Systèmes déductifs

DEA d'informatique
Université Bordeaux 1

Plan

1. Liens avec d'autres domaines de l'informatique

- Programmation fonctionnelle typée
- Programmation logique
- Modèles du parallélisme
- Traitement des langues

2. La déduction naturelle de base (conjonction, implication)

Propriétés essentielles :

- normalisation forte (propriété de la sous formule)
- décidabilité du calcul propositionnel
- confluence

2.1. Une présentation graphique (quasi arborescente)

2.1.1. Le connecteur $\wedge$

$$\dfrac{\begin{matrix}\vdots & & \vdots\\ A & & B\end{matrix}}{A \wedge B}\wedge_i \qquad \dfrac{\begin{matrix}\vdots\\ A \wedge B\end{matrix}}{A}\wedge_e 1 \qquad \dfrac{\begin{matrix}\vdots\\ A \wedge B\end{matrix}}{B}\wedge_e 2$$

2.1.2. Le connecteur $\longrightarrow$

(0,...,n occurrences de l'hypothèse libre A)

$$\dfrac{\begin{matrix}X \ldots A \ldots [A]_\alpha \ldots [A]_\alpha \ldots Y\\ \vdots\\ B\end{matrix}}{A \longrightarrow B}\,\alpha \longrightarrow_i \qquad \dfrac{\begin{matrix}\vdots & & \vdots\\ A & & A \longrightarrow B\end{matrix}}{B}\longrightarrow_e$$

2.2. Une présentation plus précise

2.2.1. Règles structurelles (contexte)

$$\frac{\Gamma, A, B, \Delta \vdash C}{\Gamma, B, A, \Delta \vdash C}\,Tr \qquad \frac{\Gamma \vdash C}{A, \Gamma \vdash C}\,Aff \qquad \frac{A, A, \Gamma \vdash C}{A, \Gamma \vdash C}\,Contr$$

2.2.2. Le connecteur $\rightarrow$

$$\frac{A, \Gamma \vdash B}{\Gamma \vdash A \rightarrow B}\rightarrow_i \qquad \frac{\Gamma \vdash A \quad \Delta \vdash A \rightarrow B}{\Gamma, \Delta \vdash B}\rightarrow_e$$

2.2.3. Le connecteur $\wedge$

$$\frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \wedge B}\wedge_i \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\wedge_e 1 \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\wedge_e 2$$

2.3. Lien avec le lambda calcul typé

2.3.1. Règles structurelles

$$\frac{\vec{g}:\Gamma, a:A, b:B, \vec{d}:\Delta \vdash t:C}{\vec{g}:\Gamma, b:B, a:A, \vec{d}:\Delta \vdash t:C}\,Tr \qquad \frac{\vec{g}:\Gamma \vdash t:C}{a:A, \vec{g}:\Gamma \vdash t:C}\,Aff \qquad \frac{a:A, a':A, \vec{g}:\Gamma \vdash t:C}{a:A, \vec{g}:\Gamma \vdash t[a' := a]:C}\,Contr$$

2.3.2. Le connecteur $\to$

$$\frac{a:A, \vec{g}:\Gamma \vdash t:B}{\vec{g}:\Gamma \vdash \lambda a.t:A \to B}\to_i \qquad \frac{\vec{g}:\Gamma \vdash u:A \quad \vec{d}:\Delta \vdash t:A \to B}{\vec{g}:\Gamma, \vec{d}:\Delta \vdash t\ u:B}\to_e$$

2.3.3. Le connecteur $\wedge$

$$\frac{\vec{g}:\Gamma \vdash u:A \quad \vec{d}:\Delta \vdash v:B}{\vec{g}:\Gamma, \vec{d}:\Delta \vdash \langle u,v\rangle:A \wedge B}\wedge_i \qquad \frac{\vec{g}:\Gamma \vdash t:A \wedge B}{\vec{g}:\Gamma \vdash \pi^1 t:A}\wedge_e 1 \qquad \frac{\vec{g}:\Gamma \vdash t:A \wedge B}{\vec{g}:\Gamma \vdash \pi^2 t:B}\wedge_e 2$$

2.4. Normalisation (ou réduction)

2.4.1. Schémas de normalisation

Redex: Introduction d'un connecteur immédiatement suivie d'une élimination de ce même connecteur.

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\delta\\ A\end{array}\qquad\begin{array}{c}\vdots\,\gamma\\ B\end{array}}{A\wedge B}\,\wedge_i}{A}\,\wedge_e 1 \qquad\qquad \pi^1\langle\delta^A,\gamma^B\rangle$$

$\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$ se réduit en $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$

$$\begin{array}{c}\vdots\,\delta\\ A\end{array} \qquad\qquad \delta^A$$

$$\dfrac{\begin{matrix}\vdots\ \delta\\ A\end{matrix} \qquad \dfrac{\begin{matrix}X \ldots A \ldots [A]_\alpha \ldots [A]_\alpha \ldots Y\\ \vdots\ \gamma\\ B\end{matrix}}{A \to B}\ \alpha \to_i}{B}\ \to_e$$

$$((\lambda a^A.\gamma^B)^{A\to B}\ \delta^A)^B$$

$\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$ se réduit en $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$

$$\begin{matrix} & \vdots\ \delta & & \vdots\ \delta & \\ X \ldots A \ldots & A & \ldots & A & \ldots Y \\ & \vdots\ \gamma & & & \\ & B & & & \end{matrix}$$

$$(\gamma^B[x^A := \delta^A])^B$$

2.4.2. Normalisation faible Toute démonstration peut être transformée en une démonstration sans redex. (**EXO**)

Degré d'une formule:

$$\begin{array}{rcl} d(p) & = & 0 \qquad p\text{: variable propositionnelle} \\ d(A \rightarrow B) & = & \max(d(A), d(B)) + 1 \\ d(A \wedge B) & = & \max(d(A), d(B)) + 1 \end{array}$$

Degré d'un redex: degré de la formule introduite puis éliminée.

Degré d'un terme: (degré max d'un redex, nbre de redex de degré max)

On réduit un redex de degré max M qui n'en contient pas d'autre (le plus haut possible dans l'arbre).

Les seuls redex qui peuvent apparaître sont de degré $\leq d(A) < M$.

La mesure décroit.

2.4.3. Normalisation forte par réductibilité

$t \in RED_T$: « t est réductible de type T »

Définition inductive de la réductibilité

- si T atomique: $t \in RED_T$ ssi t fortement normalisable
- si $T = A \to B$: $t \in RED_{A \to B}$ ssi $\forall u \in RED_A \ (t\ u) \in RED_B$
- si $T = A \wedge B$: $t \in RED_{A \wedge B}$ ssi ($\pi^1 t \in RED_A$ et $\pi^2 t \in RED_B$)

Propiétés de la réductibilité (simultanément, par induction sur le type T)

- CR1 si $t \in RED_T$ alors t est fortement normalisable
- CR2 si $t \in RED_T$ et $t \rightsquigarrow t'$ alors $t' \in RED_T$
- CR3 si t est une variable ou une élimination
  et si t ne se réduit qu'en des termes de RED_T
  alors $t \in RED_T$
  En particulier une variable ou une élimination est un terme réductible

Lemme de substitution (par induction sur le terme t) Soit t un terme non forcément réductible de variables libres $x_1, \ldots, x_n$;
soient $u_1, \ldots, u_n$ des termes réductibles de mêmes types que $x_1, \ldots, x_n$;
alors $t[\vec{x} := \vec{u}]$ est réductible.

Du lemme on déduit que tout terme est réductible — en remplaçant les x_i par eux-mêmes — et donc que tout terme est fortement normalisable par CR1.

2.5. Autres propriétés

2.5.1. Propriété de la sous formule

Toute formule d'une déduction naturelle normale est une sous formule d'une hypothèse ou de la conclusion. (**EXO**)

EXO

Propriété à montrer par induction structurelle :

si δ normale alors

1. toute formule de δ est sous formule d'une hypothèse ou de la conclusion
2. si δ se finit par une règle d'élimination, sa conclusion est sous formule d'une hypothèse

2.5.2. Confluence

La réduction est localement confluente et confluente.

Tout démonstration (et donc tout λ-terme simplement typé) a une forme normale unique.

3. Déduction naturelle pour le second ordre

Ce système a les mêmes propriétés: confluence, normalisation forte.
On définit les types ou formules à partir de variables de types X_i ainsi:

- une variable de type est un type
- si T et U sont des types $T \to U$ est un type ($T \wedge U$ aussi, mais on n'en parlera pas, il est définissable au second ordre)
- si T est un type et X une varaible de type $\Pi X.T$ est un type

3.1. Règles, termes et schémas de réduction

Règles structurelles et logiques: les mêmes que précédemment.

3.1.1. Règles de quantification

$$\dfrac{\begin{array}{c}\Gamma\\ \vdots\\ B\end{array}}{\Pi X.B}\ \Pi_i\text{(pas de } X \text{ dans une hyp. libre)} \qquad \dfrac{\vec{g}:\Gamma \vdash t:B}{\vec{g}:\Gamma \vdash \Lambda X.t:B}$$

$$\dfrac{\begin{array}{c}\Gamma\\ \vdots\\ \Pi X.B\end{array}}{B[X := T]}\ \Pi_e \qquad \dfrac{\vec{g}:\Gamma \vdash t:B}{\vec{g}:\Gamma \vdash t\{T\}:B[X := T]}$$

La restriction est nécessaire: montrer que sans elle $\forall X.X$ est démontrable. **EXO**

3.1.2. Schéma de réduction

$$\dfrac{\dfrac{\begin{array}{c}\Gamma\\ \vdots\ \gamma\\ B\end{array}}{\Pi X.B}\ \Pi_i}{B[X := T]}\ \Pi_e$$

$$t^{(\Pi X.B)}\{T\}$$

$\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$ se réduit en $\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow\downarrow$

$$\begin{array}{c}\Gamma[X := T]\\ \vdots\ \gamma[X := T]\\ B[X := T]\end{array}$$

$$t[X := T]^{B[X:=T]}$$

3.2. Définition de types de données dans le système F

Le $\perp$ est définissable par $\Pi X.X$.

$$\frac{\vec{g}:\Gamma \vdash t:\Lambda X.X}{\vec{g}:\Gamma \vdash t:T}\ \Pi_e$$

$A \wedge B$ **est définissable par** $\Pi X.((A \to (B \to X)) \to X)$

$$\dfrac{\dfrac{\dfrac{\dfrac{e:A \to (B \to X) \vdash e:A \to (B \to X) \qquad \vec{g}:\Gamma \vdash a:A}{e:A \to (B \to X), \vec{g}:\Gamma \vdash (e\ a):B \to X}\to_e \qquad \vec{d}:\Delta \vdash b:B}{e:A \to B \to X, \vec{g}:\Gamma, \vec{d}:\Delta \vdash ((e\ a)b):X}\to_e}{\vec{g}:\Gamma, \vec{d}:\Delta \vdash \lambda e.((e\ a)b):(A \to B \to X) \to X}\to_i}{\vec{g}:\Gamma, \vec{d}:\Delta \vdash \Lambda X.(\lambda e.((e\ a)b)):\Pi X.((A \to B \to X) \to X)}\Pi_i$$

$$\dfrac{\dfrac{\vec{g}:\Gamma \vdash t:\Pi X.((A \to (B \to X)) \to X)}{\vec{g}:\Gamma \vdash t\{A\}:(A \to (B \to A)) \to A}\Pi_e \qquad \dfrac{\dfrac{\dfrac{a:A \vdash a:A}{b:B, a:A \vdash a:A}aff}{a:A \vdash \lambda b.a:B \to A}\to_i}{\vdash \lambda a.(\lambda b.a):A \to (B \to A)}\to_i}{\vec{g}:\Gamma \vdash ((t\{A\})\ (\lambda a \lambda b.a)):A}\to_e$$

Le type entier est défini par $N = \Pi X.(X \rightarrow ((X \rightarrow X) \rightarrow X))$

$\bar{0} = \Lambda X \lambda x^X \lambda f^{X \rightarrow X} x$

$\bar{3} = \Lambda X \lambda x^X \lambda f^{X \rightarrow X} (f(f(f\ x)))$

EXO Donner la démonstration (ou le terme correctement typé) correspondant à la fonction successeur.

Le type liste de A est défini par $A\ list = \Pi X.X \rightarrow (A \rightarrow X \rightarrow X) \rightarrow X$

$[3; 0; 3] = \Lambda X \lambda x^X \lambda f^{A \rightarrow X \rightarrow X} ((f\ \bar{3})((f \bar{0})((f\ \bar{3})x)))$

3.3. Fonctions représentables (non traité)

Fonctions représentables sont les fonctions récursives dont la totalité est démontrable dans l'arithmétique de Peano du 2nd ordre.

Exemple de fonction totale non représentable: la fonctionde normalisation F

- F de N dans N définie par: si n est le code de Gödel d'un terme t, alors $F(n) = m$ avec m code de la forme normale de t, et sinon, $F(n) = 0$.
- $\#(n) = m$ où m est le code de Gödel de l'entier de Church $\bar{n}$
- $\flat(n) = m$ si m est le code de Gödel de l'entier de Church $\bar{n}$, et $\flat(m) = 0$ sinon
- $A(m,n) = p$ si m,n,p sont les codes de Gödel des termes m^*,n^*,p^* et si $p^* = (m^* n^*)$

$$D(n) = \flat(N(A(n,\#(n)))) + 1$$

D est récursive totale, montrons qu'il n'y a pas de terme de type $\bar{N} \to \bar{N}$ qui représente D.

Si D est représentée par le terme t de code n, alors $N(A(n,\#(n))$ est le code de Gödel de la forme normale $t\ \bar{n}$ c'est-à-dire t appliqué à l'entier de Church $\bar{n}$.

Mais $(t\bar{n})$ se réduit en l'entier de Church $\overline{D(n)}$ donc $N(A(n,\#(n))) = \#(D(n))$ et $\flat(N(A(n,\#(n)))) = D(n)$ soit $D(n) = D(n+1)$, contradiction.

Toutes les fonctions de codage et l'application sont représentables, c'est donc N qui ne l'est pas.

3.4. Ressemblance / différence avec CaML

- plus de types / par ex. arbre à branchement entiers

$$\Pi X.(X \to (((N \to X) \to X) \to X))$$

(puisque $N = \Pi X.(X \to ((X \to X) \to X))$ on a un quantificateur en position négative, ce qui n'est pas permis par CaML)

- mais toutes les fonctions terminent (pas d'opérateur de point fixe, de let rec)

4. Des systèmes intuitionnistes

La loi de Pierce $P = ((X \to Y) \to X) \to X$ n'est pas démontrable dans les systèmes vus jusqu'ici.

En logique classique elle l'est : si X est vrai, P est vraie, et si X est faux, $X \to Y$ est vrai, $((X \to Y) \to Y)$ est faux et P est aussi vraie.

Par contre on ne peut démontrer X sous l'hypothèse $((X \to Y) \to X)$ en déduction naturelle.

De même, si on s'autorise une constante $\perp$ avec la règle d'élimination (pas de règle d'introduction pour $\perp$!)

$$\frac{g : \Gamma \vdash t : \perp}{g : \Gamma \vdash \perp[t] : C} \perp$$

et qu'on définit $\neg A$ par $A \to \perp$ on ne peut démontrer
$A \vee \neg A$,
$\neg\neg A \to A$,
$(\neg B \to \neg A) \to (A \to B)$ (contraposée)...
Montrer que par contre on a $A \to \neg\neg A$ en logique intuitionniste. **EXO**

D’un point de vue informatique, en logique intuitionniste on a la correspondance

formule	type
démonstration	programme
normalisation	évaluation

D’un point de vue intuitif, il s’agit plus d’une logique de la connaissance que de la vérité:

"Il a raison ou il n’a pas raison"

n’apporte aucune connaissance mais est vrai.

5. Calcul des séquents classique

Un système qui exprime mieux la logique classique et les lois de De Morgan:

- $\neg\neg A \equiv A$
- $\neg(A \wedge B) \equiv \neg A \vee \neg B$
- $\neg(A \vee B) \equiv \neg A \wedge \neg B$

Plusieurs hypothèses, plusieurs conclusions, et la négation échange hypothèses et conclusions (donc pas de calcul de termes).

5.1. Règles

5.1.1. Règles structurelles (contexte)

$$\frac{\Gamma, A, B, \Delta \vdash \Theta}{\Gamma, B, A, \Delta \vdash \Theta}\, Tr_G \qquad \frac{\Theta \vdash \Gamma, A, B, \Delta}{\Theta \vdash \Gamma, B, A, \Delta}\, Tr_R$$

$$\frac{\Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\, Aff_G \qquad \frac{\Theta \vdash \Gamma}{\Theta \vdash A, \Gamma}\, Aff_R$$

$$\frac{A, A, \Gamma \vdash \Theta}{A, \Gamma \vdash \Theta}\, Contr_G \qquad \frac{\Theta \vdash A, A, \Gamma}{\Theta \vdash A, \Gamma}\, Contr_R$$

5.1.2. Axiome

$$A \vdash A$$

5.1.3. Coupure

$$\frac{\Gamma \vdash A,\Theta \quad A,\Gamma' \vdash \Theta'}{\Gamma,\Gamma' \vdash \Theta,\Theta'}\, CUT$$

5.1.4. Règles logiques (connecteurs)

Négation

$$\dfrac{\Gamma \vdash A, \Theta}{\neg A, \Gamma \vdash \Theta}\,\neg_R \qquad\qquad \dfrac{A, \Gamma \vdash \Theta}{\Gamma \vdash \neg A, \Theta}\,\neg_R$$

Disjonction

$$\dfrac{\Gamma \vdash A, B, \Theta}{\Gamma \vdash A \vee B, \Theta}\,\vee_R \qquad\qquad \dfrac{A, \Gamma \vdash \Theta \quad B, \Gamma' \vdash \Theta'}{A \vee B, \Gamma, \Gamma' \vdash \Theta, \Theta'}\,\vee_G$$

Conjonction

$$\dfrac{A, B, \Gamma \vdash \Theta}{A \wedge B, \Gamma \vdash \Theta}\,\wedge_G \qquad\qquad \dfrac{\Gamma \vdash A, \Theta \quad \Gamma' \vdash B, \Theta'}{\Gamma, \Gamma' \vdash A \wedge B, \Theta, \Theta'}\,\wedge_R$$

Implication

$$\dfrac{A, \Gamma \vdash B, \Theta}{\Gamma \vdash A \longrightarrow B, \Theta}\,\longrightarrow_G \qquad\qquad \dfrac{B, \Gamma \vdash \Theta \quad \Gamma' \vdash A, \Theta'}{A \longrightarrow B, \Gamma, \Gamma' \vdash \Theta, \Theta'}$$

5.2. Propriétés du calcul des séquents

5.2.1. Complétude Un séquent $A_1, \ldots, A_n \vdash B_1, \ldots, B_p$ est démontrable si et seulement si la formule $(A_1 \wedge \cdots \wedge A_n) \rightarrow (B_1 \vee \cdots \vee B_p)$ est vraie pour toute interprétation. (Il est aisé de voir que dans la variante proposée ci-après, les règles sont réversibles: pour une interprétation donnée, le séquent est conclusion est valide si et seulement si les séquents prémisses le sont.)

5.2.2. Elimination des coupures La règle de coupure est redondante: toute séquent démontrable peut être démontré sans utiliser la règle de coupure. Il s'agit de normalisation **faible**.

Idée de la démonstration :

- lorsque la formule qui disparaît vient d'être introduite de part et d'autre, on peut remplacer la coupure par une ou deux coupures plus petites. **EXO**
- étant données deux démonstrations de $\Gamma \vdash \Delta$ et $\Gamma' \vdash \Delta'$ dont les coupures sopnt de degré $\leq d$ et une formule C de degré d, on peut faire une démonstration de $\Gamma,\Gamma' - C \vdash \Delta - C,\Delta'$ dont toute les coupures sont de degré $\leq d$
- ensuite on procède par induction sur la hauteur de la démonstration

5.2.3. Propriété de la sous-formule On en déduit: toute séquent est démontrable par des séquents qui ne contiennent que des sous-formules des formules du séquent démontré, le calcul propositionnel est décidable (la règle de coupure est la seule dont les prémisses contiennent une formule qui ne soit pas sous-formule du séquent conclusion).

5.3. Une variante pour la complétude

- Pas de règles structurelles.
- Axiomes: $\mathbf{p}, \ldots \vdash \mathbf{p}, \ldots$ (une lettre en commun)
- Dans toutes les règles binaires, les contextes sont identiques :

$$\frac{A, \Gamma \vdash \Theta \quad B, \Gamma \vdash \Theta}{A \vee B, \Gamma \vdash \Theta}\vee_G \qquad \frac{\Gamma \vdash A, \Theta \quad \Gamma \vdash B, \Theta}{\Gamma \vdash A \wedge B, \Theta}\wedge_R \qquad \frac{B, \Gamma \vdash \Theta \quad \Gamma \vdash A, \Theta}{A \to B, \Gamma \vdash \Theta}$$

Pour une valuation donnée, la conclusion est valide si et seulement si les prémisses le sont.

EXO équivalence avec le calcul précédent sans coupure
EXO décidabilité
EXO complétude

5.3.1. Exemple: Recherche de démonstration pour un séquent démontrable

$$\dfrac{\dfrac{\dfrac{\mathbf{q},p,p,s \vdash \mathbf{q},t,t}{q,p,(\neg q),p,s \vdash t,t}\neg_g}{q,p,(\neg q),(p \wedge s) \vdash t,t}\wedge g \qquad \dfrac{\dfrac{r,\mathbf{q},p,p,s \vdash \mathbf{q},t}{r,q,p,(\neg q),p,s \vdash t}\neg_g}{r,q,p,(\neg q),(p \wedge s) \vdash t}\wedge g}{q,p,(\neg q),(t \rightarrow r),(p \wedge s) \vdash t}\rightarrow_g$$

5.3.2. Exemple bis: recherche de démonstration pour un séquent non démontrable

$$
\dfrac{
\dfrac{\dfrac{\vdash p,q}{(\neg p) \vdash q}\ \neg_g}{\vdash ((\neg p) \longrightarrow q)}\ \longrightarrow_d
\qquad
\dfrac{\dfrac{r \vdash}{\vdash (\neg r)}\ \neg_d \qquad s \vdash}{((\neg r) \longrightarrow s) \vdash}\ \longrightarrow_g
}{
(((\neg p) \longrightarrow q) \longrightarrow ((\neg r) \longrightarrow s)) \vdash
}\ \longrightarrow_g
$$

5.4. Calcul des séquents intuitionniste

Une restriction sur la structure des séquents (et non sur les règles): au plus une formule à droite.

Parmi les règles structurelles, seules les règles gauches restent, et la coupure devient la composition:

$$\frac{\Gamma \vdash A \quad A\Delta \vdash C}{\Gamma,\Delta \vdash C}\,CUT$$

Pour les connecteurs $\wedge$ et $\rightarrow$ on obtient:

$$\frac{A,B,\Gamma \vdash C}{A \wedge B,\Gamma \vdash C}\wedge_G \qquad \text{Conjonction} \qquad \frac{\Gamma \vdash A \quad \Gamma' \vdash B}{\Gamma,\Gamma' \vdash A \wedge B}\wedge_R$$

$$\frac{A,\Gamma \vdash B}{\Gamma \vdash A \rightarrow B}\rightarrow_G \qquad \text{Implication} \qquad \frac{B,\Gamma \vdash C \quad \Gamma' \vdash A}{A \rightarrow B,\Gamma,\Gamma' \vdash C}$$

Ce calcul est équivalent à la déduction naturelle, les règles gauches peuvent être simulées en déduction naturelle. **EXO**

6. Logique linéaire

6.1. Logique linéaire intuitionniste multiplicative

On aimerait que l'implication modélise un changement d'état:

$$A, A \to B \vdash B$$

Etat A
Transition permettant de passer de l'état A à l'état B
Etat final B.

Les logiques vues jusqu'ici ne conviennent pas, puisque

$$A, A \to B \vdash A \wedge B$$

A est toujours vrai après avoir effectué la transition!

Comme dans le passage de la logique classique à la logique intuitionniste, ce n'est pas les règles logique qu'il faut modifier mais les règles structurelles.

$$\frac{\Theta, A, B, \Gamma \vdash C}{\Theta, B, A, \Gamma \vdash C}\ E_g$$

échange

! pas d'autre règle structurelle !

$$\frac{axiome}{A \vdash A}$$

$$\frac{\Gamma \vdash X \quad X, \Gamma' \vdash C'}{\Gamma, \Gamma' \vdash C'}\ cut$$

$$\frac{A, B, \Gamma \vdash C}{A \mathbin{\underline{\triangle}} B, \Gamma \vdash C}\ \underline{\triangle}_g$$

conjonction

$$\frac{\Gamma \vdash A \quad \Gamma' \vdash B}{\Gamma, \Gamma' \vdash A \mathbin{\underline{\triangle}} B}\ \underline{\triangle}_d$$

$$\frac{B, \Gamma \vdash C \quad \Gamma' \vdash A}{\Gamma', A \multimap B, \Gamma \vdash C}\ \multimap_g$$

implication

$$\frac{A, \Gamma \vdash B}{\Gamma \vdash A \multimap B}\ \multimap_d$$

6.2. La modalité !

Il faut aussi pouvoir utiliser ad libitum une transition.

Pour cela on dispose d'une modalité notée « ! »

$!A$: « la ressource A est disponible autant qu'on le souhaite »

$$\frac{!X, !X, \Gamma \vdash C}{!X, \Gamma \vdash C}\ !_g{}^c \qquad \frac{\Gamma \vdash C}{!X, \Gamma \vdash C}\ !_g{}^a \qquad \frac{!\Gamma \vdash C}{!\Gamma \vdash !C}\ !_d$$

$!\Gamma$ désigne une suite de formules toutes précédées de la modalité « ! ».

6.3. Représentation des réseaux de Petri

Les places sont représentées par des variables propositionnelles.

Décrivons un marquage par une conjonction:
$A \mathbin{\underline{\triangle}} A \mathbin{\underline{\triangle}} B \mathbin{\underline{\triangle}} C \mathbin{\underline{\triangle}} C$ 2 jetons en A, 1 en B, 2 en C

Décrivons une transition par une implication linéaire:
$(A \mathbin{\underline{\triangle}} A \mathbin{\underline{\triangle}} B) \multimap (C \mathbin{\underline{\triangle}} C \mathbin{\underline{\triangle}} D \mathbin{\underline{\triangle}} E)$ est une transition qui prend deux jetons en A et un en B, et qui produit deux jetons en C, un en D et un en E.

Dans un réseau de Petri dont les transitions sont $t_1, \ldots, t_n$, on peut accéder d'un marquage M à un marquage M' ssi on peut démontrer en logique linéaire intuitionniste

$$M, !t_1, \ldots, !t_n \vdash M'$$

6.4. Représentation de la logique intuitionniste

On peut coder en logique linéaire les formules de la logique intuitionniste (restreinte à l'implication) ainsi:

- $p^{\#} = p$ si p est une variable propositionnelle
- $(A \rightarrow B)^{\#} = (!(A^{\#})) \multimap B^{\#}$

On alors (**EXO**) :

$H_1, \ldots, H_n \vdash C$ est démontrable en logique intuitionniste
ssi
$!H_1^{\#}, \ldots, !H_n^{\#} \vdash C^{\#}$ est démontrable en logique linéaire intuitionniste

6.5. Logique linéaire classique

La logique linéaire intuitionniste admet une version classique,
avec une négation involutive.

Une disjonction est associée à la conjonction,
et une modalité $?$ correspond à $!$.

Elle satisfait alors les lois de de Morgan.

Comme pour la logique classique es régles manipulent des séquents avec plusieurs formules à gauche et à droite.

Elles sont obtenues par symétries: la règle droite (resp. gauche) de la disjonction correspond à la règle gauche (resp. droite) de la conjonction, les règles droites (resp. gauches) de $?$ correspondent aux règles gauches (resp. droites) de $!$, et la négation tranforme une hypothèse en conclusion ou vice-versa.

6.6. Calcul de Lambek

Si de la logique linéaire intuitonniste on omet la modalité !, et qu'on supprime la règle d'échange, on obtient le calcul de Lambek.

On a alors deux implications, suivant que l'hypothèse annulée est à droite ou à gauche de la suite de formules:

$$A \vdash A$$

$$\frac{\Gamma \vdash A \quad \Delta, A, \Delta' \vdash B}{\Delta, \Gamma, \Delta' \vdash B}\ cut$$

$$\frac{\Gamma, A \vdash C}{\Gamma \vdash C \circ\!- A}\ \circ\!-_i$$

$$\frac{\Gamma, B, \Gamma' \vdash C \quad \Delta \vdash A}{\Gamma, B \circ\!- A, \Delta, \Gamma' \vdash C}\ \circ\!-_h$$

$$\frac{A, \Gamma \vdash C}{\Gamma \vdash A \multimap C}\ \multimap_i$$

$$\frac{\Delta \vdash A \quad \Gamma, B, \Gamma' \vdash C}{\Gamma, \Delta, A \multimap B, \Gamma' \vdash C}\ \multimap_h$$

$$\frac{\Gamma, A, B, \Gamma' \vdash C}{\Gamma, A \,\underline{\triangle}\, B, \Gamma' \vdash C}\ \underline{\triangle}_g$$

$$\frac{\Gamma \vdash A \qquad \Gamma' \vdash B}{\Gamma, \Gamma' \vdash A \,\underline{\triangle}\, B}\ \underline{\triangle}_d$$

Le calcul de Lambek peut-être utilisé comme une grammaire,
si on se donne une varaible propositionnelle distinguée S.

Lex: fonction de l'ensemble des terminaux dans l'ensemble des ensembles finis de formules écrites avec les connecteurs « $\multimap$ », « $\circ\!\!-$ », « $\underline{\wedge}$ ».
($\mathrm{Lex}(a)$ est un ensemble fini de formules, pour tout terminal a)

Etant donnée une telle fonction Lex le langage $\mathcal{L}(\mathrm{Lex})$ est défini par:

$$a_1 a_2 \cdots a_n \in \mathcal{L}(\mathrm{Lex})$$
ssi
$\forall i \exists t_i \in \mathrm{Lex}(a_i)$ tel que $t_1, \ldots, t_n \vdash S$ soit démontrable dans le calcul de Lambek

On décrit ainsi les langages hors-contextes (ou algébriques) . . .
. . . cf. le cours traitement symbolique des langues.

Références

[1] Jean-Yves Girard, Yves Lafont, and Paul Taylor. *Proofs and Types*. Cambridge University Press, 1988. Cours de DEA U. Paris 7 87/88, clair et vivant

[2] Jean-Yves Girard. La logique linéaire. *Pour La Science, Edition Française de "Scientific American"*, 150, avril 1990. vulgarisation bien faite: compréhensible, pas vide et vivante

[3] Jean-Yves Girard. Linear logic: its syntax and semantics. In Jean-Yves Girard, Yves Lafont, and Laurent Regnier, editors, *Advances in Linear Logic*, pages 1–42. Cambridge University Press, 1995. Excellente synthèse sur la Logique Linéaire, plus abordable que l'article original TCS 87

[4] Joachim Lambek. The mathematics of sentence structure. *American mathematical monthly*, 65:154–169, 1958. Article original remarquablement clair et facile à lire