Théorie de la démonstration

mardi 12/01/10

Lisp & Scheme sont des langages de programmation non-typés, Caml est typé.

Lectures : - notes de cours (transparents couvrant au moins les deux premiers tiers)
- notes de cours de Olivier Laurent (ancien de p.p.s., maintenant E.N.S. Lyon. Surtout L.L.)
- notes de cours de Dale Miller (cours de théo. de la dém. en M.P.R.I.)
- Proofs and types de Girard, Lafont & Taylor (couvre le corps principal du cours)
- Le point aveugle de Girard (plus récent et plus allusif).

On étudie surtout la logique propositionnelle, et peu la logique des prédicats (à l'exception de ce qui suit). On évoquera la logique du second ordre (qui mène vers Coq).

Exemple de logique des prédicats : Arithmétique de Péano.

Signature : { symboles de fonction : S^1, 0^0, $+^2$; symbole de prédicat : $=^2$ }

La construction ou la recherche de preuves est en lien avec la programmation logique (Prolog).

Axiomes : $Ax_1 : \overline{x + 0 = x}$; $Ax_2 : \overline{x + Sy = S(x+y)}$

On construit les preuves formelles avec les axiomes et les règles d'inférence.

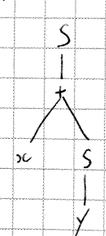
Règles d'inférence : transitivité : $\dfrac{x = y \quad y = z}{x = z}$

Règle d'inférence sans hypothèses : axiome

Par ex. trans. : $\overline{(x=y) \wedge (y=z) \Rightarrow x=z}$

[une relation est congruente si elle passe à travers les opérations du langage. Par ex. en théorie des groupes de signature $(G, \cdot, e)$, avec $\sim \subseteq G \times G$, $g_1 \sim g'_1$ et $g_2 \sim g'_2 \Rightarrow g_1 \cdot g_2 \sim g'_1 \cdot g'_2$, permet de définir le groupe quotient $G/\sim$.]

Informaticiens ont inventé arbres de syntaxe abstraite, qui permet de s'assurer de la validité d'un terme : par exemple $S(x+Sy)$ est



Règle du remplacement des égaux par les égaux : $\dfrac{A_1 = A_2}{t[A_1/x] = t[A_2/x]}$

On a donné un procédure pour décomposer un terme quelconque t_1 en $t_2 = t_1[A/x]$:

1) On isole un sous-arbre A_1 de t_1.
2) On l'enlève et on le remplace par un x "frais" (variable qui n'apparaît pas déjà ailleurs dans t_1)
3) On appelle le terme résultant de 2

On a donné $t = t_1[s_1/x]$

Par exemple, $S(x+Sy) = S(x+ry)[Sy/ry]$ On peut aussi internaliser cette procédure (substitution explicite).
$= S(ry)[x+Sy/ry]$

Notations pour la substitution : • $s[t/x]$ t est mis à la place de x dans s
• $s[x \leftarrow t]$ "x reçoit t", même sens.

On se donne une règle d'inférence simplifiée (instanciation du remplacement des égaux par les égaux) :

$$\frac{A_1 = A_2}{SA_1 = SA_2}\ S\text{-cong.}$$

On a aussi l'instanciation : $\dfrac{t_1 = t_2}{t_1[A/x] = t_2[A/x]}\ \text{Inst.}(A, x)$

Ex. de preuve :

$$\dfrac{\dfrac{\dfrac{}{x+Sy = S(x+y)}\ \text{Ax.2}}{x+SSy = S(x+Sy)}\ \text{Inst.}(Sy, y) \qquad \dfrac{\dfrac{}{x+Sy = S(x+y)}\ \text{Ax.2.}}{S(x+Sy) = SS(x+y)}\ \text{S.cong.}}{x+SSy = SS(x+y).}\ \text{Trans.}$$

Styles

Styles de système de preuves : - à la Hilbert ↓ langage de preuve : logique combinatoire
- déduction naturelle ↓ λ-calcul
- calcul des séquents ↓ système L

Syst. à la Hilbert inventé par Hilbert en 1900, calcul des séquents inventé par Gentzen en 1930 en premier, puis il a inventé la déduction naturelle, qu'il jugeait plus maniable.

Styles de règles : - additives / multiplicatives mène à la logique linéaire (85/6, Girard)
- réversibles / irréversible crucial pour le cours, mène à la logiq classiq constructive (90, Girard, Griffin, un informaticien qui a typé Call/cc).

Autres distinctions : - classique / intuitionniste
- règle dérivable / admissible

(encadré) = principal dans le cours

Règle dérivable / admissible

• Règle dérivée : macro obtenue à partir des règles existantes.

Par exemple, $\dfrac{}{x+SSy = S(S(x+y))}\ \text{Ax.3}$ est un axiome dérivé.

$$\dfrac{\dfrac{h_1}{\quad}\quad\dfrac{\dfrac{h_2}{\quad}\quad\dfrac{h_3}{\quad}}{cc}}{} \overset{\text{déf}}{=} \dfrac{\dfrac{h_1}{\quad}\quad\dfrac{h_2}{\quad}\quad\dfrac{h_3}{\quad}}{cc}\ \text{règle dérivée}$$

• Règle admissible : Toute instance close de cette règle est prouvable.
$x+S0$ n'est pas clos, $SS0$ est clos.

Par exemple $\dfrac{}{x+y = y+x}\ \text{Comm.}$ n'est pas dérivable à partir de Ax_1, Ax_2, Inst., Trans., S-Cong, etc. (car récité on a besoin de l'induction). Par contre on peut prouver $0+SS0 = SS0+0$, $S0+SS0 = SS0+S0$, avec ces règles. etc.

Systèmes à la Hilbert :

Beaucoup d'axiomes d'une seule règle d'inférence : le langage de preuve est très simple, c'est la logique combinatoire, mais les preuves sont peu structurées, peu lisibles.

Logique propositionnelle minimale :

$A ::= X \mid A \Rightarrow A$ (façon de présenter les formules : Grammaire de production (vient de l'informatique) ≠ de présentation par induction)

X : atome

Seule règle d'inférence :

$$\frac{A \Rightarrow B \quad A}{B}$$ Modus Ponens

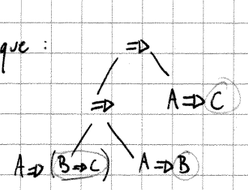
Axiomes :

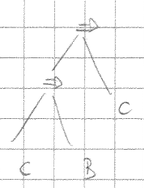
$$\frac{}{A \Rightarrow (B \Rightarrow A)}\ (K) \text{ ou } K^{A,B}$$

objectif : avoir toutes les informations avec simplement le nom des règles, ne pas avoir à regarder les formules

$$\frac{}{((A \Rightarrow (B \Rightarrow C)) \Rightarrow (A \Rightarrow B)) \Rightarrow (A \Rightarrow C)}\ (S) \text{ ou } S^{A,B,C}$$

(S) est une sorte de Modus ponens paramétrique :





(K) a à voir avec l'effacement, (S) avec la duplication. Ces opérations coûtent du temps en programmation (sauf, le calcul se fait en temps linéaire).

Mercredi 13 jan. 10

Convention : on associe à droite, $A \to B \to C$ signifie $A \to (B \to C)$

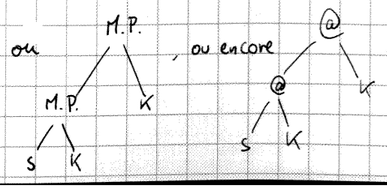
$(A \Rightarrow B \Rightarrow C) \Rightarrow (A \Rightarrow B) \Rightarrow A \Rightarrow C$ correspond à l'axiome $S^{A,B,C}$

Exemple : On veut prouver $A \Rightarrow A$. On instancie (S) avec $A, D \Rightarrow A$ et A pour se faire :

$$\dfrac{\dfrac{S^{A,D\Rightarrow A,A}}{(A\Rightarrow(D\Rightarrow A)\Rightarrow A)\Rightarrow(A\Rightarrow(D\Rightarrow A))\Rightarrow A\Rightarrow A} \quad K^{A,D\Rightarrow A}}{A\Rightarrow(D\Rightarrow A)\Rightarrow A}\ \text{M.P.} \qquad K^{A,D}$$

$$\dfrac{A\Rightarrow(D\Rightarrow A)\Rightarrow A\Rightarrow A \qquad A\Rightarrow(D\Rightarrow A)}{A\Rightarrow A}\ \text{M.P.}$$

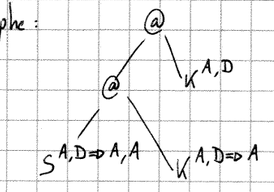
On peut résumer cette (lourde) preuve avec $(SK)K$, ou



, ou encore

.

On parle de terme de preuve, et on peut le rendre encore plus précis : $(S^{A,D\Rightarrow A,A}\ K^{A,D\Rightarrow A})\ K^{A,D}$, ce qui se traduit en graphe :



. Ce qui donne la même information que l'arbre de preuve.

Le nœud-application (qui est un nœud-modus ponens) prend deux sous-preuves en arguments, deux arguments. Il faut que ces deux arguments soient unifiables.

Pour aller vers les langages de programmation, on peut dire que K est une constante de type $A \Rightarrow (B \Rightarrow A)$, et on note $K^{A,B} : A \Rightarrow (B \Rightarrow A)$. Un terme a pour type une formule.

On présente autrement le modus ponens, en l'explicitant en nommant les preuves π_1, π_2, etc. :

$$\frac{\pi_1 : A \Rightarrow B \quad \pi_2 : A}{\pi_1 \pi_2 : B}\ \text{M.P.}$$ C'est la règle de typage.

On peut en effet appliquer $S^{A,D\Rightarrow A,A}$ à $K^{A,D\Rightarrow A}$, et on a $(S^{A,D\Rightarrow A,A}\ K^{A,D\Rightarrow A}) : (A \Rightarrow D \Rightarrow A) \Rightarrow A \Rightarrow A$, terme que l'on peut appliquer à $K^{A,D}$, et on a : $((S^{A,D\Rightarrow A,A}\ K^{A,D\Rightarrow A})\ K^{A,D}) : A \Rightarrow A$. Ce terme est correctement typé.

↳ C'est le plus simple des langage de preuve. Le langage de preuve pour la logique minimale et la logique combinatoire (exploitée par Curry). On peut définir une preuve d'une façon proche de la programmation.

$P : A$, P un programme et A un type, $\pi : A$, π une preuve formelle et A une formule.

Syntaxe de la logique combinatoire : $\pi ::= K \mid S \mid \pi\pi$, avec des types de préférence.

Calcul des séquents

Un séquent est un objet formel de la forme $\Gamma \vdash \Delta$. $\vdash$ se lit "thèse", en anglais "turnstile". Γ est un multi-ensemble de formules, Δ aussi.

Un multi-ensemble de formules accepte les répétitions, par ex. $\Gamma = \{A, A, B, C, C, C\}$. On peut le voir comme une fonction : $\mu(A) = 2$, $\mu(B) = 1$, $\mu(C) = 3$, $\mu(d) = 0$ si $d \neq A, B, C$.

Un multi sous-ensemble de X est une fonction de X dans $\mathbb{N}$.

En terme d'ensembles, $\{A,B,C\} \cup \{A,B,D\} = \{A,B,C,D\}$.

En terme de multi-ensembles, $\{A,B,C\} \cup^m \{A,B,D\} = \{A,A,B,B,C,D\}$.

Un séquent est de la forme $A_1, \dots, A_n \vdash B_1, \dots, B_m$. Dans un multi-ensemble, comme dans un ensemble, l'ordre ne compte pas : $\{A_n, \dots, A_1\} = \{A_1, \dots, A_n\}$, et il faut garder à l'esprit qu'il est possible que $i \neq j$ et $A_i = A_j$.

On manipule désormais des règles du type $\dfrac{\Gamma_1 \vdash \Delta_1 \qquad \Gamma_2 \vdash \Delta_2}{\Gamma \vdash \Delta}$

On peut voir $A_1, \dots, A_n \vdash B_1, \dots, B_m$ comme $(A_1 \wedge A_2 \wedge \dots \wedge A_n) \Rightarrow (B_1 \vee \dots \vee B_m)$, mais la notation en séquent permet d'accéder directement aux sous-formules.

conjonction vide — disjonction vide

Cas particuliers : $\vdash B$ signifie "vrai implique B", donc on peut le voir comme B.

$A \vdash$ signifie "A implique faux", qui classiquement signifie $\neg A$.

$\vdash$ signifie "vrai implique faux", donc "non-vrai", doit faux.

Les formules de la logique classique : $A ::= X \mid A \wedge A \mid A \vee A \mid \neg A$

Pas de quantification (propositionnel), X est un atome.

Ses règles : On les divise en trois groupes : Axiome et coupure, Règles d'introduction à gauche, et Règles d'introduction à droite.

$$\text{Axiome}\ \frac{}{\Gamma, A \vdash A, \Delta}$$

$$\text{Coupure}\ \frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$$

Règle qui a à voir avec le M.P. : $\dfrac{\dfrac{\vdash A}{\vdash A, B}\ \text{(Une des façon qu'on peut avoir de prouver } \vdash A, B) \qquad A \vdash B}{\vdash B}$

(Un séquent est vrai, au sens des tables de vérités, pour $\Gamma \vdash \Delta$, ssi au moins une formule de Γ est fausse ou au moins une formule de Δ est vraie.) ← non exclusif

Décomposition — Intro. à droite ↓

$$\frac{\Gamma \vdash A_1, \Delta \qquad \Gamma \vdash A_2, \Delta}{\Gamma \vdash A_1 \wedge A_2, \Delta} \qquad \frac{\Gamma \vdash A_1, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta} \qquad \frac{\Gamma \vdash A_2, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta}$$

$$\frac{\Gamma, A_1 \vdash \Delta \qquad \Gamma, A_2 \vdash \Delta}{\Gamma, A_1 \vee A_2 \vdash \Delta}$$

↑ dual de l'introduction de $\wedge$ à droite.

$$\frac{\Gamma, A_1, A_2 \vdash \Delta}{\Gamma, A_1 \wedge A_2 \vdash \Delta} \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$

↑ associativité de la conjonction : $A_1 \wedge A_2 \wedge A_3 = A_1 \wedge (A_2 \wedge A_3)$. Pas dual de $\vee$ à droite $\Rightarrow$ choix non pas pour la complétude mais pour le calcul.

On nomme ce système LK (bien que ce soit une variante du système de Gentzen), il est complet pour la logique classique : pour toute formule A, le séquent $\vdash A$ est prouvable ssi A est une tautologie classique (au sens de la distribution des valeurs de vérité).

Les règles (hormis axiome et coupure) donnent le sens des connecteurs à droite et à gauche. On a en vue l'automatisation de la recherche de preuves, qui décompose les formules en sous-formules. Lorsque l'on décompose $\vdash A_1 \vee A_2$, il y a un choix à faire : tenter de prouver $\vdash A_1$ ou tenter de prouver $\vdash A_2$.

Exemple :

Décomposition ↑

$$\dfrac{\dfrac{C_1, C_2 \vdash}{C_1 \wedge C_2 = B \vdash}}{\vdash A_1 = \neg B} \leftarrow \text{choix : on aurait pu décomposer } A_2.$$
$$\vdash A_1 \vee A_2$$

On peut ensuite décomposer C_1 ou C_2.

On ajoute quatre règles :

Affaiblissement

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta} \qquad \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Contraction

$$\frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta} \qquad \frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

On peut simuler la contraction avec la coupure (ce que fait Girard) :

$$\dfrac{\Gamma \vdash A, A, \Delta \qquad \dfrac{\Gamma, A \vdash A, \Delta}{\Gamma \vdash A, \Delta}\ Ax.}{\Gamma \vdash A, \Delta}\ \text{Coupure}$$

On obtient une macro. L'affaiblissement est admissible : d'une preuve de $\Gamma \vdash \Delta$ on peut construire une preuve de $\Gamma \vdash \Delta, A$ par un algorithme simple, qui modifie tous les axiomes en ajoutant A à droite (les règles restent applicables). L'affaiblissement est même transparent, on peut identifier ces deux preuves sans dommage.

"Inutance contraction" : légère modification de la règle de coupure (contextes différents)

Ex. :
$$\dfrac{\dfrac{\dfrac{\dfrac{}{A\vdash A}\,Ax}{A\vdash A\vee\neg A}}{\vdash \neg A, A\vee\neg A}}{\vdash A\vee\neg A, A\vee\neg A}}{\vdash A\vee\neg A}$$

La logique intuitionniste ne peut dériver cette formule : elle « bride » LK en n'autorisant qu'une formule à droite (donc la contraction droite n'a plus l'occasion de s'appliquer).

L'implication :

On ne se la donne pas primitivement car il est possible de la "simuler", $A\Rightarrow B$ est une fonction de A dans B.

$$\dfrac{\Gamma, A\vdash B, \Delta}{\Gamma\vdash A\Rightarrow B, \Delta} \qquad \dfrac{\Gamma\vdash A, \Delta \quad \Gamma, B\vdash \Delta}{\Gamma, A\Rightarrow B\vdash \Delta}$$

Ces règles sont dérivables, on les a comme une macro à partir de $A\Rightarrow B \equiv \neg(A\wedge\neg B)$

$$\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, A\vdash B, \Delta}{\Gamma, A, \neg B\vdash \Delta}}{\Gamma, A\wedge\neg B\vdash \Delta}}{\Gamma\vdash \neg(A\wedge\neg B), \Delta}}{\Gamma\vdash \neg(A\wedge\neg B), \Delta} \qquad \dfrac{\Gamma\vdash A, \Delta \quad \dfrac{\Gamma, B\vdash \Delta}{\Gamma\vdash \neg B, \Delta}}{\dfrac{\Gamma\vdash A\wedge\neg B, \Delta}{\Gamma, \neg(A\wedge\neg B)\vdash \Delta}}$$

Sens du séquent :

Quelle est la différence entre $C, A\vdash B$ et $C\vdash A\Rightarrow B$?

Par métaphore avec l'enseignement du secondaire, on voit dans le premier cas C et A comme des paramètres, et dans le second cas A comme une variable.

Par exemple, $mx^2+(3-m)x+5=0$ a pour variable x et pour paramètre m. On peut résoudre l'équation en trouvant x tel que $f(x)=0$ avec $f: x\mapsto mx^2+(3-m)x+5$. f est en fonction de x et pas de m.

On peut faire jouer des rôles différents aux variables libres. Dans le passage du premier au second séquent, A est passé de variable implicite à variable explicite. Le symbole $\vdash$ n'a donc pas complètement le même sens que l'implication.

Programme : * Complétude de LK
* Élimination des coupures

→ calcul des séquents / déduction naturelle
→ multiplicatif / additif (logique linéaire)
→ reversible / irreversible (logiques polarisées, "constructives")

mardi 1er janvier

Déduction naturelle

dont le langage de programmation sous-jacent est le λ-calcul. En calcul des séquents, on a des règles d'introduction de connecteurs à droite et à gauche. En déduction naturelle on a des règles d'élimination (à droite) et des règles d'introduction (à droite). On va les introduire et montrer comment on passe de la déduction naturelle au calcul des séquents.

Les $\Rightarrow$-elim est dérivable à partir des règles du calcul des séquents :

$$(\Rightarrow\text{-elim})\ \dfrac{\Gamma\vdash A\Rightarrow B, \Delta \qquad \Gamma\vdash A, \Delta}{\Gamma\vdash B, \Delta}$$

$$(\text{aff. d})\ \dfrac{\dfrac{\Gamma\vdash A\Rightarrow B, \Delta}{\Gamma\vdash A\Rightarrow B, \Delta, B}\,(\text{aff. d}) \qquad \dfrac{\dfrac{\Gamma\vdash A, \Delta}{\Gamma\vdash A, \Delta, B}\,(Ax) \qquad \dfrac{}{\Gamma, B\vdash B, \Delta}\,(Ax)}{\Gamma, A\Rightarrow B\vdash \Delta, B}\,(\Rightarrow g)}{\Gamma\vdash B, \Delta}\,(\text{coupure})$$

Le double trait lors de l'affaiblissement droite signale que cette règle est admissible, bien qu'elle ne fasse pas partie de nos règles. L'affaiblissement est admissible de façon vellement triviale qu'on admettra comme juste la dérivation :

$$\dfrac{\dfrac{\Gamma\vdash A\Rightarrow B, \Delta \qquad \dfrac{\Gamma, B\vdash B, \Delta \qquad \Gamma\vdash A, \Delta}{\Gamma, A\Rightarrow B\vdash \Delta, B}}{\Gamma\vdash \Delta, B}}{}$$

Le codage de la déduction naturelle vers le calcul des séquents ne sera pas traité ici, mais peut se trouver dans Proofs and Types.

Règle additive, règle multiplicative

Chaque règle existe sous deux styles, par exemple la conjonction droite :

Additive : $\dfrac{\Gamma\vdash A, \Delta \qquad \Gamma\vdash B, \Delta}{\Gamma\vdash A\wedge B, \Delta}$

Multiplicative : $\dfrac{\Gamma_1\vdash A, \Delta_1 \qquad \Gamma_2\vdash B, \Delta_2}{\Gamma_1, \Gamma_2\vdash A\wedge B, \Delta_1, \Delta_2}$

Les formules sont vues comme des ressources, en additif le même Γ peut contribuer à prouver A et B, en multiplicatif il y a une gestion plus informatique des hypothèses.

On montre que la règle $(\wedge d)$ multiplicative s'obtient à partir de son pendant additif :

$$\dfrac{\dfrac{\Gamma_1 \vdash A, \Delta_1}{\Gamma_1, \Gamma_2 \vdash A, \Delta_1, \Delta_2}(aff.dkg) \qquad \dfrac{\Gamma_2 \vdash B, \Delta_2}{\Gamma_1, \Gamma_2 \vdash B, \Delta_1, \Delta_2}(aff.dkg)}{\Gamma_1, \Gamma_2 \vdash A \wedge B, \Delta_1, \Delta_2}(\wedge d)_a$$

Les affaiblissements sont toujours vus comme des macros qui transforment des preuves en ajoutant des formules lors des axiomes.

$(\wedge d)$ mult. s'obtient grâce à $(\wedge d)$ add. et affaiblissements (weakening).

On montre la réciproque :

$$\dfrac{\dfrac{\Gamma \vdash A, \Delta \qquad \Gamma \vdash B, \Delta}{\Gamma, \Gamma \vdash A \wedge B, \Delta, \Delta}(\wedge d)_m}{\Gamma \vdash A \wedge B, \Delta}(ctr.)$$

La règle de contraction est un cas particulier de la coupure-contraction : $\dfrac{\Gamma \vdash A, A, \Delta \qquad \dfrac{}{\Gamma, A \vdash A, \Delta}(ax.)}{\Gamma \vdash A, \Delta}$

$(\wedge d)$ add. s'obtient grâce à $(\wedge d)$ mult. et contractions.

Note : Ici le double-trait souligne le fait qu'il y a plusieurs application de l'aff. d dkg, ou ctr. d. dkg.

En se privant de l'affaiblissement et de la contraction, on obtiendra deux connecteurs différents : la conjonction additive deviendra & ("with"), et la multiplicative devient ⊗ ("tenseur"). Ils correspondent d'un point de vue sémantique au produit cartésien et au produit tensoriel.

On distinguera également la disjonction en deux styles, puis en deux connecteurs distincts.

Réversible, irréversible

$(\vee d)$ en irréversible : $\dfrac{\Gamma \vdash A_1, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta} \qquad \dfrac{\Gamma \vdash A_2, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta}$

$(\vee d)$ en réversible : $\dfrac{\Gamma \vdash A_1, A_2, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta}$

Se donner l'un ou l'autre ne change pas le pouvoir expressif (via affaiblissement et contraction).

De façon informelle, $\vee d$ en irréversible impose un choix : lorsqu'on remonte la preuve, on choisit de prouver A_1 ou A_2 (du moins sans "backtracking"). En réversible, lorsqu'on remonte la preuve on ne perd aucune information. $\Gamma \vdash A_1, A_2, \Delta$ se lit d'ailleurs de la même façon que $\Gamma \vdash A_1 \vee A_2, \Delta$.

→ Si l'hypothèse de la règle peut être déduite de la conclusion, la règle est réversible.

$$(aff.)\ \dfrac{\Gamma \vdash A_1 \vee A_2, \Delta}{\Gamma \vdash A_1 \vee A_2, A_1, A_2, \Delta} \qquad \dfrac{\dfrac{}{\Gamma, A_1 \vdash A_1, \Delta} \qquad \dfrac{}{\Gamma, A_2 \vdash A_2, \Delta}(vg)}{\Gamma, A_1 \vee A_2 \vdash A_1, A_2, \Delta}(\text{contraction coupure})$$
$$\Gamma \vdash A_1, A_2, \Delta$$

On obtient une macro dont l'interface est une preuve de $\Gamma \vdash A_1 \vee A_2, \Delta$: à partir de la conclusion on peut obtenir l'hypothèse.

Pour prouver qu'une règle est irréversible, il faut faire appel à la sémantique.

Atomisation de l'axiome

Un atome est indécomposable, c'est une variable de formule en quelque sorte.

$$\dfrac{}{\Gamma, A \vdash A, \Delta}(Ax.) \qquad \dfrac{}{\Gamma, X \vdash X, \Delta}(Ax.\ at.)$$

Prop. : $(Ax.)$ est admissible à partir de $(Ax.\ at.)$

Preuve : Par récurrence sur la taille de A. Pour mémoire, $A ::= X \mid A \vee A \mid A \wedge A \mid \neg A$.

Si A est atomique c'est trivial, c'est la règle $(Ax.\ at.)$.

Si A est complexe, et si son connecteur principal est $\neg$: $\dfrac{\dfrac{\Gamma, A \vdash A, \Delta}{\Gamma \vdash A, \neg A, \Delta}}{\Gamma, \neg A \vdash \neg A, \Delta}$ et par récurrence ça fonctionne la complexité de A est inférieure à celle de $\neg A$.

Si c'est $\vee$: $\dfrac{\dfrac{\Gamma, A \vdash A, \Delta}{\Gamma, A \vdash A \vee B, \Delta}(vd^1) \qquad \dfrac{\Gamma, B \vdash B, \Delta}{\Gamma, B \vdash A \vee B, \Delta}(vd^2)}{\Gamma, A \vee B \vdash A \vee B, \Delta}(vg)$ Technique de preuve : retarder l'irréversible quand c'est possible.

Pour simuler la $\to$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, A \vdash A, \Delta \qquad \Gamma, \neg B \vdash \neg B, \Delta}{\Gamma, A, \neg B \vdash A \wedge \neg B, \Delta}\wedge d}{\Gamma, A, \neg B \vdash A \wedge \neg B, \Delta}\neg g}{\Gamma, \neg(A \to B), A \wedge \neg B \vdash \Delta}\neg g}{\Gamma, \neg(A \wedge \neg B) \vdash \neg(A \to B), \Delta}\neg d}{\Gamma, A \to B \vdash A \to B, \Delta}trad$$

Si c'est $\wedge$: $\dfrac{\dfrac{\Gamma, A \vdash A, \Delta}{B, \Gamma, A \vdash A, \Delta}(aff.) \qquad \dfrac{\Gamma, B \vdash B, \Delta}{A, \Gamma, B \vdash B, \Delta}(aff.)}{\Gamma, A, B \vdash A \wedge B, \Delta}(\wedge d)$ Et par hypothèse de récurrence le résultat est démontré.
$$\Gamma, A \wedge B \vdash A \wedge B, \Delta \quad (\wedge g)$$

Vocabulaire : · les systèmes de calcul des séquents sont désignés par L, les systèmes de déduction naturelle par N (provient de Gentzen).

· Intuitionniste : J Classique : K

L (LJ) (LK) ← Notre point de vue

N (NJ) LJ

↑ Th. Soly débute avec NJ et passe à LJ avec les opérateurs de contrôle.

Théorème : LK est complet, c'est-à-dire que quelque soit la formule A, $\vdash A$ est prouvable ssi A est valide (au sens de ses tables de vérité).

déf : Pour toute valuation de ses atomes, la formule est valide si la table de vérité donne vrai.
Une formule est satisfiable s'il existe une valuation des atomes pour laquelle la table de vérité donne vrai.
Valide → pour toute valuation... Satisfiable → il existe une valuation...

Preuve : 1) On choisit la présentation de LK.

Dans le système donné, toutes les règles sont réversibles, sauf (cut), que l'on remplace par $\dfrac{\Gamma \vdash A_1, A_2, \Delta}{\Gamma \vdash A_1 \vee A_2, \Delta}$, qui est réversible.

On choisit de prendre (Ax. at.) plutôt que (Ax.).

2) Pour montrer $(\vdash A) \Rightarrow (\models A)$ ("soundness"), on admet l'axiome général et la règle (cut/contraction), c'est-à-dire qu'on n'a pas besoin de restreindre notre système.

Pour montrer $(\models A) \Rightarrow (\vdash A)$, on se restreint à (Ax. at.) et on exclut la règle (cut).

rem. : $\dfrac{\Gamma \vdash A, \Delta \quad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$ (cut/contr.) La contraction est une instance de cette règle : $\dfrac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}$ (contr) $\dfrac{}{\Gamma, A \vdash A, \Delta}$ (Ax)

(cut) est toute instance des (cut/contr.) différente de (contr.).

· On montre par récurrence sur la taille de la preuve de $\vdash A$ que $(\vdash A) \Rightarrow (\models A)$. "Trouver la bonne charge inductive".

On montre $(\Gamma \vdash \Delta) \Rightarrow (\models (\Gamma \vdash \Delta)) =_{def.} \models (\bigwedge \Gamma) \Rightarrow (\bigvee \Delta)$ (disjonction des formules de Δ ; conjonction des formules de Γ)

Par ex. si la dernière règle de la preuve de A est $(\neg d)$, alors A est de la forme $\neg B$ et : $\dfrac{\Gamma, B \vdash \Delta}{\Gamma \vdash \neg B, \Delta}$.

Par récurrence $\models (\bigwedge \Gamma \wedge B) \Rightarrow (\bigvee \Delta)$, or $(X \wedge Z \Rightarrow Y)$ a la même table de vérité que $(X \Rightarrow \neg Z \vee Y)$, donc $\models (\bigwedge \Gamma) \Rightarrow (\bigvee \Delta \vee \neg B)$.

On traite les autres règles de façon similaire.

· On démontre la complétude en s'inspirant de la preuve de Henkin. On montre $(\models A) \Rightarrow (\vdash A)$. Pour se faire on construit la preuve de A en utilisant les règles qu'on s'est données (la contraction et les règles de décomposition à droite et à gauche) : c'est une preuve en construction.

À partir de la conclusion, $\vdash A$, on construit une preuve, un arbre partiel, et l'on s'arrête aux feuilles, $\Gamma_1 \vdash \Delta_1, \dots, \Gamma_n \vdash \Delta_n$. On nomme Π cette tentative de preuve dont on n'est pas encore assuré de la validité.

Observations fondamentales : · Comme toute les règles sont réversibles, et par "soundness", $\models A$ ssi $(\models (\Gamma_1 \vdash \Delta_1)$ et .... et $\models (\Gamma_n \vdash \Delta_n))$. (réversible ; soundness)

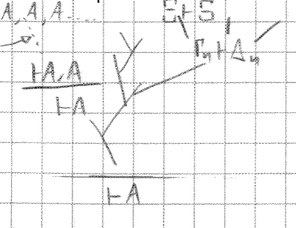
· Π a un nombre de branches fini, et chaque branche est finie. En continuant systématiquement la décomposition des formules, on obtient un essai de preuves dont les feuilles sont constituées uniquement d'atomes.

· Un séquent qui ne contient que des atomes n'est valide que si et seulement si c'est un axiome, donc qu'il existe un atome qui se trouve des deux côtés du signe $\vdash$.

Pour Ξ et Ξ' constitués uniquement d'atomes, $\models (\Xi \vdash \Xi') \Rightarrow \exists X \in \Xi \cap \Xi'$.

↳ on le montre par contraposé, si le séquent est de la forme $X_1, \dots, X_n \vdash Y_1, \dots, Y_p$, avec $X_i \neq Y_j$ pour $1 \leq i, j \leq p$, alors la valuation $X_1, \dots, X_n$ = vrai, et $Y_1, \dots, Y_p$ = faux, rend ce séquent non-valide.

Souci : Puisqu'on a la contraction, il existe potentiellement des branches infinies. Le problème sera contourné par le retrait de la règle (cut/contraction).



Si $\Gamma \vdash \Delta$ est prouvable en LK

⇓ en LKrev

⇓ en LKrev - (cut/contr.) par le cor. 1.

⇓ en LK - cut

LK : $\dfrac{}{\Gamma, \textcircled{A} \vdash \textcircled{A}, \Delta}\, A_x$ $\qquad$ $\dfrac{\Gamma \vdash \textcircled{A}, \Delta \quad \Gamma, \textcircled{A} \vdash \Delta}{\Gamma \vdash \Delta}$ cut/contr.

Formule entourée = Formule active dans la règle.

$\dfrac{\Gamma, \textcircled{A} \vdash \Delta}{\Gamma \vdash \neg \textcircled{A}, \Delta}$ $\quad$ $\dfrac{\Gamma \vdash \textcircled{A}, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash \textcircled{A \wedge B}, \Delta}$ $\quad$ $\dfrac{\Gamma \vdash \textcircled{A}, \Delta}{\Gamma \vdash \textcircled{A \vee B}, \Delta}$ $\quad$ $\dfrac{\Gamma \vdash \textcircled{B}, \Delta}{\Gamma \vdash \textcircled{A \vee B}, \Delta}$

$\dfrac{\Gamma \vdash \textcircled{A}, \Delta}{\Gamma, \neg \textcircled{A} \vdash \Delta}$ $\quad$ $\dfrac{\Gamma, \textcircled{A}, \textcircled{B} \vdash \Delta}{\Gamma, \textcircled{A \wedge B} \vdash \Delta}$ $\quad$ $\dfrac{\textcircled{A}, \Gamma \vdash \Delta \quad \Gamma, \textcircled{B} \vdash \Delta}{\Gamma, \textcircled{A \vee B} \vdash \Delta}$

On définit désormais des règles de réécriture sur les preuves, en dissociant les cas où la formule de coupure vient d'être introduite à droite et à gauche des autres cas. On traite ces cas en premier (cas logiques).

Une formule n'est pas active dans un séquent mais dans une règle : $\dfrac{\dfrac{B_1, B_2, \textcircled{A} \vdash \Delta \quad B_1, B_2, \textcircled{B} \vdash \Delta}{B_1, B_2, \textcircled{A \vee B} \vdash \Delta}\, \vee g}{B_1 \wedge B_2, A \vee B \vdash \Delta}\, \wedge g$

Actives dans $\vee g$ — actives dans $\wedge g$

Si à un moment dans une preuve on croise ce bloc : $\dfrac{\dfrac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta} \quad \dfrac{\Gamma \vdash A, \Delta}{\Gamma, \neg A, \Delta}}{\Gamma \vdash \Delta}$

On peut sans dommages le remplacer par un autre bloc, qui a les mêmes interfaces,

à savoir : $\dfrac{\Gamma, A \vdash \Delta \quad \Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}$

C'est une règle de réécriture de la preuve pour le cas où la formule coupée vient d'être introduite à droite et à gauche par des règles concernant la négation.

On traite désormais la règle de réécriture logique pour le cas de ou :

$$\dfrac{\dfrac{\overset{1}{\vdots}\ \Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \quad \dfrac{\overset{2}{\vdots}\ \Gamma, A \vdash \Delta \quad \overset{3}{\vdots}\ \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}}{\Gamma \vdash \Delta} \quad \rightsquigarrow \quad \dfrac{\overset{1}{\vdots}\ \Gamma \vdash A, \Delta \quad \overset{2}{\vdots}\ \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$$

On a ici un effacement de la preuve de $\Gamma, B \vdash \Delta$, ce qui peut avoir un certain coût en informatique. La modification n'est pas forcément strictement locale. On efface l'arbre de preuve 3, et on branche 1 et 2 sur la nouvelle interface : on obtient bien une preuve de même conclusion (de même sortie).

Pour le cas du et : $\dfrac{\dfrac{\overset{1}{\vdots}\ \Gamma \vdash A, \Delta \quad \overset{2}{\vdots}\ \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \quad \overset{3}{\vdots}\ \dfrac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}}{\Gamma \vdash \Delta}\, 4$

$$\rightsquigarrow \quad \dfrac{\dfrac{\overset{1}{\vdots}\ \Gamma \vdash A, \Delta \quad \overset{3}{\vdots}\ \Gamma, A, B \vdash \Delta}{\Gamma, B \vdash A, \Delta}}{\Gamma, B \vdash \Delta} \quad \overset{2}{\vdots}\ \Gamma \vdash B, \Delta}{\Gamma \vdash \Delta}$$

La procédure n'est ici pas déterministe : on a coupé 1 sur 3, et la résultante sur 2. On aurait pu couper 2 sur 3, et la résultante sur 1. On a dû choisir un ordre sur les coupures.

Une possibilité pour rester déterministe est d'accepter la règle de multi-coupure : $\dfrac{\Gamma, A_1, A_2 \vdash \Delta \quad \Gamma \vdash A_1, \Delta \quad \Gamma \vdash A_2, \Delta}{\Gamma \vdash \Delta.}$ (cut / multi-cut)

On peut faire un parallèle avec la substitution individuelle ($s[t/x]$) et la substitution parallèle ($s[\vec{t}/\vec{x}]$). Dans un cas la substitution est simultanée, pas dans l'autre. $s[t_1/x][t_2/y]$ peut être différent de $s[t_1, t_2/x, y]$, typiquement si y apparaît dans t_1.

On traite désormais toutes les autres règles, toutes les situations où la règle de coupure n'a pas pour principale au moins une formule qui vient d'être introduite. Il y a de nombreux cas, que l'on nomme cas commutatifs, car on fait commuter l'ordre des règles (on fait remonter la coupure).

$$\dfrac{\dfrac{\overset{1}{\vdots}\ \Gamma_1, B_1 \vdash A, \Delta \quad \overset{2}{\vdots}\ \Gamma_1, B_2 \vdash A, \Delta}{\Gamma_1, B_1 \vee B_2 \vdash A, \Delta} \quad \overset{3}{\vdots}\ \Gamma_1, B_1 \vee B_2, A \vdash \Delta}{\Gamma_1, B_1 \vee B_2 \vdash \Delta}$$

Que l'on transforme en dupliquant 3, ce qui peut également dupliquer le nombre de coupures. On s'assurera que la preuve termine en éliminant les coupures les plus hautes dans la preuve (non surmontée d'autres coupures).

$$\dfrac{\dfrac{\vdots 1}{\Gamma_1, B_1 \vdash A, \Delta} \quad \dfrac{\vdots 3}{\Gamma_1, B_1 \vee B_2, A \vdash \Delta}}{\Gamma_1, B_1, B_1 \vee B_2 \vdash \Delta} \qquad \dfrac{\dfrac{\vdots 2}{\Gamma_1, B_2 \vdash A, \Delta} \quad \dfrac{\vdots 3}{\Gamma_1, B_1 \vee B_2, A \vdash \Delta}}{\Gamma_1, B_2, B_1 \vee B_2 \vdash \Delta}$$

on accepte cette règle, qui se simplifie avec des affaiblissements

$$\dfrac{\Gamma_1, B_1 \vee B_2, B_1 \vee B_2 \vdash \Delta}{\Gamma_1, B_1 \vee B_2 \vdash \Delta}$$

Le procédé d'élimination des coupures peut entraîner l'introduction de nouvelles contractions.

On traite un autre cas :

$$\dfrac{\dfrac{\Gamma \vdash A, B, \Delta \qquad \Gamma, B \vdash A, \Delta}{\Gamma \vdash A, \Delta} \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$$

Faire commuter ces deux coupures nécessite de connaître la complexité de A et de B. La règle sera ainsi en fait : commencer par éliminer la coupure la plus haute (celle qui a pour principale B).

↳ Cet exemple illustre la discipline nécessaire :

Le procédé d'élimination des coupures est "plus simple" si l'on essaye systématiquement d'éliminer les coupures les plus éloignées de la racine.

Toutefois, si la coupure qui surmonte une autre coupure est une coupure-contraction, la procédure diffère :

$$(ctr.)\ \dfrac{\dfrac{\vdots 1}{\Gamma \vdash A, A, \Delta}}{\dfrac{\Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}} \quad \dfrac{\vdots 2}{\Gamma, A \vdash \Delta} \qquad \rightsquigarrow \qquad \dfrac{\dfrac{\vdots 1}{\Gamma \vdash A, A, \Delta} \quad \dfrac{\vdots 2}{\Gamma, A \vdash \Delta}}{\dfrac{\Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}} \quad \dfrac{\vdots 2}{\Gamma, A \vdash \Delta}$$

Ici aussi, il faut dupliquer une preuve. Ce cas est bien un cas particulier de ce qui a été présenté plus haut : les deux coupures n'ont pas la même (occurrence de) formule pour principale.

$$\dfrac{\dfrac{\Gamma \vdash \boxed{A}, \boxed{A}, \Delta \quad (A)\Gamma \vdash \Delta, A}{\Gamma \vdash \boxed{A}, \Delta}\ cut_1 \qquad \Gamma, \boxed{A} \vdash \Delta}{\Gamma \vdash \Delta}\ cut_2 \qquad \dfrac{}{(A) \vdash A}\ ax.$$

Schématiquement, une coupure logique est : $\dfrac{\Gamma \vdash A, \Delta}{\Gamma \vdash \Delta}\ \text{intro. d.} \quad \dfrac{\Gamma, A \vdash \Delta}{}\ \text{intro. g.}$

Dans tous les autres cas on fait commuter la coupure vers le haut, ce qui duplique parfois une branche de la preuve. Ce sont les règles commutatives.

Dans les deux cas il peut arriver qu'on efface une branche de la preuve.

On verra que si la coupure n'est surmontée ni à droite ni à gauche par une règle qui n'introduit pas la formule active de la coupure, on est confronté au non-déterminisme. Ce non-déterminisme rend la théorie de la démonstration incohérente : toute preuve en vaut une autre.

mardi 26 jan.

- Langage des termes pour LK
- Décrire l'élimination des coupures à travers ce langage
- "Incohérence" de l'élimination des coupures de LK.

Les preuves sont notées c, c', et par réécriture sur les preuves on a $c \mapsto c'$, qui se lit "la preuve que décrit c se transforme en la preuve que décrit c' par une étape d'élimination des coupures". On peut alors utiliser la théorie de la réécriture et ses outils.

L'incohérence se démontrera ainsi : quelque soient π_1 et π_2 prouvant le même séquent, il existe une preuve de ce même séquent π qui par élimination des coupures peut mener à π_1 ou à π_2 selon le choix qu'on effectue (à savoir dans quel ordre on élimine les coupures, et dans quel ordre on fait commuter les coupures ayant pour prémisses des règles n'introduisant pas la formule de coupure). On aura, en faisant le chemin inverse, $\pi_1 = \pi_2$, ce qui détruit la distinction entre les preuves qu'on avait pu introduire (// tous les programmes du même type sont égaux).

LK pol.

formules actives.

$$\dfrac{}{(A), A \vdash (A)}\ Ax. \qquad \text{et} \qquad \dfrac{}{A, (A) \vdash (A)}\ Ax.$$

sont deux preuves prouvablement égales, mais sont différentes.

On les distingue en nommant toutes les hypothèses (procédé syntaxique) et conclusions :

On note $\langle x \mid \alpha \rangle$ la preuve $\dfrac{}{\underset{x}{A}, \underset{y}{A} \vdash \underset{\alpha}{A}}\ Ax.$ et $\langle y \mid \alpha \rangle$ la preuve $\dfrac{}{\underset{x}{A}, \underset{y}{A} \vdash \underset{\alpha}{A}}\ Ax.$. En fait on va noter les indices à gauche des formules : $\overline{x : A, y : A \vdash \alpha : A}$ et $\overline{x : A, y : A \vdash \alpha : A}$. Ces indices se comportent comme des variables : ordinaires à gauche et de continuation à droite.

On décore tous les séquents $A_1, \dots, A_n \vdash B_1, \dots B_n$ de la façon suivante : $x_1 : A_1, \dots, x_n : A_n \vdash \alpha_1 : B_1, \dots, \alpha_n : B_n$.

On transmet ensuite toute cette décoration par un terme c (avec $x_1, \dots, x_n$ distincts et $\alpha_1, \dots, \alpha_n$ aussi).

On suppose donnée une collection $x_1, \dots, x_n, \dots, x, y$ de variables ordinaires et une collection de variables de continuation $\alpha_1, \dots, \alpha_n, \beta, \gamma$, etc. Ces deux collections sont disjointes.

Exemple : $\dfrac{\vdots\ \pi}{\Gamma \vdash A_1, \Delta}$ / $\Gamma \vdash A_1 \vee A_2, \Delta$

Déclaration de variable

On note $c : x_1 : A_1, \dots, x_n : A_n \vdash \alpha_1 : B_1, \dots$ et par fois $c : \Gamma \vdash \Delta$, les ensembles Γ et Δ étant typés.

Parallèle avec logique combinatoire. On avait vu que toute preuve π d'une formule A en logique minimale (présentation à la Hilbert) peut être transcrite en un terme $s : A$ avec $s ::= K^{A,B} \mid S^{A,B,C} \mid ss$.

Par récurrence on sait qu'on a

Terme $\quad \vdots\ \pi$ Type (formule)

$$\frac{c : \Gamma \vdash \alpha_1 : A_1, \Delta}{\langle \mathrm{inl}(\mu\alpha_1 . c) \mid \alpha \rangle : \Gamma \vdash \alpha : A_1 \vee A_2, \Delta.}$$

ces termes doivent aider un outil automatique, donc avoir toutes les informations

Rappel : α et α_1 sont ici les occurrences actives.

Jugements de typage : $c : (\Gamma \vdash \Delta)$

$r : (\Gamma \vdash A \mid \Delta)$ On précise la dernière formule introduite dans le séquent.

En fait on a : $$\frac{c : (\Gamma \vdash \alpha_1 : A_1, \Delta)}{\Gamma \vdash \mathrm{inl}(\mu\alpha_1 . c) : A_1 \vee A_2 \mid \Delta}$$

La notation introduite est plus homogène, mais on opte pour $\Gamma \vdash v : A \mid \Delta$.

En dérivant toutes les étapes entre $\Gamma \vdash A_1, \Delta$ et $\Gamma \vdash A_1 \vee A_2, \Delta$, on a $\langle \overbrace{\mathrm{inl}(\mu\alpha_1 . c)}^{\text{règle proprement dite.}} \mid \alpha \rangle$, le terme dans son ensemble étant la désactivation de $A_1 \vee A_2$.

$\underbrace{\mu\alpha_1 . c}_{\text{activation de } A_1}$

L'activation et la désactivation sont des formes de coercion (terme informatique). ou coercition?

Par exemple soient $3 : \mathrm{nat}$, $\pi : \mathrm{real}$, et deux additions ($+ : \mathrm{nat} \times \mathrm{nat} \to \mathrm{nat}$ et $+ : \mathrm{real} \times \mathrm{real} \to \mathrm{real}$). On doit transformer 3 d'entier en réel, ce qui peut nécessiter un réel travail informatique :

entier réel

$$\frac{\dfrac{3 : \mathrm{nat}}{3 : \mathrm{real}} \qquad \pi : \mathrm{real} \qquad + : \mathrm{real} \times \mathrm{real} \to \mathrm{real}}{3 + \pi : \mathrm{real}.}$$

coercion.

On se donne les règles :

Activation : $$\frac{c : (\Gamma \vdash \alpha : A, \Delta)}{(\Gamma \vdash \mu\alpha . c : A \mid \Delta)}$$

μ est un lieur, α passe de variable libre potentielle dans c à variable liée dans $\mu\alpha . c$.

Désactivation : $$\frac{\Gamma \vdash r : A \mid \Delta}{\langle r \mid \alpha \rangle : (\Gamma \vdash \alpha : A, \Delta)}$$

x étant une variable fraîche pour r, c'est-à-dire n'étant pas libre dans r.

On ne considère dans le système l'activation que si elle est suivie d'une règle. Notre système est pour le moment :

axiome désactivation

Commande : $c : \Gamma \vdash \Delta$ avec $c ::= \langle v \mid \alpha \rangle \mid \langle r \mid \alpha \rangle$

Expression : $\Gamma \vdash v : A \mid \Delta$ que l'on peut aussi noter $v : (\Gamma \vdash A \mid \Delta)$, avec $v ::= \mathrm{inl}(\mu\alpha . c)$

Ces transcriptions ne sont pas encore fidèles : un terme correctement typé, par contre, donnit compléterait la preuve.

(Pour le moment, $\mathrm{inl}(\mu\alpha . c)$ ne dit pas quelle est la formule introduite ajoutée par disjonction à droite de la formule qui type α.)

On se donne aussi : $$\frac{c : (\Gamma \vdash \alpha_2 : A_2, \Delta)}{\Gamma \vdash \mathrm{inr}(\mu\alpha_2 . c) : A_1 \vee A_2 \mid \Delta}$$

$$\frac{\overset{\alpha_1}{c_1 : (\Gamma \vdash A_1, \Delta)} \qquad \overset{\alpha_2}{c_2 : (\Gamma \vdash A_2, \Delta)}}{\langle \mu\alpha_1 . c_1, \mu\alpha_2 . c_2 \mid \alpha \rangle : \Gamma \vdash \alpha : A_1 \wedge A_2, \Delta.}$$

$$\frac{\overset{\alpha_1}{c_1 : (\Gamma \vdash \alpha_1 : A_1, \Delta)} \qquad \overset{\alpha_2}{c_2 : (\Gamma \vdash \alpha_2 : A_2, \Delta)}}{\dfrac{\Gamma \vdash \mu\alpha_1 . c_1 : A_1 \mid \Delta \qquad \Gamma \vdash \mu\alpha_2 . c_2 : A_2 \mid \Delta}{\Gamma \vdash (\mu\alpha_1 . c_1, \mu\alpha_2 . c_2) : A_1 \wedge A_2 \mid \Delta}}$$

$$\langle \mu\alpha_1 . c_1, \mu\alpha_2 . c_2 \mid \alpha \rangle : \Gamma \vdash \underset{\alpha}{A_1 \wedge A_2}, \Delta.$$

$$\frac{c : (\Gamma, x : A \vdash \Delta)}{\Gamma \vdash (\tilde{\mu} x . c)^{\cdot} : \neg A \mid \Delta}$$

$$\langle (\tilde{\mu} x . c)^{\cdot} \mid \alpha \rangle : \Gamma \vdash \alpha : \neg A, \Delta$$

Activation à gauche : $$\frac{c : (\Gamma, x : A \vdash \Delta)}{\Gamma \mid \tilde{\mu} x . c : A \vdash \Delta}$$

Désactivation à gauche : $$\frac{\Gamma \mid e : A \vdash \Delta}{\langle x \mid e \rangle : (\Gamma, x : A \vdash \Delta)}$$

Règles à gauche $$\frac{c : (\Gamma, x_1 : A_1, x_2 : A_2 \vdash \Delta)}{\tilde{\mu}(x_1, x_2) . c : (\Gamma \mid \overset{\alpha}{A_1 \wedge A_2} \vdash \Delta)}$$

pour α frais dans A.

$$\langle x \mid \tilde{\mu}(x_1, x_2) . c \rangle : (\Gamma, x : A_1 \wedge A_2 \vdash \Delta)$$

G. Curien & Herbelin
The Duality of Computation

$$\frac{c_1 : (\Gamma, x_1 : A_1 \vdash \Delta) \qquad c_2 : (\Gamma, x_2 : A_2 \vdash \Delta)}{\tilde{\mu}\left(\mathrm{inl}(x_1) \to c_1, \mathrm{inl}(x_2) \to c_2\right) : (\Gamma \mid A_1 \vee A_2 \vdash \Delta)}$$

$$\frac{c : (\Gamma \vdash \alpha : A \mid \Delta)}{\tilde{\mu}\alpha^{\cdot} . c : (\Gamma \mid \neg A \vdash \Delta)}$$

$c ::= \langle x \mid \alpha \rangle \mid \langle v \mid \alpha \rangle \mid \langle x \mid e \rangle \mid \langle \mu\alpha . c \mid \tilde{\mu} x . c \rangle$

$v ::= \mathrm{inl}(\mu\alpha . c) \mid \mathrm{inr}(\mu\alpha . c) \mid (\mu\alpha . c, \mu\alpha . c) \mid (\tilde{\mu} x . c)^{\cdot}$

$e ::= \tilde{\mu}(x_1, x_2) . c \mid \tilde{\mu}\left(\mathrm{inl}(x_1) \to c_1, \mathrm{inr}(x_2) \to c_2\right) \mid \tilde{\mu}\alpha^{\cdot} . c$

en fait moralement $c ::= \langle v \mid e \rangle$.

Reste la coupure / contraction :

$$\frac{c_1 : (\Gamma \vdash \alpha : A, \Delta) \qquad c_2 : (\Gamma, x : A \vdash \Delta)}{\langle \mu\alpha . c_1 \mid \tilde{\mu} x . c_2 \rangle : (\Gamma \vdash \Delta)} \qquad \frac{c_1 : (\Gamma \vdash \alpha : A, \Delta) \quad c_2 : (\Gamma, x : A \vdash \Delta)}{(\Gamma \vdash \mu\alpha . c_1 : A \mid \Delta) \quad (\Gamma \mid \tilde{\mu} x . c_2 : A \vdash \Delta)}$$

$$\langle \mu\alpha . c_1 \mid \tilde{\mu} x . c_2 \rangle : (\Gamma \vdash \Delta)$$

(1) Établir le lien précis entre les termes de preuves et les arbres de preuve.

Un terme de cette syntaxe ne correspond à un arbre de preuve que s'il est bien typé. Un exemple de règle de typage : $\dfrac{c : (\Gamma, x_1 : A_1, x_2 : A_2 \vdash \Delta)}{\tilde{\mu}(x_1, x_2) . c : (\Gamma \mid A_1 \wedge A_2 \vdash \Delta)}$, et toutes les règles que l'on vient de donner.

Si $c : x_1 : A_1, \dots, x_m : A_m \vdash \alpha_1 : B_1, \dots, \alpha_n : B_n$, alors on peut effacer dans la preuve de bon typage (le jugement) de c : toutes les informations x, α, c, v, e, ainsi que les distinctions entre les trois formes de séquents. Il faut également éliminer les étapes de la forme $\dfrac{\Gamma \vdash A \mid \Delta}{\Gamma \vdash A, \Delta}$ (les activations ne sont elles jamais explicites : elles sont toujours associées à une règle de preuve) et $\dfrac{\Gamma \mid A \vdash \Delta}{\Gamma, A \vdash \Delta}$. On obtient alors un arbre de preuve de la forme $A_1, \dots, A_m \vdash B_1, \dots, B_n$.

Dans l'autre sens : tout arbre de preuve π d'un séquent $A_1, \dots, A_m \vdash B_1, \dots, B_n$ peut être décoré, c'est à dire transformé en jugement de $c : x_1 : A_1, \dots, x_m : A_m \vdash \alpha_1 : B_1, \dots, \alpha_n : B_n$. Il y a de l'arbitraire dans le choix du nom des variables. Mais ce processus de décoration est unique dans le sens suivant : si $c_1 : x_1 : A_1, \dots, x_m : A_m \vdash \alpha_1 : B_1, \dots, \alpha_n : B_n$ et $c_2 : x'_1 : A_1, \dots, x'_m : A_m \vdash \alpha'_1 : B_1, \dots, \alpha'_n : B_n$ décorent la même preuve π, alors c_1 et c_2 ne diffèrent que par le nom des variables : $c_2 = c_1[x'_1/x_1, \dots, \alpha'_n/\alpha_n]$

c décore π ssi π = effacement (c).

Lemme (réciproque du précédent) Si $c : (x : A, \Gamma \vdash \Delta)$ est correctement typé, alors $c[y/x]$ pour y frais pour c et des variables déclarées dans Γ, est une preuve de $y : A, \Gamma \vdash \Delta$.

Lemme (version collective) Si $c : (x_1 : A_1, \dots, x_m : A_m \vdash \alpha_1 : B_1, \dots, \alpha_n : B_n)$ et si $x'_1, \dots, x'_m$ sont distincts deux à deux, et si $\alpha'_1, \dots, \alpha'_n$ sont distincts deux à deux, alors $c[x'_1/x_1, \dots, x'_n/x_m, \alpha'_1/\alpha_1, \dots, \alpha'_n/\alpha_n]$ est une preuve de $x'_1 : A_1, \dots \vdash \dots, \alpha'_n : B_n$.

On utilise le lemme pour montrer l'existence de c décorant π pour tout π. Par récurrence, le lemme servira pour les règles binaires : elles sont décorables par induction, et on les nommera ou les unifie, ce qui permettra d'appliquer la règle ("harmonisation").

mer. 27 jan.

$$\begin{aligned} c &::= \langle x \mid \alpha \rangle \mid \langle v \mid \alpha \rangle \mid \langle x \mid e \rangle \mid \langle \mu\alpha . c \mid \tilde{\mu} x . c \rangle \\ v &::= (\tilde{\mu} x . c)^\star \mid (\mu\alpha . c, \mu\alpha . c) \mid \mathrm{inl}(\mu\alpha . c) \mid \mathrm{inr}(\mu\alpha . c) \\ e &::= \tilde{\mu}\alpha^\star . c \mid \tilde{\mu}(x, x) . c \mid \tilde{\mu}(\mathrm{inl}(x) \to c, \mathrm{inr}(x) \to c) \end{aligned}$$

} Langage de preuve pour LK.

Proposition : - Soit π un arbre de preuve de LK d'un séquent $\vec{A} \vdash \vec{B}$

- Soit une décoration de ce séquent : $\vec{x} : \vec{A} \vdash \vec{\alpha} : \vec{B}$

Alors il existe un unique terme $c : (\vec{x} : \vec{A} \vdash \vec{\alpha} : \vec{B})$ tel que effacement $(c) = \pi$.

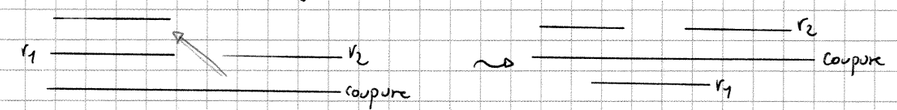
$c : (\vec{x} : \vec{A} \vdash \vec{\alpha} : \vec{B})$ est un arbre de preuve décoré.

Exemple : Un terme $\langle x \mid e \rangle : (\Gamma, x : A \vdash \Delta)$ est forcément conclusion d'une instance de déactivation ayant pour prémisse $e : (\Gamma \mid A \vdash \Delta)$. ou $(\Gamma \mid e : A \vdash \Delta)$

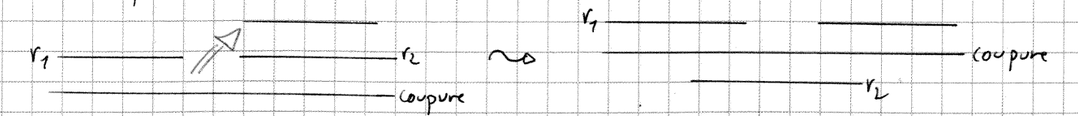
Rappel de la structure des règles d'élimination des coupures :

$$\dfrac{\dfrac{}{\Gamma \vdash A, \Delta}\, r_1 \qquad \dfrac{}{\Gamma, A \vdash \Delta}\, r_2}{\Gamma \vdash \Delta}\ \text{coupure}$$

- Si r_1 est une règle introduisant A et r_2 est une règle introduisant A → étape d'élimination logique, 3 cas possibles (1 par connecteur).
- Si r_1 introduit une formule de Γ ou une formule de Δ → étape de coupure commutative (3+3 cas)



Ici, l'élimination des coupures fait remonter la preuve de droite dans la preuve de gauche. Si c'est r_2 qui introduit une formule autre que A, l'élimination des coupures se fait dans l'autre sens : la preuve de gauche remonte dans la preuve de droite.



• Si r_1 est un axiome introduisant A, on ne fait rien, car c'est une instance de contraction :

$$\dfrac{\dfrac{}{\Gamma, A \vdash \Delta, A}\,Ax \qquad \Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

• Si r_1 est un axiome introduisant une autre formule, par exemple B, on a :

$$\dfrac{\dfrac{}{\Gamma, B \vdash B, A, \Delta}\,Ax \qquad \begin{matrix}\vdots \pi \\ \Gamma, B, A \vdash B, \Delta\end{matrix}}{\Gamma, B \vdash B, \Delta}\,\text{coupure} \quad \rightsquigarrow \quad \dfrac{}{\Gamma, B \vdash B, \Delta}\,Ax.$$

C'est un effacement : la preuve π disparaît. On formule ce cas avec le langage de termes :

$$\dfrac{\dfrac{}{\langle y \mid \beta \rangle : (\Gamma, y : B \vdash \alpha : A, \beta : B, \Delta)}\,Ax. \qquad c : (\Gamma, y : B, x : A \vdash \beta : B, \Delta)}{\langle \mu\alpha . \langle y \mid \beta \rangle \mid \tilde{\mu} x . c \rangle : (\Gamma, y : B \vdash \beta : B, \Delta)}\,\text{coupure.}$$

Ce terme se réécrit en : $\langle y \mid \beta \rangle : (\Gamma, y : B \vdash \beta : B, \Delta)$. On voit donc qu'un terme peut avoir deux types. Par contre, un terme et son type désignent univoquement un arbre de preuve.

• Si r_1 est une instance de cut/contraction

$$r_1\,\dfrac{\dfrac{\Gamma \vdash A, B, \Delta \qquad \Gamma, B \vdash A, \Delta}{\Gamma \vdash A, \Delta}}{\Gamma \vdash \Delta}\,wk_2 \qquad \Gamma, A \vdash \Delta$$

Si r_1 est une instance de coupure, on réduit d'abord cette coupure avant de regarder l'instance de coupure wk_2.

Si r_1 est une contraction, on fait une duplication, on le formule avec le langage de termes :

$$\dfrac{\dfrac{c : (\Gamma \vdash \alpha_1 : A, \alpha_2 : A, \Delta)}{\langle \mu\alpha_2 . c \mid \tilde{\mu} x . \langle x \mid \alpha_1 \rangle \rangle : (\Gamma \vdash \alpha_1 : A, \Delta)}\,\text{contraction} \qquad d : (\Gamma, y : A \vdash \Delta)}{\langle \mu\alpha_1 . \langle \mu\alpha_2 . c \mid \tilde{\mu} x . \langle x \mid \alpha_1 \rangle \rangle \mid \tilde{\mu} y . d \rangle : (\Gamma \vdash \Delta)}\,\text{coupure}$$

$\dfrac{}{\langle x \mid \alpha_1 \rangle : (\Gamma, x : A \vdash \alpha_1 : A, \Delta)}\,ax.$

Traison !

On effectue un pas d'élimination des coupures :

on fait d'abord apparaître à droite introduisant $\alpha_2 : A$ avant la coupure

duplication de cette preuve.

$$\dfrac{\dfrac{c : (\Gamma \vdash \alpha_1 : A, \alpha_2 : A, \Delta) \qquad d : (\Gamma, y : A \vdash \Delta)}{\langle \mu\alpha_1 . c \mid \tilde{\mu} y . d \rangle : (\Gamma \vdash \alpha_2 : A, \Delta)}\,wk \qquad d : (\Gamma, y : A \vdash \Delta)}{\langle \mu\alpha_2 . \langle \mu\alpha_1 . c \mid \tilde{\mu} y . d \rangle \mid \tilde{\mu} y . d \rangle : (\Gamma \vdash \Delta)}\,\text{coupure}$$

Théorème : Le processus d'élimination des coupures termine avec succès (jusqu'à ce qu'il n'y ait plus de coupures).

- **Sens faible** : il existe une stratégie d'élimination des coupures qui termine
- **Sens fort** : quelque soit la stratégie adoptée, le processus termine.

On a pour ce langage les deux, mais on ne le démontre pas.

En plus des règles d'élimination des coupures, on peut se donner la règle suivante : variable libre

$$\langle \mu\alpha_2 . c \mid \tilde{\mu} x . \langle x \mid \alpha_1 \rangle \rangle \longrightarrow c \qquad \text{à condition que } \alpha_2 \notin \mathrm{FV}(c).$$

Du point de vue de la transformation de preuves :

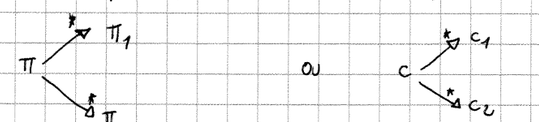
Puisque $\alpha_2 \notin \mathrm{FV}(c)$, α_2 ne joue aucun rôle dans c, et on peut l'introduire par aff.

$$\dfrac{\dfrac{c : (\Gamma \vdash \alpha_1 : A, \Delta)}{c : (\Gamma \vdash \alpha_1 : A, \alpha_2 : A, \Delta)}\,\text{aff.} \qquad \dfrac{}{\langle x \mid \alpha_1 \rangle : (\Gamma, x : A \vdash \alpha_1 : A, \Delta)}\,Ax}{\langle \mu\alpha_2 . c \mid \tilde{\mu} x . \langle x \mid \alpha_1 \rangle \rangle : (\Gamma \vdash \alpha_1 : A, \Delta)}\,\text{coupure}$$

En fait cette transformation revient à : $\dfrac{\dfrac{\vdots \pi}{\Gamma \vdash A, \Delta}}{\dfrac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}\,ctr.}\,\text{aff.} \quad \rightsquigarrow \quad \begin{matrix}\vdots \pi \\ \Gamma \vdash A, \Delta\end{matrix}$.

C'est une simplification qui n'est pas une étape de réduction des coupures. On la nomme (weak/cont.)

On se considère muni de toutes les règles d'élimination des coupures et de (weak/cont). On peut montrer le résultat annoncé plus tôt : si π_1 et π_2 prouvent le même séquent $\Gamma \vdash \Delta$, il existe une preuve π de $\Gamma \vdash \Delta$ à partir de laquelle on peut, par élimination des coupures, rejoindre au choix π_1 ou π_2. On le note formellement :



Preuve :

$$\pi \begin{cases} \vdots \pi_1 \\ c_1 : \Gamma \vdash \Delta \end{cases}\,\text{aff.}\,d \qquad \begin{cases} \vdots \pi_2 \\ c_2 : \Gamma \vdash \Delta \end{cases}\,\text{aff.}\,g.$$

$$\dfrac{c_1 : (\Gamma \vdash \alpha : A, \Delta) \qquad c_2 : (\Gamma, x : A \vdash \Delta)}{\langle \mu\alpha . c_1 \mid \tilde{\mu} x . c_2 \rangle : (\Gamma \vdash \Delta)}\,wk$$

Si on fait remonter systématiquement π_2 sur π_1, on finit par arriver sur un axiome dont A n'est pas principale, ce qui se réduit en effaçant π_2. Modulo la règle de (weak/cont), π_1 n'a pas changé, et π s'est réduite en π_1. Faire remonter obstinément π_1 sur π_2 réduit π en π_2.

Note : C'est la patte mitique de Lafont.

Lemme : Soient $c:(\Gamma \vdash \Delta)$ et $d:(\Gamma', x:B \vdash \Delta')$, avec $\alpha \notin FV(c)$, alors $\langle \mu\alpha.c \mid \tilde{\mu}x.d \rangle \rightsquigarrow_D^* c$.

On a : $\dfrac{\dfrac{c:(\Gamma \vdash \Delta)}{c:(\Gamma \vdash \alpha:B, \Delta)}\,aff. \qquad d:(\Gamma', x:B \vdash \Delta')}{\Gamma'' \vdash \Delta''}$

avec $\Gamma(A)$ = nombre d'occurrences de A dans Γ, et
$\Gamma''(A) = \sup(\Gamma(A), \Gamma'(A))$ et
$\Delta''(A) = \sup(\Delta(A), \Delta'(A))$ pour tout A.

Ce lemme et le lemme symétrique démontrent le théorème.

Preuve : Par récurrence sur la taille de c.

On ne traite qu'un cas, si $c = \langle (\tilde{\mu}x.e)^\bullet \mid \alpha \rangle$:

$$\dfrac{\dfrac{\dfrac{\pi_1 :}{e:(\Gamma, x:A \vdash \beta:B, \Delta)}\,aff.}{\langle (\tilde{\mu}x.e)^\bullet \mid \alpha \rangle : \Gamma \vdash \beta:B, \alpha:\neg A, \Delta}\,id \qquad d:(\Gamma', y:B \vdash \Delta)\,(\pi_2 :)}{\langle \mu\beta.\langle (\tilde{\mu}x.e)^\bullet \mid \alpha \rangle \mid \tilde{\mu}y.d \rangle : \Gamma'' \vdash \Delta'', \alpha:\neg A}\,wk$$

On le réduit en :

$$\dfrac{e:(\Gamma, x:A \vdash \beta:B, \Delta) \quad (\pi_1) \qquad d:(\Gamma', y:B \vdash \Delta') \quad (\pi_2)}{\langle \mu\beta.e \mid \tilde{\mu}y.d \rangle : (\Gamma'', x:A \vdash \Delta'')}\,wk$$
$$\dfrac{}{\langle (\tilde{\mu}x.\langle \mu\beta.e \mid \tilde{\mu}y.d \rangle)^\bullet . \alpha \rangle : (\Gamma'' \vdash \alpha:\neg A, \Delta'')}\,id$$

Ce qui se réduit, par hypothèse d'induction, en $\dfrac{\Gamma'', A \vdash \Delta'' \quad (\pi_1)}{\Gamma'' \vdash \neg A, \Delta''}$, qui type le terme $c = \langle (\tilde{\mu}x.e)^\bullet \mid \alpha \rangle$

On reformule l'exemple avec les contextes exacts :

$$\dfrac{\dfrac{\dfrac{\Gamma, A \vdash \Delta}{\Gamma, A \vdash B, \Delta}}{\Gamma \vdash \neg A, B, \Delta} \qquad \Gamma, B \vdash \neg A, \Delta}{\Gamma \vdash \neg A, \Delta} \quad \rightsquigarrow$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma, A \vdash \Delta \quad (c_1)}{\Gamma, A \vdash B, \Delta}\,aff.}{\Gamma, B \vdash \neg A, \Delta}\,wk}{\Gamma, A \vdash \neg A, \Delta}\,id}{\Gamma \vdash \neg A, \neg A, \Delta}\,ctr.}{\Gamma \vdash \neg A, \Delta}$$

Cet ensemble se réduit par récurrence en c_1.

On a alors :

$$\dfrac{\dfrac{\dfrac{\Gamma, A \vdash \Delta \quad (c_1)}{\Gamma, A \vdash \neg A, \Delta}}{\Gamma \vdash \neg A, \neg A, \Delta}}{\Gamma \vdash \neg A, \Delta}$$

qui se réduit en c_1 par (weak/cont) : on n'a pas modifié fondamentalement c_1, on n'a introduit que des contractions sur des formules qui étaient affaiblies.

La paire critique de Lafont apparaît en annexe du Proofs and Types.

Simplification de la syntaxe :

On ne fait plus ensemble activation et application de la règle : on introduit donc $\mu\alpha.c$ et $\tilde{\mu}x.c$, et on peut simplifier les autres termes.

Commande : $\Gamma \vdash \Delta$
V = Expression : $\Gamma \vdash A \mid \Delta$
E = Contexte : $\Gamma \mid A \vdash \Delta$

On a alors : $c ::= \langle v \mid e \rangle$

$$v ::= x \mid e^\bullet \mid (v, v) \mid inl(v) \mid inr(v) \mid \mu\alpha.c$$
$$e ::= \alpha \mid \tilde{\mu}\alpha^\bullet.c \mid \tilde{\mu}(x,x).c \mid \tilde{\mu}(inl(x) \to c, inr(x) \to c) \mid \tilde{\mu}x.c$$

On a beaucoup plus de termes, mais des règles de typage plus simples.

Coupure : $\dfrac{v:(\Gamma \vdash A \mid \Delta) \qquad e:(\Gamma \mid A \vdash \Delta)}{\langle v \mid e \rangle : (\Gamma \vdash \Delta)}$

$\langle x \mid \alpha \rangle : (\Gamma, x:A \vdash \alpha:A, \Delta)$
$\mu\alpha.\langle x \mid \alpha \rangle : (\Gamma, x:A \vdash A \mid \Delta)$ équiv
$x : (\Gamma, x:A \vdash A \mid \Delta)$

Axiomes : $\dfrac{\text{aussi } (\Gamma, x:A \vdash x:A \mid \Delta)}{x : (\Gamma, x:A \vdash A \mid \Delta)}$ $\qquad \dfrac{}{\alpha : (\Gamma \mid A \vdash \alpha:A, \Delta)}$ désactive

$$\dfrac{\text{active à droite} \quad \text{active à gauche}}{\langle x \mid \alpha \rangle : (\Gamma, x:A \vdash \alpha:A, \Delta)}$$

Activations : $\dfrac{c:(\Gamma \vdash \alpha:A, \Delta)}{\mu\alpha.c : (\Gamma \vdash A \mid \Delta)} \qquad \dfrac{c:(\Gamma, x:A \vdash \Delta)}{\tilde{\mu}x.c : (\Gamma \mid A \vdash \Delta)}$

En faisant on a : $\tilde{\mu}x.\langle x \mid \alpha \rangle = \alpha$ qui correspond à la règle η en λ-calcul : $\lambda x\,\Pi\,x =_\eta x$, avec $x \notin FV(\Pi)$.

Exemple de règle logique : $\dfrac{v\,(\Gamma \vdash A_1 \mid \Delta)}{inl(v) : (\Gamma \vdash A_1 \vee A_2 \mid \Delta)}$

Le gain de finesse dans le contrôle qui permet la confluence. mardi 2 fév.
Tout terme écrit dans cette syntaxe décore un arbre qui n'est plus tout à fait une preuve de LK à chaque fois. Par contre, toute preuve de LK se décore dans cette syntaxe.

Expressions : $(\Gamma \mid A \vdash \Delta)$ $\approx$ programme

Contextes : $(\Gamma \vdash A \mid \Delta)$ $\approx$ bibliothèque

Commandes : $(\Gamma \vdash \Delta)$

Dans l'autre syntaxe, dans la preuve de correspondance bi-univoque entre termes et preuves, on a besoin de deux conditions supplémentaires : $\langle v \mid \alpha \rangle \quad \alpha \notin FV(v)$

$\langle x \mid e \rangle \quad x \notin FV(e)$

$$\frac{c : (\Gamma, y : A \vdash \Delta, \alpha : B)}{\Gamma \vdash (\tilde{\mu} y . c)^{\cdot} : \neg A \mid \Delta, \alpha : B)} \qquad d : (\Gamma, x : B \vdash \neg A, \Delta)$$
$$\frac{\langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle : (\Gamma \vdash \beta : \neg A, \Delta, \alpha : B)}{\langle \mu \alpha . \langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle \mid \tilde{\mu} x . d \rangle : (\Gamma \vdash \neg A, \Delta)}$$

Un cas d'élimination des coupures :

Cas de commutation $(\neg D)$ commutation vers la gauche.

introduction de la négation à droite

$$\frac{c : (\Gamma, y : A \vdash \Delta, \alpha : B) \qquad d : (\Gamma, x : B \vdash \beta' : \neg A, \Delta)}{\langle \mu \alpha . c \mid \tilde{\mu} x . d \rangle : (\Gamma, y : A \vdash \Delta, \beta' : \neg A)}$$
$$\frac{\langle (\tilde{\mu} y . \langle \mu \alpha . c \mid \tilde{\mu} x . d \rangle)^{\cdot} \mid \beta' \rangle : (\Gamma \vdash \beta : \neg A, \beta' : \neg A, \Delta)}{\langle \mu \beta' \langle (\tilde{\mu} y . \langle \mu \alpha . c \mid \tilde{\mu} x . d \rangle)^{\cdot} \mid \beta' \rangle \mid \tilde{\mu} y . \langle y \mid \beta \rangle \rangle : (\Gamma \vdash \beta : \neg A, \Delta)}$$ contradiction

$$\frac{\dfrac{\langle \mu \alpha . \langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle \mid \tilde{\mu} x . d \rangle}{\text{intro. de la neg.}}}{\text{coupure}} \rightsquigarrow \frac{\langle \mu \beta' . \langle (\tilde{\mu} y . \dfrac{\langle \mu \alpha . c \mid \tilde{\mu} x . d \rangle}{\text{coupure}})^{\cdot} \mid \beta' \rangle \mid \tilde{\mu} y . \langle y \mid \beta \rangle \rangle}{\text{intro. de la négation.}}$$

On se donne une nouvelle règle : $\langle \mu \alpha . c \mid e \rangle \rightarrow c[e/\alpha]$, qui est une substitution explicite (cf. Expliciter substitution Abadi, Cardelli, Curien, Lévy, 1992). Ce n'est pas le résultat de la substitution, mais l'indication de la commande de la substitution (un déclencheur).

$\tilde{\mu} x . d$ est présent dans les deux termes et ne change pas (c'est la preuve à droite), on le note désormais e. On reprend la transformation de façon plus fine :

$\langle \mu \alpha . \langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle \mid e \rangle$

$\langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle [e/\alpha]$ car $\langle \mu \alpha . c \mid e \rangle \rightarrow c[e/\alpha]$

$\langle (\tilde{\mu} y . c)^{\cdot} [e/\alpha] \mid \beta [e/\alpha] \rangle$ par distribution : $\langle v \mid e \rangle [e'/\alpha] \rightarrow \langle v [e'/\alpha] \mid e [e'/\alpha] \rangle$.

$\langle (\tilde{\mu} y . (c[e/\alpha]))^{\cdot} \mid \beta \rangle$ car $\beta \neq \alpha$! car $(\tilde{\mu} y . c)[e/\alpha] \rightarrow \tilde{\mu} y . (c[e/\alpha])$ si $y \notin FV(e)$.

De même : $\langle \mu \beta' . \langle \tilde{\mu} y . \langle \mu \alpha . c \mid e \rangle^{\cdot} \mid \beta' \rangle \mid \tilde{\mu} y . \langle y \mid \beta \rangle \rangle$ car $\langle v \mid \tilde{\mu} x . e \rangle \rightarrow e [v/x]$

$\langle y \mid \beta \rangle [\langle \mu \beta' . \langle \tilde{\mu} y . \langle \mu \alpha . c \mid e \rangle^{\cdot} \mid \beta' \rangle / y]$

$\langle \mu \beta' . \langle \tilde{\mu} y . \langle \mu \alpha . c \mid e \rangle^{\cdot} \mid \beta' \rangle \mid \beta \rangle$

$\langle (\tilde{\mu} y . \langle \mu \alpha . c \mid e \rangle)^{\cdot} \mid \beta' \rangle [\beta / \beta']$, le β' a été choisi frais, du dernier instant.

$\langle (\tilde{\mu} y . \langle \mu \alpha . c \mid e \rangle)^{\cdot} \mid \beta \rangle \rightarrow \langle (\tilde{\mu} y . (c[e/\alpha]))^{\cdot} \mid \beta \rangle$

Pour toutes les règles commutatives d'élimination des coupures : $t_1 \longrightarrow t_2$ → : ancienne syntaxe

t_3 ⇢ : syntaxe avec simplifiée (avec déclencheur).

$\mu \alpha . \langle (\tilde{\mu} y . c)^{\cdot} \mid \beta \rangle$

trace syntaxique du fait que la formule de coupure n'était pas principale dans la règle d'avant : la formule oubliée a varié.

Autres cas d'élimination des coupures :

* Règle logique pour la négation :

$$\frac{c : (\Gamma, x : A \vdash \Delta)}{\langle (\tilde{\mu} x . c)^{\cdot} \mid \alpha \rangle : (\Gamma \vdash \alpha : \neg A, \Delta)} \qquad \frac{d : (\Gamma \vdash \alpha : A, \Delta)}{\langle y \mid \tilde{\mu} \alpha^{\cdot} . d \rangle : (\Gamma, y : \neg A \vdash \Delta)}$$
$$\langle \mu \alpha . \langle (\tilde{\mu} x . c)^{\cdot} \mid \alpha \rangle \mid \tilde{\mu} y . \langle y \mid \tilde{\mu} \alpha^{\cdot} . d \rangle \rangle : \Gamma \vdash \Delta$$

$\langle \mu \alpha . \langle (\tilde{\mu} x . c)^{\cdot} \mid \alpha \rangle \mid \tilde{\mu} y . \langle y \mid \tilde{\mu} \alpha^{\cdot} . d \rangle \rangle$ ⟶ ne désactive/active pas

$\langle \mu \alpha . \langle (\tilde{\mu} x . c)^{\cdot} \mid \alpha \rangle \mid \tilde{\mu} \alpha^{\cdot} . d \rangle$

$\langle (\tilde{\mu} x . c)^{\cdot} \mid \tilde{\mu} \alpha^{\cdot} . d \rangle$ qu'on écrit $\langle e^{\cdot} \mid \tilde{\mu} \alpha^{\cdot} . d \rangle$ ⟶

$$\frac{c : (\Gamma, x : A \vdash \Delta) \qquad d : (\Gamma \vdash \alpha : A, \Delta)}{(\Gamma \vdash (\tilde{\mu} x . c)^{\cdot} : A \mid \Delta) \qquad (\Gamma \mid \mu \tilde{\alpha}^{\cdot} . d : \neg A \vdash \Delta)}$$
$$\langle (\tilde{\mu} x . c)^{\cdot} \mid \mu \tilde{\alpha}^{\cdot} . d \rangle : (\Gamma \vdash \Delta)$$

$\langle \mu \alpha . d \mid e \rangle$

$d[e/\alpha]$, on ajoute la règle qui permet de passer de $\langle e^{\cdot} \mid \tilde{\mu} \alpha^{\cdot} . d \rangle$ à $d[e/\alpha]$.

$$\frac{d : (\Gamma \vdash \alpha : A, \Delta) \qquad \tilde{\mu} x . c}{(\Gamma \vdash \mu \alpha . d : A \mid \Delta) \qquad (\Gamma \mid e : A \vdash)}$$
$$\langle \mu \alpha . d \mid e \rangle : (\Gamma \vdash \Delta)$$

$d[e/\alpha]$: on coupe directement A dans d à l'aide de e :

* Règle logique pour la conjonction :

$\langle (v_1, v_2) \mid \tilde{\mu} (x_1, x_2) . c \rangle \rightarrow c[v_1/x_1, v_2/x_2]$

On complique encore la syntaxe : on peut recourir aux substitutions simultanées.

* Règles logiques pour la disjonction :

$\langle inl(v_1) \mid \tilde{\mu} (inl(x_1) . c_1 \mid inr(x_2) . c_2) \rangle \rightarrow c_1 [v_1 / x_1]$.

$\langle inr(v_2) \mid \tilde{\mu} (inl(x_1) . c_1 \mid inr(x_2) . c_2) \rangle \rightarrow c_2 [v_2 / x_2]$

On ajoute donc à c, $c[\sigma]$, à v, $v[\sigma]$ et à e, $e[\sigma]$, avec $\sigma := e/\alpha \mid v/x \mid (\sigma, v/x) \mid (\sigma, e/\alpha)$

Les règles de typage

Le terme $c[e/\alpha]$ est une coupure : $\dfrac{c : (\Gamma \vdash \alpha : A, \Delta) \qquad e : (\Gamma \mid A \vdash \Delta)}{c[e/\alpha] : (\Gamma \vdash \Delta)}$.

Donc en terme d'arbres de preuve, réduire $\langle(\mu\alpha.c)|e\rangle$ en $c[e/\alpha]$ ne change rien, on reste sur le même arbre de preuve. En fait on fait commuter activation et coupure, ce qui est invisible dans un arbre de preuve.

$$\dfrac{\dfrac{c:(\Gamma\vdash\Delta,A)}{\mu\alpha.c:(\Gamma\vdash\Delta\,|\,\alpha:A)}\text{ activation} \qquad e:(\Gamma\,|\,A\vdash\Delta)}{\langle(\mu\alpha.c)|e\rangle:(\Gamma\vdash\Delta)} \rightsquigarrow \dfrac{c:(\Gamma\vdash A,\Delta) \qquad e:(\Gamma|A\vdash\Delta)}{c[e/\alpha]:(\Gamma\vdash\Delta)}$$

Règle de multi-coupure:

$$\dfrac{v:(\Gamma,x_1:A_1,\dots x_m:A_m\vdash B\,|\,\alpha_1:B_1,\dots\alpha_n:B_n\vdash\Delta) \qquad \overbrace{\Gamma\vdash\vec{v_i}:\vec{A_i}\,|\,\Delta}^{\text{m jugements}} \qquad \overbrace{\Gamma\,|\vec{e_j}:\vec{B_j}\vdash\Delta}^{\text{n jugements}}}{v[\vec{v}/\vec{x},\vec{e}/\vec{\alpha}]:(\Gamma\vdash B|\Delta)}$$

Commentaire:

$\langle\mu\alpha.\langle v|B\rangle|e\rangle\longrightarrow\langle v[e/\alpha]|B\rangle$, que l'on écrit $\langle v'|\beta\rangle$.

avec $\alpha\neq\beta$, $\beta\notin FV(v)$, $\beta\in FV(e)$ et $\alpha\in FV(v)$. Donc $\beta\in FV(v[e/\alpha])$.

Il y a ici, implicite dans la syntaxe, une contraction

$$\dfrac{\Gamma\vdash v':B\,|\,\beta:B,\Delta \qquad \overline{|\beta:B\vdash\beta:B}}{\langle v'|B\rangle:\Gamma\vdash\beta:B,\Delta}\text{ ctr.}$$

Trivial du point de vue récursif, il y a un jeu au niveau logique par l'introduction d'une contraction.

La paire de Lafont:

$$\dfrac{\dfrac{c:(\Gamma\vdash\Delta)}{c:(\Gamma\vdash\alpha:A,\Delta)} \qquad \dfrac{d:(\Gamma\vdash\Delta)}{d:(\Gamma,x:A\vdash\Delta)}}{\langle\mu\alpha.c|\tilde{\mu}x.d\rangle:(\Gamma\vdash\Delta)}\quad\text{pour }\alpha\notin FV(c)\text{ et }x\notin FV(d)$$

$\langle\mu\alpha.c|\tilde{\mu}x.d\rangle$ se réduit en $c[\tilde{\mu}x.d/\alpha]\longrightarrow c$ ou en $d[\mu\alpha.c/x]\longrightarrow d$

On restreint notre syntaxe pour éliminer ce cas, et ne plus pouvoir réduire $\langle\mu\alpha.c|\tilde{\mu}x.d\rangle$ en d.

On conserve donc la règle $\langle\mu\alpha.c|e\rangle\longrightarrow c[e/\alpha]$, règle "call by name".

C.B.N. : $(\lambda x\,M)\,N\longrightarrow M[N/x]$

C.B.V. : $(\lambda x\,M)\,V\longrightarrow M[V/x]$ si V est déjà évalué (normal?)

On a donc : $\langle V^\circ|\tilde{\mu}x.d\rangle\longrightarrow c[V/x]$, règle "call by value", on restreint la règle auparavant admise.

On modifie encore la grammaire:

$c::=\langle v|e\rangle$

$V::=x\;|\;e^\cdot\;|\;(V,V)\;|\;inl(V)\;|\;inr(V)$

$v::=V^\circ\;|\;\mu\alpha.c$

↳ On divise les expressions en deux sous-ensembles.

On conserve les règles de substitution explicite, et les règles logiques (CBV):

$\langle(e^\cdot)^\circ|\tilde{\mu}\alpha^\cdot\,c\rangle\longrightarrow c[e/\alpha]$

$\langle(V_1,V_2)^\circ|\tilde{\mu}(x_1,x_2).c\rangle\longrightarrow c[V_1/x_2,V_2/x_1]$

$\langle inl(V_1)^\diamond|\tilde{\mu}(inl(x_1).c_1\,|\,inr(x_2).c_2)\longrightarrow c_1[V_1/x_1]$.

Cette syntaxe correspond à une présentation focalisée de la logique classique.

↳ Ce calcul est une astuce pour éviter la paire de Lafont

Théorème: Ce calcul, LKQ, est confluent (ce qui implique la cohérence des preuves: deux preuves sans coupure ne peuvent pas être rendues égales par l'élimination des coupures).

Confluence: relation de réécriture $\longrightarrow$. $t\to t'$, $t\to t''$ alors par confluence il existe t^* tel que $t'\to^* t^*$, $t''\to^* t^*$.

On introduira le séquent: $\Gamma\vdash V:A;\Delta$ et une nouvelle coercion: $\dfrac{\Gamma\vdash V:A;\Delta}{\Gamma\vdash V^\circ:A\,|\,\Delta}$.

mercredi 3 fév.

Prouver focalisées

Initié par Andreoli, dans une perspective plus logique que fonctionnelle: recherche de preuves et non par transformation de preuve.

Recherche de preuve → Programmation logique (par ex. Prolog).

Transformation de preuve → Programmation fonctionnelle (par ex. Caml).

Andreoli travaille sur la logique linéaire, et voulait réduire l'espace de recherche des preuves : il a vu pour cela une discipline de recherche de preuves.

On reformule sa discipline en termes de logique classique:

L'idée est d'exploiter une différence entre la gauche et la droite du séquent: ce qui est à gauche est réversible, ce qui est à droite est irréversible. Le terrain est préparé par les règles qu'on s'est données:

$$\frac{\Gamma, A, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash A \vee B, \Delta} \qquad \frac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \qquad \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A} \qquad \frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta}$$

Réversible — Irréversible. — réversibles

Discipline de recherche de preuve focalisée : algorithme réduisant les choix, en fixant les alternances de phases droites et gauches:

- Phase droite: • Choisir une formule à droite (on la distingue en la plaçant de le bénitier), celle sur laquelle on focalise. (on peut la distinguer en la gardant sur copie)
  - Décomposer cette formule de manière héréditaire dans chaque branche de l'arbre de preuve partiel.
  - Lorsque l'on trouve dans le bénitier une négation, on la décompose et l'on passe à la phase gauche.
    un atome, on passe à la phase gauche.

note : Le choix de la formule dans la phase droite est déjà irréversible: rien ne dit que la preuve termine par l'introduction de cette formule (celle sur laquelle on focalise)

ex.: on passe à la phase gauche. on continue dans cette branche la phase droite.

$$\dfrac{\dfrac{\dfrac{\Gamma; A \vdash \Delta}{\Gamma \vdash \neg A; \Delta} \qquad \vdots \quad \Gamma \vdash B; \Delta}{\Gamma \vdash \neg A \wedge B; \Delta}}{\Gamma \vdash (\neg A \wedge B) \vee C; \Delta}$$

↑ choix de chercher cette preuve et non $\Gamma \vdash C; \Delta$.

↑ choix de la formule, qu'on place dans le bénitier.

$$\dfrac{\Gamma \vdash (\neg A \wedge B) \vee C, \Delta}{}$$

- Phase gauche : • Choisir une formule à gauche.
  - La décomposer une fois.
  - Boucler (choix, et un pas de décomposition)
  - Arrêt et retour en phase droite quand on veut.
- À tout moment, on peut clore une branche si le séquent feuille est un axiome.

L'algorithme commence en phase gauche ou en phase droite.

Ex. :

$$\dfrac{\dfrac{\vdots}{\Gamma, \neg A', B, C' \vdash \Delta}}{\dfrac{\Gamma, \neg A' \wedge B, C' \vdash \Delta \qquad \Gamma, \neg A' \wedge B, D \vdash \Delta}{\Gamma, (\neg A' \wedge B), C' \vee D \vdash \Delta}}$$

↳ Stratégie présentée sous sa forme la plus libérale: focalisation faible.

La focalisation forte :

Faible:

- Seuls les axiomes atomiques sont autorisés.

$$\frac{}{A \vee B \vdash A \vee B}\ Ax.$$

- Les phases gauches doivent être maximales. Ainsi toutes les phases droites se déroulent sur des séquents $\exists \vdash A; \Delta$, où $\exists$ n'est composé que d'atomes
- La stratégie doit commencer par une phase gauche

Forte:

$$\frac{}{A \vdash A;}\ Ax \qquad \frac{}{B \vdash B;}\ Ax$$

$$\frac{A \vdash A \vee B;}{A \vdash A \vee B} \qquad \frac{B \vdash A \vee B;}{B \vdash A \vee B}$$

$A \vee B \vdash A \vee B$.

Si A et B sont des atomes.

Cette focalisation nous mènera vers une autre syntaxe (le système L synthétique).

Théorèmes à prouver: Les ajouts au crayon de mine rendent les théorèmes plus forts

1. La stratégie de focalisation est complète.
   Formellement : si $\Gamma \vdash \Delta$ est prouvable dans LK, alors $\Gamma \vdash \Delta$ est prouvable en suivant une stratégie focalisée (forte) !
2. On va construire un système logique pour LK qui admet une élimination des coupures confluente. On veut un sous-ensemble de LK qui contienne toutes les preuves sans coupures obtenues par recherche focalisée (faible) et où l'on puisse éliminer les coupures de manière confluente.

On revient, pour prouver ces théorèmes, au langage de termes établi plus tôt (en fait à une de ses variation):

$$C ::= \langle V \mid e \rangle \;\big|\; [C\ q,q\ C]$$
$$V ::= x \;\big|\; \mu\alpha.\, c \;\big|\; V^\circ$$
$$\overline{V} ::= e^\bullet \;\big|\; (V,V) \;\big|\; inl(V) \;\big|\; inr(V) \qquad \text{Phase droite}$$
$$e ::= \alpha \;\big|\; \tilde{\mu} q.\, c \qquad \text{Interface entre les phases}$$
$$q ::= x \;\big|\; \alpha^\bullet \;\big|\; (q,q) \;\big|\; [q,q]$$

On ne donne pas de règle pour le moment

q est un ensemble de motifs (pattern), ce qui sert en informatique pour définir les fonctions par "pattern-matching".

ex.: $\tilde{\mu}\,(inl(x_1).c_1 \mid inr(x_2).c_2)$ réécrit des ormais $\tilde{\mu}[x_1,x_2].[c_1\ x_1,x_2\ c_2]$.

* Dans l'idée, $\overline{V}$ va être associé à la phase droite:

$$\frac{\Gamma \vdash V_1 : A_1 ; \Delta \qquad \Gamma \vdash V_2 : A_2 ; \Delta}{\Gamma \vdash (V_1,V_2) : A_1 \wedge A_2 ; \Delta} \qquad \frac{\Gamma \vdash V_1 : A_1 ; \Delta}{\Gamma \vdash inl(V_1) : A_1 \vee A_2 ; \Delta}$$

idem avec inr.

activée d'un pôle dans la bénition.

$$\frac{\Gamma \mid e : A \vdash \Delta}{\Gamma \vdash e^\bullet : \neg A ; \Delta}$$

* Passage dans la recherche de preuve de droite à gauche par la négation:

phase gauche ↑

$$\frac{c : (\Gamma, q : A \vdash \Delta)}{\Gamma \mid \tilde{\mu} q.\, c : A \vdash \Delta}$$
$$\frac{\Gamma \mid \tilde{\mu} q.\, c : A \vdash \Delta}{\Gamma \vdash (\tilde{\mu} q.\, c)^\bullet : \neg A ; \Delta}$$

phase droite ↓

$$\frac{C_1 : (\Gamma, q_1 : A_1 \vdash \Delta) \qquad C_2 : (\Gamma, q_2 : A_2 \vdash \Delta)}{[C_1\ q_1,q_2\ C_2] : (\Gamma, [q_1,q_2] : A_1 \vee A_2 \vdash \Delta)}$$

$$\frac{C_3 : (\Gamma, q_3 : A_3 \vdash \Delta) \qquad C_4 : (\Gamma, q_4 : A_4 \vdash \Delta)}{[C_3\ q_3,q_4\ C_4] : (\Gamma, [q_3,q_4] : A_3 \vee A_4 \vdash \Delta)}$$

$$\frac{[C_1\ q_1,q_2\ C_2] : (\Gamma, [q_1,q_2] : A_1 \vee A_2 \vdash \Delta) \qquad [C_3\ q_3,q_4\ C_4] : (\Gamma, [q_3,q_4] : A_3 \vee A_4 \vdash \Delta)}{[[C_1\ q_1,q_2\ C_2]\ [q_1,q_2],[q_3,q_4]\ [C_3\ q_3,q_4\ C_4]] : (\Gamma, [[q_1,q_2],[q_3,q_4]] : (A_1 \wedge A_2) \vee (A_3 \wedge A_4) \vdash \Delta)}$$

* La phase gauche est associée à C:

$$\frac{C : (\Gamma, q_1 : A_1, q_2 : A_2 \vdash \Delta)}{C : (\Gamma, (q_1,q_2) : A_1 \wedge A_2 \vdash \Delta)}$$

noter la différence avec $\tilde{\mu}(x_1,x_2).c$, qui fait passer d'une commande à un contexte, ici on obtient une commande.

$$\frac{C_1 : (\Gamma, q_1 : A_1 \vdash \Delta) \qquad C_2 : (\Gamma, q_2 : A_2 \vdash \Delta)}{[C_1\ q_1,q_2\ C_2] : (\Gamma, [q_1,q_2] : A_1 \vee A_2 \vdash \Delta)}$$

$$\frac{C : (\Gamma \vdash \alpha : A ; \Delta)}{C : (\Gamma, \alpha^\bullet : \neg A \vdash \Delta)}$$

* Passage de gauche à droite:

Mardi 16 fév. → Séance d'exercices.

Fin février → Partiel (2 mars?)

→ Avant le 4 mars → 2 mai 18

phase droite ↓

$$\frac{\Gamma \vdash V : A ; \Delta}{\langle V^\circ \mid \alpha \rangle : (\Gamma \vdash \alpha : A, \Delta)}$$
$$\frac{\langle V^\circ \mid \alpha \rangle : (\Gamma \vdash \alpha : A, \Delta)}{\langle V^\circ \mid \alpha \rangle : (\Gamma, \alpha^\bullet : \neg A \vdash \Delta)}$$

phase gauche ↑

Pour n'avoir que des formes normales, on restreint C à $\langle V^\circ \mid \alpha \rangle$. Un terme en forme normale est un terme auquel aucune réécriture ne peut s'appliquer. ↳ c'est une coupure/contraction qui n'est pas une coupure.

• Avantages et inconvénients de cette syntaxe avec contre-motifs par rapport à celle avec $\tilde{\mu}\alpha$, $\tilde{\mu}(x_1,x_2)$, etc.

Avantages: • Colle bien avec la description informelle de la discipline de focalisation.
• L'élimination des coupures est plus élégante.

Inconvénients: • Les étapes de commutation sont plus délicates.
• $[C^{q,q}\ C]$ est peu clair et conceptuellement instable.

⇒ C'est un quotient sur les preuves:

$$\left.\begin{array}{c} c : (\Gamma, x_1 : A_1, x_2 : A_2, x_3 : A_3, x_4 : A_4 \vdash \Delta) \\ \hline c : (\Gamma, (x_1,x_2) : A_1 \wedge A_2, x_3 : A_3, x_4 : A_4 \vdash \Delta) \\ \hline c : (\Gamma, (x_1,x_2) : A_1 \wedge A_2, (x_3,x_4) : A_3 \wedge A_4 \vdash \Delta) \end{array}\right\} \text{cette preuve est identifiée à la preuve qui aurait introduit } A_3 \wedge A_4 \text{ puis } A_1 \wedge A_2.$$

On verra que ce quotient est insuffisant, avec les disjonctions notamment: $[[c_1,c_2],[c_3,c_4]]$ n'est pas identifié avec $[[c_1,c_3],[c_2,c_4]]$, alors qu'on souhaiterait que ce soit le cas. ?

Parce que $((A_1 \vee A_2) \vee (A_3 \vee A_4)) \leftrightarrow ((A_1 \vee A_3) \vee (A_2 \vee A_4))$

• Règles de réduction:

$$\langle \mu\alpha.\, c \mid e \rangle \;\to\; c[e/\alpha] \qquad \text{C.B.N.}$$

$$\langle V^\circ \mid \tilde{\mu} q.\, c \rangle \;\to\; c[V/q] \qquad \text{C.B.V.}$$

- Règles de substitution :

$$C[(V_1,V_2)/(q_1,q_2)] \to C[V_1/q_1, V_2/q_2]$$

$$C[e^\circ/\alpha^\circ] \to C[e/\alpha]$$

$$[C_1{}^{q_1,q_2}C_2][\mathrm{inl}(V_1)/[q_1,q_2]] \to C_1[V_1/q_1] \qquad [C_1{}^{q_1,q_2}C_2][\mathrm{inr}(V_2)/[q_1,q_2]] \to C_2[V_2/q_2]$$

La substitution explicite décompose les contre-motifs et fait les commutations.

L'opérateur $\mu\alpha.c$ est presque dérivable : il ne l'est pas en soit, mais lorsqu'il est sous la forme $\langle \mu\alpha.c \mid e\rangle$:

$$\langle (e^\circ)^\bullet \mid \tilde{\mu}\alpha^\circ.c\rangle \to c[e^\circ/\alpha^\circ] \to c[e/\alpha] \leftarrow \langle \mu\alpha.c \mid e\rangle$$

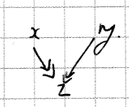
(la règle n'est plus valable lorsqu'il est coupé)

$\mu\alpha.c$ est codable faiblement en ce sens que tout contexte où il est utilisé peut être remplacé par un terme qui n'utilise pas μ sans changer le comportement. Une expression qui contient $\mu\alpha.c$ ne peut être que de la forme $\langle v \mid e\rangle$. Ainsi $\langle \mu\alpha.c \mid e\rangle$ est remplacé par $\langle (e^\circ)^\bullet \mid \tilde{\mu}\alpha^\circ.c\rangle$, ce qui est regrettable au niveau logique : on introduit une négation à chaque fois qu'on réintroduit μ.

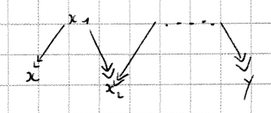
On conserve donc pour simplicité ce symbole μ dans notre syntaxe.

mardi 9 fév

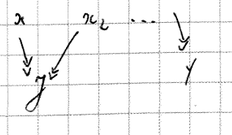
Fait : Si R est confluente, si $x R^* y$ par la clôture réflexive, symétrique et transitive de R, alors il existe z tel que $x R^* z$ et $y R^* z$:



Preuve : par induction sur le nombre de "pics" graphiques :



par confluence, il existe z tel que :

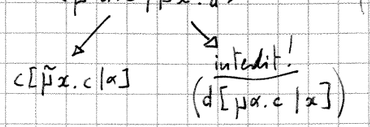


↳ Nous inférons pour obtenir une logique classique "constructive", c.à.d. dans laquelle la théorie de transformation des preuves n'est pas "incohérente" (dans le sens qu'il existe π_1 et π_2 telles que $\pi_1 \neq \pi_2$).

La confluence implique la cohérence de la théorie des preuves :

deux preuves sans coupures π_1 et π_2 distinctes sont telles que $\pi_1 \neq \pi_2$: on ne peut pas prouver qu'elles sont égales.

Ici on a la confluence par l'interdiction : $\langle \mu\alpha.c \mid \tilde{\mu}x.d\rangle$ (dans la paire critique de l'opérateur, le même terme se chevauche lui-même).



Système de réécriture des termes

Théorème de Klop : (Combinatory Reduction System : son système) Tout système (avec liens) orthogonal est confluent.

(Alternative au syst. C.R.S. de Klop : Higher Order Rewriting Systems, de Nipkow).

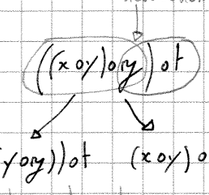
Un système est orthogonal : s'il est linéaire à gauche : dans chaque membre gauche de règle de réécriture, il n'y a pas de répétition de variable.

ex de règle non linéaire : $f(x,x) \to x$

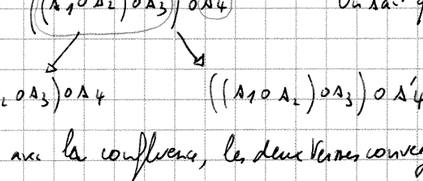
ex de règle linéaire : $f(x,y) \to h(k(x),y)$.

- s'il respecte le non chevauchement des règles

ex l'associativité se chevauche avec elle-même :



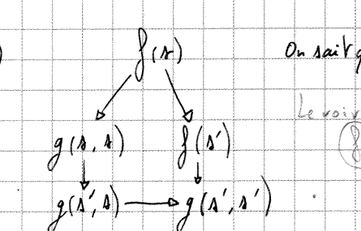
ex. de non chevauchement :



On n'a ici aucun problème avec la confluence, les deux termes convergent vers $A_1\circ(A_2\circ A_3)\circ A'_4$.

ex de règle duplicante : $f(x) \to g(x,x)$

pas de chevauchement



Dans ce genre de cas, on se ramène à la confluence en trouvant une relation $\Rightarrow$ telle que $\to \subseteq \Rightarrow \subseteq \to^*$ et qui soit confluente, ce qui implique que $\to$ est confluente également. On procède par réductions indépendantes sur les termes. ↳ on ne rentre pas dans la structure des règles, le système est confluent pour des raisons générales, parce qu'il est orthogonal.

Syntaxe : $c ::= \langle V \mid e \rangle \mid c[\sigma]$

$V ::= x \mid e^\bullet \mid (V,V) \mid \mathrm{inl}(V) \mid \mathrm{inr}(V) \mid V[\sigma]$

$v ::= V^\circ \mid \mu\alpha.c \mid v[\sigma]$

$e ::= \alpha \mid \tilde{\mu}x.c \mid \tilde{\mu}\alpha^\bullet.c \mid \tilde{\mu}(x_1,x_2).c \mid \tilde{\mu}(\mathrm{inl}(x).c_1 \mid \mathrm{inr}(x).c_2) \mid e[\sigma]$.

Règles :

$\langle \mu\alpha.c \mid e \rangle \longrightarrow c[e/\alpha]$ C.B.N.

$\langle V^\circ \mid \tilde{\mu}x.c \rangle \longrightarrow c[V/x]$ C.B.V. Réduit que si V déjà valeur

$\langle e^{\bullet\circ} \mid \tilde{\mu}\alpha^\bullet.c \rangle \longrightarrow c[e/\alpha]$

$\langle (V_1,V_2)^\circ \mid \tilde{\mu}(x_1,x_2).c \rangle \longrightarrow c[V_1/x_1, V_2/x_2]$

$\langle \mathrm{inl}(V_1)^\circ \mid \tilde{\mu}(\mathrm{inl}(x_1).c_1, \mathrm{inr}(x_2).c_2) \rangle \longrightarrow c_1[V_1/x_1]$

idem inr

+ toutes les règles de substitution, sans surprise.

$\langle \mu\alpha_1.c_1 \mid \tilde{\mu}\alpha_2.c_2 \rangle$ → $c_1[\tilde{\mu}\alpha_2.c_2/\alpha_1]$; $C_2[\mu\alpha_1.c_1/\alpha_2]$

Toutes ces règles sont linéaires à gauche. Les méta-variables sont $c, V, v, e, V_1, e_1, \dots$, et elles ne sont jamais imprimées à gauche.

Toutes les règles ne sont pas linéaires à droite, certaines sont dupliantes. Par ex. $\langle v \mid e \rangle[W/x] \longrightarrow \langle v[W/x] \mid e[W/x] \rangle$.

Il n'y a pas de chevauchement dans les règles. La seule possibilité de chevauchement était $\langle \mu\alpha.c \mid \tilde{\mu}\alpha.c' \rangle$, mais on ne peut plus réécrire cette terme que vers la gauche désormais. Le système ancien a la même syntaxe, sauf que V et v n'existent pas, remplacé par $v ::= x \mid e^\bullet \mid \mu\alpha.c \mid (v,v) \mid \mathrm{inl}(v) \mid \mathrm{inr}(v)$. Ses règles sont identiques sauf que $\langle V^\circ \mid \tilde{\mu}x.c \rangle \longrightarrow c[V/x]$ est remplacé par $\langle v \mid \tilde{\mu}x.c \rangle \longrightarrow c[v/x]$. Ce système est non orthogonal (chevauchement) et non confluent. Le fait de polariser / focaliser ce système l'amène confluent.

Agenda :
- normalisation / élimination des coupures pour les preuves focalisées
- complétude des preuves focalisées
  - ↳ Si $\Gamma \vdash \Delta$ est prouvable dans LK, alors il est prouvable par une preuve focalisée (sans coupure).

Preuve : On traduit toute preuve de LK en une preuve du système L focalisant. Cette traduction peut introduire des coupures.

$$\frac{}{\Gamma, A \vdash A, \Delta}$$ ⇝ $\langle x^\circ \mid \alpha \rangle$, adaptation du langage de preuve de LK qui donnait $\langle x \mid \alpha \rangle$.

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma \vdash \Delta}$$ ⇝ $\langle \mu\alpha.c_1 \mid \tilde{\mu}x.c_2 \rangle$ induction, alors $\exists c_1, c_2$ t.q. si la feuille de $\Gamma \vdash A, \Delta$ admet $\alpha : A$, elle non?

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta}$$ ⇝ $\langle x \mid \tilde{\mu}\alpha^\bullet.c \rangle$ nous donnait le langage de preuve de LK, on l'adapte en $\langle x^\circ \mid \tilde{\mu}\alpha^\bullet.c \rangle$.

$$\frac{\Gamma, A_1, A_2 \vdash \Delta}{\Gamma, A_1 \wedge A_2 \vdash \Delta}$$ ⇝ $\langle x^\circ \mid \tilde{\mu}(x_1,x_2).c \rangle$ ajout de la nouvelle variable par rapport à l'ancienne

↳ peu à dire pour ces règles, c'est du côté droit que la focalisation joue pleinement et complexifie la traduction.

$$\frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \neg A, \Delta}$$ ⇝ $\langle (\tilde{\mu}x.c)^{\bullet\circ} \mid \alpha \rangle$ α valeur

$$\frac{\Gamma \vdash A_1, \Delta}{\Gamma \vdash A_2 \vee A_1, \Delta}$$ ⇝ $\langle \mathrm{inl}(\mu\alpha_1.c) \mid \alpha \rangle$ n'est pas un terme de notre syntaxe.

(activer ; introduire ; désactiver)

en info., $(\lambda x.M)N \neq \mathrm{let}\ x = N \mathrm{in}\ M$, car on a le droit d'appeler $(\lambda x.M)N$ par nom.

let implique en sémantique opérationnelle d'évaluer N d'abord. On veut essayer d'introduire dans notre syntaxe quelque chose comme "$\mathrm{let}\ x = \mu\alpha_1.c_1$ in $\langle \mathrm{inl}(x)^\circ \mid \alpha \rangle$". $\langle \mathrm{inl}(x)^\circ \mid \alpha \rangle$ est dans notre syntaxe, et par définition $\mathrm{let}\ x = v$ in c est une macro pour $\langle v \mid \tilde{\mu}x.c \rangle$. On n'évalue c que lorsque v est une valeur, c'est exactement le mécanisme dont on a besoin : on force le $\mu\alpha_1.c$ à se calculer, et lorsque c'est fait on peut le substituer dans le terme, dupliquer, copier la valeur. On obtient :

$\langle \mu\alpha_1.c_1 \mid \tilde{\mu}x.\langle \mathrm{inl}(x)^\circ \mid \alpha \rangle \rangle$ On introduit une coupure, qui sera réduite avec la règle concernant $\langle \mu\alpha.c \mid e \rangle$ et non celle concernant $\langle V^\circ \mid \tilde{\mu}x.c \rangle$.

$$\frac{\dfrac{\dfrac{\dfrac{\dfrac{}{\langle x \mid \alpha \rangle : \Gamma, A_1 \vdash A_1, \Delta}\,Ax}{\mu\alpha.\langle x \mid \alpha \rangle : \Gamma, A_1 \vdash A_1 \mid \Delta}}{\mathrm{inl}(\mu\alpha.\langle x \mid \alpha \rangle) : \Gamma, A_1 \vdash A_1 \vee A_2 \mid \Delta}}{\langle y \mid \mathrm{inl}(\mu\alpha.\langle x \mid \alpha \rangle) \rangle : \Gamma, A_1 \vdash y : A_1 \vee A_2, \Delta} \qquad \Gamma \vdash A_1, \Delta, A_1 \vee A_2}{\tilde{\mu}x.\langle y \mid \mathrm{inl}(\mu\alpha.\langle x \mid \alpha \rangle) \rangle : \Gamma \vdash x : A_1 \mid \Delta, A_1 \vee A_2}$$ aff.

$$\frac{}{\Gamma, A_1 \vdash A_1 ; \Delta}\,Ax. \qquad \frac{\Gamma, A_1 \vdash A_1 ; \Delta}{\Gamma, A_1 \vdash A_1 \vee A_2 ; \Delta} \qquad \frac{\Gamma, A_1 \vdash A_1 \vee A_2 ; \Delta}{\Gamma \mid A_1 \vdash A_1 \vee A_2, \Delta}$$

$$\frac{\dfrac{x^\circ : \Gamma, x : A_1 \vdash \alpha : A_1 \mid \Delta}{\mathrm{inl}(x^\circ) : \Gamma, x : A_1 \vdash \alpha : A_1 \vee A_2 \mid \Delta}}{\langle \mathrm{inl}(x^\circ) \mid \alpha \rangle : \Gamma, x : A_1 \vdash \alpha : A_1 \vee A_2 \mid \Delta} \qquad \tilde{\mu}x.\langle \mathrm{inl}(x^\circ) \mid \alpha \rangle : \Gamma \mid x : A_1 \vdash \alpha : A_1 \vee A_2$$

$\langle \mu\beta.c \mid \tilde{\mu}x.\langle y \mid \mathrm{inl}(\mu\alpha.\langle x \mid \alpha \rangle) \rangle \rangle : \Gamma \vdash A_1 \vee A_2, \Delta$ $\qquad \Gamma \vdash A_1 \vee A_2, \Delta.$

let $x = \mu\alpha.c_1$ in c_2 est une macro pour $\langle \mu\alpha.c_1 \mid \tilde{\mu}x.c_2 \rangle$ qui force à calculer d'abord $\mu\alpha.c_1$, puis ensuite à substituer la valeur obtenue à x dans c_2. On note $\tilde{\mu}x.c_2$ en réservé et on l'appelle la variable de continuation α. On se permet réduire $\langle \mu\alpha.c_1 \mid \tilde{\mu}x.c_2 \rangle$ qu'en $c_1[\tilde{\mu}x.c_2/\alpha]$, mais on doit calculer d'abord c_1 avant d'effectuer la substitution.

$$\frac{\Gamma \vdash A_1, \Delta \qquad \Gamma \vdash A_2, \Delta}{\Gamma \vdash A_1 \wedge A_2, \Delta} \quad \rightsquigarrow \quad c_1 : (\Gamma \vdash \alpha_1 : A_1, \Delta) \text{ et } c_2 : (\Gamma \vdash \alpha_2 : A_2, \Delta)$$

$$\text{let } x_1 = \mu\alpha_1.c_1 \text{ in } \left(\text{let } x_2 = \mu\alpha_2.c_2 \text{ in } \left(\langle (x_1, x_2)^\diamond \mid \alpha \rangle\right)\right)$$

Le terme initial était $\langle (\mu\alpha_1.c_1, \mu\alpha_2.c_2) \mid \alpha \rangle$.

$$\langle \mu\alpha_1.c_1 \mid \tilde{\mu}x_1.\langle \mu\alpha_2.c_2 \mid \tilde{\mu}x_2.\langle (x_1, x_2)^\diamond \mid \alpha \rangle \rangle \rangle.$$

↳ la traduction de LK dans LK focalisée n'est pas anodine : elle choisit un ordre d'évaluation : on évalue c_1 puis c_2, on aurait pu prendre l'ordre inverse.

On a vu la confluence, la complétude de LK par rapport à LK focalisée avec coupures → il manque

Thm : L'élimination des coupures termine dans LK focalisée et un autre ingrédient pour montrer que la complétude de la recherche de preuve focalisée.

Soit $\Gamma \vdash \Delta$ prouvable dans LK

alors $\Gamma \vdash \Delta$ prouvable dans LK focalisée avec coupures

alors par une preuve en forme normale. Alors ... par un preuve sans coupure focalisée.

Ce qui nous manque est : toute preuve en forme normale correspond à une preuve focalisée sans coupure.

mer. 10 fév.

Formes normales : On vérifie que les formes normales sont sans coupure.

On n'a besoin d'examiner que les membres gauches des termes dans les règles de réécriture.

On le démontre en étudiant les cas pour la structure des termes :

Si le terme est de la forme $\langle v \mid e \rangle$: $v \neq \mu\alpha.c$, sinon la règle $\langle \mu\alpha.c \mid e \rangle \to c[e/\alpha]$ s'applique.

Donc v est de la forme V°, on regarde le terme e.

si $e = \alpha$, le terme est $\langle V^\circ \mid \alpha \rangle$, qui correspond à la contraction.

si $e = \tilde{\mu}x.c$, on applique la règle $\langle V^\circ \mid \tilde{\mu}x.c \rangle \to c[V/x]$.

si $e = \tilde{\mu}\alpha^\bullet.c$, on étudie les différents cas :

si $V = x$, le terme est $\langle x^\circ \mid \tilde{\mu}\alpha^\bullet.c \rangle$, qui correspond à une contraction

si $V = e^\bullet$, le terme est $\langle e^{\bullet\circ} \mid \tilde{\mu}\alpha^\bullet.c \rangle$, qui se réécrit.

si $V = (V_1, V_2)$, $\mathrm{inl}(V_1)$ ou $\mathrm{inr}(V_1)$, ces cas sont exclus par nos règles de typage.

$\langle (V_1, V_2)^\circ \mid \tilde{\mu}\alpha^\bullet.c \rangle$ n'est pas un cas possible.

si $e = \tilde{\mu}(x_1, x_2).c$, on regarde la forme de V, tous les cas se traitent de la même façon, sauf

$\langle x^\circ \mid \tilde{\mu}(x_1, x_2).c \rangle$.

si $e = \tilde{\mu}(\mathrm{inl}(x_1))$, on fait le même traitement et le seul cas qui reste est $\langle x^\circ \mid \tilde{\mu}[\mathrm{inl}(x_1).c_1 \mid \mathrm{inr}(x_2).c_2] \rangle$.

Ces cas correspondent à des règles de contraction : $\langle V^\circ \mid \alpha \rangle$, $\langle x^\circ \mid \tilde{\mu}\alpha^\bullet.c \rangle$, $\langle x^\circ \mid \tilde{\mu}(x_1, x_2).c \rangle$ et

$\langle x^\circ \mid \tilde{\mu}[\mathrm{inl}(x_1).c_1 \mid \mathrm{inr}(x_2).c_2] \rangle$. $\langle v \mid e \rangle$, c'a.d. $\langle V^\circ \mid e \rangle$

Normalisation : faible : $\forall s\ \exists$ chemin $s \to s_1 \to \ldots \to s_n$, s_n en évaluation forme normale.

forte : $\forall s \to \exists$ chemin infini $s \to s_1 \to \ldots \to s_n \to s_{n+1} \to \ldots$

On démontre la normalisation faible, et pour se faire on introduit les notions de résidu d'un radical et de radical créé par une réduction.

On l'illustre sur une règle de réécriture plus simple : système de $\begin{cases} h(x) \to g(x) \\ f(g(x)) \to h(x) \end{cases}$ $\quad P(s,t)$ ne se réduit pas.

signature : $\Sigma = \{f^1, g^1, h^1, P^2\}$

deux radicaux

Exemple de résidu : $P(h(x), f(g(x)))$

$P(g(x), f(g(x)))$ $\qquad$ $P(h(x), h(x))$

résidu du radical présent dans le premier terme

$P(g(x), h(x))$

Il peut y avoir duplication des radicaux : si R est un radical sous-terme de N, $(\lambda x.xx)N \to NN$ duplique ce radical.

Exemple de création : $f(h(x)) \to f(g(x)) \to h(x)$

Ce radical n'est pas un résidu d'un radical de $f(h(x))$, c'est une création

Le λ-calcul non-typé ne normalise pas : $(\lambda x.xx)(\lambda y.yy) \to (\lambda y.yy)(\lambda y.yy)$ Il y a ici création de β-redex : ce n'était pas un radical, $(\lambda y.yy)(\lambda y.yy)$ est un radical créé par la substitution de $\lambda y.yy$ sur la première occurrence de x.

Le λ-calcul non-typé ne termine pas car il peut toujours créer des radicaux (thm des diaphonies permet de limiter la création des radicaux que l'on peut discipliner).

Preuve de normalisation faible du système L (typé) :

On supprime la règle $\langle \mu\alpha.c \mid e \rangle \to c[e/\alpha]$, car $\langle \mu\alpha.c \mid e \rangle$ peut se coder comme $\langle e \mid \tilde{\mu}\alpha.c \rangle$. On peut alors supprimer dans la syntaxe la classe v et modifier c en $\langle V^\circ \mid e \rangle$. Les deux systèmes sont équiexpressifs.

On se passe des substitutions explicites : on définit une opération qui termine, que l'on note $c\{\sigma\}$, et définie par :

- $V\{\sigma\} : (V_1, V_2)\{\sigma\} = (V_1\{\sigma\}, V_2\{\sigma\})$, le reste est à l'habitude. (On note = plutôt que → car la transformation est instantanée.)
- $\langle v^\circ \mid e \rangle\{\sigma\} = \langle V\{\sigma\}^\circ \mid e\{\sigma\} \rangle$ si on n'est pas dans un des cas suivant :

  * $\langle x^\circ \mid \tilde{\mu}x.c \rangle\{\sigma, V/x\} \neq \langle x\{\sigma, V/x\}^\circ \mid \tilde{\mu}y.c\{\sigma, V/x\} \rangle = \langle V^\circ \mid \tilde{\mu}y.c\{\sigma, V/x\} \rangle$ Création de radical, car $\langle x^\circ \mid \tilde{\mu}x.c \rangle$ correspond à une contraction. Cette coupure créé a pour principal la même formule que la coupure éliminée (le type associé au radical créé est le même que le type du radical créateur).

    On adopte donc : $\langle x^\circ \mid \tilde{\mu}x.c \rangle\{\sigma, V/x\} = c\{\sigma, V/x\}$.

  * $\langle x^\circ \mid \tilde{\mu}(x_1, x_2).c \rangle\{\sigma, (V_1, V_2)/x\} = c\{\sigma, V_1/x, V_2/x\}$

La définition de $c\{\sigma\}$, $e\{\sigma\}$, $v\{\sigma\}$ est bien fondée car la taille de c, e ou v décroit avec la réécriture pour tout σ.

Sur le même modèle : * $\langle x^\circ \mid \tilde{\mu}\alpha^\cdot.c \rangle\{\sigma, e^\cdot/x\} = c\{\sigma, e/x\}$

* $\langle x^\circ \mid \tilde{\mu}(\mathrm{inl}(x_1).c_1 \mid \mathrm{inr}(x_2).c_2) \rangle\{\sigma, \mathrm{inl}(V_1)/x\} = c_1\{\sigma, V_1/x_1\}$

Les nouvelles règles de réécriture sont donc : $\langle V^\circ \mid \tilde{\mu}x.c \rangle \to c\{V/x\}$

$\langle e^{\cdot\circ} \mid \tilde{\mu}\alpha^\cdot.c \rangle \to c\{e/\alpha\}$

$\langle (V_1, V_2)^{\times\circ} \mid \tilde{\mu}(x_1, x_2).c \rangle \to c\{V_1/x_1, V_2/x_2\}$

$\langle \mathrm{inl}(V_1) \mid \tilde{\mu}(\mathrm{inl}(x_1).c_1 \mid \mathrm{inr}(x_2).c_2) \rangle \to c_1\{V_1/x_1\}$

Pas de règle de réécriture pour $c\{\sigma\}$, car cette substitution opère immédiatement.

À tout radical on associe un degré, qui est égal à la taille du type de la formule coupée.

Si selon notre système de réécriture $s \to t$, si r est un radical de s et r' un radical de t résidu de r', alors
- le degré de r est identique à celui de r'.
- si r est un radical créé dans t par la réduction d'un radical r' de s, alors le degré de r est strictement inférieur à celui de r'.

La nouvelle substitution a écarté beaucoup de cas de création. La seule qui reste est celle-ci :

$\langle V^\circ \mid \alpha \rangle\{\sigma, (\tilde{\mu}x.c)/\alpha\} = \langle V^\circ\{\sigma\} \mid \tilde{\mu}x.c \rangle$

$\langle (V_1, V_2)^\circ \mid \alpha \rangle\{\sigma, \tilde{\mu}(x_1, x_2).c/\alpha\} = \langle (V_1, V_2)\{\sigma, \tilde{\mu}(x_1, x_2).c/\alpha\} \mid \tilde{\mu}(x_1, x_2).c \rangle$ et c.

(on ne substitue à une variable α que dans ce cas.)

Or ce cas ne se présente, avant donner nos règles de réécriture, qu'après avoir réduit un terme de la forme $\langle e^\circ \mid \tilde{\mu}\alpha^\cdot.c \rangle$, où c contient un sous-terme de la forme $\langle V^\circ \mid \alpha \rangle$, et e est de la forme $\tilde{\mu}x.c$.

Le degré du radical $\langle e^\circ \mid \tilde{\mu}\alpha^\cdot.c \rangle$ est la taille de la formule coupée, de type τ → ρ.

$\langle V^\circ \mid e \rangle$, c'est-à-dire la taille de ρ. décroit.

↳ Seul cas de création possible.

La preuve elle-même : à chaque étape de réécriture, on choisit de réduire un radical de degré maximal tel que tous ses sous-radicaux sont de degré strictement inférieur.

Si $s \to t$ selon cette stratégie, alors poids(t) < poids(s), avec poids(s) = multiensemble des degrés de ses radicaux. (↑ extension multi-ensemble de l'ordre sur ℕ.)

(Extension multi-ensemble d'une relation d'ordre : soit une relation d'ordre « sur X, $(\mathcal{M}(X), \ll_{ext})$, l'ordre $\ll_{ext}$ sur le multi-ensemble obtenu à partir de X, est défini comme suit : si $y_j > u$, $y_j > v$, $\{x, y, y_j, y_j\} > \{x, y, u, v, y_j\}$. On peut remplacer un ensemble par un multi-ensemble d'ensembles plus petits arbitrairement grand.

Par le lemme de König, si « est bien fondé, $\ll_{ext}$ aussi.)

Si S est un radical de s, poids(s) = {… degré(S) …}, si S est maximal, tous les sous-termes de S qui sont des radicaux sont de degré strictement plus petit que le degré de S.

Si $s \to_S t$, le poids de t est composé des degrés des résidus de s et des degrés des résidus créés. On sait que les degrés des résidus créés sont inférieurs au degré de S. Les résidus peuvent être indépendants de S,

on les nomme R_1, les résidus R_2 contiennent S, et les R_3 sont contenus dans S.

poids $(s) = \langle R_1, \text{degré}(S), R_2, R_3 \rangle \longrightarrow$ poids $(t) = \langle R_1, R_2, \text{résidus nés}, R_3', R_3' \dots \rangle$

Les résidus R_3 sont dupliqués en des résidus R_3', mais par hypothèse de stratégie ils sont de degrés plus petit que S.

Donc poids (s) > poids (t), et la procédure termine.

Machines abstraites pour le λ-calcul

$M ::= x \mid \lambda x.M \mid MM$ Langage de preuve pour la logique minimale présentée en déduction naturelle.

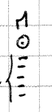
On y injecte en avant que langage de programmation, et on considère l'exécution du λ-calcul sur une machine abstraite. On décrit une machine Call by Value :

$\langle MM \mid [] \rangle$

code (v) ↑ ↑ pile (ici vide), on empile, et dépile en partant de la dernière chose mise dans la pile.

CBV : avant de calculer MN, on doit calculer N.

$\langle MN \mid e \rangle \longrightarrow \langle N \mid M \odot e \rangle$



Pour V une valeur, $\langle V \mid M \odot e \rangle \longrightarrow \langle M \mid V.e \rangle$, c'est le swap sur une pile.

On peut voir le code comme le sommet de la pile :

	swap (échange)	
V	→	M
M		V
⊙		·
e		e

Si $MN = \Big((\lambda x(\lambda y P))Q\Big)(\lambda y.y)y$, on évalue $(\lambda y.y)y \to y$, puis $(\lambda x(\lambda y P))Q \to \lambda y\, P[Q/x]$ avant d'évaluer les deux arguments. M et N doivent être évalués avant application.

(↳ ne contredit pas ce qui est dit plus tard, i.e bbo?)

$\langle \lambda x.M \mid V.e \rangle \longrightarrow \langle M[V/x] \mid e \rangle$

On a cette syntaxe : $c ::= \langle M \mid e \rangle$

$M ::= x \mid \lambda x.M \mid MM$

Syntaxe pour le code : syntaxe intermédiaire (entre le langage de programmation et un vrai compilateur)

en fait $c ::= \langle M \mid e \rangle$

$M ::= V^\circ \mid MM$

$V ::= x \mid \lambda x.M$

$e ::= [] \mid M \odot e \mid V.e$

Ce langage se code dans le système L. Le type de $\lambda x.M$ est $A \to B$, qui se code $\vdash (A \otimes \vdash B))$.
Signifie NJ → LK. ≈ $\vdash (A \parr \vdash B)$

Typage du λ-calcul : $\dfrac{}{\Gamma, x:A \vdash x:A}$ $\qquad \dfrac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$ $\qquad \dfrac{\Gamma, x:A \vdash M : B}{\Gamma \vdash \lambda x.M : A \to B}$

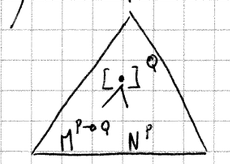
Typage du langage intermédiaire : $\Gamma \mid [] : R \vdash [] : R$ $\qquad \dfrac{\Gamma \vdash V : P; \quad \Gamma \mid e : Q \vdash R}{\Gamma \mid V.e : P \to Q \vdash R}$

$$\frac{\Gamma \vdash M : P \to Q \qquad \Gamma \mid e : Q \vdash R}{\Gamma \mid M \odot e : P \vdash R}$$

→ Partiel le 2 mars.

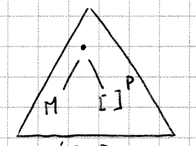
R est un type spécial qui représente le résultat final.

On commente la règle $\langle MN \mid e \rangle \to \langle N \mid M \odot e \rangle$.



E[MN]

→



E'[N]

?

Remplir le contexte

mardi 16 fév : ex (vers fin du cahier) et ce qui suit : (+ fiche pdf)

Syntaxe pour LK avec ∧, ∨, ¬, → :

$c ::= \langle v \mid e \rangle$

$V ::= x \mid \text{inl}(v) \mid \text{inr}(V) \mid \lambda x.v$

$v ::= \mu\alpha.c \mid V^\circ$

$e ::= \alpha \mid \tilde{\mu} x.c \mid \tilde{\mu}\left(\text{inl}(x_1)c_1 \mid \text{inr}(x_2)c_2\right) \mid V.e$

Langage de terme pour → : $\dfrac{\Gamma, x:A \vdash v : B \mid \Delta}{\Gamma \vdash (\lambda x.v) : A \to B ; \Delta}$ $\qquad \dfrac{\Gamma \vdash V : A \; ; \Delta \qquad \Gamma \mid e : B \vdash \Delta}{\Gamma \mid (V.e) : A \to B \vdash \Delta}$

$$\frac{\Gamma \vdash V : A \; ; \Delta}{c : (\Gamma \vdash \alpha : A, \Delta)} \longrightarrow \frac{\Gamma \vdash V : A \; ; \Delta \qquad \overline{\qquad}^{Ax}}{\Gamma \vdash V^\circ : A \mid \Delta \qquad \mid \alpha : A \vdash \alpha : A} \qquad \langle V^\circ \mid \alpha \rangle : (\Gamma \vdash \alpha : A, \Delta)$$

en fait, pour démarrer une phase positive : coercion $\dfrac{\Gamma \vdash V : A \; ; \Delta, \alpha : A}{\Gamma \vdash V^\circ : A \mid \alpha : A, \Delta}$ cont. $\dfrac{}{\Gamma \mid \alpha : A \vdash \alpha : A, \Delta}$

$\langle V^\circ \mid \alpha \rangle : (\Gamma \vdash \alpha : A, \Delta)$

mer. 17 Déc.

Relation entre (machine abstraite pour exécuter) le λ-calcul et le calcul des séquents (système L focalisant)

λ-calcul: $M ::= x \mid MM \mid \lambda x.M$ Machine abstraite en C.B.V. (existe aussi en C.B.N.) Caml (Haskell)

Règles : $\langle MN \mid e \rangle \to \langle N \mid M \circ e \rangle$ on évalue d'abord N jusqu'à ce que ce soit une valeur (notée $V^{(v)}$).

$\langle V^{(v)} \mid M \circ e \rangle \to \langle M \mid V.e \rangle$

$\langle \lambda x.P \mid V.e \rangle \to \langle P[V/x] \mid e \rangle$ β-réduction, on dépile V, P devient du code exécutable.

Terme : $M ::= V^{(v)} \mid MM$

Valeur : $V ::= x \mid \lambda x M$ } distinction qui vient de Plotkin (T.C.S., 1975).

x est une valeur car on ne remplace x que par une valeur. $\lambda x M$ est une valeur, on le voit avec un exemple de Caml : on déclare une fonction : let toto x = (4×5)+x (correspond à $\lambda x.(4\times5)+x$)

toto : nat → nat, et non $\lambda x\ 20+x$ → une fonction est un texte auquel on ne touche pas, c'est une valeur. On ne calculera 4×5 que si on demande par exemple toto 38. Le calcul n'effectue que lorsque l'on donne un argument (qui est une valeur, sinon on calcule d'abord l'argument).

Contexte : $e ::= [] \mid M \circ e \mid V.e$ cf. Curien, Herbelin, The duality of computation (2000)

État : $c ::= \langle M \mid e \rangle$

↳ On peut définir ces constructions comme opérateurs dérivés du système L. On commence par se donner des règles de typage. On a quatre catégories de termes (c, M, V, e), on se donne donc :

État : $c : (\Gamma \vdash)$ — Si on se donne R le type des résultats finaux, on a

Terme : $\Gamma \vdash M : A \mid$ — en fait : $c(\Gamma \vdash R)$ et $\Gamma \mid e : A \vdash R$.

Valeur : $\Gamma \vdash V : A$; — Du point de vue logique, R = faux (de plus du point de vue des tables de

Contexte : $\Gamma \mid e : A \vdash$ — vérité, $\Gamma \vdash$ équivaut à $\Gamma \vdash$ faux).

On n'a jamais plus d'une formule à droite → typique de la logique intuitionniste. On deviendra classique avec les opérateurs de contrôle.

Pour introduire des valeurs :

$$\frac{}{\Gamma, x : A \vdash x : A ;} \qquad \frac{\Gamma, x : A \vdash M : B \mid}{\Gamma \vdash \lambda x.M : A \to B ;} \begin{array}{l} \leftarrow \text{Pas une valeur} \\ \leftarrow \text{Valeur} \end{array} \Big\} \text{from toto}$$

Pour les termes :

$$\frac{\Gamma \vdash V : A ;}{\Gamma \vdash V^{(v)} : A \mid} \qquad \frac{\Gamma \vdash M_1 : A \to B \mid \quad \Gamma \vdash M_2 : A \mid}{\Gamma \vdash M_1 M_2 : B \mid}$$

M_1, M_2 et $M_1 M_2$ ne sont pas supposées être des valeurs.

$$\frac{\Gamma \vdash M : A \mid \qquad \Gamma \mid e : A \vdash R}{\langle M \mid e \rangle : (\Gamma \vdash R)}$$

$\Gamma \mid e : A \vdash R \qquad \Gamma \mid [] : R \vdash R \qquad \mid \alpha : A \vdash \alpha : A$ (?)

$$\frac{}{\langle M \mid [] \rangle} \text{ Etat initial}$$

1ère règle de réécriture

$$\frac{\Gamma \vdash M : A \to B \mid \quad \Gamma \mid e : B \vdash R}{\Gamma \mid M \circ e : A \vdash R} \qquad \frac{\Gamma \vdash V : A ; \quad \Gamma \mid e : B \vdash R}{\Gamma \mid (V.e) : A \to B \vdash R} \to \text{intro g de} \to.$$

2ème règle de la machine

On regarde les deux premières règles : $\langle MV \mid e \rangle \to \langle V \mid M \circ e \rangle \to \langle M \mid V.e \rangle$ lorsque le second argument est une valeur. On fait une analyse de typage :

$M : A \to B$ et $V : A$ puisque V est l'argument de M. Donc $MV : B$. Puisque c'est une coupure, e a V de type B également.

$$\langle \underbrace{MV}_{B} \mid \underbrace{e}_{B} \rangle \to^* \langle \underbrace{M}_{A\to B} \mid \underbrace{\overbrace{V}^{A}.\ \underbrace{e}_{B}}^{A\to B} \rangle$$

La règle de calcul impose le typage.

On associe la règle de typage et l'opération d'exécution d'une machine abstraite (lien remontant à la thèse de Herbelin, 1995).

Calcul des séquents (intro g., intro d), déduction naturelle (intro d, élim. d) → système hybride.

$$\frac{\Gamma \vdash V : A ; \quad \Gamma \mid e : B \vdash R}{\Gamma \mid (V.e) : A \to B \vdash R} \to g \qquad \frac{\Gamma \vdash M_1 : A \to B \mid \quad \Gamma \vdash M_2 : A \mid}{\Gamma \vdash M_1 M_2 : B \mid} \to \text{elim.}$$

On ajoute deux instructions, constructions de termes : $M := \ldots \mid \mathcal{C}(M)$, et $V := \ldots \mid e^*$

L'instruction unaire $\mathcal{C}(M)$ vient de Felleisen, qui participe à Scheme (langage fonctionnel non typé), fin 1970.

En 1990, Griffin, qui travaillait avec les types, a réussi à typer ces deux opérations, qui se sont avérées être ce qui manquait pour passer d'intuitionniste à classique.

On donne l'exécution, la règle de calcul : $\langle \mathcal{C}(M) \mid e \rangle \to \langle M \mid e^* \cdot [] \rangle$ « Capture » (gèle)

$\langle e^* \mid V \cdot e' \rangle \to \langle V \mid e \rangle$ « Restore » (dégèle)

$\mathcal{C}$ est un opérateur qui fait une capture de pile. Il fait un terme de cette pile qu'il nous faut nommer de la pile qu'il a vidé. On pourra restaurer cette pile

La seconde règle élimine la pile courante et restaure la pile capturée par $\mathcal{C}$.

↳ On brise le flot normal de calcul, on insère des annotations qui modifient le cours normal de l'exécution.

Raison pour laquelle on nomme ces deux règles un opérateur de contrôle (l'opérateur de contrôle $\mathcal{C}$ de Felleisen).

Il en existe d'autres, comme Call/cc : $\langle \text{Call/cc}(M) \mid e \rangle \to \langle M \mid e^* \cdot e \rangle$

Les règles de typage sont : $\dfrac{\Gamma \vdash M : (A \to R) \to R}{\Gamma \vdash \mathcal{C}(M) : A \mid}\ \neg_e$ (On peut le voir comme $\neg(\neg A) \to A$)

$\dfrac{\Gamma \mid e : A \vdash R}{\Gamma \vdash e^* : A \to R \mid}\ \neg_i$ (Intro de la négation à droite)

↳ Règles de déduction naturelle.

On vérifie les règles d'exécution (avec $\neg\neg A \equiv (A \to R) \to R$) :

$$\langle \underbrace{\mathcal{C}\underbrace{(M)}_{\neg\neg A}}_{A} \mid \underbrace{e}_{A} \rangle \to \langle \underbrace{M}_{\neg\neg A} \mid \underbrace{e^*}_{A \to R} \cdot \underbrace{[]}_{R} \rangle \quad (A \to R) \to R$$

$$\langle \underbrace{e^*}_{A \to R} \mid \overset{A}{\overset{\sqcap}{V}} \cdot \overset{R}{\overset{\sqcap}{e'}} \rangle \to \langle \underbrace{V}_{A} \mid \underbrace{e}_{A} \rangle \qquad A \to R$$

Résumé :

λ-calcul + contrôle ($\mathcal{C}$) machine abstraite en CBV Système de typage.

Ce système de typage a le « subject réduction », la préservation du typage par réduction : $A \to t$, $\Delta : A \Rightarrow t : A$

On compile le tout en calcul des séquents : (la machine abstraite en cBV et le système de typage vers le système L focalisant).

$c := \langle v \mid e \rangle$ ○ : ce qu'on doit coder

$v := V^0 \mid v\,v \mid \mathcal{C}(M)$ On définit les opérateurs comme des opérateurs dérivés, des macros dans le système focalisé.

$V := x \mid \lambda x.M \mid e^*$

$e := [] \mid M \odot e \mid V.e$

↑ la variable de continuation

$\mathcal{C}(M) =_{def} \mu\beta.\langle M \mid \beta^* \cdot [] \rangle$ $\langle \mathcal{C}(M) \mid \beta \rangle \to \langle M \mid \beta^* \cdot [] \rangle$

$e^* =_{def} \lambda x\, \mu\alpha.\langle x \mid e \rangle$ pour α fraîche, correspond à l'idée de jeter e' dans la récriture de x.

$\mathcal{C}(M) =_{def} \mu\beta.\langle M \mid \lambda x\, \mu\alpha.\langle x \mid \beta \rangle \cdot [] \rangle$

$\langle e^* \mid V \cdot e' \rangle = \langle \lambda x\, \mu\alpha.\langle x \mid e \rangle \mid V \cdot e' \rangle \to \langle (\mu\alpha.\langle x \mid e \rangle)[V/x] \mid e' \rangle$ en utilisant la règle du λ, qui est $\langle \lambda x.P \mid V.e \rangle \to \langle P[V/x] \mid e \rangle$

On a ensuite $\langle (\mu\alpha.\langle x \mid e \rangle)[V/x] \mid e' \rangle \to \langle \mu\alpha.\langle V \mid e \rangle \mid e' \rangle \to \langle V \mid e \rangle [e'/\alpha]$ par μ-réduction. Or α n'apparaît ni dans V ni dans e, donc $\langle V \mid e \rangle [e'/\alpha] \to \langle V \mid e \rangle$.

Correspond à notre règle $\langle e^* \mid V.e' \rangle \to \langle V \mid e \rangle$.

On justifie la règle de typage :

$$\dfrac{\dfrac{\dfrac{\dfrac{x : A \vdash x : A\ ;}{\langle x \mid e \rangle : x : A, \Gamma \vdash R}\ \text{Ax}^+ \qquad \Gamma \mid e : A \vdash R}{\langle x \mid e \rangle : (x : A, \Gamma \vdash R, \alpha : R)}\ \text{aff.}}{x : A, \Gamma \vdash \mu\alpha.\langle x \mid e \rangle : R}\ \text{activation}}{\Gamma \vdash \lambda x\, \mu\alpha.\langle x \mid e \rangle : A \to R}$$

revient à appliquer $\neg_i$.

aff., on n'a plus intuitionniste : deux formules à droite. Peut valoir : le R est comme rien, le vide.

$= e^*$

On justifie le codage de e^* du point de vue du typage :

$$\dfrac{\Gamma \vdash M : (A \to R) \to R \qquad \dfrac{}{\beta : A \vdash \beta : A}\ \text{Ax.}}{\vdash \beta^* : A \to R ; \beta : A}$$

on utilise vraiment la possibilité qu'on a d'avoir deux formules à droite.

non, cf. page suivante.

$$\dfrac{\dfrac{\dfrac{}{\Gamma \mid \beta : A \vdash \beta : A}\text{ax.}}{\Gamma \vdash \beta^* : A \to R ; \beta : A} \quad \dfrac{}{\Gamma \mid [\,] : R \vdash [\,] : R}\text{ax.}}{\Gamma \vdash M : (A \to R) \to R \qquad \Gamma \mid \beta^* . [\,] : (A \to R) \to R \vdash \beta : A, [\,] : R}$$

$$\dfrac{\langle M \mid \beta^* . [\,] \rangle : \Gamma \vdash \beta : A, [\,] : R}{\Gamma \vdash \mu\beta . \langle M \mid \beta^* . [\,] \rangle : A \mid [\,] : R} \approx \Gamma \vdash \mathcal{C}(M) : A \mid [\,] : R$$

(revient à l'intro) — ↑ comme rien

Pour $M \circ e$ (et $V.e$) :

La règle : $\langle V \mid M \circ e \rangle \to \langle M \mid V.e \rangle$ avec une variable : $\langle x \mid M \circ e \rangle \to \langle M \mid x.e \rangle$

La déf. : $M \circ e =_{déf.} \tilde{\mu} x . \langle M \mid x.e \rangle$, x n'apparait pas dans M ni dans e.

Vérification : $\langle V \mid \tilde{\mu} x . \langle M \mid x.e \rangle \rangle \to$ (puisque V est une valeur) $\langle M \mid x.e \rangle [V/x] \to \langle M \mid V.e \rangle$ car x frais, ni dans M ni dans e

Pour MN :

Déf. : $MN =_{déf.} \mu\alpha . \langle N \mid \tilde{\mu} x . \langle M \mid x.\alpha \rangle \rangle$

$$\dfrac{\dfrac{\dfrac{\Gamma, x : \; \vdash \Delta, \alpha : B}{\langle M \mid x.\alpha \rangle : (\Gamma, x : \; \vdash \Delta, \alpha : B)}}{\Gamma \mid \tilde{\mu} x . \langle M \mid x.\alpha \rangle : \; \vdash \Delta, \alpha : B}}{\dfrac{\langle N \mid \tilde{\mu} x . \langle M \mid x.\alpha \rangle \rangle : (\Gamma \vdash \alpha : B, \Delta)}{(\Gamma \vdash \mu\alpha . \langle N \mid \tilde{\mu} x . \langle M \mid x.\alpha \rangle \rangle : B)}}\text{ activat°}$$

Vérifier que ce terme a le type qui convient → exercice.

$\tilde{\mu} \approx$ let, c'est-à-dire qu'on évalue d'abord N

$\langle MN \mid e \rangle \to \langle N \mid \tilde{\mu} x . \langle M \mid x.e \rangle \rangle$ en une étape

$=_{déf} \langle N \mid M \circ e \rangle$ Donc le codage convient.

Pour λ :

$\lambda x . v$ correspond à l'intro à droite de la flèche. Or on n'a pas la flèche en connecteur, donc on reprend $(A \to B) =_{déf} \neg (A \wedge \neg B)$ (codage par valeur, $\wedge$ ou $\otimes$).

$\lambda x . v =_{déf.} \left(\left(\tilde{\mu} (x, \alpha^\circ) . \langle v \mid \alpha \rangle \right)^\bullet \right)^\circ$ ← résumé faux

$V.e$ correspond à l'intro à gauche de l'implication. On pose : $V.e = \tilde{\mu} \alpha^\circ . \langle (V, e^\circ)^\circ \mid \alpha \rangle$.

On introduit l'opérateur $\bullet$ la semaine prochaine.

$V.e$ est de type $A \to B$, donc $\neg (A \otimes \neg B)$: $\tilde{\mu} \alpha^\circ . \langle \underbrace{(V, e^\circ)^\circ}_{\otimes} \mid \alpha \rangle$ On retrouve la même granularité. ($\otimes = \wedge$)

↑ contexte, donc introduction

mar 23 fév.

- Calcul des séquents classique (focalisé = synt. L focalisant) intro. g. / d, séquents, multi ou mono-conclusion (int. dans ce cas).
- $\lambda\mu$-calcul (Parigot) : déduction naturelle + multi-conclusion.
- Déduction naturelle (int. ou classique)

$\langle MN \mid e \rangle \to \langle N \mid M \circ e \rangle$

$\langle V \mid M \circ e \rangle \to \langle M \mid V.e \rangle$

$\langle \lambda x . M \mid V.e \rangle \to \langle M [V/x] \mid e \rangle$

$\langle \mathcal{C} M \mid e \rangle \to \langle M \mid e^* . [\,] \rangle$

$\langle e^* \mid V.e' \rangle \to \langle V \mid e \rangle$

On code $\lambda x . v$ avec $\left(\left(\tilde{\mu} (x, \alpha)^\bullet . \underbrace{\langle v / \alpha \rangle}_{\text{désactivation}} \right) \right)^\bullet$ (!)

∈ à la 1ère syntaxe donnée, pas à la dernière.

On reformule la règle :

$$\dfrac{\Gamma, x : A \vdash v : B \mid}{\Gamma \vdash \lambda x . v : \underbrace{\neg (A \wedge \neg B)}_{A \to B}} ;$$

$A \to B$, codage par valeur de l'implication.

On a, pour l'intro. à gauche de connecteurs, $\tilde{\mu} \alpha^\circ$, $\tilde{\mu} (x_1, x_2)$ et $\tilde{\mu} (\text{inl}(x_1), \text{inr}(x_2))$ (ainsi que $\tilde{\mu} x . c$) ← synt. L focalisant

On avait proposé pour variante $e ::= x \mid \tilde{\mu} x . c \mid \tilde{\mu} q . c$

$q ::= \alpha^\circ \mid x \mid (q_1, q_2) \mid [q_1, q_2]$ ← contre-motifs

On regarde comment on définit $\tilde{\mu} (x, \alpha^\circ) . c$ en syntaxe focalisante :

$\tilde{\mu} (x, \alpha^\circ) . c = \text{let } y = \alpha^\circ \text{ in } \tilde{\mu} (x, y) . c$ ← mais ce n'est pas une commande ! Mais l'idée est bonne.

$= \tilde{\mu}(x, y) . (\text{let } y = \alpha^\circ \text{ in } c)$ est conforme, on le reformule :

$=_{déf.} \tilde{\mu}(x, y) . \langle y \mid \tilde{\mu} \alpha^\circ . c \rangle$

Dans la dérivation, on fait en même temps l'intro. de la $\neg$ sur B et la conjonction (à gauche) de A et $\neg B$.

$$\frac{\Gamma, x:A \vdash v:B \mid \qquad}{\langle v|\alpha\rangle : \Gamma, x:A \vdash \alpha:B} \quad \frac{\qquad \mid \alpha:B \vdash \alpha:B}{\qquad} \text{Ax.}$$

$$\frac{\Gamma \mid \tilde{\mu}(x,y).\langle y|\tilde{\mu}\alpha.\langle v|\alpha\rangle\rangle : A \wedge B \vdash}{\Gamma \vdash (\tilde{\mu}(x,y).\langle y|\tilde{\mu}\alpha.\langle v|\alpha\rangle\rangle)^{\bullet} : \neg(A \wedge B)} \quad = \Gamma \mid (\tilde{\mu}(x,\alpha^{\circ}).\langle v|\alpha\rangle) : A \wedge B \vdash \quad = \lambda x.v$$

Abréviation : $V^{\bullet} =_{\text{def}} \tilde{\mu}\alpha^{\circ}.\langle V^{\circ}|\alpha\rangle$ (à froid)

Justification du codage employé pour $\lambda x.v$:

Déf : $\forall e = (V, e^{\circ})^{\bullet} = \tilde{\mu}\alpha^{\circ}.\langle (V,e^{\circ})^{\circ}|\alpha\rangle$

On réécrit : $\langle (\tilde{\mu}(x,\alpha^{\circ}).\langle v|\alpha\rangle)^{\bullet} \mid (V,e^{\circ})^{\bullet}\rangle \to \langle (V,e^{\circ})^{\circ} \mid \tilde{\mu}(x.\alpha^{\circ}).\langle v|\alpha\rangle\rangle$ (car)

On a donc $\langle e^{\circ}|V^{\bullet}\rangle = \langle e^{\circ}|\tilde{\mu}\alpha^{\circ}.\langle V^{\circ}|\alpha\rangle\rangle \to \langle V^{\circ}|e\rangle$

• intervertir le rôle entre gauche et droite.

Ensuite, $\langle (V,e^{\circ})^{\circ} \mid \tilde{\mu}(x,\alpha^{\circ}).c\rangle \to c[V/x, e/\alpha]$

Donc $\langle (V,e^{\circ})^{\circ} | \tilde{\mu}(x,\alpha^{\circ}).\langle v|\alpha\rangle\rangle \to \langle v|\alpha\rangle[e/\alpha, V/x] \to \langle v[V/x]|e\rangle$

On détaille ce passage :

$\langle (V,e^{\circ})^{\circ} | \tilde{\mu}(x,y).\langle y^{\circ}|\tilde{\mu}\alpha^{\circ}.c\rangle\rangle \to \langle y^{\circ}|\tilde{\mu}\alpha^{\circ}.c\rangle[V/x, e^{\circ}/y]$

$\to \langle (e^{\circ})^{\circ} \mid \tilde{\mu}\alpha^{\circ}.c[V/x]\rangle$

$\to c[V/x, e/\alpha]$

Calcul des séquents, langage de preuves, système L focalisant (Lfoc) ← Logique, théorie de la démonstration.

λ-calcul, machine abstraite C.B.V, langage de programmation, ← (par ex. Caml) Programmation

$\lambda x.M : A \to B$ renvoit à une fonction de A dans B.

Curry-Howard : formaliser le lien entre ces deux mondes (via une traduction formelle).

Partiel, Mardi 3 mars 9-11h.

Documents autorisés : - Chap. 3 en Anglais
- Notes complémentaires
- Corps principal du poly (règles inversibles, styles, etc.)

Programme : → Quotients sur les preuves → Syst. L synthétique.
→ Logique linéaire (réseaux de preuves) (intro. à la logiq lin. auparavant.)

Quotient :

Jusqu'ici, on peut décomposer $\Gamma, (A_1 \wedge A_2) \wedge (A_3 \wedge A_4) \vdash \Delta$ de deux façons, alors que l'ordre des étapes à gauche n'a pas d'importance → On a trop de détails, on va faire en sorte de pouvoir passer directement de ce séquent à $\Gamma, A_1, A_2, A_3, A_4 \vdash \Delta$, grâce aux contre-motifs. On typera alors le séquent d'arrivé avec $\Gamma \mid \tilde{\mu}((x_1,x_2),(x_3,x_4)).c : (A_1 \wedge A_2) \wedge (A_3 \wedge A_4) \vdash \Delta$, on aura un terme unique pour deux preuves.

Logiq linéaire :

La contraction et l'affaiblissement ne sont plus libres. Les formules sur lesquelles on a le droit de faire l'un ou l'autre doivent être marquées par une modalité : ex : $\frac{\vdash \Gamma}{\vdash \Gamma, ?A}$ aff. ?A se lit "pourquoi pas A".

De même, $\frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A}$ ctr. Le duel de ? est ! (bien sûr), il signifie "autant de fois que vous voulez", c'est un théorème.

$\frac{!A_1, \dots, !A_n \vdash A}{!A_1, \dots, !A_n \vdash !A}$ ← on ne peut utiliser A qu'une fois
← on peut utiliser A autant qu'on veut.

Terminologie :

Présentation focalisée de LK a déjà deux conjonctions et deux disjonctions (à droite / à gauche), réversible ou pas. On introduit un nom différent pour le $\vee$ irréversible ($\oplus$) et pour le $\vee$ réversible ($\bar{\wedge}$, dont le dual, $\wedge$, est réversible à droite).

On n'a pas, ici, le dual de $\wedge = \vee$ ($\bar{\wedge} \neq \vee$).

$\vee$ irréversible à d. $\oplus$ plus ; $\wedge$ irréversible à d. $\otimes$ tenseur ← apparaissent dans Lfoc.

(dual $\otimes$ $\vee$ réversible à d. = $\wedge$ réversible à g. $\parr$ par ; $\wedge$ réversible à d. & with (dual $\oplus$ $\vee$ réversible à g. = $\parr$)

Dans tout ce qu'on a fait pour Lfoc, on rebaptise $\wedge$ $\otimes$ et $\vee$ $\oplus$ (ainsi que $\neg$ $\neg^*$), A en P

$P \otimes Q \vdash \neg^* R$

$\vdash \overline{P \otimes Q}, \neg^* R$

$\vdash \bar{P} \parr \bar{Q}, \neg^* R$

$A ::= X \mid A \wedge A \mid A \vee B \mid \neg A$

$P ::= X \mid P \otimes P \mid P \oplus P \mid \neg^+ P$ ← positifs, les seuls qu'on a utilisé.

$N ::= X^{\perp} \mid N \parr N \mid N \& N \mid \neg^- N$

On a une dualité grâce aux lois de De Morgan : $\begin{cases} \overline{X} = X^\perp \\ \overline{P \otimes Q} = \overline{P} \parr \overline{Q} \\ \overline{P \oplus Q} = \overline{P} \,\&\, \overline{Q} \end{cases}$

↳ On peut ramener les séquents en monolatère : $\Gamma \vdash \Delta \rightsquigarrow \vdash \overline{\Gamma}, \Delta$, on fait alors apparaître les formules négatives.

ex. : règle pour $\otimes$ à gauche : $\dfrac{\Gamma, \overbrace{P, Q}^{\text{formules positives}} \vdash \Delta}{\Gamma, P \otimes Q \vdash \Delta} \rightsquigarrow \dfrac{\vdash M, N, \Delta}{\vdash M \parr N, \Delta}$ pour $\overline{P} = M$ et $\overline{Q} = N$, M et N étant négatives.

Les règle à gauche deviennent règles d'introductions du connecteur dual à droite.

$\vee$ réversible à gauche : $\dfrac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta}$ $\wedge$ irréversible à droite. $\dfrac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta}$

↳ irréversibilité du $\wedge$ à droite ne vient pas de la règle mais de la recherche de preuve (focalisation).
Décomposer une formule à gauche peut se faire n'importe quand, juste une fois, alors que la décomposition à droite doit être obtenue.

$\otimes$ tenseur : $\dfrac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2 \vdash B, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \otimes B, \Delta_1, \Delta_2}$ Règle réversible sera : $\dfrac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \parr B, \Delta}$

↳ on doit prendre la décision de couper en deux les conclusions. → Choix.

On peut résumer LK en termes de réversibilité / irréversibilité sans considérer les plus de linéarité que pour LL, mais il faut dans ce cas se doter d'un protocole de recherche de preuves.

$\&$, réversible "additif" : $\dfrac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \& B, \Delta}$

$\otimes$, irréversible "multiplicatif" : $\dfrac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2 \vdash B, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \otimes B, \Delta_1, \Delta_2}$

$\parr$, réversible "multiplicatif" : $\dfrac{\vdash \Gamma, A, B}{\vdash \Gamma, A \parr B}$

$\oplus$, irréversible "additif" : $\dfrac{\vdash \Gamma, A}{\vdash \Gamma, A \oplus B}$

$!(A \times B)$ isomorphe à $(!A) \otimes (!B)$

Terminologie en L.L. : - multiplicatif / additif $\otimes$ $\parr$ / $\oplus$ $\&$ (classes stables par dualité).
- exponentiel ! et ? non duaux ?

explication ↳ $e^{m+n} = e^m \times e^n$

↳ modèles des espaces de cohérence.

Dans LK l'exponentiel était caché : $\neg P$ correspond à $!\overline{P}$.

mer. 24 fév.

Question du quotient sur les preuves

On nomme autrement $A, B \ldots \rightsquigarrow P, Q ::= X \mid P \otimes Q \mid P \oplus Q \mid \neg P$ (∧ ∨ ¬)

$\dfrac{c : (\Gamma, x_1 : P_1, x_2 : P_2, x_3 : P_3, x_4 : P_4 \vdash \Delta)}{\Gamma \mid e : (P_1 \otimes P_2) \otimes (P_3 \otimes P_4) \vdash \Delta}$ On souhaite faire abstraction des étapes.

On voudrait avoir le droit d'écrire $e = \tilde{\mu}((x_1, x_2), (x_3, x_4)).c$, soit en l'ajoutant à la syntaxe soit en le codant.

Deux façons de coder : - $e_1 = \tilde{\mu}(y, y).\langle y^\circ \mid \tilde{\mu}(x_3, x_4).\langle y^\circ \mid \tilde{\mu}(x_1, x_2).c \rangle\rangle$

↳ $\dfrac{\Gamma, P_1, \ldots, P_4 \vdash \Delta}{\Gamma, P_1, P_2, P_3 \otimes P_4 \vdash \Delta}$ $\dfrac{}{\Gamma, P_1 \otimes P_2, P_3 \otimes P_4 \vdash \Delta}$ $\dfrac{}{\Gamma, (P_1 \otimes P_2) \otimes (P_3 \otimes P_4) \vdash \Delta}$

- $e_2 = \tilde{\mu}(y, y).\langle y^\circ \mid \tilde{\mu}(x_3, x_4).\langle y^\circ \mid \tilde{\mu}(x_1, x_2).c \rangle\rangle$ renverse l'ordre de y et y.

↳ 3 descriptions possibles de la même preuve, la 1ère va d'un seul coup. → but : décrire illégales en elles-mêmes, on ne veut pas différencier ces deux termes. → On les rend égaux à e.

Second problème : formaliser la focalisation forte.

Rappel : * Phase droite : - Choix de P à droite $\Gamma \vdash P, \Delta$, $P \leftarrow$ on garde un copie de la formule focalisée.
- Décomposition de P jusqu'à atteindre • une négation $\Gamma \vdash \neg Q, \Delta \rightarrow \Gamma, Q \vdash \Delta$, passe à droite
• un axiome (atomique par la focalisation forte)

* Phase gauche : Décomposer les formules à gauche dans l'ordre que l'on veut et ceci tant que l'on veut.

↳ restriction à « jusqu'à ce que ce ne soit plus possible », ainsi on va à gauche plus que des formules atomiques lorsque cette phase est finie. (Si formule avec négation, on la passe à droite mais on continue la décomposition à gauche si possible)

↳ technique complète pour LK, recherche de preuve plus simple et plus efficace

L'ancienne méthode est plus efficace pour le calcul sur les preuves → Programmation fonctionnelle (vs prog. logique).

À la fin d'une phase gauche, le séquent est de type $x_1 \dots x_n \vdash \Delta$ pour x_i des atomes, on note Ξ un multi-ensemble d'atomes.

Nouvelles règles : $\dfrac{}{\Xi, x:X \vdash x:X ; \Delta}\ Ax.$ Δ peut contenir des formules complexes.

ex : passe à droite si P_3 est atomique $\left(\dfrac{x_1 \dots x_n, P_3 \vdash P_1 \otimes P_2, \neg P}{x_1 \dots x_n \vdash P_3 \multimap P_1 \otimes P_2, \neg P}\right.$ $\dfrac{x_1, \dots, x_n \vdash P_4, P_1 \otimes P_2, \neg P}{x_1 \dots x_n, \neg P_4 \vdash P_1 \otimes P_2, \neg P}$ décon pour $\neg P_4$ immédiatement, puis phase droite (on peut refocaliser sur $\neg P$).

pour $P = P_3 \oplus \neg P_4$, on passe immédiatement à gauche pour décomposer ce qui vient d'y être introduit et n'est pas atomique.

$\left(\dfrac{\dfrac{x_1, \dots x_n, P_3 \oplus \neg P_4 \vdash P_1 \otimes P_2, \neg P}{x_1, \dots, x_n \vdash \neg P ; P_1 \otimes P_2, \neg P}}{x_1, \dots, x_n \vdash \neg P, P_1 \otimes P_2}\right.$ ↑ choix de la focalisation, on copie la formule.

Syntaxe intermédiaire : → bien adaptée à la focalisation, mais mauvaises propriétés

$c ::= \langle v \mid e \rangle \mid [c^{q,q} c]$

$v ::= V^0 \mid \mu\alpha.c$ $\qquad \Xi \vdash v : P \mid \Delta$

$V ::= x \mid (V_1, V_2) \mid \mathrm{inl}(V_1) \mid \mathrm{inr}(V_2) \mid e^\bullet$ $\qquad \Xi \vdash V : P ; \Delta$

$e ::= \alpha \mid \tilde{\mu} q.c$ $\qquad + \ q ::= \alpha^\bullet \mid x \mid (q,q) \mid [q,q]$ (contre-motifs)

V $\dfrac{\Xi \vdash V_1 : P_1 ; \Delta \qquad \Xi \vdash V_2 : P_2 ; \Delta}{\Xi \vdash (V_1, V_2) : P_1 \otimes P_2 ; \Delta}$ $\qquad \uparrow \dfrac{\Xi \mid e : P \vdash \Delta}{\Xi \vdash e^\bullet : \neg P ; \Delta}$ → par toutes les règles relatives à V, on impose que le contexte à gauche soit atomique

On intervertit la phase droite, qui aurait été une phase gauche qui décompose P.

v → pas de changements, on atomise les contextes à gauche.

e $::= \alpha \mid \tilde{\mu} x.c \mid \tilde{\mu}\alpha^\bullet.c \mid \tilde{\mu}(x_1, x_2).c \mid \tilde{\mu}(\mathrm{inl}(x_1).c_1, \mathrm{inr}(x_2).c_2)$ ← Ancienne syntaxe.

↳ nouvelle syntaxe utilise q pour quotienter les preuves.

Rem : ⇒ les variables n'ont plus que des types atomiques, donc les termes en deux sont exclus : dans $\langle y^a \mid \tilde{\mu}(x_1, x_2).c \rangle$, il faudrait que y ait le type tenseur → impossible.

Par contre, un typage entrainé e, c'est un terme des nuls, bien typé. → tendance à la preuve plus canonique.

$\Xi \mid \alpha : P \vdash \alpha : P, \Delta$

Anciennes règles : $\dfrac{c : (x_1 : P_1, x_2 : P_2, \Gamma \vdash \Delta)}{\Gamma \mid \tilde{\mu}(x_1, x_2).c : P_1 \otimes P_2 \vdash \Delta} \rightsquigarrow$ Nouvelles règles $\dfrac{c : (x_1 : P_1, x_2 : P_2, \Gamma \vdash \Delta)}{c : ((x_1, x_2) : P_1 \otimes P_2, \Gamma \vdash \Delta)}$

↳ les phases gauches n'agissent qu'au niveau des commandes.

pas ?

$\dfrac{c : (\Gamma \vdash \alpha : P, \Delta)}{\Gamma \mid \tilde{\mu}\alpha^\bullet.c : \neg^\perp P \vdash \Delta} \rightsquigarrow \dfrac{c : (\Gamma \vdash \alpha : P, \Delta)}{c : (\Gamma, \alpha^\bullet : \neg P \vdash \Delta)}$?

↳ très proche dans l'esprit de l'ancienne, le $\tilde{\mu}$ servant à activer.

$\dfrac{c_1 : (\Gamma, x_1 : P_1 \vdash \Delta) \qquad c_2 : (\Gamma, x_2 : P_2 \vdash \Delta)}{\Gamma \mid \tilde{\mu}(\mathrm{inl}(x_1).c_1 \mid \mathrm{inr}(x_2).c_2) : P_1 \oplus P_2 \vdash \Delta} \rightsquigarrow \dfrac{c_1 : (\Gamma, q_1 : P_1 \vdash \Delta) \qquad c_2 : (\Gamma, q_2 : P_2 \vdash \Delta)}{[c_1^{q_1,q_2}, c_2] : (\Gamma, [q_1, q_2] : P_1 \oplus P_2 \vdash \Delta)}$

↳ complication des commandes

↳ distinction : $c ::= \langle v \mid e \rangle$ par souci de typage : $c : (\Xi \vdash \Delta)$

$\mathcal{C} ::= c \mid [\mathcal{C}^{q,q} \mathcal{C}]$ $\qquad \mathcal{C} : (\Gamma \vdash \Delta)$ ← décomposition progressive à gauche.

↳ on fait $\tilde{\mu} q.\mathcal{C}$ et non $\tilde{\mu} q.c$.

$\dfrac{c : (\Gamma, x : P \vdash \Delta)}{\Gamma \mid \tilde{\mu} x.c : P \vdash \Delta} \rightsquigarrow \dfrac{\mathcal{C} : (\Xi, q : P \vdash \Delta)}{(\Xi \mid \tilde{\mu} q.\mathcal{C} : P \vdash \Delta)}$

↳ relation très forte entre contre-motifs et termes : $(q,q) \to \otimes$, $[q,q] \to \oplus$ et $\alpha^\bullet \to \neg$.

exemple : $P = X \otimes (Y \otimes \neg Q)$

Si P est à gauche (on ne note pas les contextes) la commande reste identique.

$\dfrac{c : (x : X, y : Y \vdash \alpha : Q)}{c : (x : X, y : Y, \alpha^\bullet : \neg Q \vdash)}$

$\dfrac{c : (x : X, (y, \alpha^\bullet) : Y \otimes \neg Q \vdash)}{c : ((x, (y, \alpha^\bullet)) : X \otimes (Y \otimes \neg Q) \vdash)}$

Un terme ne décrit pas une preuve, mais un terme et son séquent décoré (par des contre-motifs) permet de reconstruire la preuve d'une manière unique.

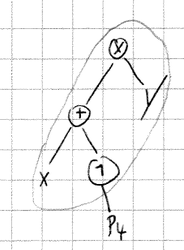
↳ les contre-motifs sont une annotation de l'arbre de preuve.

⇒ Lorsque $q : P$ apparaît dans un séquent prouvable du système intermédiaire, alors q décrit la structure de tête positive de P. (à gauche)

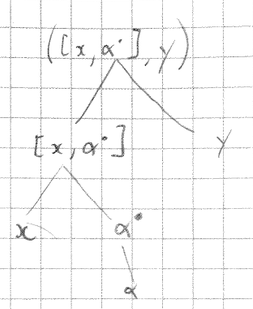
On n'aura jamais $(q_1, q_2) : X$.

Si $P = X$, $x : X$

$P = P_1 \otimes P_2$



Arbre de tête positif.
Bloc (positif) de tête ⊗



Lacune de ce système intermédiaire du point de vue du quotient des preuves.

Les ⊗ ne posent pas de problème, les ⊕ en posent.

Pour $i = 1,2$ et $j = 3,4$. $c_{ij} : (\Gamma, x_i : P_i, y_j : P_j \vdash \Delta)$

P_i et P_j atomiques

$$\frac{c_{13} : (P_1, Q_3 \vdash) \quad c_{14} : (P_1, Q_4 \vdash) \qquad c_{23} : (P_2, Q_3 \vdash) \quad c_{24} : (P_2, Q_4 \vdash)}{[c_{13}\ c_{14}] : ([\ /3\ /4] : P_1, Q_3 \oplus Q_4 \vdash) \qquad P_2, Q_3 \oplus Q_4 \vdash}$$

$$\frac{P_1 \oplus P_2, Q_3 \oplus Q_4 \vdash}{(P_1 \oplus P_2) \otimes (Q_3 \oplus Q_4) \vdash}$$

on décompose $Q_3 \oplus Q_4$, puis $P_1 \oplus P_2$.

↳ Deux preuves que l'on voudrait identifier : $\star [[c_{13}, c_{14}], [c_{23}, c_{24}]]$

ont deux écritures différentes : $\star [[c_{13}, c_{23}], [c_{14}, c_{24}]]$

↳ $[\zeta^{q_1, q_2}\ \zeta]$ nous a été pour quotienter les contre-motifs pour bien leur rôle, les commandes posent problème.

On ajoute l'équivalent à droite des contre-motifs, les motifs (ρ).

Contre-motif unique $([x_1, x_2], [y_1, y_2])\ (P_1 \oplus P_2) \otimes (Q_1 \oplus Q_2)$, mais deux commandes.

On perd ensemble les 4 commandes (c_{13}, c_{23}, c_{14} et c_{13}) de deux façons différentes.

On remplace $V := x \mid (V_1, V_2) \mid \mathrm{inl}(V_1) \mid \mathrm{inr}(V_2) \mid e^\cdot$ par : $V := \rho\{\tilde{V}_i \mid i \in \rho\}$ joue le rôle de σ, substitution.

valeurs élémentaires → $\tilde{V} := x \mid e^\cdot$ (fin de phase droite)

forme / motifs → $\rho := x \mid (\rho, \rho) \mid \mathrm{inl}(\rho) \mid \mathrm{inr}(\rho) \mid \alpha^\cdot$

↑ mémorise ce qui se passe lors des phases droites.

$p = ((x, y), z)$ est un motif qui filtre $t = ((3, 7+5), 12)$

$t = \sigma(p)$ où σ est une substitution.

Règles : $\dfrac{}{x \in x}$ $\dfrac{x \in \rho_1}{x \in (\rho_1, \rho_2)}$ $\dfrac{x \in \rho_2}{x \in (\rho_1, \rho_2)}$ $\dfrac{x \in \rho}{x \in \mathrm{inl}(\rho)}$ $\dfrac{x \in \rho}{x \in \mathrm{inr}(\rho)}$ $\dfrac{}{\alpha^\cdot \in \alpha^\cdot}$

Sa formule : "p joue contre q" q est orthogonal à p (noté ⊥) → nécessite que p et q aient le même type.

$$\frac{}{x \perp x} \qquad \frac{}{\alpha^\cdot \perp \alpha^\cdot} \qquad \frac{q_1 \perp p_1 \quad q_2 \perp p_2}{(q_1, q_2) \perp (p_1, p_2)} \qquad \frac{q_1 \perp p_1}{[q_1, q_2] \perp \mathrm{inl}(p_1)} \qquad \frac{q_2 \perp p_2}{[q_1, q_2] \perp \mathrm{inr}(p_2)}$$

par la de l'élimination d'une coupure sur $P_1 \oplus P_2$ vers une coupure sur P_1.

$\mathrm{inl}(p_1)$ est forcément de type $p \oplus q$, donc ~~$x \neq \mathrm{inl}(p_1)$~~, car x est de type X (atomique)

Orthogonalité : étape relationnelle entre motifs et contre-motifs engendrée par ces règles.

Pour approfondir : The duality of computation under focus, Curien & Munch.

↳ On arrive au système L synthétique

↳ Terme de Girard : dans cette logique polarisée, un connecteur est synthétique s'il regroupe un paquet maximal de connecteurs de même polarité.

Nouvelle syntaxe : $c := \langle v \mid e \rangle$

$v := V^\circ \mid \mu\alpha.c$

$V := p\{\tilde{V}_i \mid i \in p\}$

$\tilde{V} := x \mid e^\cdot$

$e := \alpha \mid \tilde{\mu} q \cdot \{c_p \mid q \perp p\}$ ← contrôle de ce qui est dans cet ensemble par l'orthogonalité.

p et q comme avant : $p := x \mid (p, p) \mid \mathrm{inl}(p) \mid \mathrm{inr}(p) \mid \alpha^\cdot$

$q := x \mid (q, q) \mid [q, q] \mid \alpha^\cdot$

Théo. : Tout séquent prouvable en LK admet une preuve que l'on peut écrire dans le système L synthétique (on reste complet).

Pour le problème précédent, on peut ici bien en quotienter en identifiant les deux preuves avec $\tilde{\mu} q. \{c_{13}, c_{14}, c_{23}, c_{24}\}$.

Règles de calcul : $\star \langle \mu\alpha.c \mid e \rangle \longrightarrow c[e/\alpha]$

$\langle p\{\tilde{V}_i \mid i \in p\}^\circ \mid \tilde{\mu} q. \{c_p \mid q \perp p\} \rangle \longrightarrow c_p[\ ?$

autre formulation : Par la loi de typage, $p \perp q$.

$\star \langle p\{y/x \ldots e^\cdot/\alpha^\cdot\} \mid \tilde{\mu} q. \{c_p \mid q \perp p\} \rangle \longrightarrow c_p[\ldots/x \ldots e/\alpha]$ Seule règle qui gère tout

Partie contrôle : déclenchée par $\tilde{\mu}$ contre une valeur. Partie logique réglée par orthogonalité.

Commutations prises en charge par substitution.

→ Nouvelle programmation fait son apparition : programmation par objets. Permet sélection d'un champ.
On aurait pu écrire $\mu q.\ (c\ p \mid q \perp p)$ sous la forme $\mu q.\ (p := c\ p)$, le p sélectionne alors un champ.

mar. 2 mars → exam.
mer. 3 mars

Logique linéaire

Support de cours → Introduct° to linear logic & ludics (I & II)

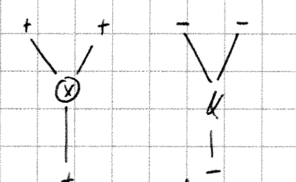
Sémantique dénotationnelle

↳ Modèle des espaces cohérents (précède logiq linéaire en réalité) → modèle du système F (λ-calcul du 2nd ordre). Invent° par Girard dans sa thèse d'état (73), repris en 86.

Syntaxe logique linéaire : $A ::= X \mid \bar{X} \mid A \otimes A \mid A \& A \mid A \oplus A \mid A ⅋ A \mid !A \mid ?A \mid 1 \mid \top \mid \bot \mid 0$

(atome et son dual ; tenseur, with : conjonctions ; plus, par : disjonction ; bien sûr, pourquoi pas : exponentiels ; un, top : vrai ; bottom, zéro : faux)

↳ on ne regarde plus la polarité des formules.



$A \otimes (B \& C)$ → serait mal formé en logique linéaire polarisée (Olivier Laurent).
(⊕tif au-dessus de $B \& C$, ⊕tif sous $A\otimes$)

$\sim \vdash \sim \ \Rightarrow\ \vdash \sim^{\perp}, \sim$ — On n'utilise plus qu'un côté du séquent, qui contient donc des formules des deux polarités ($\vdash \Gamma$).

$\Gamma \vdash \Delta \Leftrightarrow \vdash \bar{\Gamma}, \Delta$

Définition : Pour toute formule A de la logique linéaire, on définit sa duale, originellement notée $A^{\perp}$, puis $A^{\sim}$, mais on adoptera $\bar{A}$. $\bar{\bar{A}} = A$

- $\bar{X} = \bar{X}$ et $\bar{\bar{X}} = X$
- $\overline{A \otimes A} = \bar{A} ⅋ \bar{A}$
- $\overline{A \oplus A} = \bar{A} \& \bar{A}$
- $\overline{!A} = ?\bar{A}$
- $\bar{1} = \bot$
- $\bar{\top} = 0$

> MLL : Fragment multiplicatif : ⊗, ⅋, 1, ⊥
> ALL : additif : ⊕, &, 0, ⊤
> MALL : additif multiplicatif : tout sauf ? et !
> LL : Logique linéaire : MALL + !, ?

Tous ces fragments sont clos par duaux.

La négation n'est pas un opérateur primitif, elle n'apparaît que pour les atomes et est admissible : on peut calculer $\bar{A}$ à partir de A, par induction (et non par une macro).

Règles : Pas d'aff. et de contraction sans ! et ? (MALL et MLL n'en ont pas : $\Gamma, A \vdash A, \Delta$ est interdit).

$$\frac{}{\vdash \bar{A}, A}\ Ax.$$, on pourrait même se restreindre aux axiomes atomiques : $$\frac{}{\vdash X, \bar{X}}\ Ax.at.$$

Conjonctions :
$$\frac{\vdash A, \Gamma_1 \quad \vdash B, \Gamma_2}{\vdash A \otimes B, \Gamma_1, \Gamma_2} \qquad \frac{\vdash A, \Gamma \quad \vdash B, \Gamma}{\vdash A \& B, \Gamma}$$
Style multiplicatif — Style additif. On peut dériver une règle avec l'autre à l'aide de l'aff. ou de la contr.

Disjonctions :
$$\frac{\vdash \Gamma, A}{\vdash \Gamma, A \oplus B} \qquad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \oplus B} \qquad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A ⅋ B}$$

La règle pour 1 correspond au cas 0aire du tenseur (⊗) : $\frac{}{\vdash 1}$
⊤ du with (&) : $\frac{}{\vdash \Gamma, \top}$
0 du plus (⊕) : le 0 n'a pas de règle
⊥ du par (⅋) : $\frac{\vdash \Gamma}{\vdash \Gamma, \bot}$

Pour LL, on ajoute :
$$\frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}\ \text{dereliction} \quad \text{et} \quad \frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A}\ \text{contraction}$$
$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A}\ \text{affaiblissement} \quad \text{et} \quad \frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A}\ \text{promotion} \qquad \left(\begin{array}{l}\Gamma = A_1, \dots, A_n \\ ?\Gamma = ?A_1, \dots, ?A_n\end{array}\right)$$

ex :
$$\frac{\dfrac{\dfrac{\vdash \Gamma, A}{\vdash \Gamma, A, ?A}\ \text{aff.}}{\vdash \Gamma, ?A, ?A}\ \text{der.}}{\vdash \Gamma, ?A}\ \text{contr.}$$) dereliction.

La dérélicition : si on a une copie de A, on a au moins une copie de A ($!A$).
$?A$ → on a n copies de A ($n \in [0, \dots \infty]$). Mais combien ?
$!A$ → A autant de fois que l'on veut.

On regarde ces deux règles dans les séquents à deux côtés : $\frac{A \vdash B}{!A \vdash B}$ dereliction $\frac{\vdash \Gamma}{!A \vdash \Gamma}$ aff.

Avec une infinité d'euros je peux acheter une cigarette $\frac{!\Gamma \vdash A}{!\Gamma \vdash !A}$ promotion
" " je peux acheter une infinité de cigarettes

Illustration (due à Yves Lafont) de MALL pour illustrer l'idée de formule comme ressource.

Dans un restaurant : Menu 17€
- Quiche ou salade
- Poulet ou poisson*
- Banane ou "Surprise du chef"**

Pas de dessert en fait. (* = Profiteroles ou tarte tatin selon l'humeur)

* = Sole ou truite selon arrivages

$17€ \vdash (Qui. \,\&\, Sa.) \otimes (Pou \,\&\, (So. \oplus Trui.)) \otimes D$ décrit les choix du client et du gérant du restau.

Gérant : $\dfrac{5€ \vdash Q \,\&\, S}{5€ \vdash Q} \quad \dfrac{5€ \vdash S}{}$ $\qquad$ $\dfrac{8€ \vdash P \,\&\, (S \oplus T)}{8€ \vdash P} \quad \dfrac{8€ \vdash S \oplus T}{8€ \vdash S}$ $\qquad$ $4€ \vdash D$

↑ ce jour là il y a de la sole, il y a livraison.

Le client choisit un chemin dans la preuve, le gérant fait la preuve.

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \oplus B} \quad \frac{\Gamma \vdash B}{\Gamma \vdash A \oplus B}$$

Choix de la règle laissée à celui qui construit la preuve (le gérant).
Pas de choix au niveau de l'hypothèse.

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

Choix de l'hypothèse, mais pas de la règle.
Choix laissé à celui qui teste la règle (le client).

→ Interprétation de la logique en termes de jeux.

Pour le par : $B \vdash A \rightsquigarrow \dfrac{\vdash \bar{B}, A}{\vdash \bar{B} \parr A}$

Quant à l'implication : $A \multimap B$ implication linéaire (CBV)

$A \Rightarrow B$ " intuitionniste / classique (CBN)

$\bar{A} \parr B \equiv A \multimap B$

$!A \multimap B \equiv A \Rightarrow B$

On peut voir $!A$ comme $A \otimes A \otimes A \ldots$ (nb infini de fois)

Sémantique dénotationnelle de la logique linéaire

Sens (vient de la linguistique) → sémantique d'un programme = dire ce qu'il fait.

(logique de Hoare) :
- sémantique opérationnelle : dire ce qu'il fait en décrivant ses règles de programme, comment il s'exécute.
- " axiomatique : quelles propriétés a le programme (approximation de ce qu'il fait).
- " dénotationnelle : un programme est un point dans un espace (interprétation géométrique) / élément dans un ensemble (structuré)

Structure sur ces ensembles : $\leq$ (relation d'ordre), topologie, espace vectoriel.

La programmation, contrairement à la logique, peut considérer des programmes incomplets.

↑ information partielle (ordre partiel, non-total)

ex : factorielle : let rec fact n = if n = 0 alors 1 / n × fac (n-1)) → la fonction factorielle est le ⊕pt point fixe d'une fonctionnelle de type F : (nat→nat) → (nat→nat) (Scott).

Déf. : x est un point fixe de $f : D \to D$ si $f(x) = x$.

Tarski : dans tout treillis complet (contient sup & inf finis & infinis) toute fonction monotone a un plus grand point fixe et un plus petit point fixe.

Kleene : (cas particulier du théo. précédent) dans tout ensemble partiellement ordonné [qui est complet pour les parties dirigées, toute fonction f telle que $f(\sup(x_n)) = \sup(f(x_n))$] dans lequel toute suite croissante a une borne supérieure, toute fonction $f : D \to D$ continue (au sens précédent) a un plus petit point fixe.

Les théo. proches mais aux démonstrations très différentes.

ex : fact (3) = 3 × fact (2)
= 3 × 2 × fact (1)
= 3 × 2 × 1 × fact (0) ← connaître fact(0) ne demande qu'une partie de la déf. de fact.
= 3 × 2 × 1 × 1
= 6

on fait le type de la fonctionnelle qui est
F : (nat → nat) → (nat → nat)
fct° partielle, fct° totale.

ϕ : nat → nat , définie nulle part

$fact_0(0) = 1$, et $fact_0$ indéfinie partout ailleurs.

$fact_1(0) = 1$, $fact_1(1) = 1$, indéfinie partout ailleurs

$fact_3(0) = 1$, $fact_3(1) = 1$, $fact_3(2) = 2$ et $fact_3(3) = 6$, indéfinie partout ailleurs

↳ $\phi \subseteq fact_0 \subseteq fact_1 \subseteq fact_2 \subseteq fact_3$.

$fact_0 = \bot$, défini nulle part.

$fact_n = F(fact_{n-1})$, on applique façon uniforme F.

$f \subseteq g$ nat → nat si $\forall x$ $f(x)$ défini → $g(x) = f(x)$.

$[\![fact]\!] \in (nat \to nat)$ l'interprétation de fact appartient à un ensemble ordonné complet, qui est le sup. des $fact_i$. → suite croissante de fonctions partielles, dont la limite est la fonction factorielle.

Modèle de la logique linéaire :

formule → espace cohérent (graphe) (type) ≈ nat → nat

preuve → un élément (une clique) de cet espace. (programme) ≈ une fonction partielle de nat dans nat

Espace cohérent : (non pas un ensemble mais la description d'un ensemble)

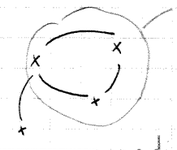
$(E, \frown)$ les éléments de E sont des événements, $\frown$ est une relation symétrique & réflexive.

(Un espace ≈ un graphe, un événement ≈ nœuds) $\subseteq E \times E$

clique : Soit $(E, \frown)$ un espace cohérent, une clique x de E est un événement sous-"ensemble" de E tel que

$\forall e_1, e_2 \in x$, $e_1 \frown e_2$.

$\frown$ est une relation de cohérence.



mardi 9 mars. Correct du partiel (voir fin + impression) pr dimanche

Théorème du point fixe de Kleene :

Hypothèse : 1) $(D, \leqslant)$, pour $\leqslant$ un ordre partiel tel que $\forall x_1 \leqslant \dots \leqslant x_n \leqslant \dots$ $\bigvee x_i$ existe (plus petit majorant).

et $\exists \bot \in D$ t.q. $\forall x \in D$, $\bot \leqslant x$

2) $f : D \to D$ t.q. $\forall x_1 \leqslant \dots \leqslant x_n \leqslant \dots$

a) $x \leqslant y \to f(x) \leqslant f(y)$

b) $f(\bigvee x_i) = \bigvee f(x_i)$ (continuité)

On a alors : $\bigvee f^n(\bot) = x_0$, le sup. des puissances n de $\bot$ est le plus petit point fixe de f.

$\bot$ est le ⊕pt d'énergie, donc $\bot \leqslant f(\bot)$, et la fonction est croissante, donc $f(\bot) \leqslant f(f(\bot))$, etc., et cette suite croissante a un ⊕ petit majorant (par hypothèse). Exo → montrer que c'est le ⊕pt point fixe.

↳ def fonctorielle

Espace de cohérence : $(E, \frown)$, E étant un ensemble d'événements, $\frown$ une relation de cohérence (symétrique et réflexive sur E).

Configuration (clique) : $x \subseteq E$ tel que $\forall e_1, e_2 \in x$, $e_1 \frown e_2$.

$D(E)$ = ensemble des configurations de E, ordonné par inclusion.

↳ L'ordre partiel qui nous intéresse est $(D(E), \subseteq)$

$(E, \frown)$ est une présentation concrète d'un ordre partiel.

L'interprétation de A, $[\![A]\!] = (E, \frown)$

$\pi : \vdash A$ ⇝ $[\![\pi]\!] = D([\![A]\!])$

Si π se réduit en π' par élimination des coupures, $[\![\pi]\!] = [\![\pi']\!]$, la réduction est invariante de la sémantique.

Exemple : • Booléens, $E = \{V, F\}$, $V \frown V$, $F \frown F$ ($V \not\frown F$)

$D(E) = \{\phi, \{V\}, \{F\}\}$ $\phi \subseteq \{V\}$ et $\phi \subseteq \{F\}$ ⇝ V, F ↓ ⊥ } domaine plat :

• Entiers naturels : $E = \{0, 1, \dots\}$, $i \frown i$ pour tout $i \in \mathbb{N}$

$D(E) = \{\phi, \{0\}, \{1\}, \dots\}$ 0 1 2 3 ... ↓ ⊥

• Nat × Bool $(n, b) \in nat \times bool$ pour $n \in nat$, $b \in bool$, par exemple $(4, F)$.

nat × bool → nat & bool.

On fait le with de deux structures d'éléments : $E \& E'$ $[\![A \& B]\!] = [\![A]\!] \&$ (symétrique) $[\![B]\!]$

(sémantique, c'est sa définition)

Événements : $E \cup E'$ (union disjointe) $= \{e.1 \mid e \in E\} \cup \{e'.2 \mid e' \in E'\}$, on tagge les événements pour ne pas les mélanger.

$e_1 \underset{E}{\frown} e_2$ pour $e_i \in E$, $e_1 \cdot 1 \frown e_2 \cdot 1$ dans $E \& E'$.

$e'_1 \underset{E'}{\frown} e'_2$ pour $e_i \in E'$, $e'_1 \cdot 2 \underset{E \& E'}{\frown} e'_2 \cdot 2$

e.1 est toujours cohérent avec e'.2. $e.1 \underset{E \& E'}{\frown} e'.2$

↳ relation de cohérence de $E \cup E'$ est la plus petite qui convient à ce qui précède.

rem : $\#(E \& E') = \#E + \#E'$, car c'est une union disjointe (cardinal de $E = \#E$).

↑ le connecteur est additif !

$$\frac{e_1 \frown_E e_2}{e_1 \cdot 1 \frown_{E \& E'} e_2 \cdot 1} \qquad \frac{e'_1 \frown_{E'} e'_2}{e'_1 \cdot 2 \frown_{E \& E'} e'_2 \cdot 2}$$

$$\frac{}{e.1 \frown_{E \& E'} e'.2}$$

$\mathcal{D}(\text{nat} \,\&\, \text{bool}) = \{ \emptyset, \dots, \{n.1\}, \dots, \{V.2\}, \{F.2\}, \{n.1, V.2\}, \{n.1, .2\} \}$.

En couples : $\mathcal{D}(\text{nat} \,\&\, \text{bool}) = \{ (\bot,\bot), (n,\bot), (\bot,V), (\bot,F), (n,V), (n,F) \}$.

D'où $\mathcal{D}(\text{nat} \,\&\, \text{bool}) = \mathcal{D}(\text{nat}) \times \mathcal{D}(\text{bool})$, le produit cartésien ensembliste, qui est lui-même ordonné. Produit cartésien dans une certaine catégorie (les esp. partiellement ordonnés).

mer. 10 mars (j'ai trouvé un étage avec I. ? N.)

$(E, \bot)$ ↝ formules $\qquad A ::= A \otimes A \mid A \,\&\, A \mid A \oplus A \mid A \parr A \mid\, !A \mid ?A \mid 0 \mid 1 \mid \bot \mid \top$

$\mathcal{D}(E)$ ↝ preuves

$A \multimap B = \overline{A} \parr B = \overline{A \otimes \overline{B}}$ implication linéaire

$A \to B = (!A) \multimap B = ?\overline{A} \otimes B$ implication

Interprétation des formules

& • $[\![A \,\&\, B]\!]$ = produit de $[\![A]\!]$ et de $[\![B]\!]$, $(E, \frown) \,\&\, (E', \frown') = (E'', \frown'')$ pour $E'' = (E.1) \cup (E'.2)$ $= (E) \uplus (E')$

$\mathcal{D}\big((E, \frown) \,\&\, (E', \frown')\big) \approx \mathcal{D}(E) \times \mathcal{D}(E')$, à isomorphisme près.

$\{e.1 \mid e \in x\} \cup \{e'.2 \mid e' \in x'\} \longleftrightarrow (x, x') \in \mathcal{D}(E) \times \mathcal{D}(E')$, pour $x \in \mathcal{D}(E)$ et $x' \in \mathcal{D}(E')$ } bijection

$y \longmapsto \big(\{e \mid e.1 \in y\}, \{e' \mid e'.2 \in y\}\big)$ }

⊗ • $\mathcal{D}(E \otimes E')$ = ... déf possible, mais complexe

$(E, \frown) \otimes (E', \frown') = (E'', \frown'')$

pour $E'' = E \times E'$

et $\frown''$ définie par : $\dfrac{e_1 \frown e_2 \quad e'_1 \frown' e'_2}{(e_1, e'_1) \frown'' (e_2, e'_2)}$ (voir cette règle comme une règle réversible)

Dual • On pose au préalable la

définition : Si $(E, \frown)$ est un espace de cohérence, on définit son dual, $\overline{(E, \frown)}$ comme $(E, \frown')$, les mêmes événements et la relation d'incohérence $\frown'$ définie par : $e \frown' e'$ si $e = e'$ ou $\neg(e \frown e')$. C'est le complémentaire de la relation, qui contient la réflexivité, et la symétrie (par être une relation de cohérence). On pose $\frown' = \asymp$, $\overline{(E, \frown)} = (E, \asymp)$. $e \asymp e'$ ssi $e = e'$ ou $\neg(e \frown e')$

↳ le dual n'est pas un connecteur mais on le définit en termes sémantiques.

⊕ • $E \oplus E' = \overline{\overline{E} \,\&\, \overline{E'}}$, le ⊕ est lié à un choix, mais comme le & il fait l'union disjointe des ensembles d'événements.

$$\frac{\dfrac{e_1 \frown e_2}{e_1 \cdot 1 \frown e_2 \cdot 2} \qquad \dfrac{e'_1 \frown e'_2}{e'_1 \cdot 1 \frown e'_2 \cdot 2} \qquad \dfrac{}{e.1 \frown e'.2}}{\&}$$

Pour $E'' = E \,\&\, E'$, $e''_1 \asymp e''_2$ ne peut être vrai que si e''_1 et e''_2 proviennent de la même composante, du même ensemble d'événements, et n'y sont pas en relation de cohérence.

$$\frac{e_1 \asymp_E e_2}{e_1 \cdot 1 \asymp_{E''} e_2 \cdot 1} \quad \text{et} \quad \frac{e'_1 \asymp_{E'} e'_2}{e'_1 \cdot 2 \asymp_{E''} e'_2 \cdot 2}$$

⇒ Donc $e''_1 \frown_{E \oplus E'} e''_2 = e''_1 \asymp_{\overline{E} \,\&\, \overline{E'}} e''_2$

$$\frac{\dfrac{e_1 \frown_E e_2 \quad \text{idem avec} = \quad e'_1 \frown_{E'} e'_2}{e_1 = e_2 \quad e'_1 = e'_2}}{\dfrac{e_1 \cdot 1 = e_2 \cdot 1}{e_1 \cdot 1 \frown_{E \oplus E'} e_2 \cdot 1} \qquad \dfrac{e'_1 \cdot 2 = e'_2 \cdot 2}{e'_1 \cdot 2 \frown_{E \oplus E'} e'_2 \cdot 2}}$$

Donc $E \parr E' = \overline{\overline{E} \otimes \overline{E'}} = (E'', \frown_{E \parr E'})$ pour $E'' = E \times E'$.

$\frown$ = cohérence stricte, $e_1 \frown e_2$ ssi $e_1 \neq e_2$ et $e_1 \frown e_2$

$\smile$ = incohérence stricte, $e \smile e_2$ ssi $e_1 \asymp e_2$ et $e_1 \neq e_2$.

Théorèmes : 1) $\mathcal{D}(E \to E')$ est par définition en bijection avec $\{ f : \mathcal{D}(E) \to \mathcal{D}(E') \mid f \text{ est linéaire} \}$

2) $\mathcal{D}(E \to E')$ ——— $\{ f : \mathcal{D}(E) \to \mathcal{D}(E') \mid f \text{ est stable} \}$

↑ G. Berry, 1977.

Déf. : • f est continue si pour tout $x_1 \subseteq x_2 \dots \subseteq x_n \dots$, $f(\bigcup_{i\geq 1} x_i) = \bigcup_{i\geq 1} f(x_i)$ et f est croissante

• f est croissante si $x \subseteq y \Rightarrow f(x) \subseteq f(y)$

• f est stable si f est continue et pour tout $x_1, x_2 \in \mathcal{D}(E)$, si $x_1 \cup x_2 \in \mathcal{D}(E)$, (d'où $x_1 \circ x_2$) alors $f(x_1 \cap x_2) = f(x_1) \cap f(x_2)$.

rem. : une intersection de configurations est une configuration, car tout sous-ensemble d'une configuration est une configuration.

• f est linéaire si f est stable et si pour tout $x_1, x_2 \in \mathcal{D}(E)$, si $x_1 \cup x_2 \in \mathcal{D}(E)$, alors $f(x_1 \cup x_2) = f(x_1) \cup f(x_2)$ (f est affine) et si $f(\emptyset) = \emptyset$.

Prouvons que c'est une bonne définition : on pose $x_1 \cup x_2 = x$. On a $x_1 \subseteq x$ et $x_2 \subseteq x$, et puisque la fonction est monotone (croissante), $f(x_1) \subseteq f(x)$ et $f(x_2) \subseteq f(x)$

$x_1 \cup x_2 \in \mathcal{D}(E) \iff$ (il existe $y \in \mathcal{D}(E)$ t.q. $x_1 \subseteq y$ et $x_2 \subseteq y$).

Donc $f(x_1) \cup f(x_2)$ a bien un majorant.

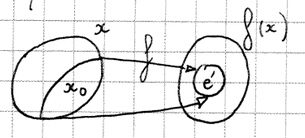
Proposition : Si f est stable, alors $\forall x \in \mathcal{D}(E)$, $\forall e' \in f(x)$, il existe $x_0 \in \mathcal{D}(E)$ tel que (pour simplifier, E est toujours dénombrable)

1) x_0 est fini
2) $x_0 \subseteq x$
3) $e' \in f(x_0)$
4) $\forall y < x_0$, $e' \notin f(y)$

($\to$ 5) x_0 est la seule partie de $\mathcal{D}(E)$ qui a les propriétés 1-4)

5) se déduit de 1-4.

On a f est vu comme un programme qui utilise x. Jusqu'à quel besoin ce programme a besoin de x pour trouver e' ? Cette proposition dit qu'on peut justement trouver cette partie de x, x_0, qui permet de trouver e', et par 4 cette partie est minimale, et donc unique.



Preuve : • Si $e' \in f(x)$, il existe $y_0 < x$ tel que y_0 est fini et $e' \in f(y_0)$.

$x = \bigcup_{i\geq 1} x_i$, donc $e' \in \bigcup f(x_i) = f(\bigcup x_i)$ car f est stable, donc il existe i tel que $e' \in f(x_i)$, $y_0 = x_i$. ↳ vérifie 1)

• $y_0 = \{e_1, \dots, e_n\}$ on a $y_1 = \{e_1, \dots, \hat{e_i}, \dots, e_n\}$, on a ôté un élément à y_0.

Algorithme : $y := y_0$; regarde pour chaque $e \in y$ si $e' \in f(y)$; dès qu'on en trouve un on pose $y_i = y_0 / \{e\}$; recommence.

↳ vérifie 3) et 4), car l'ensemble de départ est fini.

• Si x_0 et $x_0^{\hat{}}$ vérifient 1-4, $x_0 \cap x_0^{\hat{}} = y$ vérifie 1-4, car

$f(y) = f(x_0) \cap f(x_0^{\hat{}})$, et puisque $e' \in f(x_0)$ et $e' \in f(x_0^{\hat{}})$, $e' \in f(y)$.

$y \subseteq x_0$, si $y = x_0$, $x_0 \cap x_0^{\hat{}} = x_0$, et $x_0 \subseteq x_0^{\hat{}}$, donc $x_0 = x_0^{\hat{}}$

↳ vérifie 5), $x_0 = x_0^{\hat{}}$ s'ils vérifient 1-4.

Proposition : Si f est linéaire, $\forall x \in \mathcal{D}(E)$, $\forall e' \in f(x)$, il existe $e \in x$ tel que

1) $e' \in f(\{e\})$
2) $e' \notin f(\emptyset)$
3) e est unique satisfaisant 1 et 2.

lundi 12 avril ↳ 9h-12
Examen !
(note finale ≠ moyenne (partiel, examen)) ↳ plutôt max (partiel, exam)

Le renforcement de la stabilité, la partie de x minimale suffisante pour ce calcul est en fait un seul événement de x.

Preuve : f est linéaire, donc stable, et il existe x_0 vérifiant 1-5 de la proposition précédente. Deux cas :

• x_0 est un singleton, et c'est fini.

• x_0 n'est pas un singleton, on a deux cas : - $x_0 = \emptyset$

- $x_0 = x_1 \cup x_2$ pour $x_1 \neq \emptyset$ et $x_2 \neq \emptyset$, $x_1 < x_0$ et $x_2 < x_0$

Si $x_0 = \emptyset$, $f(x_0) = \emptyset$ car f est linéaire, et il n'est pas possible que $e' \in \emptyset$ → cas exclu.

Sinon, $e' \in f(x_0) = f(x_1 \cup x_2)$ par déf., et $= f(x_2) \cup f(x_1)$ car f est continue. Donc

$e' \in f(x_1)$ ou $e' \in f(x_2)$ → alors x_0 n'est pas minimal, puisque $x_1 \in x_0$ ou $x_2 \in x_0$ et $e' \in f(x_1)$ ou $e' \in f(x_2)$.

! · $!(E,\subset) = (E',\subset')$ pour $E' = \{x \mid x \in \mathcal{D}(E) \text{ et } \forall x \text{ est fini}\}$

$\subset'$: $x_1 \subset' x_2$ ssi $x_1 \cup x_2 \in \mathcal{D}(E)$

Preuve du théorème 2 : $\mathcal{D}(E \to E') =_{\text{déf.}} \mathcal{D}(!E \multimap E') = \{f : \mathcal{D}(E) \to \mathcal{D}(E') \mid f \text{ est stable}\}$.

On construit une bijection $\text{trace}(f) \longleftrightarrow f \qquad \Psi \mapsto \text{fun}(\Psi)$

déf : $\text{trace}(f) = \{(x,e') \mid x \in \mathcal{D}(E),\ x \text{ est fini},\ e' \in f(x) \text{ et } \forall y < x,\ e' \notin f(y)\}$

$\text{fun}(\Psi)(x) = \{e' \mid \exists x_0 \underset{\text{unique en fait}}{\leqslant} x,\ (x_0,e') \in \Psi\}$

→ montre que $\text{trace}(f)$ est une configuration et que $\text{fun}(\Psi)(x)$ est correctement défini. $\text{fun}(\text{trace}(f)) = f$ et $\text{trace}(\text{fun}(\Psi)) = \Psi$.

· fun est croissante (trivial), continue, car le x_0 de la déf. est fini (→ exo.), et c'est bien une configuration de E', et elle est stable (voir *)

↳ On ne veut pas : $x_0 \leqslant x$, e'_0 tels que $(x_0,e'_0) \in \Psi$ et $x_1 \leqslant x$, e'_1 tels que $(x_1,e'_1) \in \Psi$ } et $e'_0 \smile e'_1$ → $\text{fun}(\Psi)(x)$ ne serait alors pas une configuration.

↳ il faut qu'on montre que sous ces conditions, $(x_0,e'_0) \smile (x_1,e'_1)$

On doit montrer par que $\text{fun}(\Psi)$ soit une configuration : $\dfrac{x_0 \subset x_1 \qquad e'_0 \smile e'_1}{(x_0,e'_0) \smile (x_1,e'_1)}$

C'était pour $(!E) \multimap E'$, pour $E \multimap E'$ on a $\dfrac{e_0 \subset_E e_1 \qquad e'_0 \smile_{E'} e'_1}{(e_0,e'_0) \smile_{E \multimap E'} (e_1,e'_1)}$

Par définition, on a approximativement : $(e_0,e'_0) \subset_{E \multimap E'} (e_1,e'_1) \Rightarrow \big[(e_0 \subset e_1) \Rightarrow (e'_0 \subset e'_1)\big]$.

* $\text{fun}(\Psi)$ est stable implique $\text{fun}(\Psi)(x_1 \cap x_2) = \text{fun}(\Psi)(x_1) \cap \text{fun}(\Psi)(x_2)$.

Inclusion $\subseteq$ facile, puisque $\text{fun}(\Psi)$ est croissante.

Inclusion $\supseteq$: supposons $e' \in \text{fun}(\Psi)(x_1) \cap \text{fun}(\Psi)(x_2)$. Alors il existe $x_0 \ll x_1$ et $(x_0,e') \in \Psi$ et il existe $x'_0 \ll x_2$ tel que $(x'_0,e') \in \Psi$. On cherche y_0 tel que $y_0 \ll x_1 \cap x_2$ tel que $(y_0,e') \in \Psi$.

On complète la déf de $\subset_{E \multimap E'}$: $(e_0,e'_0) \subset_{E \multimap E'} (e_1,e'_1) \Rightarrow \big[(e_0 \subset e_1) \Rightarrow \big((e'_0 \subset e'_1) \text{ et } (e'_0 = e'_1 \Rightarrow e_0 = e_1)\big)\big]$

↳ donc $x_0 = x'_0$, car $(x_0,e') \in \Psi \subset (x'_0,e') \in \Psi$, et $e' = e'$.

Déf : $e_1 \asymp e_2 \iff e_1 = e_2$ ou $\neg(e_1 \subset e_2)$

$e_1 \subset e_2 \iff e_1 = e_2$ ou $\neg(e_1 \asymp e_2)$

· $E \otimes E'$

Événements : $E \times E'$

$(e_1,e'_1) \subset (e_2,e'_2)$ ssi $(e_1 \subset e_2)$ et $(e'_1 \subset e'_2)$

· $E \parr E'$

Événements : $E \times E'$

$(e_1,e'_1) \asymp (e_2,e'_2)$ ssi $(e_1 \underset{E}{\asymp} e_2)$ et $(e'_1 \underset{E'}{\asymp} e'_2)$

· $E \multimap E' = \bar{E} \parr E'$

Événements : $E \times E'$

$(e_1,e'_1) \subset (e_2,e'_2)$ ssi $(e_1 \subset e_2) \Rightarrow \big((e'_1 \subset e'_2) \text{ et } \big((e'_1 = e'_2) \Rightarrow (e_1 = e_2)\big)\big)$

· $\bar{E} \parr E'$

$(e_1,e'_1) \asymp (e_2,e'_2)$ ssi $(e_1 \subset_E e_2)$ et $(e'_1 \underset{E'}{\asymp} e'_2)$.

On montre le sens $\Rightarrow$ de $\multimap$:

· Supp. $(e_1,e'_1) \subset (e_2,e'_2)$ et $(e_1 \subset e_2)$

Montrons $e'_1 \subset e'_2$ par l'absurde : supp. $e'_1 \smile e'_2$, alors puisq. $(e_1 \subset e_2)$, $(e_1,e'_1) \asymp (e_2,e'_2)$, d'où $(e_1 = e_2)$ et $(e'_1 = e'_2)$ → absurde, $e'_1 \subset e'_2$.

· Supp. enfin $e'_1 = e'_2$, on montre $e_1 = e_2$ par l'absurde : supp. $e_1 \neq e_2$, par le même raisonnement que précédemment on démontre $e_1 = e_2$, d'où contradiction.

En logique $A \simeq B$, A est isomorphe à B, si

· A et B sont équiprouvables : $A \vdash B$ et $B \vdash A$

· si $\pi_1 : (A \vdash B)$ et $\pi_2 : (B \vdash A)$, $\dfrac{\overset{\vdots \pi_1}{A \vdash B} \quad \overset{\vdots \pi_2}{B \vdash A}}{A \vdash A}$ cut-elim $\rightsquigarrow$ $\dfrac{}{A \vdash A}\ Ax$ (ou une version expansée de l'axiome)

On a $A \otimes 1 \simeq A$

$(A \otimes_1 B) \otimes_3 C \otimes_5 D = A \otimes_4 B \otimes_4 C \otimes_4 D$: le tenseur n-aire permet de s'passer de l'associativité.

1 est le cas 0-ain du tenseur : $A \otimes_2 (\otimes_0 ()) = \otimes_1 A$

D'où 1 interprété comme $(\{*\}, \subset)$, avec $* \subset *$ il n'y a qu'un événement. $\bot$ est interprété de la même façon

$\top$ et 0 sont interprétés par $(\emptyset, \supset)$

$$\begin{cases} 1 = (\{*\}, \supset) = \bot \\ 0 = (\emptyset, \supset) = \top \end{cases}$$

Interprétation des preuves

Un atome est interprété par un espace de cohérence choisi. Une valuation φ associe aux atomes des espaces: $\varphi(X) = E_1$, $\varphi(Y) = E_2$, etc.

$[\![A \otimes B]\!]_\varphi = [\![A]\!]_\varphi \otimes [\![B]\!]_\varphi$, idem pour les autres connecteurs. On omet de mentionner la valuation pour alléger l'écriture.

$$\dfrac{\begin{array}{c}\vdots \pi \\ \vdash A_1, \dots, A_n\end{array}}{}$$ $\rightsquigarrow$ clique / configuration de $[\![A_1 \parr \dots \parr A_n]\!]$, donc un élément de $\mathcal{D}([\![A_1 \parr \dots \parr A_n]\!])$

$\pi \rightsquigarrow_{\text{par élim. des coupures}} \pi'$ $\rightsquigarrow$ $[\![\pi]\!] = [\![\pi']\!]$.

Utilité générale de construire un modèle de preuves:

si on peut établir que $[\![\pi]\!] \neq [\![\pi']\!]$ dans le modèle, alors il n'existe pas de transformation sur les preuves par élimination des coupures qui va de π à π'.

(raisonnable si le syst. n'est pas confluent: sinon pour prouver la même chose il suffit de normaliser π et π', et de voir que leurs formes normales sont différentes).

$$\pi \left\{ \dfrac{\begin{array}{cc}\vdots \pi_1 & \vdots \pi_2 \\ \vdash \Gamma_1, A & \vdash \Gamma_2, B\end{array}}{\vdash \Gamma_1, \Gamma_2, A \otimes B} \right.$$

On a $x \in \Gamma_1 \parr A$, les éléments de x sont du type (γ_1, e) pour $\gamma_1 \in \Gamma_1$ et $e \in A$. Idem avec $\gamma \in \Gamma_2 \parr B$, (γ_2, e') pour $\gamma_2 \in \Gamma_2$ et $e' \in B$.

↳ Donc $[\![\pi_1]\!] = x \in \Gamma_1 \parr A$, $[\![\pi_2]\!] = \gamma \in \Gamma_2 \parr B$,

et $[\![\pi]\!] = \left\{ (\gamma_1, \gamma_2, (e, e')) \mid (\gamma_1, e) \in [\![\pi_1]\!] \text{ et } (\gamma_2, e') \in [\![\pi_2]\!] \right\}$

"$[\![\pi]\!] = [\![\pi_1]\!] \times [\![\pi_2]\!]$"

$$\pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash \Gamma, A, B\end{array}}{\vdash \Gamma, A \parr B} \right.$$

$(\gamma, e, e') \in [\![\pi_1]\!]$, $(\gamma, (e, e')) \in [\![\pi]\!]$ $[\![\pi]\!]$ est la même chose que $[\![\pi_1]\!]$ à réarrangement près: $[\![\pi]\!] = \left\{ (\gamma, (e, e')) \mid (\gamma, e, e') \in [\![\pi_1]\!] \right\}$.

$$\pi \left\{ \dfrac{\begin{array}{cc}\vdots \pi_1 & \vdots \pi_2 \\ \vdash \Gamma, A & \vdash \Gamma, B\end{array}}{\vdash \Gamma, A \& B} \right. \qquad (\gamma_2, e') \in [\![\pi_2]\!] \qquad (\gamma_1, e) \in [\![\pi_1]\!].$$

$$[\![\pi]\!] = \left\{ (\gamma_1, e.1) \mid (\gamma_1, e) \in [\![\pi_1]\!] \right\} \cup \left\{ (\gamma_2, e'.2) \mid (\gamma_2, e') \in [\![\pi_2]\!] \right\}$$

Vérifie que $[\![\pi]\!]$ donne bien une cliq. Il y a pas deux elem incohérents dans $[\![\pi]\!]$. Sa parenté par $\Gamma \parr (A \& B)$ $\mathcal{P}_0$

$$\pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash \Gamma, A\end{array}}{\vdash \Gamma, A \oplus B} \right. \qquad [\![\pi]\!] = \left\{ (\gamma, e.1) \mid (\gamma, e) \in [\![\pi_1]\!] \right\} \qquad [\![\pi']\!] = \left\{ (\gamma, e'.2) \mid (\gamma, e') \in [\![\pi_2]\!] \right\} \qquad \left. \dfrac{\begin{array}{c}\vdots \pi_2 \\ \vdash \Gamma, B\end{array}}{\vdash \Gamma, A \oplus B} \right\} \pi'$$

$$\pi \left\{ \dfrac{}{\vdash 1} \right. \qquad [\![\pi]\!] = \{*\} \in 1 \qquad\qquad \pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash \Gamma\end{array}}{\vdash \Gamma, \bot} \right. \qquad [\![\pi]\!] = \left\{ (\gamma, *) \mid \gamma \in [\![\pi_1]\!] \right\}$$

($\bot$ = "élément neutre de $\parr$" 0 = vrai faux classique, pas de règle K par d'interprétation)

$$\pi \left\{ \dfrac{}{\vdash \Gamma, \top} \right. \qquad [\![\pi]\!] = \emptyset \in \mathcal{D}(\emptyset)$$

$\emptyset$ est l'unique clique de $\emptyset$

1 a deux cliques: $\emptyset$ et $\{*\}$.

Déreliction:
$$\pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash \Gamma, A\end{array}}{\vdash \Gamma, ?A} \right. \qquad (\gamma, e) \in [\![\pi_1]\!] \qquad [\![\pi]\!] = \left\{ (\gamma, !e) \mid (\gamma, e) \in [\![\pi_1]\!] \right\}.$$

Promotion:
$$\pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash ?\Gamma, A\end{array}}{\vdash ?\Gamma, !A} \right. \qquad [\![?\Gamma]\!] = [\![(?A_1) \parr \dots \parr (?A_n)]\!] = [\![?(A_1 \oplus \dots \oplus A_n)]\!] \text{ à isomorphisme près.}$$

nota : $(!A) \otimes (!B) \approx !A \& B$, $e^m \times e^n = e^{m+n}$

$\cdot\ (?A) \parr (?B) \approx ?(A \oplus B)$

$[\![\pi_1]\!] \ni (\xi_1, e)$, pour $\xi_1 = \langle \dots \gamma_1 \dots \rangle$, pour $\gamma_1 \in \Gamma$. Or $\xi_1, \dots, \xi_n \in ?\Gamma$, et donc ξ aussi $\in ?\Gamma$

$(\xi, \langle e_1, \dots, e_n \rangle) \in [\![?\Gamma \parr !A]\!]$

↳ $[\![\pi]\!] = \left\{ (\xi, x) \mid \exists \xi_1, e_1, \dots, \xi_n, e_n \quad x = \{e_1, \dots, e_n\}, \xi = \xi_1 \cup \dots \cup \xi_n, \langle \xi_1, e_1 \rangle \in [\![\pi_1]\!], \dots \right.$
$\left. \dots \langle \xi_n, e_n \rangle \in [\![\pi_1]\!] \right\}$

Contraction
$$\pi \left\{ \dfrac{\begin{array}{c}\vdots \pi_1 \\ \vdash \Gamma, ?A, ?A\end{array}}{\vdash \Gamma, ?A} \right. \qquad (\gamma, x_1, x_2) \in [\![\pi_1]\!]$$

$$[\![\pi]\!] = \left\{ (\gamma, x_1 \cup x_2) \mid (\gamma, x_1, x_2) \in [\![\pi_1]\!] \right\}$$

Affaiblissement $\pi\left\{\dfrac{\vdots\,\pi_1 \quad \vdash\Gamma}{\vdash\Gamma,?A}\right.$ $\qquad [\![\pi]\!] = \{(\gamma,\emptyset) \mid \gamma\in[\![\pi_1]\!]\}$

Une étape d'élimination des coupures pour justifier :

La coupure est multiplicative : $\pi\left\{\dfrac{\vdots\,\pi_1 \quad \vdash\Gamma_1,A \qquad \vdots\,\pi_2 \quad \vdash\Gamma_2,\bar A}{\vdash\Gamma_1,\Gamma_2}\right.$ (pas de contraction cachée dernière)

$$[\![\pi]\!] = \{(\gamma_1,\gamma_2) \mid \exists e\ (\gamma_1,e)\in[\![\pi_1]\!] \text{ et } (\gamma_2,e)\in[\![\pi_2]\!]\}$$
((γ,δ) ; (δ) ; (δ'))

A et $\bar A$ ont les mêmes événements.

// composition des relations : $R\subseteq X\times Y$ et $S\subseteq Y\times Z$, $(S\circ R)\subseteq X\times Z$,

$$S\circ R = \{(x,y) \mid \exists y \text{ t.q. } (x,y)\in R \text{ et } (y,y)\in S\}$$

On montre que c'est cohérent : supposons par l'absurde que $(\gamma_1,\gamma_1')\frown(\gamma_2,\gamma_2')$, c'est le cas si $[\![\pi]\!]$ n'est pas un conf. Alors $\gamma_1\asymp\gamma_2$ et $\gamma_1'\asymp\gamma_2'$. $(\gamma_1,e_1)\underset{\Gamma_1\otimes A}{\supset}(\gamma_2,e_2)$ et $(\gamma_1',e_1)\underset{\Gamma_2\otimes\bar A}{\supset}(\gamma_2',e_2)$ car $(\gamma_1,\gamma_1')\in[\![\pi]\!]$ et $(\gamma_2,\gamma_2')\in[\![\pi]\!]$

On raisonne par cas : • si $e_1\underset{A}{\asymp}e_2$, alors puisque $\gamma_1\asymp\gamma_2$, $(\gamma_1,e_1)\asymp(\gamma_2,e_2)$, or ils sont également cohérents, donc $\gamma_1=\gamma_2$ et $e_1=e_2$. ($e_1\underset{A}{\asymp}e_2 \Leftrightarrow e_1\underset{\bar A}{\supset}e_2$) Donc $e_1\underset{\bar A}{\asymp}e_2$, et puisque $\gamma_1'\asymp\gamma_2'$, $(\gamma_1',e_1)\underset{\Gamma_2\otimes A}{\asymp}(\gamma_2',e_2)$, or on les avait supposé cohérents, donc ils sont égaux, et $\gamma_1'=\gamma_2'$. Donc $(\gamma_1,\gamma_1')\supset(\gamma_2,\gamma_2')$ $\to$ incohérence.

• le raisonnement est symétrique si on suppose $e_1\supset_A e_2$.

$\pi\left\{\dfrac{\vdots}{A\vdash B}\right.$: $[\![\pi]\!]$ est un conf de $\bar A\otimes B = A\multimap B$, d'où $[\![\pi]\!]\in\mathcal{D}(A\multimap B)$ = fonctions linéaires de $\mathcal{D}(A)$ dans $\mathcal{D}(B)$

Théo. Quelques soient E,E',E'' des espaces de cohérence, quelques soient $f\in\mathcal{D}(E\multimap E')$ et $g\in\mathcal{D}(E'\multimap E'')$, on a : $\mathrm{trace}(g\circ f) = \mathrm{trace}(g)\circ\mathrm{trace}(f)$ Relie composée de fct° et composée des relat°

(trace linéaire)

Déf : $\mathrm{trace}(\beta) = \{(e,e') \mid e'\in\beta(e)\}$

$$\text{ctr}\ \frac{\vdots\,\pi_1 \quad \vdash\Gamma_1,?A,?A}{\vdash\Gamma_1,?A} \qquad \frac{\vdots\,\pi_2 \quad \vdash\Gamma_2,!\bar A}{\vdash\Gamma_1,\Gamma_2} \rightsquigarrow \frac{\vdots\,\pi_1 \quad \vdash\Gamma_1,?A}{\vdash\Gamma_1,\Gamma_2}$$
Pb avec la duplication de π_2 : gestion des contextes

mer 17 mars 10.

Contraction : $\dfrac{\vdots\,\pi_1 \quad \vdash\Gamma,?A,?A}{\vdash\Gamma,?A}\Big\}\pi$ $\qquad [\![\pi]\!] = \{(\gamma,x_1\cup x_2) \mid (\gamma,x_1,x_2)\in[\![\pi_1]\!]\}$

Promotion : $\dfrac{\vdots\,\pi_1 \quad \vdash ?\Gamma,A}{\vdash ?\Gamma,!A}\Big\}\pi$ $\qquad [\![\pi]\!] = \{(\xi,x) \mid \xi=\xi_1\cup\dots\cup\xi_n,\ x=\{e_1,\dots,e_n\},(\xi_1,x_1),\dots,(\xi_n,e_n)\in[\![\pi_1]\!]\}$

Invariance de la sémantique : Si π se réduit en éliminant les coupures sur π', $[\![\pi]\!]=[\![\pi']\!]$.

On illustre avec la règle la plus compliquée, qui met en jeu contraction et promotion, et vient de deplique un preuve :

$$\pi\left\{\begin{array}{l}\pi_1'\left\{\dfrac{\vdots\,\pi_1 \quad \vdash\Gamma_1,?A,?A}{\vdash\Gamma_1,?A}\right. \qquad \dfrac{\vdots\,\pi_2 \quad \vdash ?\Gamma_2,\bar A}{\vdash ?\Gamma_2,!\bar A}\\ \qquad\qquad \vdash\Gamma_1,?\Gamma_2\end{array}\right\}\pi_2' \rightsquigarrow \pi'\left\{\dfrac{\dfrac{\vdots\,\pi_1 \quad \vdash\Gamma_1,?A,?A \qquad \vdots\,\pi_2' \quad \vdash ?\Gamma_2,!\bar A}{\vdash\Gamma_1,?\Gamma_2,?A}\,\pi_3 \qquad \vdots\,\pi_2' \quad \vdash ?\Gamma_2,!\bar A}{\vdash\Gamma_1,?\Gamma_2,?\Gamma_2}\,\pi_4 \quad \text{contraction de toutes les formules de } ?\Gamma_2.}{\vdash\Gamma_1,?\Gamma_2}\right.$$

On ne montre en fait que $[\![\pi]\!]\subseteq[\![\pi']\!]$

sens de lecture ↑ π ; ↓ π'

Soit $(\gamma_1,\xi)\in[\![\pi]\!]$, il existe x tel que $(\gamma_1,x)\in[\![\pi_1']\!]$ et $(\xi,x)\in[\![\pi_2']\!]$. En fait $x=x_1\cup x_2$ et $(\gamma_1,x_1,x_2)\in[\![\pi_1]\!]$, à droite $x=\{e_1,\dots,e_n\}$ et $\xi=\xi_1\cup\dots\cup\xi_n$ tels que $(\xi_1,e_1)\dots(\xi_n,e_n)\in[\![\pi_2]\!]$

↳ il faut montrer que $(\gamma_1,\xi)\in[\![\pi']\!]$. On doit transférer l'information de la contraction de $?A$ dans π sur la contraction des formules de $?\Gamma_2$ dans π'.

Soit $[\![1,n]\!]=I\sqcup J$, $x_1=\{e_i \mid i\in I\}$ et $x_2=\{e_j \mid j\in J\}$

On pose $\xi^1=\bigcup_{i\in I}\xi_i$ et $\xi^2=\bigcup_{j\in J}\xi_j$, donc $\xi=\xi^1\cup\xi^2$

Donc $(\xi^1,x_1)\in[\![\pi_2']\!]$ $(\gamma_1,x_1,x_2)\in[\![\pi_1]\!]$, d'où $(\gamma_1,\xi^1,x_2)\in[\![\pi_3]\!]$.

Symétriquement, $(\xi^2,x_2)\in[\![\pi_2']\!]$, donc $(\gamma_1,\xi^1,\xi^2)\in[\![\pi_4]\!]$.

Par interprétation, $(\gamma_1,\xi^1\cup\xi^2)\in[\![\pi']\!]$, et c'est terminé car $\xi^1\cup\xi^2=\xi$.

Réseaux de preuves

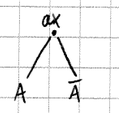
Innovation de la Logique Linéaire, dûe à Girard en 86, cf. son papier original Linear Logic.
On passe des arbres de preuves aux graphes, ce qui offre deux avantages:

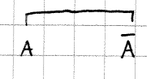
- le graphe représente un quotient sur les preuves: plusieurs preuves ont le même graphe associé
- on peut définir l'élimination des coupures dans les graphes.

Il y a un bémol: cette pratique n'est acceptée que pour MELL. Il n'y a pas de théorie canonique des réseaux de preuves en MALL, qui est encore un sujet de recherche. Les constantes ne se représentent pas très bien non plus.
MELL permet de coder tout le λ-calcul, mais pas la disjonction classique ($\oplus$)

Déf. par récurrence de comment associer à un arbre de preuve de MELL un graphe que l'on nomme un réseau de preuve

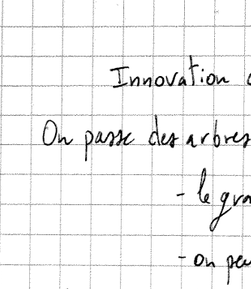
$$\frac{}{\vdash A, \bar{A}}\ Ax$$

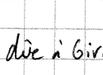




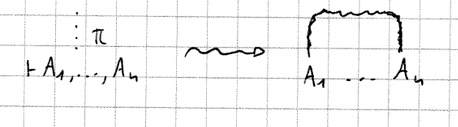
Si on veut vraiment un graphe — Est la notation plus souvent utilisée.

Invariant:



(dag: graphe dans lequel les flèches sont orientées (ici, toujours vers le bas), sans cycle orienté  mais  Nos réseaux de preuve sont des dag, et on a un sens de parler de nœud minimum).

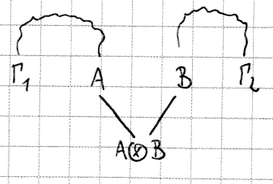
(Dans un dag, appelons conclusion un nœud minimal, qui n'est pas source d'une flèche)

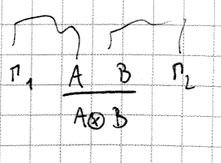


Un graphe représentant une preuve π a pour conclusions les conclusions de π

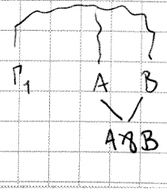
Un réseau de preuve est un dag dont tous les nœuds sont étiquetés par des formules (sauf ax et cut)

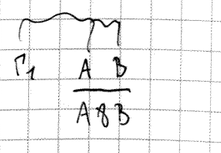
$$\frac{\vdash \Gamma_1, A \qquad \vdash \Gamma_2, B}{\vdash \Gamma_1, \Gamma_2, A \otimes B}$$





$$\frac{\vdash \Gamma_1, A, B}{\vdash \Gamma_1, A \parr B}$$

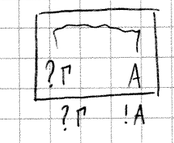




notation de Girard

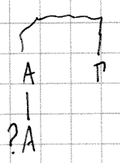
exponentiels :

$$\text{promo.}\ \frac{\vdash ?\Gamma, A}{\vdash ?\Gamma, !A}$$

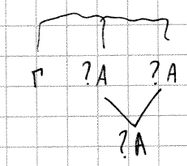


la boîte identifie un sous-graphe.

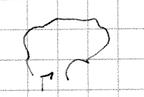
$$\text{derel.}\ \frac{\vdash \Gamma, A}{\vdash \Gamma, ?A}$$



$$\text{contr.}\ \frac{\vdash \Gamma, ?A, ?A}{\vdash \Gamma, ?A}$$

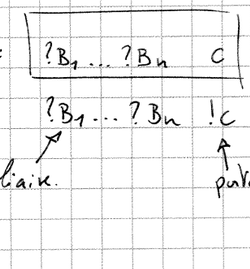


$$\frac{\vdash \Gamma}{\vdash \Gamma, ?A}\ ff$$

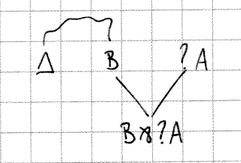


on pose un point isolé à l'extérieur du graphe.

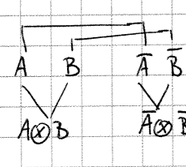
Terminologie:



ex: $\dfrac{\dfrac{\vdash B, \Delta}{\vdash ?A, B, \Delta}}{\vdash B \parr ?A, \Delta}$



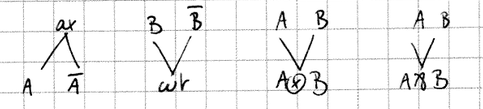
ex. de dag qui n'est pas un réseau de preuve:

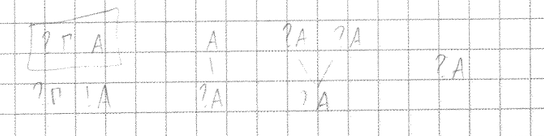


On appelle les graphes qui se sont par de preuves des structures de preuve.

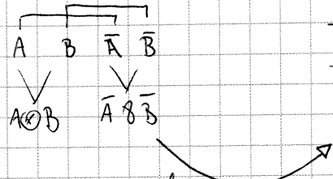
(explicit° formelle des structures de preuve dans Introduction to L.L. de Ludics, de Curien, II, 1ère section).

Réduit:





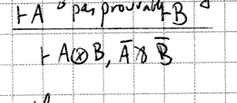
Par contre est un réseau:



séquentialisation d'un réseau de preuve.

$$\frac{\dfrac{\overline{\vdash A, \bar{A}} \qquad \overline{\vdash B, \bar{B}}}{\vdash A \otimes B, \bar{A}, \bar{B}}}{\vdash A \otimes B, \bar{A} \parr \bar{B}}$$

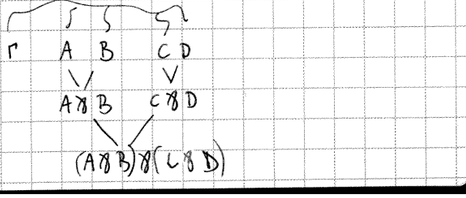
positif négatif, donc réversible



Quotient: // passage syst. à localisation → syst. syntaxique s'était fait par quotient sur les preuves.

$$\frac{\vdash A, B, C, D, \Gamma}{\vdash A \parr B, C, D, \Gamma}\ \text{ou}\ \frac{\vdash A, B, C, D, \Gamma}{\vdash A, B, C \parr D, \Gamma}$$
$$\vdots$$
$$\frac{\vdots}{\vdash A \parr B, C \parr D, \Gamma}$$
$$\vdash (A \parr B) \parr (C \parr D), \Gamma$$

sont représentés par



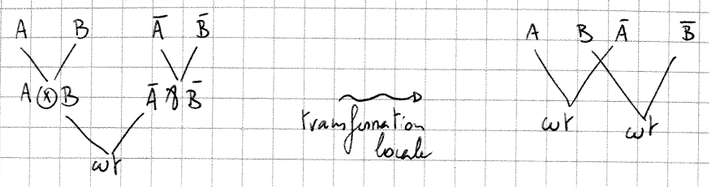
Élimination des coupures

Si, il existe une coupure commutative.

Il y a plus de commutations car elle entre à travers dans les réseaux. Il y a plus que les cas logiques.

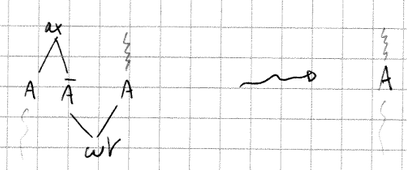
MLL

⊗/⅋

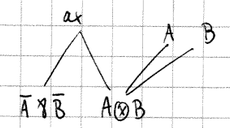


Le graphe est non planaire, on ne peut pas le représenter sans croisement, car $A \otimes B \neq B \otimes A$.

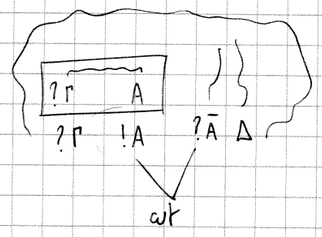
Ax / wr



⚠ est interdit

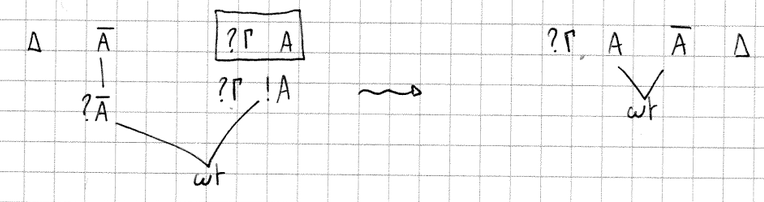


ELL

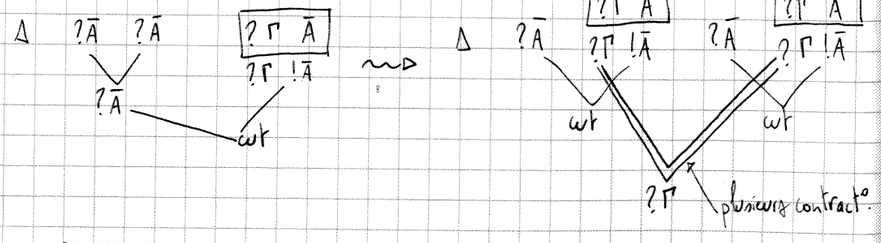


Traité par cas selon la règle qui précède ?Ā : contracté, "déreliction", affaibli, si on a formé une boîte pour une formule de Δ.

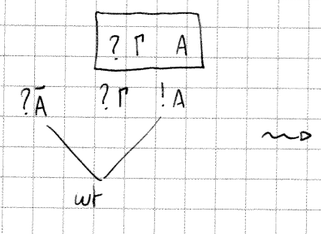
4 cas : Déreliction :



· Contraction :

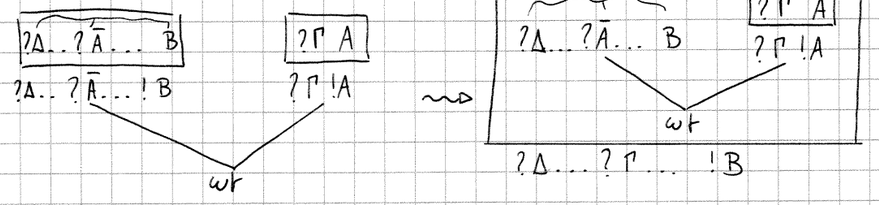


· Affaiblissement :



?Γ introduit par affaiblissements + effacement de la boîte.

· Si ?Ā sort d'une boîte Cas où l'on permute.



instance correcte de la promotion, pour vérifier les formules sont ?.

Au programme : critère de correction des réseaux.