
\Gamma', \neg A, \Gamma'' \vdash \Delta \arrow[d] \\
\Gamma', \Gamma'' \vdash \Delta, A
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
\Gamma', (A \wedge B), \Gamma'' \vdash \Delta \arrow[d] \\
\Gamma', A, B, \Gamma'' \vdash \Delta
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
& \Gamma', (A \vee B), \Gamma'' \vdash \Delta \arrow[dl] \arrow[dr] & \\
\Gamma', A, \Gamma'' \vdash \Delta & & \Gamma', B, \Gamma'' \vdash \Delta
\end{tikzcd}$$

$$\begin{tikzcd}[column sep=tiny]
& \Gamma', (A \rightarrow B), \Gamma'' \vdash \Delta \arrow[dl] \arrow[dr] & \\
\Gamma', \Gamma'' \vdash \Delta, A & & \Gamma', B, \Gamma'' \vdash \Delta
\end{tikzcd}$$

ÉLIMINATION DES COUPURES.

Voici l'énoncé du **Hauptsatz** de Gentzen :

Tout séquent prouvable dans LK
est aussi prouvable sans utilisation de la règle de coupure.

Les coupures dont il est question sont les coupures logiques : il n'est évidemment pas question de dire quoi que ce soit de général au sujet des coupures mettant en cause d'éventuels axiomes ou règles propres à une théorie particulière!

L'élimination des coupures rend une preuve complètement explicite, mais "très exponentiellement" plus haute que la preuve initiale.

Voyons ces deux points plus en détail.

- **Propriété de la sous formule.**

Les formules qui interviennent dans une preuve sans coupure d'un séquent Σ sont des sous formules des éléments de Σ.

La vérification se fait en observant, à l'œil nu, les règles de LKP distinctes de la règle de coupure. L'importance de cette propriété est évidente pour la recherche d'une preuve. En particulier, on en déduit que :

LK est cohérent

car il est clair que l'on ne dispose d'aucune sous–formule pour construire une preuve sans coupure du séquent vide $\vdash$.

- Désignons par h la hauteur d'une preuve P et par d son degré, c'est–à–dire la plus grande hauteur des formules de coupure de P, alors, la hauteur de la preuve sans coupure obtenue à partir de P vaut, au pire, $H(d,h)$, où

$$H(0,h) = h \qquad H(d+1,h) = 4^{H(d,h)}$$

ce qui veut dire que le gain d'un degré coûte une exponentielle ...

Deux cas d'élimination sont très immédiats.

Les axiomes s'éliminent d'eux mêmes, au contact d'une coupure, par exemple :

$$\dfrac{\dfrac{}{A \vdash A} \qquad \begin{array}{c} G \\ \vdots \\ A, \Lambda \vdash \Pi \end{array}}{A, \Lambda \vdash \Pi}$$

se réduit simplement en G elle–même.

La règle d'affaiblissement donne lieu à une élimination encore plus sévère, par exemple :

$$\dfrac{\begin{array}{c} D_A \\ \vdots \\ \Gamma \vdash \Delta, A \end{array} \qquad \begin{array}{c} G \\ \vdots \\ \dfrac{\Lambda \vdash \Pi}{A, \Lambda \vdash \Pi}\ (A\_g) \end{array}}{\Gamma, \Lambda \vdash \Delta, \Pi}\ \text{(Coupure)}$$

se réduit en

$$\begin{array}{c} G \\ \vdots \\ \dfrac{\Lambda \vdash \Pi}{\Gamma, \Lambda \vdash \Delta, \Pi}\ (A\_g, d) \end{array}$$

L'idée générale de l'élimination est assez simple. On fait "remonter" les coupures en procédant à des opérations élémentaires :

- **Commutations** : la coupure se rapproche strictement de la région des feuilles, c'est–à–dire des axiomes,
- **Cas clefs** : la coupure se trouve décomposée en des coupures portant sur des formules strictement plus simples.

Malheureusement, cette idée n'est pas suffisante pour conclure et, la démonstration complète comporte des détails délicats.

Lorsque la formule de coupure est restée passive dans les dernières règles des deux preuves auxquelles on applique la coupure, la transformation est une **commutation**, par exemple :

$$\dfrac{\text{(E) }(\varrho\_?)\ \dfrac{\begin{array}{c} D_A \\ \text{☟} \\ \Gamma \vdash \Delta, A \end{array}}{\Gamma' \vdash \Delta', A} \qquad \dfrac{\begin{array}{c} G_A \\ \text{☟} \\ A, \Lambda \vdash \Pi \end{array}}{A, \Lambda' \vdash \Pi'}\ \text{(E) }(\varsigma\_?)}{\Gamma', \Lambda' \vdash \Delta', \Pi'}\ \text{(Coupure)}$$

se réduit en

$$\dfrac{\dfrac{\dfrac{\begin{array}{c} D_A \\ \text{☟} \\ \Gamma \vdash \Delta, A \end{array} \qquad \begin{array}{c} G_A \\ \text{☟} \\ A, \Lambda \vdash \Pi \end{array}}{\Gamma, \Lambda \vdash \Delta, \Pi}\ \text{(Coupure)}}{\Gamma', \Lambda \vdash \Delta', \Pi}\ \text{(E) }(\varsigma\_?)}{\Gamma', \Lambda' \vdash \Delta', \Pi'}\ \text{(E) }(\varrho\_?)$$

ou en la preuve obtenue en transposant $\varrho\_?$ et $\varsigma\_?$.

Un cas clef se présente lorsque les règles précédant la coupure sont une règle à droite et une règle à gauche relatives au même opérateur, par exemple :

$$\dfrac{(\wedge\_d)\ \dfrac{\begin{array}{c} D_A \\ \text{☟} \\ \Gamma \vdash \Delta, A \end{array} \quad \begin{array}{c} D_B \\ \text{☟} \\ \Lambda \vdash \Pi, B \end{array}}{\Gamma, \Lambda \vdash \Delta, \Pi, (A \wedge B)} \qquad \dfrac{\begin{array}{c} G_{A,B} \\ \text{☟} \\ A, B, \Theta \vdash \Psi \end{array}}{(A \wedge B), \Theta \vdash \Psi}\ (\wedge\_g)}{\Gamma, \Lambda, \Theta \vdash \Delta, \Pi, \Psi}\ \text{(Coupure)}$$

admet deux réductions possibles :

$$\dfrac{\begin{array}{c} D_B \\ \text{☟} \\ \Lambda \vdash \Pi, B \end{array} \qquad \dfrac{\begin{array}{c} D_A \\ \text{☟} \\ \Gamma \vdash \Delta, A \end{array} \quad \begin{array}{c} G_{A,B} \\ \text{☟} \\ A, B, \Theta \vdash \Psi \end{array}}{B, \Gamma, \Theta \vdash \Delta, \Psi}\ \text{(Coupure) (E)}}{\Gamma, \Lambda, \Theta \vdash \Delta, \Pi, \Psi}\ \text{(Coupure) (E)}$$

et celle que l'on obtient en transposant les rôles de D_A et D_B.

Le cas d'un échange est facile à comprendre : il faut remonter dans la sous preuve qui se termine par l'échange en question, jusqu'au niveau où la formule de coupure est apparue et y appliquer la coupure ...

Le cas d'une contraction est vraiment très problématique car la méthode que nous avons appliquée jusqu'à présent ne produit pas l'effet désiré, par exemple

$$\dfrac{\begin{array}{c} D_A \\ \vdots \\ \Gamma \vdash \Delta, A \end{array} \qquad \dfrac{\begin{array}{c} G_A \\ \vdots \\ A, A, \Lambda \vdash \Pi \end{array}}{A, \Lambda \vdash \Pi}\ (C\_g)}{\Gamma, \Lambda \vdash \Delta, \Pi}\ \text{(Coupure)}$$

deviendrait

$$\dfrac{\dfrac{\begin{array}{c} D_A \\ \vdots \\ \Gamma \vdash \Delta, A \end{array} \qquad \dfrac{\begin{array}{c} D_A \\ \vdots \\ \Gamma \vdash \Delta, A \end{array} \quad \begin{array}{c} G_A \\ \vdots \\ A, A, \Lambda \vdash \Pi \end{array}}{A, \Gamma, \Lambda \vdash \Delta, \Pi}\ \text{(Coupure)}}{\Gamma, \Gamma, \Lambda \vdash \Delta, \Delta, \Pi}\ \text{(Coupure)}}{\Gamma, \Lambda \vdash \Delta, \Pi}\ \text{(E)}\ (C\_g,\, d)$$

ce qui est très loin d'être satisfaisant!

Modification des règles logiques.

Nous avons vu qu'il était possible de rendre inversibles des règles de LK qui ne l'étaient pas, grâce aux règles structurelles; il est aussi possible, avec les mêmes moyens, de faire le contraire!

Voici un cas typique :

$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, (A \vee B)}\ (\vee\_d)$$

est équivalent au couple de règles

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, (A \vee B)}\ (\vee B\_d) \qquad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, (A \vee B)}\ (A\vee\_d)$$

L'équivalence est facile à vérifier :

$(\vee B\_d)$ and $(A\vee\_d)$ sont prouvables à partir de $(\vee\_d)$, par exemple :

$$\cfrac{\cfrac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A, B}\ (A\_d)}{\Gamma \vdash \Delta, (A \vee B)}\ (\vee\_d)$$

réciproquement :

$$\cfrac{\cfrac{\cfrac{\cfrac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A, (A \vee B)}\ (A\vee\_d)}{\Gamma \vdash \Delta, (A \vee B), A}\ (E\_d)}{\Gamma \vdash \Delta, (A \vee B), (A \vee B)}\ (\vee B\_d)}{\Gamma \vdash \Delta, (A \vee B)}\ (C\_d)$$

SYSTÈME **LJ** (Logique intuitionniste).

Les séquents ont la forme $\Gamma \vdash A$ *où* A *est* **une seule** *formule.*

Identité.

$$\frac{}{p \vdash p}$$

Coupure.

$$\frac{\Gamma \vdash A \quad A, \Lambda \vdash B}{\Gamma, \Lambda \vdash B}$$

Règles structurelles.

$$\frac{\Gamma \vdash B}{A, \Gamma \vdash B} \quad (A\_g)$$

$$\frac{A, A, \Gamma \vdash B}{A, \Gamma \vdash B} \quad (C\_g)$$

$$\frac{\Gamma \vdash A}{\sigma(\Gamma) \vdash A} \quad (E\_g)$$

pour toute permutation σ

Règles logiques.

$$(\mathbf{1}\_d) \quad \frac{}{\vdash \mathbf{1}}$$

$$(\wedge\_d) \quad \frac{\Gamma \vdash A \quad \Lambda \vdash B}{\Gamma, \Lambda \vdash (A \wedge B)}$$

$$\frac{A, B, \Gamma \vdash C}{(A \wedge B), \Gamma \vdash C} \quad (\wedge\_g)$$

$$\frac{}{\mathbf{0} \vdash A} \quad (\mathbf{0}\_g)$$

$(\vee B\_d)$ $$\frac{\Gamma \vdash A}{\Gamma \vdash (A \vee B)}$$

$(A\vee\_d)$ $$\frac{\Gamma \vdash B}{\Gamma \vdash (A \vee B)}$$

$$\frac{A, \Gamma \vdash C \quad B, \Gamma \vdash C}{(A \vee B), \Gamma \vdash C}$$ $(\vee\_g)$

$(\rightarrow\_d)$ $$\frac{A, \Gamma \vdash B}{\Gamma \vdash (A \rightarrow B)}$$

$$\frac{\Gamma \vdash A \quad B, \Lambda \vdash C}{(A \rightarrow B), \Gamma, \Lambda \vdash C}$$ $(\rightarrow\_g)$

0 (faux) est une constante : **on définit** la négation par $\neg A = (A \rightarrow \mathbf{0})$.

À propos de LJ.

LJ satisfait la propriété d'élimination des coupures et tous les axiomes d'identité sont prouvables à partir des axiomes d'identité atomiques.

La condition imposée aux membres droits des séquents intuitionnistes est une limitation de l'usage des règles structurelles : les séquents prouvables dans LJ sont donc prouvables dans LK (ce qui implique que LJ est cohérent), mais pas nécessairement le contraire.

Par exemple, on peut prouver :

$$\vdash (A \to \neg\neg A)$$
$$\vdash ((\neg A \vee \neg B) \to \neg(A \wedge B))$$

dans LJ, mais en général pas leurs réciproques.

LJ est constructive, plus précisément :

$\vdash (A \vee B)$ *est prouvable ssi* $\vdash A$ *ou* $\vdash B$ *est prouvable.*

La logique intuitionniste, et les nombreux développements auxquels elle a donné lieu (isomorphisme de Curry–Howard, système NLJ de la déduction naturelle, sémantique dénotationnelle, ...) est le prototype, et certainement la source d'inspiration principale de la logique linéaire.

La page suivante montre les règles de NLJ : ce système possède des propriétés remarquables de normalisation (analogue de l'élimination des coupures) et c'est aussi le prototype de la notion de réseaux de preuves, qui trouve son accomplissement en logique linéaire.

SYSTÈME **NLJ** (Déduction naturelle).

Identité.

$$A$$

Constructions logiques.

$(\mathbf{1}\_I)$ $$\frac{}{\mathbf{1}}$$

$$\frac{(A \wedge B)}{A} \quad (\wedge B\_E)$$

$(\wedge\_I)$ $$\frac{A \quad B}{(A \wedge B)}$$

$$\frac{(A \wedge B)}{B} \quad (A\wedge\_E)$$

$$\frac{\mathbf{0}}{A} \quad (\mathbf{0}\_E)$$

$(A\vee\_I)$ $$\frac{B}{(A \vee B)}$$

$(\vee B\_I)$ $$\frac{A}{(A \vee B)}$$

$$\frac{(A \vee B) \quad \begin{matrix}[\![A]\!]\\ \vdots \\ C\end{matrix} \quad \begin{matrix}[\![B]\!]\\ \vdots \\ C\end{matrix}}{C} \quad (\vee\_E)$$

$(\rightarrow\_I)$ $$\frac{\begin{matrix}[\![A]\!]\\ \vdots \\ B\end{matrix}}{(A \rightarrow B)}$$

$$\frac{A \quad (A \rightarrow B)}{B} \quad (\rightarrow\_E)$$

0 (faux) est une constante et **on définit** $\neg A = (A \rightarrow \mathbf{0})$.

Le système de déduction naturelle NLJ.

Le système **NLJ** est une représentation du système **NJ**, qui, au moins dans son principe, porte bien son nom car il simule très précisément le raisonnement naturel, lorsque le tiers–exclu n'est pas en cause.

Une preuve est un arbre, dont les feuilles sont les "hypothèses" et la racine la formule à démontrer, et dont les embranchements sont des règles.

Pour des raisons techniques, en relation avec la contraction et l'affaiblissement, une hypothèse A se présente sous la forme d'un ensemble fini d'occurrences de A que l'on appelle *un paquet de type* A : un tel paquet, quoique typé, peut fort bien être vide (un artifice quelconque, par exemple une indexation, permet de simuler facilement la structure en paquets).

Les hypothèses évoluent de deux façons différentes au cours de la construction d'un preuve :

- par juxtaposition, avec regroupement éventuel de paquets : ceci se produit dans l'application d'une règle binaire, $(\wedge\_I)$ et $(\rightarrow\_E)$;
- par "décharge", c'est–à–dire effacement, d'un paquet d'hypothèses lors de l'application de la règle $(\rightarrow\_I)$: le paquet déchargé disparaît de l'inventaire des hypothèses, mais il est essentiel de signaler à quelle application de la règle $(\rightarrow\_I)$ la disparition a eu lieu.

Ces opérations interviennent toutes les deux lors de l'application de la règle $(\vee\_E)$.

Remarques sur les règles.

• Dans la règle d'**Identité**, A est à la fois un paquet d'hypothèse à un seul élément, et la conclusion.

• Les règles d'**Introduction** simulent les règles à droite de **LJ**.

• Les règles d'**Elimination** en traduisent les règles à gauche, par le truchement de l'application d'une coupure (la seule qui soit capable d'éliminer ...).

LOGIQUE LINÉAIRE DE GIRARD (1987).

Dans ce système, la règle de contraction disparaît : les séquents ne sont plus composés d'ensembles mais de multi–ensembles. De même, la règle d'affaiblissement ...

Les règles de LKP et de LKPI relatives au même opérareur binaire ne sont plus équivalentes, puisque ces équivalences dépendent fortement de l'affaiblissement et de la contraction! Ceci entraine la duplication de la conjonction en $\otimes$ et $\&$, et de la disjonction en $\parr$ et $\oplus$:

> $\otimes$ (*fois*) et $\parr$ (*par*) sont dites *multiplicatives*
> $\&$ (*avec*) et $\oplus$ (*plus*) sont dites *additives*.

Une unité est introduite pour chacun de ces connecteurs, respectivement : $\mathbf{1}$ et $\perp$, $\top$ et $\mathbf{0}$.
Les *exponentielles* $!$ (*bien sûr!*) et $?$ (*pourquoi pas?*) établissent une relation entre les deux mondes : bien entendu, ceci tient au fait que des règles pour ces opérateurs sont une contraction et un affaiblissement!

Antagonisme linéaire.

Toute formule atomique p advient avec son antagoniste $p^\perp$, et cet antagonisme s'étend à chaque formule linéaire grâce aux égalités suivantes (issues des équivalences de de Morgan) :

$$p^{\perp\perp} = p$$

$$\begin{array}{rclcrcl}
\perp^\perp &=& \mathbf{1} & \qquad & \mathbf{1}^\perp &=& \perp \\
(A\parr B)^\perp &=& (A^\perp \otimes B^\perp) & & (A\otimes B)^\perp &=& (A^\perp \parr B^\perp) \\
\top^\perp &=& \mathbf{0} & & \mathbf{0}^\perp &=& \top \\
(A\& B)^\perp &=& (A^\perp \oplus B^\perp) & & (A\oplus B)^\perp &=& (A^\perp \& B^\perp) \\
(?A)^\perp &=& !A^\perp & & (!A)^\perp &=& ?A^\perp
\end{array}$$

On vérifie que $A^{\perp\perp} = A$ pour toute formule linéaire A.

L'implication linéaire multiplicative $\multimap$ (*devient*) est définie par

$$(A \multimap B) = (A^\perp \parr B)$$

joue un rôle important (contrairement à son analogue additif).

La dualité gauche/droite de la logique classique est remplacée ici par l'antagonisme précédent : il devient naturel de considérer des séquents unilatéraux $\vdash \Gamma$ où Γ est un multi–ensemble.

SYSTÈME **LL** (Logique linéaire).

Les séquents sont de la forme $\vdash \Gamma$ *où* Γ *est une suite finie de formules.*

Antagonisme.

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

Coupure.

$$\frac{\vdash \Gamma, A \quad \vdash \Delta, A^{\perp}}{\vdash \Gamma, \Delta}$$

Règles structurelles.

$$\text{(C)} \qquad \frac{\vdash \Gamma, ?\,A, ?\,A}{\vdash \Gamma, ?\,A} \qquad \Bigg| \qquad \frac{\vdash \Gamma}{\vdash \Gamma, ?\,A} \qquad \text{(A)}$$

Echange : pour toute permutation σ

$$\frac{\vdash \Gamma}{\vdash \sigma(\Gamma)}$$

L'axiome d'antagonisme est plus couramment appelé *axiome d'identité*.

La règle d'échange traduit le fait que l'ordre dans lequel on écrit les formules d'un séquent $\vdash \Gamma$ n'est pas significatif, mais il faut insister sur le fait que la règle de contraction ne s'applique qu'à des formules particulières de la forme $?\,A$.

L'absence d'une règle de contraction applicable librement signifie que Γ représente **un multi–ensemble de formules**, c'est–à–dire un ensemble d'occurrences de formules!

De même, on ne peut procéder à un affaiblissement que par une formule de la forme $?\,A$.

En quelques mots, la logique linéaire gère les ressources avec parcimonie :

– une ressource du type $?\,A$ est réutilisable indéfiniment, comme une donnée inaltérable et toujours disponible (par exemple, un théorème ou une valeur enregistrée en mémoire);
– une ressource A qui n'est pas de cette forme, représente une donnée consommable, c'est–à–dire révisable : par exemple, le contenu d'une variable est perdu après une affectation.

Règles logiques.

Opérateurs asynchrones	**Opérateurs synchrones**

Fragment multiplicatif.

	Opérateurs asynchrones	Opérateurs synchrones	
$(\bot)$	$\dfrac{\vdash \Gamma}{\vdash \Gamma, \bot}$	$\dfrac{}{\vdash \mathbf{1}}$	$(\mathbf{1})$
$(\parr)$	$\dfrac{\vdash \Gamma, A, B}{\vdash \Gamma, (A \parr B)}$	$\dfrac{\vdash \Gamma, A \quad \vdash \Delta, B}{\vdash \Gamma, \Delta, (A \otimes B)}$	$(\otimes)$

Fragment additif.

	Opérateurs asynchrones	Opérateurs synchrones	
$(\top)$	$\dfrac{}{\vdash \Gamma, \top}$		(rien pour **0**)
$(\&)$	$\dfrac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, (A \& B)}$	$\dfrac{\vdash \Gamma, A}{\vdash \Gamma, (A \oplus B)}$	$(\oplus B)$
		$\dfrac{\vdash \Gamma, B}{\vdash \Gamma, (A \oplus B)}$	$(A \oplus)$

Exponentielles.

	Opérateurs asynchrones	Opérateurs synchrones	
$(?)$	$\dfrac{\vdash \Gamma, A}{\vdash \Gamma, ?A}$	$\dfrac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A}$	$(!)$

Un opérateur ω est dit *asynchrone* ssi il vérifie la propriété de commutation suivante :

Toute preuve D d'un séquent $\vdash \Gamma$ qui est conlusion de la règle pour ω peut se transformer en une preuve D' du même séquent, qui se termine par cette règle.

Il est dit *synchrone* dans le cas contraire.

Cette distinction joue un rôle essentiel dans la recherche de preuve dont il sera question par la suite.

Quelques équivalences prouvables dans LL.

L'équivalence linéaire (multiplicative) est définie par

$$A \multimapboth B = (A \multimap B) \otimes (B \multimap A)$$

Chaque unité est élément neutre pour son opérateur.

$$(A \parr \bot) \multimapboth A \qquad\qquad (A \otimes \mathbf{1}) \multimapboth A$$
$$(A \,\&\, \top) \multimapboth A \qquad\qquad (A \oplus \mathbf{0}) \multimapboth A$$

Chaque unité additive est absorbante pour un opérateur multiplicatif.

$$(A \parr \top) \multimapboth \top \qquad\qquad (A \otimes \mathbf{0}) \multimapboth \mathbf{0}$$

Les additifs sont idempotents.

$$(A \,\&\, A) \multimapboth A \qquad\qquad (A \oplus A) \multimapboth A$$

Les opérateurs sont commutatifs et associatifs, par exemple :

$$(A \otimes B) \multimapboth (B \otimes A) \qquad\qquad ((A \otimes B) \otimes C) \multimapboth (A \otimes (B \otimes C))$$

Distributivité.

$$(A \parr (B \,\&\, C)) \multimapboth ((A \parr B) \,\&\, (A \parr C)) \qquad\qquad (A \otimes (B \oplus C)) \multimapboth ((A \oplus B) \otimes (A \oplus C))$$

Semi–distributivité.

$$(A \otimes (B \,\&\, C)) \multimap ((A \otimes B) \,\&\, (A \otimes C)) \qquad\qquad ((A \parr B) \oplus (A \parr C)) \multimap (A \parr (B \oplus C))$$

Les exponentielles font passer d'un additif à un multiplicatif.

$$!\top \multimapboth \mathbf{1} \qquad\qquad ?\mathbf{0} \multimapboth \bot$$
$$!(A \,\&\, B) \multimapboth (!A \otimes !B) \qquad\qquad ?(A \oplus B) \multimapboth (?A \parr ?B)$$

Enfin, voici quelques propriétés qui se déduisent de propriétés précédentes par de simples traductions.

$$(\mathbf{1} \multimap A) \multimapboth A \qquad (A \multimap \bot) \multimapboth A^\bot \qquad (\mathbf{0} \multimap A) \multimapboth \top \qquad (A \multimap \top) \multimapboth \top$$

$$(A \multimap B) \multimapboth (B^\bot \multimap A^\bot)$$

$$(A \multimap (B \multimap C)) \multimapboth ((A \otimes B) \multimap C) \qquad\qquad (A \multimap (B \,\&\, C)) \multimapboth ((A \multimap B) \,\&\, (A \multimap C))$$

$$(A \multimap (B \parr C)) \multimapboth ((A \multimap B) \parr C) \qquad\qquad ((A \oplus B) \multimap C) \multimapboth ((A \multimap C) \,\&\, (B \multimap C))$$

Exemples de preuves dans LL.

Dans les deux preuves qui suivent, on a négligé de mentionner les applications de la règle d'échange.

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$\vdash A^{\perp}, A$}
\LeftLabel{$(\oplus B)$}
\UnaryInfC{$\vdash A^{\perp}, (A \oplus B)$}
\LeftLabel{(?)}
\UnaryInfC{$\vdash A^{\perp}, ?(A \oplus B)$}
\LeftLabel{(!)}
\UnaryInfC{$\vdash !(A^{\perp}), ?(A \oplus B)$}
\AxiomC{}
\UnaryInfC{$\vdash B^{\perp}, B$}
\RightLabel{$(A \oplus)$}
\UnaryInfC{$\vdash B^{\perp}, (A \oplus B)$}
\RightLabel{(?)}
\UnaryInfC{$\vdash B^{\perp}, ?(A \oplus B)$}
\RightLabel{(!)}
\UnaryInfC{$\vdash !(B^{\perp}), ?(A \oplus B)$}
\RightLabel{$(\otimes)$}
\BinaryInfC{$\vdash (!(A^{\perp}) \otimes !(B^{\perp})), ?(A \oplus B), ?(A \oplus B)$}
\RightLabel{(C)}
\UnaryInfC{$\vdash (!(A^{\perp}) \otimes !(B^{\perp})), ?(A \oplus B)$}
\RightLabel{$(\parr)$}
\UnaryInfC{$\vdash ((?\,A \parr ?\,B) \multimap ?(A \oplus B))$}
\end{prooftree}
$$

$$
\begin{prooftree}
\AxiomC{}
\UnaryInfC{$\vdash A^{\perp}, A$}
\LeftLabel{(?)}
\UnaryInfC{$\vdash A^{\perp}, ?\,A$}
\LeftLabel{(A)}
\UnaryInfC{$\vdash A^{\perp}, ?\,A, ?\,B$}
\AxiomC{}
\UnaryInfC{$\vdash B^{\perp}, B$}
\RightLabel{(?)}
\UnaryInfC{$\vdash B^{\perp}, ?\,B$}
\RightLabel{(A)}
\UnaryInfC{$\vdash B^{\perp}, ?\,A, ?\,B)$}
\RightLabel{(\&)}
\BinaryInfC{$\vdash (A^{\perp} \& B^{\perp}), ?\,A, ?\,B$}
\RightLabel{(!)}
\UnaryInfC{$\vdash !(A^{\perp} \& B^{\perp}), ?\,A, ?\,B$}
\RightLabel{$(\parr)$}
\UnaryInfC{$\vdash !(A^{\perp} \& B^{\perp}), (?\,A \parr ?\,B)$}
\RightLabel{$(\parr)$}
\UnaryInfC{$\vdash (?(A \oplus B) \multimap (?\,A \parr ?\,B))$}
\end{prooftree}
$$

Rechant :

> LL *satisfait la propriété d'élimination des coupures*,
> Tout axiome est prouvable à partir des axiomes atomiques.

Interprétation de la logique intuitionniste dans la logique linéaire.

Avant d'évoquer quelques aspects de la logique linéaire, il est intéressant de voir succintement en quel sens celle–ci englobe la logique intuitionniste.

Pour cela, on associe une formule A^* de LL à chaque formule A de LJ, par l'induction suivante :

$A^* = A$ pour toute formule atomique A

$\mathbf{0}^* = \mathbf{0}$

$(A \wedge B)^* = A^* \& B^*$

$(A \vee B)^* = !A^* \oplus !B^*$

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

Alors, on peut vérifier que ce plongement de LJ dans LL est correct et fidèle :

$\Gamma \vdash A$ *est prouvable dans LJ ssi* $\vdash ?{\Gamma^*}^{\perp}, A^*$ *est prouvable dans LL.*

(Dans la présentation bilatérale de LL, ce dernier séquent s'écrit $!\Gamma^* \vdash A^*$)

Il n'y a pas d'analogue simple pour la logique classique mais il faut citer *la logique unifiée* LU qui est un système comportant trois fragments remarquables, équivalents respectivement à LL, LJ et LK.

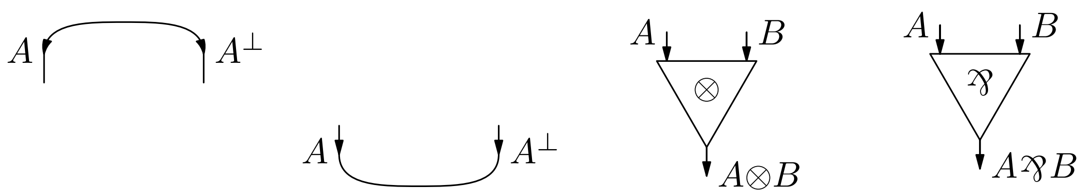
Réseaux de preuves.

Dans un calcul de séquents à la Gentzen (pour ne rien dire d'un système à la Hilbert), les preuves sont séquentielles car il faut choisir un ordre pour l'application des règles; or, celui–ci n'a rien d'intrinsèque et constitue le défaut principal du système.

Une propriété très remarquable des preuves dans LL est qu'elles peuvent se représenter sous la forme de **réseaux**, qui ne souffrent pas de ce défaut : leur prototype, NLJ, se trouve encore une fois dans la logique intuitionniste.

Nous nous limiterons ici au fragment multiplicatif ($\otimes$, $\parr$), qui est à la fois le plus simple et le plus convaincant.

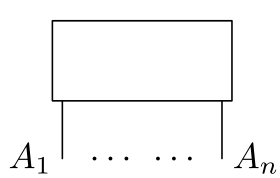
Un réseau typé est un graphe construit avec des éléments :



qui sont respectivement *un lien axiome* (antagonisme), *un lien coupure*, et des cellules pour $\otimes$ et $\parr$.

Les branchements doivent se faire conformément aux étiquettes et aux orientations indiquées. Par la suite nous négligerons de noter les orientations et nous enchaînerons librement des liens (de façon conforme). Nous négligerons aussi les formules qui servent d'étiquettes, mais les seuls réseaux convenables sont bien entendu *les réseaux typables*.

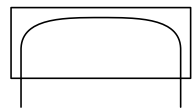
Nous allons montrer comment associer un réseau



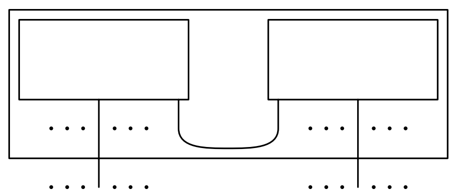
dont tous les ports libres sont figurés et sortants, à une preuve de $\vdash\ A_1, \ldots, A_n$, puis nous étudierons un critère de correction, c'est–à–dire une condition nécessaire et suffisante pour qu'un réseau typable soit un réseau associé à une preuve.

La construction du réseau associé à une preuve se fait par induction, de la façon figurée ci–desous :

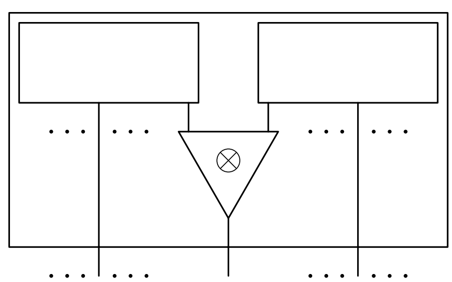
Axiome :



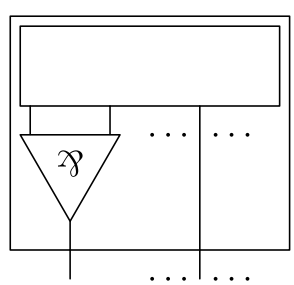
Coupure :



Règle $(\otimes)$ **:**

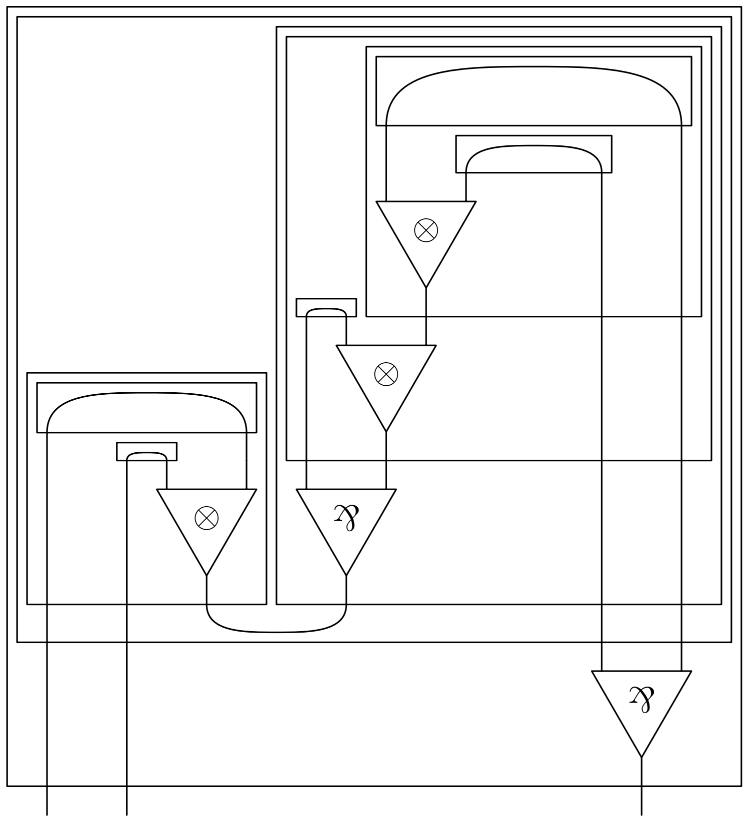


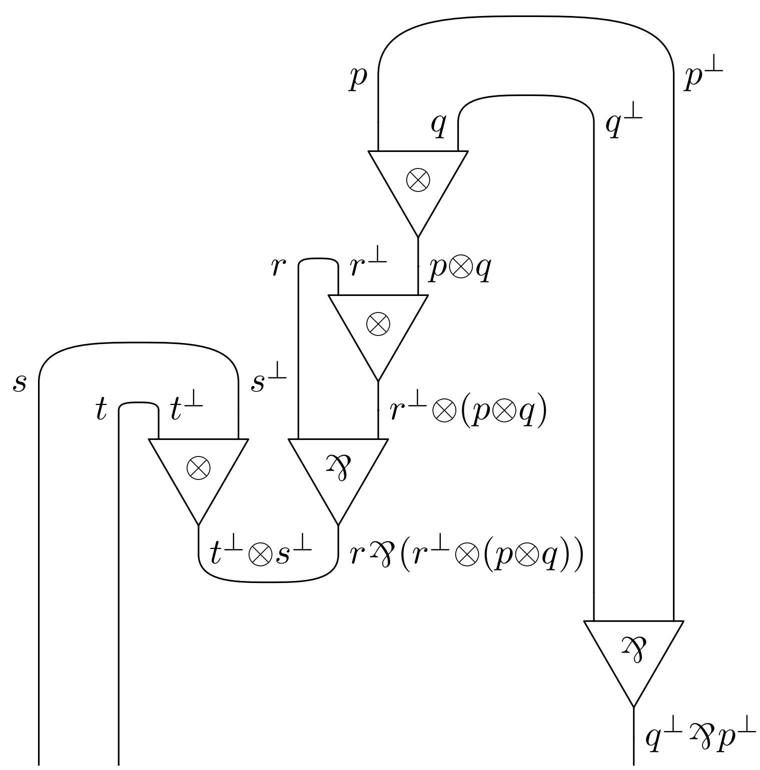
Règle $(\parr)$ **:**



Réciproquement, l'analyse par "mise en boîte" permet de typer un réseau, lorsqu'il est typable (la condition à vérifier est, lorsque l'on rencontre un lien coupure étiqueté X et Y, de pouvoir unifier $X^{\perp}$ et Y). Il est alors facile de constater qu'un réseau typable est le réseau d'une preuve ssi il est analysable par mise en boîte, car cette dernière fournit une procédure de "séquentialisation" du réseau.

Voici l'exemple d'Yves Lafont, où l'on peut prendre $t = r$ et $s = (r^{\perp} \otimes (p \otimes q))$:

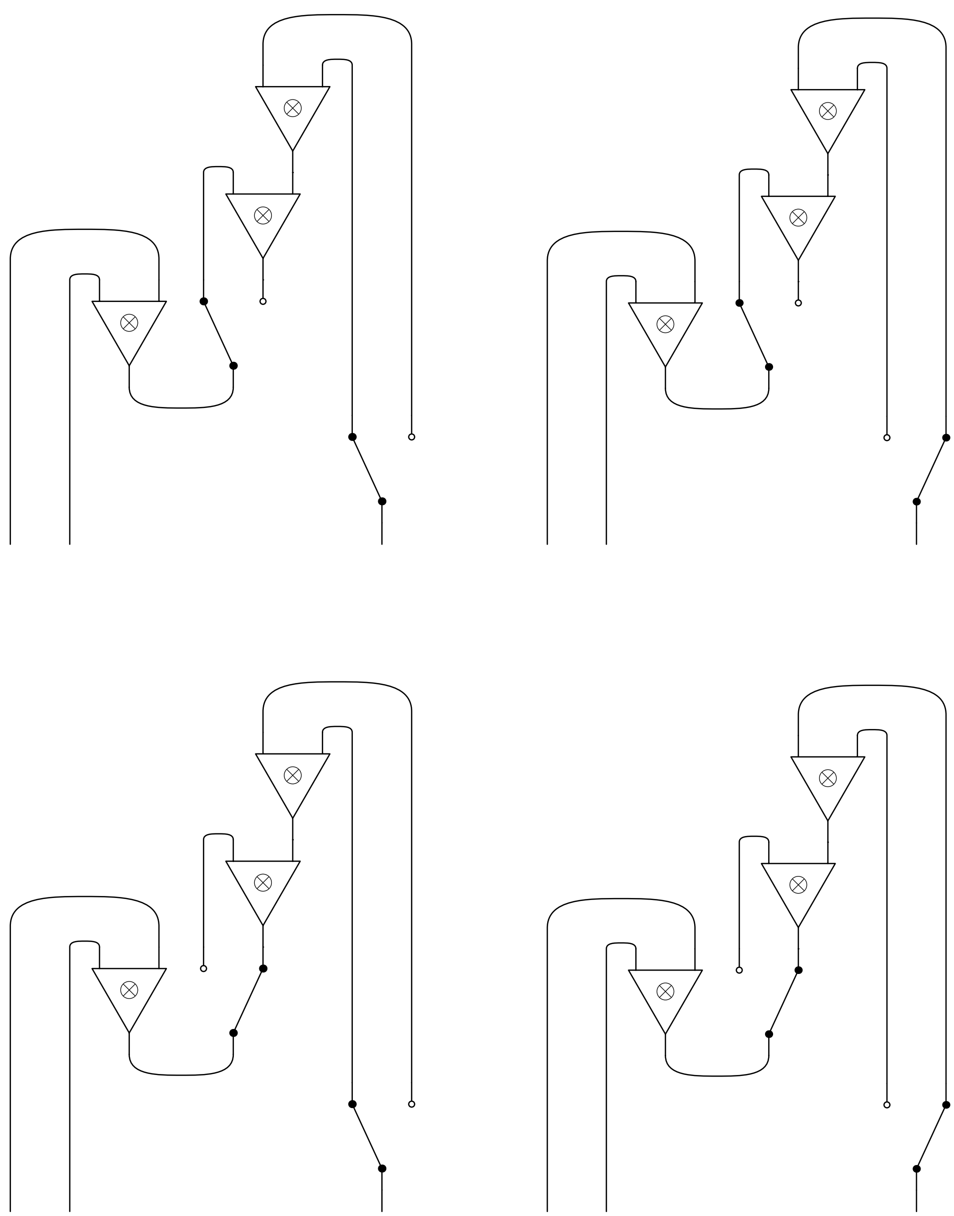




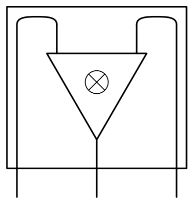
Si l'on transforme les cellules ⅋ en commutateurs à deux positions, permettant de relier l'un des deux ports entrants au port sortant, on peut exprimer **le critère de correction** de la façon suivante :

Un réseau typable est un réseau associé à une preuve ssi pour toutes les positions possibles des commutateurs, les graphes obtenus sont **connexes et sans cycle**.

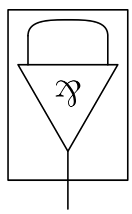
En voici l'illustration sur notre exemple :



La preuve du critère de correction se fait en considérant des réseaux qui comportent, en plus des cellules habituelles, des boîtes, dans l'esprit de l'analyse par emboîtement ci–dessus. Pour des raisons techniques, il faut modifier un peu la façon d'emboîter; les boîtes précédentes correspondant à la coupure et à la règle pour $\parr$ sont conservées, mais les axiomes ne sont plus considérés isolément et donnent lieu à ce qui suit :

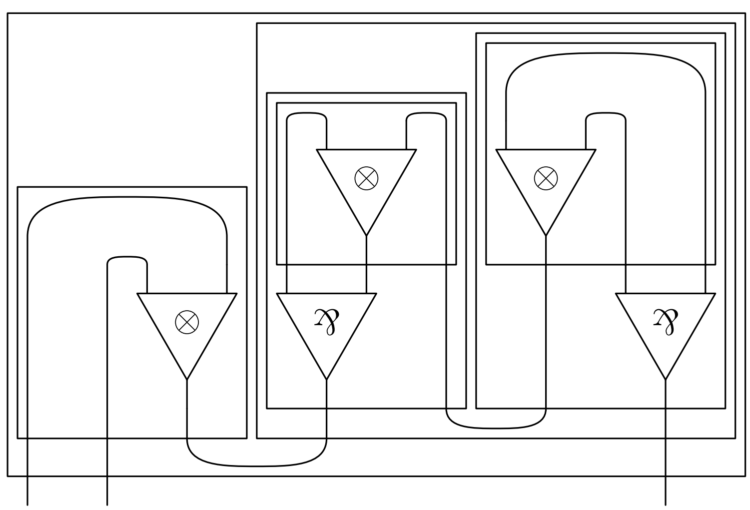


représentant $$\dfrac{\dfrac{}{\vdash p^{\perp}, p} \qquad \dfrac{}{\vdash q, q^{\perp}}}{\vdash p^{\perp}, (p\otimes q), q^{\perp}}$$



représentant $$\dfrac{\dfrac{}{\vdash p, p^{\perp}}}{\vdash (p \parr p^{\perp})}$$

Voici comment notre réseau se trouve maintenant analysé :

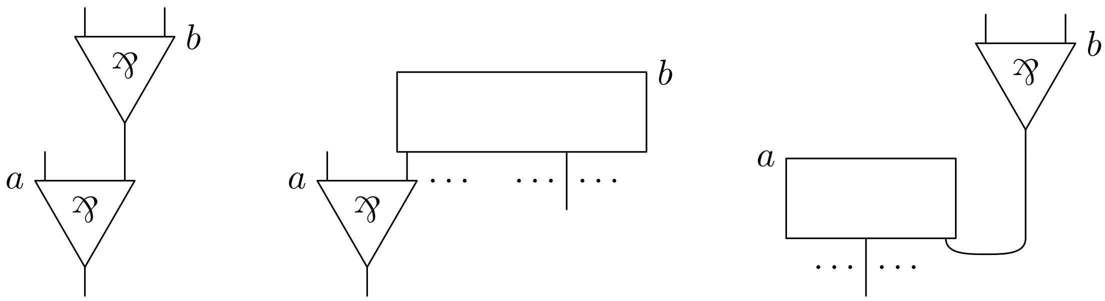


Pour procéder à cette opération, il a été nécessaire de représenter le lien qui unit les deux cellules $\otimes$ par l'enchaînement d'un lien axiome et d'un lien coupure : ce retour à la définition des liens permet de mettre en boîte toute cellule $\otimes$.

Si l'on oublie les contenus, on obtient des "réseaux avec boîtes" pour lesquels l'énoncé du critère fait sens, et on peut énoncer :

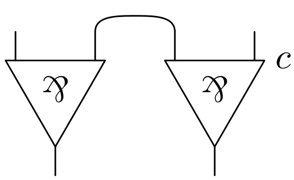
Si un réseau avec boîtes π, ne se résumant pas à un lien axiome, satisfait les conditions du critère et est irréductible par emboîtement, alors π est une boîte.

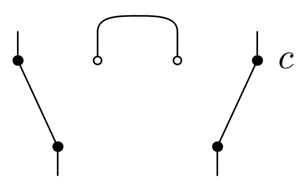
- π étant irréductible pour $\otimes$ ne comporte aucune cellule $\otimes$.
- Considérons la relation "b domine a" décrite par les figures suivantes :



La condition d'acyclicité satisfaite par π implique que celui–ci ne comporte aucun cycle pour cette relation : on peut donc trouver un élément c de π qui n'est dominé par aucun autre.

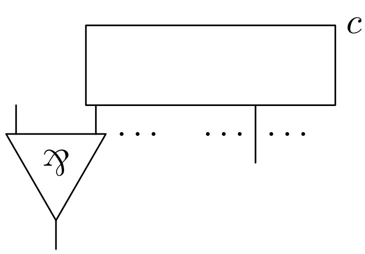
- Si c était une cellule $\parr$, l'un de ses ports entrants ne pourrait être connecté à un port libre de π, car une position du commutateur déconnecterait cette branche; c n'étant pas dominée et π étant irréductible pour $\parr$, on devrait alors avoir la configuration :

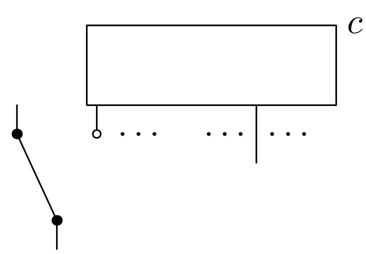


mais


n'est pas connexe.

- c est donc une boîte dont les ports ne pourraient être liés qu'à des ports entrants de cellules $\parr$:



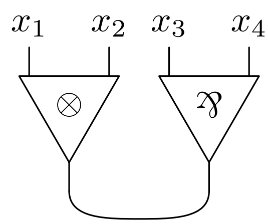
mais


n'est pas connexe.

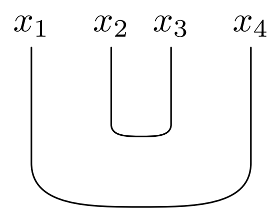
- c est donc une boîte seule au monde dans π!

Normalisation des réseaux.

L'élimination des coupures se traduit de façon très simple pour les réseaux qui nous intéressent ici, en effet :

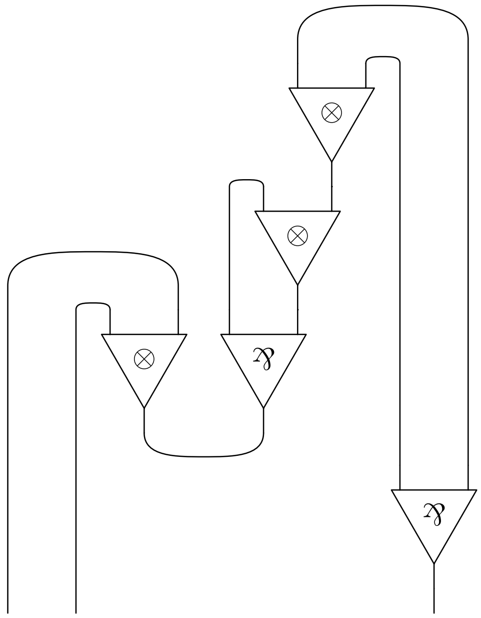


se réduit en

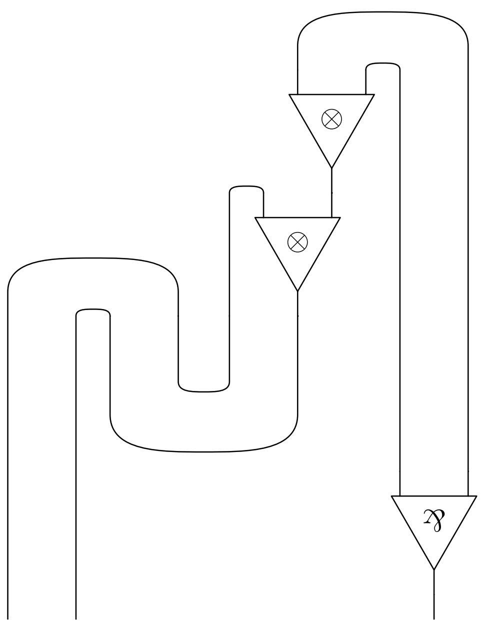


Cette réduction correspond au seul cas clef qui se présente dans le petit fragment de LL où nous nous sommes placés : un fait remarquable, dû à l'absence de toute règle évoquant une contraction, est que l'élimination des coupures diminue la hauteur des preuves. Un exemple est présenté sur la page suivante.

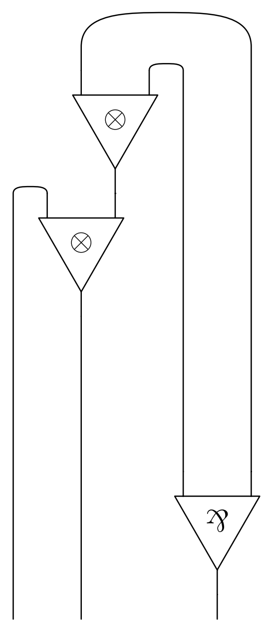
La normalisation de notre réseau lui fait l'effet suivant :



$$\Downarrow$$



$$=$$



SÉMANTIQUES DE LL.

Il y a deux traditions sémantiques :

- la première est une *théorie des modèles* qui s'intéresse à la prouvabilité des formules : la sémantique des phases est de cette espèce;
- la seconde s'intéresse aux preuves et, dans sa forme *dénotationnelle*, à ce qui est implicite dans une preuve.

La sémantique des phases.

Un espace de phases $(M, \perp)$ est la donnée d'un monoïde commutatif (noté multiplicativement et d'élément neutre noté 1) et d'une partie $\perp \subseteq M$.

Pour $X \subseteq M$ et $Y \subseteq M$, on définit $X \multimap Y \subseteq M$ comme un quotient

$$n \in X \multimap Y \text{ ssi pour tout } m \in X \text{ on a } mn \in Y.$$

En particulier, *l'orthogonal de* $X \subseteq M$ est $X^\perp = X \multimap \perp$.
Un fait est $X \subseteq M$ égal à son biorthogonal : $X^{\perp\perp} = X$.
On voit facilement que toute partie de la forme $Y^\perp$ est un fait.

Les parties définies ci–dessous sont des faits lorsque X et Y en sont : les notations choisies permettent alors d'interpréter toute formule de LL (sans quantificateur) comme un fait d'un espace de phases, dès que l'on a choisi une interprétation des atomes :

$$\begin{array}{rclrcl} & & & \mathbf{1} &=& \{1\}^\perp \\ X \parr Y &=& (X^\perp \otimes Y^\perp)^{\perp\perp} & X \otimes Y &=& \{mn \mid m \in X \wedge n \in Y\}^{\perp\perp} \\ \top &=& M & \mathbf{0} &=& \emptyset^{\perp\perp} \\ X \& Y &=& X \cap Y & X \oplus Y &=& (X \cup Y)^{\perp\perp} \\ ?X &=& (X^\perp \cap I)^\perp & !X &=& (X \cap I)^{\perp\perp} \end{array}$$

où I est l'ensemble des idempotents de M qui appartiennent à $\mathbf{1} = \{1\}^\perp$.

Cette sémantique est correcte et complète, c'est–à–dire :

Une formule (du fragment de LL qui nous intéresse) est prouvable ssi dans tout espace de phases, son interprétation contient 1.

La preuve est plus technique que délicate.

Ce résultat rassurant commence à recevoir des applications.

Espaces cohérents.

Issus d'une simplification des domaines de Scott, ils sont à la base d'*une sémantique dénotationnelle*, tout d'abord destinée à la logique intuitionniste, mais la richesse des opérations qu'ils permettaient ouvrait des possibilités qui, paraît-il, sont à l'origine de la logique linéaire; ainsi, la logique linéaire serait–elle apparue sous l'espèce d'une algèbre linéaire sur les espaces cohérents!

Un espace cohérent est un graphe réflexif.

Un espace cohérent X est donc un ensemble noté $|X|$, muni d'une relation binaire réflexive et symétrique notée $x \mathrel{\overset{\frown}{\smile}} y\ [X]$ (ou simplement $x \mathrel{\overset{\frown}{\smile}} y$), appelée *cohérence*.

Il est intéressant de définir les relations suivantes, relativement à un espace cohérent :

- Cohérence stricte : $x \frown y$ ssi $x \mathrel{\overset{\frown}{\smile}} y$ et $x \neq y$;
- Incohérence : $x \asymp y$ ssi $\neg(x \frown y)$;
- Incohérence stricte : $x \smile y$ ssi $\neg(x \mathrel{\overset{\frown}{\smile}} y)$.

Chacune de ces relations (avec ses propriétés respectives) peut servir à la définition d'un espace cohérent.
Nous allons définir sur les espaces cohérents, des opérations analogues aux opérations linéaires (en négligeant les exponentielles).

Négation.

Soit X un espace cohérent, alors la négation linéaire définit un espace cohérent $X^{\perp}$ tel que $\left|X^{\perp}\right| = |X|$ et

- $x \mathrel{\overset{\frown}{\smile}} y\ [X^{\perp}]$ ssi $x \asymp y\ [X]$

ce qui pourrait aussi se définir par

- $x \frown y\ [X^{\perp}]$ ssi $\neg(x \mathrel{\overset{\frown}{\smile}} y)\ [X]$

Les opérateurs multiplicatifs.

Soient X et Y deux espaces cohérents, alors $\otimes$ et $\parr$ définissent des espaces cohérents tels que $|X \otimes Y| = |X \parr Y| = |X| \times |Y|$ et

- $(x, y) \frown\!\!\!\!\smile (x', y')$ $[X \otimes Y]$ ssi $x \frown\!\!\!\!\smile x'$ $[X]$ et $y \frown\!\!\!\!\smile y'$ $[Y]$
- $(x, y) \frown (x', y')$ $[X \parr Y]$ ssi $x \frown x'$ $[X]$ ou $y \frown y'$ $[Y]$

et en posant $X \multimap Y = X^{\perp} \parr Y$, il vient

- $(x, y) \frown (x', y')$ $[X \multimap Y]$ ssi $x \frown\!\!\!\!\smile x'$ $[X]$ implique $y \frown y'$ $[Y]$

Les unités multiplicatives n'ont pas d'interprétation très satisfaisante.

On pose $|\mathbf{1}| = |\perp| = \{0\}$: les deux espaces cohérents sont donc identiques, mais on fait mine de les distinguer en écrivant $\mathbf{1}^{\perp} = \perp$ et $\perp^{\perp} = \mathbf{1}$.

Les opérateurs additifs.

Les opérations additives utilisent la *somme directe d'ensembles* (union disjointe) définie par : $E + F = (E \times \{0\}) \cup (F \times \{1\})$.

Soient X et Y deux espaces cohérents, alors $\&$ et $\oplus$ définissent des espaces cohérents Z tels que $|Z| = |X \& Y| = |X \oplus Y| = |X| + |Y|$ et

- $(x, 0) \frown\!\!\!\!\smile (x', 0)$ $[Z]$ ssi $x \frown\!\!\!\!\smile x'$ $[X]$
- $(y, 1) \frown\!\!\!\!\smile (y', 1)$ $[Z]$ ssi $y \frown\!\!\!\!\smile y'$ $[Y]$
- $(x, 0) \frown (y, 1)$ $[X \& Y]$
- $(x, 0) \smile (y, 1)$ $[X \oplus Y]$

Les unités additives ne sont pas meilleures que les précédentes.

En effet, on pose $|\top| = |\mathbf{0}| = \emptyset$ et on écrit $\top^{\perp} = \mathbf{0}$ et $\mathbf{0}^{\perp} = \top$.

Les constructions sont satisfaisantes dans la mesure où elles vérifient les identités de de Morgan et où les équivalences prouvables dans LL se traduisent en isomorphismes d'espaces cohérents.

Regardons plus spécialement le cas de l'implication.

On note

- $a \sqsubseteq X$ le fait que a est *une clique* de X c'est–à–dire une partie $a \subseteq |X|$ dont les éléments sont cohérents deux à deux;
- $\mathcal{C}l(X)$ l'ensemble des cliques de X.

Applications linéaires.

Une aplication linéaire d'un espace cohérent X vers un espace cohérent Y est une application $F : \mathcal{C}l(X) \to \mathcal{C}l(Y)$ vérifiant

1) $\bigcup_{i\in I} b_i \sqsubseteq X$ implique $F(\bigcup_{i\in I} b_i) = \bigcup_{i\in I} F(b_i)$;
2) $a \cup b \sqsubseteq X$ implique $F(a \cap b) = F(a) \cap F(b)$.

quelles que soient a, b et $b_i \sqsubseteq X$ pour tout élément i d'un ensemble I.

Ces propriétés signifient que F commute aux sommes disjointes. La propriété 1) implique qu'une application linéaire F est entièrement déterminée par sa "restriction" $F : |X| \to \mathcal{C}l(Y)$ puisque

$$F(a) = F(\bigcup_{x\in a}\{x\}) = \bigcup_{x\in a} F(x)$$

Propriété.

Il existe une correspondance biunivoque entre les applications linéaires d'un espace cohérent X vers un espace cohérent Y et l'ensemble des cliques $\mathcal{C}l(X \multimap Y)$.

En effet :

- toute $F : |X| \to \mathcal{P}(|Y|)$ admet un graphe $Tr(F) \subseteq |X| \times |Y|$ (*la trace* de F) :

$$(x, y) \in Tr(F) \text{ ssi } y \in F(x)$$

qui se trouve être une clique de $X \multimap Y$ lorsque F est linéaire;

- toute $A \subseteq |X| \times |Y|$ est le graphe d'une application $A(-) : |X| \to \mathcal{P}(|Y|)$:

$$y \in A(x) \text{ ssi } (x, y) \in A$$

qui se trouve être linéaire lorsque A est une clique de $X \multimap Y$.

Les propriétés $Tr(A(-)) = A$ et $Tr(F)(-) = F$ sont élémentaires.

Interprétation de LL.

Pour simplifier, nous exclurons les exponentielles.

Interprétation des formules et des séquents.

Une telle interpétation est déterminée par la donnée, pour chaque formule atomique p d'un espace cohérent p^* : l'espace cohérent A^* associé à chaque formule A se définit alors par induction, de la façon suivante :

$$A^{\perp *} = A^{*\perp}$$

$$\begin{aligned} \perp^* &= \perp & \qquad \mathbf{1}^* &= \mathbf{1} \\ (A \parr B)^* &= A^* \parr B^* & \qquad (A \otimes B)^* &= A^* \otimes B^* \\ \top^* &= \top & \qquad \mathbf{0}^* &= \mathbf{0} \\ (A \& B)^* &= A^* \& B^* & \qquad (A \oplus B)^* &= A^* \oplus B^* \end{aligned}$$

Un séquent $\vdash A_1, \ldots, A_n$ est interprété par un espace cohérent

$$(\vdash A_1, \ldots, A_n)^* = (A_1, \ldots, A_n)^* = (A_1^*, \ldots, A_n^*),$$

dont la valeur est $A_1^* \parr \ldots \parr A_n^*$. Plus précisément :

Si $X_1, \ldots, X_n$ sont des espaces cohérents, $X = (X_1, \ldots, X_n)$ est l'espace cohérent tel que

- $|X| = |X_1| \times \cdots \times |X_n|$, on représentera ses éléments par des suites $x_1 \ldots x_n$;
- $x_1 \ldots x_n \frown y_1 \ldots y_n$ ssi il existe i tel que $x_i \frown y_i$.

Il est instructif de traduire les définitions qui suivent en termes d'applications linéaires : il suffit de remarquer que, par exemple $(\Gamma, A)^* = \Gamma^{*\perp} \multimap A^*$. Les opérations en question deviennent alors des constructions canoniques sur les applications.

En particulier, **la coupure s'interprète comme la composition** des applications; or, la valeur de $F(G(x))$ ne conserve pas le souvenir de $G(x)$, ni même de F et de G : ceci signifie que les coupures se trouvent "éliminées" de l'interprétation ainsi définie.

Interprétation des preuves de LL (sans les exponentielles).

Toute preuve ρ ☞ $\vdash \Gamma$ dans ce fragment de LL s'interprète comme une clique $\rho^* \sqsubseteq \Gamma^*$ par induction, de la façon suivante :

$$\frac{}{\rho \text{ ☞ } \vdash A, A^{\perp}} \qquad \rho^* = \{xx \mid x \in [A^*]\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, A \quad \lambda \text{ ☞ } \vdash A^{\perp}, \Delta}{\rho \text{ ☞ } \vdash \Gamma, \Delta} \qquad \rho^* = \{\underline{x}\,\underline{z} \mid \exists y(\underline{x}y \in \pi^* \wedge y\underline{z} \in \lambda^*\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma}{\rho \text{ ☞ } \vdash \Gamma, \bot} \qquad \rho^* = \{\underline{x}0 \mid \underline{x} \in \pi^*\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, A, B}{\rho \text{ ☞ } \vdash \Gamma, (A \parr B)} \qquad \rho^* = \{\underline{x}(y,z) \mid \underline{x}yz \in \pi^*\}$$

$$\frac{}{\rho \text{ ☞ } \vdash \Gamma, \top} \qquad \rho^* = \emptyset$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, A \quad \lambda \text{ ☞ } \vdash \Gamma, B}{\rho \text{ ☞ } \vdash \Gamma, (A \& B)} \qquad \rho^* = \{\underline{x}(y,0) \mid \underline{x}y \in \pi^*\} \cup \{\underline{x}(z,1) \mid \underline{x}z \in \lambda^*\}$$

$$\frac{}{\rho \text{ ☞ } \vdash \mathbf{1}} \qquad \rho^* = \{0\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, A \quad \lambda \text{ ☞ } \vdash B, \Delta}{\rho \text{ ☞ } \vdash \Gamma, (A \otimes B), \Delta} \qquad \rho^* = \{\underline{x}(y,z)\underline{t} \mid \underline{x}y \in \pi^* \wedge z\underline{t} \in \lambda^*\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, A}{\rho \text{ ☞ } \vdash \Gamma, (A \oplus B)} \qquad \rho^* = \{\underline{x}(y,0) \mid \underline{x}y \in \pi^*\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma, B}{\rho \text{ ☞ } \vdash \Gamma, (A \oplus B)} \qquad \rho^* = \{\underline{x}(z,1) \mid \underline{x}z \in \pi^*\}$$

$$\frac{\pi \text{ ☞ } \vdash \Gamma}{\rho \text{ ☞ } \vdash \sigma(\Gamma)} \qquad \rho^* = \{\sigma(\underline{x}) \mid \underline{x} \in \pi^*\}$$

Recherche de preuves par focalisation (LL sans les quantificateurs).

Une recherche de preuve est une tentative pour construire une preuve d'un séquent donné, ce qui se fait naturellement en utilisant les règles "à l'envers" c'est–à–dire, de bas en haut : on parle souvent d'une décomposition lorsque l'on fait la rétro–application d'une règle.

Opérateurs asynchrones et formules négatives.

Les formules négatives ont l'une des formes suivantes :

$$p \qquad \bot \qquad (A \parr B) \qquad \top \qquad (A \& B) \qquad ?A$$

où p est un atome (il faut choisir!).

Lorsqu'une formule négative est présente dans un séquent, on peut la décomposer immédiatement, car les opérateurs logiques ci–dessus sont asynchrones.

Opérateurs synchrones et formules positives.

Les formules positives ont l'une des formes suivantes :

$$p^\perp \qquad \mathbf{1} \qquad (A \otimes B) \qquad \mathbf{0} \qquad (A \oplus B) \qquad !A$$

Lorsque toutes les formules d'un séquent sont positives, il faut bien se résoudre à décomposer l'une d'entre elles! mais ceci ne peut pas toujours se faire de façon déterministe car les règles des opérateurs synchrones sont ce qu'elles sont.

Le principe de recherche de preuves par **focalisation** en découle : on exécute une itération des trois opérations suivantes

- décomposition négative complète;
- choix d'une formule (nécessairement positive) : on dit que l'*on se focalise* sur cette formule;
- on exécute *la section focalisée* :
  - décomposition itérée de la formule sur laquelle on est focalisé, puis focalisation sur les sous–formules ainsi obtenues;
  - enfin, sortie de cette section focalisée, lorsque toutes les sous–formules de la formule sur laquelle on s'était initialement focalisé sont devenues négatives.

Pour simplifier la présentation des règles, nous utiliserons des multi–ensembles de formules (et pas de règle d'échange explicite).

Les séquents $\vdash \Pi \mid \Gamma \mid \Delta$ sont divisés en trois champs :

- le premier est *le champ classique*, utilisé pour les exponentielles et la règle d'insertion;
- le second est le champ banal;
- le dernier est *le foyer* : il ne comporte donc qu'une formule au plus.

SYSTÈME DE RECHERCHE FOCALISÉE.

$$\frac{}{\vdash \Pi \mid p \mid p^{\perp}}$$

Changement de phase.

(Foc.) $$\frac{\vdash \Pi \mid \Gamma \mid P}{\vdash \Pi \mid \Gamma, P \mid}$$

$$\frac{\vdash \Pi \mid \Gamma, N \mid}{\vdash \Pi \mid \Gamma \mid N}$$ (Ban.)

Règles logiques.

Opérateurs asynchrones

$$\frac{\vdash \Pi \mid \Gamma \mid}{\vdash \Pi \mid \Gamma, \perp \mid}$$

$$\frac{\vdash \Pi \mid \Gamma, A, B \mid}{\vdash \Pi \mid \Gamma, (A \parr B) \mid}$$

$$\frac{}{\vdash \Pi \mid \Gamma, \top \mid}$$

$$\frac{\vdash \Pi \mid \Gamma, A \mid \quad \vdash \Pi \mid \Gamma, B \mid}{\vdash \Pi \mid \Gamma, (A \& B) \mid}$$

$$\frac{\vdash \Pi, A \mid \Gamma \mid}{\vdash \Pi \mid \Gamma, ?A \mid}$$

Opérateurs synchrones

$$\frac{}{\vdash \ \mid \mid \mathbf{1}}$$

$$\frac{\vdash \Pi \mid \Gamma \mid A \quad \vdash \Pi \mid \Delta \mid B}{\vdash \Pi \mid \Gamma, \Delta \mid (A \otimes B)}$$

(rien pour **0**)

$$\frac{\vdash \Pi \mid \Gamma \mid A}{\vdash \Pi \mid \Gamma \mid (A \oplus B)}$$

$$\frac{\vdash \Pi \mid \Gamma \mid B}{\vdash \Pi \mid \Gamma \mid (A \oplus B)}$$

$$\frac{\vdash \Pi \mid \mid A}{\vdash \Pi \mid \mid !A}$$

Règle d'insertion.

Si Γ ne contient pas de formule négative non–atomique :

$$\frac{\vdash \Pi, A \mid \Gamma, A \mid}{\vdash \Pi, A \mid \Gamma \mid}$$

Dans l'application que nous allons faire de ce système, il ne sera pas question d'exponentielle, aussi, négligerons–nous d'écrire le premier champ qui serait toujours vide.

Application à la programmation logique.

Définisons d'abord deux classes de formules :

Formules héréditairement négatives.

$$N ::= p \mid (N \parr N) \mid (N \& N)$$

Formules bipôlaires.

$$P ::= p^{\perp} \mid (M \otimes M) \mid (M \oplus M)$$
$$M ::= N \mid P$$

Un programme est défini par un ensemble de formules $(p \parr P^{\perp})$ où p est un atome négatif et P une formule bipôlaire : pour respecter les traditions de la programmation logique, une telle formule sera notée $(p \circ\!\!- P)$.

L'intervention d'une formule du programme se fait par le truchement d'une règle qui, en fait, est une coupure propre :

$$\frac{\vdash \Gamma \mid P}{\vdash \Gamma, p \mid} \quad (p \circ\!\!- P)$$

Les philosophes de la table ronde.

n philosophes $\varphi_0, \ldots, \varphi_{n-1}$ sont installés autour d'une table ronde où ils se trouvent séparés par n fourchettes $f_0, \ldots, f_{n-1}$: la forme de la table nous permet de numéroter tout ce monde *modulo* n.

Décrivons un programme qui correspond au problème de ces jeunes gens :

- Lorsque φ_i peut se saisir des fourchettes f_i et f_{i+1} qui l'environnent, il se trouve dans l'état m_i de philosophe qui peut manger (ce qu'il ne manque jamais de faire) :

$$\varphi_i \multimapinv f_i^\perp \otimes m_i \otimes f_{i+1}^\perp$$

- Puis, il repose sagement les deux fourchettes et redevient le simple φ_i :

$$m_i \multimapinv f_i \parr \varphi_i \parr f_{i+1}$$

Les phases bien séparées de la recherche de preuve focalisée permettent de simuler les simultanéités nécessaires au bon déroulement de ce banquet, de façon assez convaincante. Voici le début d'une réunion à deux :

$$\dfrac{\dfrac{\dfrac{}{\vdash f_1 \mid f_1^\perp} \qquad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash \varphi_0, f_1, \varphi_1, f_0 \mid}{\vdash \varphi_0, f_1 \parr \varphi_1, f_0 \mid}}{\vdash \varphi_0, f_1 \parr \varphi_1 \parr f_0 \mid}}{\vdash \varphi_0 \mid f_1 \parr \varphi_1 \parr f_0}}{\vdash \varphi_0, m_1 \mid}\ (m_1 \multimapinv f_1 \parr \varphi_1 \parr f_0)}{\vdash \varphi_0 \mid m_1} \qquad \dfrac{}{\vdash f_0 \mid f_0^\perp}}{\vdash f_0, \varphi_0 \mid m_1 \otimes f_0^\perp}}{\vdash f_0, \varphi_0, f_1 \mid f_1^\perp \otimes m_1 \otimes f_0^\perp}}{\vdash f_0, \varphi_0, f_1, \varphi_1 \mid}\ (\varphi_1 \multimapinv f_1^\perp \otimes m_1 \otimes f_0^\perp)$$

Logique linéaire polarisée.

Les phénomènes de réversibilité et de focalisation conduisent à une analyse des exponentielles :

$$? A = \uparrow \flat A$$
$$! A = \downarrow \sharp A$$

où $\flat$ et $\sharp$ chantent dans le rôle des vraies exponentielles et où $\uparrow$ et $\downarrow$ sont des *décalages*.

Les nouvelles exponentielles entretiennent les mêmes relations que les anciennes :

$$(\flat A)^{\perp} = \sharp(A^{\perp})$$
$$(\sharp A)^{\perp} = \flat(A^{\perp})$$

Les décalages $\uparrow$ et $\downarrow$, sont des changeurs de polarité (c'est–à–dire de signe) qui ont chacun leur changement favori, comme l'explique la définition suivante.

Dans un premier formalisme dit **logique linéaire polarisée** (LLP) dans sa forme, en quelque sorte, **neutre**, les formules sont définies de la façon suivante :

$$N ::= p \mid \perp \mid (N \parr N) \mid \top \mid (N \& N) \mid \sharp N \mid \uparrow P$$
$$P ::= p^{\perp} \mid \mathbf{1} \mid (P \otimes P) \mid \mathbf{0} \mid (P \oplus P) \mid \flat P \mid \downarrow N$$

Les règles de la LLP sont présentées sur la page suivante : les formules négatives ont des règles réversibles, les positives sont celles sur lesquelles on se focalisera, les décalages font passer d'une phase de recherche à une autre.

Les potentialités, au moins combinatoires, laissent entrevoir d'autres possibilités, dont certaines sont actuellement très étudiées :

- La LLPN (négative), dont les formules sont définies par

$$N ::= p \mid \perp \mid (N \parr N) \mid \top \mid (N \& N) \mid \flat \uparrow P \mid \uparrow P$$
$$P ::= p^{\perp} \mid \mathbf{1} \mid (P \otimes P) \mid \mathbf{0} \mid (P \oplus P) \mid \sharp \downarrow N \mid \downarrow N$$

et dont les séquents sont formés de formules négatives et d'au plus une positive.

- La LLPP (positive), dont les formules sont celles de LLP et dont les séquents sont formés de formules positives et d'au plus une négative.

...

SYSTÈME **LLP** (Logique linéaire polarisée).

Les séquents sont de la forme $\vdash \Gamma$ *où* Γ *est un ensemble fini d'occurrences de formules* : N et M sont négatives, P et Q sont positives.

Antagonisme.

$$\frac{}{\vdash p, p^{\perp}}$$

Règles structurelles.

$$(\mathrm{C})\quad \frac{\vdash \Gamma, \flat P, \flat P}{\vdash \Gamma, \flat P} \qquad\qquad \frac{\vdash \Gamma}{\vdash \Gamma, \flat P}\quad (\mathrm{A})$$

Règles logiques.

$$(\bot)\quad \frac{\vdash \Gamma}{\vdash \Gamma, \bot} \qquad\qquad \frac{}{\vdash \mathbf{1}}\quad (\mathbf{1})$$

$$(\parr)\quad \frac{\vdash \Gamma, M, N}{\vdash \Gamma, (M \parr N)} \qquad\qquad \frac{\vdash \Gamma, P \quad \vdash \Delta, Q}{\vdash \Gamma, \Delta, (P \otimes Q)}\quad (\otimes)$$

$$(\top)\quad \frac{}{\vdash \Gamma, \top} \qquad\qquad \text{(rien pour } \mathbf{0}\text{)}$$

$$(\&)\quad \frac{\vdash \Gamma, M \quad \vdash \Gamma, N}{\vdash \Gamma, (M \& N)} \qquad\qquad \frac{\vdash \Gamma, P}{\vdash \Gamma, (P \oplus Q)}\quad (\oplus Q) \qquad \frac{\vdash \Gamma, Q}{\vdash \Gamma, (P \oplus Q)}\quad (P \oplus)$$

Exponentielles.

$$(\sharp)\quad \frac{\vdash \flat \Gamma, N}{\vdash \flat \Gamma, \sharp N} \qquad\qquad \frac{\vdash \Gamma, P}{\vdash \Gamma, \flat P}\quad (\flat)$$

Règles de décalage.

$$(\uparrow)\quad \frac{\vdash \Gamma, P}{\vdash \Gamma, \uparrow P} \qquad\qquad \frac{\vdash \Gamma, N}{\vdash \Gamma, \downarrow N}\quad (\downarrow)$$

Logique linéaire non commutative.

Un séquent linéaire $\vdash \ \Gamma$ est constitué d'un multi–ensemble et l'ordre dans lequel on énumère ses éléments n'est pas significatif : ceci tient au fait que nous n'avons pas encore restreint l'usage de la règle d'échange.

Au point où nous en sommes, il est cependant naturel de tenter de donner à Γ une structure plus fine, par exemple celle d'ensemble ordonné d'occurrences de formules, et plusieurs tentatives ont été faites en ce sens.

La logique NL dont les règles sont présentées plus loin est intéressante parce qu'elle est une extension assez simple de la LL.

- **Deux nouveaux connecteurs multiplicatifs** apparaissent : la conjonction non–commutative notée $\odot$ et la disjonction non–commutative notée $\triangleleft$, reliées par les égalités

$$(A \odot B)^{\perp} = B^{\perp} \triangleleft A^{\perp} \qquad\qquad (A \triangleleft B)^{\perp} = B^{\perp} \odot A^{\perp}$$

Deux implications sont alors définissables :

$$A \multimap\!\!\bullet\, B = A^{\perp} \triangleleft B \qquad\qquad A \bullet\!\!- B = B \triangleleft A^{\perp}$$

- Un séquent de NL est de la forme $\vdash \ \Gamma$ où Γ est un ensemble d'occurrences de formules muni d'une relation d'ordre (pas total en général).

Un tel Γ est construit récursivement à partir des formules par les opérations suivantes :

$$\Gamma_1, \Gamma_2 \qquad\qquad \Gamma_1 ; \Gamma_2$$

où les séparateurs , et ; sont associatifs et où , est le séparateur commutatif habituel. L'image n'est donc plus celle d'une simple liste commutative mais celle d'un arbre, possédant des embranchements commutatifs et d'autres qui ne le sont pas.

Si l'on désigne par $\Gamma[X]$ un séquent de NL dont une feuille est figurée par la variable syntaxique X et si Δ est un autre séquent de NL, alors $\Gamma[\Delta]$ désignera le séquent obtenu en greffant Δ en X dans le séquent initial : cette notion est nécessaire pour exprimer les règles propres à NL.

SYSTÈME **NL** (Logique linéaire non nécessairement commutative).

Toutes les règles de LL, à l'exception de la règle d'échange, sont aussi des règles de NL (le séparateur principal doit être ,) seule la nature des séquents les distingue.

Les règles propres à NL sont les suivantes :

Règles structurelles.

(Entropie) $\dfrac{\vdash \Gamma[\Delta;\Pi]}{\vdash \Gamma[\Delta,\Pi]}$ | $\dfrac{\vdash \Gamma,\Delta}{\vdash \Gamma;\Delta}$ (Co–entropie)

(Devant) $\dfrac{\vdash \Gamma[?\,A,\Pi]}{\vdash [?\,A;\Pi]}$ | $\dfrac{\vdash \Gamma[\Pi,?\,A]}{\vdash \Gamma[\Pi;?\,A]}$ (Derrière)

Règles multiplicatives.

($\odot$) $\dfrac{\vdash \Gamma;A \quad \vdash B;\Delta}{\vdash \Gamma;(A\odot B);\Delta}$ | $\dfrac{\vdash \Gamma;A;B}{\vdash \Gamma;(A\triangleleft B)}$ ($\triangleleft$)

Le système NL vérifie, comme il se doit, la propriété d'élimination des coupures.

Voici quelques formules qui sont prouvables dans NL :

$$(A\triangleleft B)\multimap(A⅋B) \qquad (A\otimes B)\multimap(A\odot B) \qquad (A \multimap\!\bullet\; B)\multimap(A\multimap B)$$

$$!(A\& B) \circ\!\!-\!\!\circ (!\,A\odot !\,B) \qquad ?(A\oplus B) \circ\!\!-\!\!\circ (?\,A\triangleleft ?\,B)$$