

Une introduction à la réalisabilité

Alexandre Miquel

24 octobre 2007

Introduction (point de vue du hacker)

De nombreuses situations où un programme mal typé est correct vis-à-vis de l'exécution :

```
let ma_fonction_inutile n =  
  if n * n + 1 = 0 then 42 else true
```

This expression has type `bool` but is here used with type `int`

Pourtant `ma_fonction_inutile` retourne toujours un `bool` quand on l'applique à un `int`...

Deux notions différentes de correction :

- 1 Correction vis-à-vis du typage
- 2 Correction vis-à-vis de l'exécution \rightsquigarrow **Réalisabilité**

Le système T : parties communes

Syntaxe

| | |
|---------------|---------------------------------------------------------------------------|
| Types | $A, B ::= \text{nat} \mid A \rightarrow B$ |
| Termes | $M, N ::= x \mid \lambda x. M \mid MN$ $\mid 0 \mid s \mid \text{rec}$ |

Règles de réduction

| | | |
|-----------------------------|---------|---------------------------------|
| $(\lambda x. M)N$ | \succ | $M\{x := N\}$ |
| $\text{rec } M_0 M_1 0$ | \succ | M_0 |
| $\text{rec } M_0 M_1 (s N)$ | \succ | $M_1 N (\text{rec } M_0 M_1 N)$ |

Le système T : typage (1/2)

$$\frac{}{\Gamma \vdash x : A} \quad (x:A) \in \Gamma \qquad \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x. M : A \rightarrow B}$$
$$\frac{\Gamma \vdash M : A \rightarrow B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$
$$\frac{}{\Gamma \vdash 0 : \text{nat}} \qquad \frac{}{\Gamma \vdash s : \text{nat} \rightarrow \text{nat}}$$
$$\frac{}{\Gamma \vdash \text{rec} : A \rightarrow (\text{nat} \rightarrow A \rightarrow A) \rightarrow \text{nat} \rightarrow A}$$

- Le typage porte sur des termes *ouverts* \rightsquigarrow contextes de typage
- Justification simple : dérivation de typage
- Inférence/vérification de type décidables (dirigées par la syntaxe)
- La réduction n'est jamais mentionnée...
... quelle garantie vis-à-vis du calcul ?

Le système T : typage (2/2)

3 lemmes garantissent la correction vis-à-vis du calcul :

1. Subject reduction

Si $\Gamma \vdash M : A$ et $M \succ M'$, alors $\Gamma \vdash M' : A$

2. Valeurs de type nat

Si $\Gamma \vdash M : \text{nat}$ et si M est une valeur, alors $M = s^n 0$

(valeur = forme normale close)

3. Normalisation (forte)

Si $\Gamma \vdash M : A$, alors M est (fortement) normalisable

$1 + 2 + 3 \Rightarrow$ Tout programme clos : nat se réduit sur un naturel

Remarque : dans le cas du système T , pas besoin de **lemme de progression**

Le système T : réalisabilité

Relation binaire $M \Vdash A$ (M terme **clos**)

Définition de la réalisabilité

- 1 $M \Vdash \text{nat}$ si $M \succ^* s^n 0$
- 2 $M \Vdash A \rightarrow B$ si $N \Vdash A$ entraîne $MN \Vdash B$ (pour tout N)

- Termes clos \rightsquigarrow pas de contexte
- Définition purement calculatoire : **syntaxe = boîte noire**
- Pas de correction à démontrer : tout est dans la définition !
(Ensemble des réalisateurs de A clos par réduction et anti-réduction)
- Pas de justification élémentaire (telle qu'une dérivation)
 \rightsquigarrow recours à une **justification externe** : preuve
- Relation $M \Vdash A$ **indécidable**, non récursivement énumérable

Le système T : typage et réalisabilité

Lemme d'adéquation

Si $x_1 : A_1, \dots, x_k : A_k \vdash M : B$, alors pour tous N_1, \dots, N_k
 $N_1 \Vdash A_1, \dots, N_k \Vdash A_k$ entraîne $M\{x_1 := N_1; \dots; x_k := N_k\} \Vdash B$

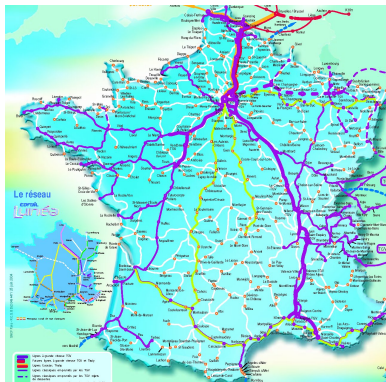
Preuve : induction immédiate sur la dérivation.

Le cas du λ repose sur la propriété de clôture par anti-réduction.

Cas particulier (contexte vide) : $\vdash M : A$ implique $M \Vdash A$

- Typage + adéquation \rightsquigarrow
tout programme clos : nat se réduit vers un naturel
(Sans passer par 1 + 2 + 3. En fait, 3 (SN) se prouve par réalisabilité.)
- Seuls de petits réalisateurs sont construits à la main...
... pour le reste, on passe par typage + lemme d'adéquation

Réalisabilité et typage (métaphore)



Première partie I

Réalisabilité intuitionniste (Kleene)

Réalisabilité de Kleene : les bases

Mathématiser l'idée de constructivité au sens de Brouwer

- 1908. Brouwer : *De la non fiabilité des principes de la logique*
(Principes de l'intuitionnisme)
- 1936. Church : *An Unsolvable Problem of Elementary Number Theory*
(Application du λ -calcul à l'*Entscheidungsproblem*)
- 1936. Turing : *On Computable Numbers, with an Application to the Entscheidungsproblem*
- 1936. Kleene : *λ -definability and recursiveness*
- 1945. Kleene : *On the Interpretation of Intuitionistic Number Theory*

Relation de réalisabilité : $n \Vdash A$

- n = entier naturel (simple donnée, ou numéro de fonction récursive)
- A = formule close de l'arithmétique (Heyting)

Réalisabilité de Kleene : les ingrédients

Réalisabilité paramétrée par :

- Injection récursive : $(n, p) \mapsto \langle n, p \rangle : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$

Par ex. : $\langle n; p \rangle = 2^n \times 3^p$

- Énumération $(\phi_n)_{n \in \mathbb{N}}$ de toutes les **fonctions récursives partielles** d'arité 1 (i.e. de \mathbb{N} dans \mathbb{N})

Application de Kleene : $n \cdot p = \phi_n(p)$ (si défini)

Langage de l'arithmétique (1er ordre)

- Symbole de prédicat : $=$
- Symboles de constante/fonction : $0, s, +, \times$
(Éventuellement : toutes les fonctions primitives récursives)

Réalisabilité de Kleene : définition informelle

- $n \Vdash t_1 = t_2 \quad \equiv \quad n = 0 \quad \text{et} \quad t_1 = t_2 \text{ vrai}$
- $n \Vdash \top \quad \equiv \quad n = 0$
- $n \Vdash \perp \quad \equiv \quad \text{contradiction}$
- $n \Vdash A \Rightarrow B \quad \equiv \quad p \Vdash A \text{ implique } n \cdot p \Vdash B \quad (\text{pour tout } p)$
- $n \Vdash A \wedge B \quad \equiv \quad n = \langle p; q \rangle, \text{ avec } p \Vdash A \text{ et } q \Vdash B$
- $n \Vdash A \vee B \quad \equiv \quad (n = \langle 0; p \rangle, p \Vdash A) \text{ ou } (n = \langle 1; q \rangle, q \Vdash B)$
- $n \Vdash \forall^{\mathbb{N}} x A(x) \quad \equiv \quad n \cdot p \Vdash A(p) \text{ pour tout } p$
- $n \Vdash \exists^{\mathbb{N}} x A(x) \quad \equiv \quad n = \langle p, q \rangle, \text{ avec } q \Vdash A(p)$

- $n \cdot p \Vdash \dots$ signifie « $n \cdot p$ défini **et** $n \cdot p \Vdash \dots$ »
- $\neg A$ défini comme $A \Rightarrow \perp$

Réalisabilité de Kleene : ensembles de réalisateurs

Si on note $A^* = \{n \in \mathbb{N} \mid n \Vdash A\}$, la définition devient :

$$(t_1 = t_2)^* = \{0\} \text{ si } t_1 = t_2 \text{ vrai} / \emptyset \text{ sinon}$$

$$(\top)^* = \{0\}$$

$$(\perp)^* = \emptyset$$

$$(A \Rightarrow B)^* = A^* \rightarrow B^* = \{n \in \mathbb{N} \mid \forall p (p \in A^* \Rightarrow n \cdot p \in B^*)\}$$

$$(A \wedge B)^* = A^* \times B^* = \{\langle p; q \rangle \mid p \in A^*, q \in B^*\}$$

$$(A \vee B)^* = A^* + B^* = \{0\} \times A^* \cup \{1\} \times B^*$$

$$(\forall^{\mathbb{N}} x A(x))^* = \prod_{p \in \mathbb{N}} (A(p))^* = \{n \in \mathbb{N} \mid \forall p \in \mathbb{N} n \cdot p \in (A(p))^*\}$$

$$(\exists^{\mathbb{N}} x A(x))^* = \sum_{p \in \mathbb{N}} (A(p))^* = \{\langle p, q \rangle \mid q \in (A(p))^*\}$$

Réalisabilité de Kleene : définition formelle

Une transformation **syntactique**

qui à chaque formule $A(x_1, \dots, x_k)$ (k var. libres)

associe une autre formule $n \Vdash A(x_1, \dots, x_k)$ ($k + 1$ var. libres)

$$n \Vdash t_1 = t_2 \equiv (n = 0) \wedge (t_1 = t_2)$$

$$n \Vdash \top \equiv n = 0$$

$$n \Vdash \perp \equiv \perp$$

$$n \Vdash A \Rightarrow B \equiv \forall^{\mathbb{N}} p (p \Vdash A \Rightarrow n \cdot p \Vdash B)$$

$$n \Vdash A \wedge B \equiv \exists^{\mathbb{N}} p \exists^{\mathbb{N}} q (n = \langle p, q \rangle \wedge p \Vdash A \wedge q \Vdash B)$$

$$n \Vdash A \vee B \equiv \exists^{\mathbb{N}} p (n = \langle 0, p \rangle \wedge p \Vdash A) \vee \exists^{\mathbb{N}} q (n = \langle 1, q \rangle \wedge q \Vdash B)$$

$$n \Vdash \forall^{\mathbb{N}} x A(x) \equiv \forall^{\mathbb{N}} p (n \cdot p \Vdash A(p))$$

$$n \Vdash \exists^{\mathbb{N}} x A(x) \equiv \exists^{\mathbb{N}} p \exists^{\mathbb{N}} q (n = \langle p, q \rangle \wedge q \Vdash A(p))$$

Réalisabilité de Kleene : exemple

$$n \Vdash \forall^{\mathbb{N}} x \exists^{\mathbb{N}} y (x = 2y \vee x = 2y + 1) \equiv$$

$$\begin{aligned} & \forall^{\mathbb{N}} x \exists^{\mathbb{N}} y \exists^{\mathbb{N}} z [\\ & \quad n \cdot x = \langle y, z \rangle \wedge \\ & \quad (\exists^{\mathbb{N}} p (z = \langle 0; p \rangle \wedge p = 0 \wedge x = 2y) \vee \\ & \quad \exists^{\mathbb{N}} q (z = \langle 1; q \rangle \wedge q = 0 \wedge x = 2y + 1)) \\ &] \end{aligned}$$

Réalisabilité de Kleene : le résultat

Théorème

À chaque preuve $\pi : (\text{HA} \vdash A)$ (où A est une formule close) on sait associer un entier naturel n_π tel que $\text{HA} \vdash n_\pi \Vdash A$

Preuve. Il s'agit de compiler la preuve π en un code n_π de fonction récursive à travers la correspondance de Curry-Howard. Seules difficultés :

- La lourdeur du système formel (déduction à la Hilbert)
- La lourdeur des codages (gödelite)

Transformation $\pi \mapsto n_\pi$ calculable... mais impraticable

In Gödel numbers we trust?

Corollaires

Propriété du témoin faible

$HA \vdash \exists^{\mathbb{N}} x A(x) \rightsquigarrow$ deux entiers p, q tels que $q \Vdash A(p)$

Propriété de la disjonction faible

$HA \vdash A \vee B \rightsquigarrow$ deux entiers p, q tels que

$p = 0$ et $q \Vdash A$ ou $p = 1$ et $q \Vdash B$

Calculabilité des fonctions totales

$HA \vdash \forall^{\mathbb{N}} x \exists^{\mathbb{N}} y A(x, y) \rightsquigarrow$ deux fonct. réc. totales f, g telles que

$g(n) \Vdash A(n, f(n))$ pour tout $n \in \mathbb{N}$

Attention ! A réalisable $\not\Rightarrow$ A prouvable (dans HA)

(Pour l'équivalence, on utilise la **\mathcal{P} -réalisabilité**)

Réalisable, mais pas prouvable

Principe de Markov (MP)

$$\Vdash \forall^{\mathbb{N}} x (A(x) \vee \neg A(x)) \wedge \neg \neg \exists^{\mathbb{N}} x A(x) \Rightarrow \exists^{\mathbb{N}} x A(x)$$

- Le prédicat $T(n, x, z) \equiv$ « z est la trace du calcul de $n \cdot x$ »
- La fonction $U(z) =$ « résultat du calcul de trace z »

sont définissables dans HA (et prim. réc.)

Permet d'exprimer : $n \cdot x = y \equiv \exists z (T(n, x, z) \wedge U(z) = y)$

Thèse de Church (CT)

$$\Vdash \forall^{\mathbb{N}} x \exists^{\mathbb{N}} y A(x, y) \Rightarrow \exists^{\mathbb{N}} n \forall^{\mathbb{N}} x \exists z (T(n, x, z) \wedge A(x, U(z)))$$

Pour en finir avec la gödelite (1/2)

Algèbre combinatoire partielle (PCA)

Un ensemble \mathcal{A} muni de :

- 1 Une fonction partielle $(\cdot) : \mathcal{A} \times \mathcal{A} \rightarrow \mathcal{A}$ (application)
- 2 Un élément $\mathbf{K} \in \mathcal{A}$ t.q. pour tous $x, y \in \mathcal{A}$:

$$(\mathbf{K} \cdot x) \cdot y \text{ défini} \quad \wedge \quad (\mathbf{K} \cdot x) \cdot y = x$$

- 3 Un élément $\mathbf{S} \in \mathcal{A}$ t.q. pour tous $x, y, z \in \mathcal{A}$:

$$(x \cdot z) \cdot (y \cdot z) \text{ défini} \quad \Rightarrow \quad ((\mathbf{S} \cdot x) \cdot y) \cdot z \text{ défini} \quad \wedge \\ ((\mathbf{S} \cdot x) \cdot y) \cdot z = (x \cdot z) \cdot (y \cdot z)$$

(On suppose $\mathbf{S} \cdot x$ et $(\mathbf{S} \cdot x) \cdot y$ toujours définis)

Permet de recoder : couples + entiers (codage à la Church)

Pour en finir avec la gödelite (2/2)

Exemples de PCAs :

- ω + application de Kleene (partielle)
- λ -termes clos ($/ \beta$ -reduction) + application (totale)
- λ -termes clos **normaux** + application normalisée (**partielle**)
- Variantes : termes clos normalisables du système T , etc.

Théorème (pour une PCA \mathcal{A} quelconque)

À chaque preuve $\pi : (HA \vdash A)$ (A formule close)
on sait associer un combinateur $c_\pi \in \mathcal{A}$ tel que $c_\pi \Vdash A$

- Principe de Markov (MP) et/ou Thèse de Church (CT)
pas toujours réalisables dans \mathcal{A} ...

Extension catégorique : ω -sets (A-sets)

Catégorie des ω -sets :

- **Objets** : $X = (|X|, \Vdash_X)$ avec
 - $\Vdash_X \subseteq \omega \times |X|$
 - $\forall x \in |X| \exists n \in \omega \quad n \Vdash_X x$
- **Flèches** : fonctions $f : |X| \rightarrow |Y|$ t.q. :
 - $\exists n \in \omega \quad \forall x \in |X| \quad \forall p \in \omega \quad (p \Vdash_X x \Rightarrow n \cdot p \Vdash_Y f(x))$

- ω -Set = ccc (en fait : quasi-topos)
- Sous-catégories intéressantes : **PER**, **Set**
- Produit dépendant : $\prod(x \in X, Y_x)$
- Modèle du calcul des constructions (voire de CIC) :

$$\llbracket \text{Prop} \rrbracket = \mathbf{PER} \qquad \llbracket \text{Type} \rrbracket = \omega\text{-Set} \quad (\text{en gros})$$

- Généralisation aux PCAs : **A-sets**

Quantificateurs paramétriques

On ajoute au langage

- Des quantificateurs **paramétriques** \forall, \exists
- Un nouveau symbole de prédicat $\text{Nat}(_)$

Extension de la réalisabilité

- $n \Vdash \forall x A(x) \equiv \forall p \in \mathbb{N} \ n \Vdash A(p)$
- $n \Vdash \exists x A(x) \equiv \exists p \in \mathbb{N} \ n \Vdash A(p)$
- $n \Vdash \text{Nat}(p) \equiv n = p$

Lemme de décomposition

$$\begin{aligned} n \Vdash \forall^{\mathbb{N}} x A(x) &\Leftrightarrow n \Vdash \forall x (\text{Nat}(x) \Rightarrow A(x)) \\ n \Vdash \exists^{\mathbb{N}} x A(x) &\Leftrightarrow n \Vdash \exists x (\text{Nat}(x) \wedge A(x)) \end{aligned}$$

- Présentation de HA modifiée en conséquence
- Récurrence valable uniquement avec $\forall^{\mathbb{N}}$

Les dessous de tapis

Commutation \forall/\vee (Remarque de Girard)

$$n \Vdash \forall x (A(x) \vee B(x)) \quad \Leftrightarrow \quad n \Vdash \forall x A(x) \vee \forall x B(x)$$

- Équivalence **fausse** en logique classique !
- Ne marche plus si on remplace \forall par $\forall^{\mathbb{N}}$ (ouf !)
- Une logique de la réalisabilité ? (encore plus intuitionniste)

Pauvreté de la (double) négation

$$(\neg A)^* = \begin{cases} \emptyset & \text{si } A \text{ réalisable} \\ \mathbb{N} & \text{sinon} \end{cases} \quad (\neg\neg A)^* = \begin{cases} \mathbb{N} & \text{si } A \text{ réalisable} \\ \emptyset & \text{sinon} \end{cases}$$

- Réalisabilité triviale sur les formules classiques
- Conséquence du (mauvais) choix de design : $\perp^* = \emptyset$

Deuxième partie II

Réalisabilité classique (Krivine)

Le λ_c calcul (Krivine)

Syntaxe

| | |
|------------------|--------------------------------------------------------------------------------|
| Termes | $t, u ::= x \mid \lambda x. t \mid tu \mid \mathfrak{c} \mid k_\pi \mid \dots$ |
| Piles | $\pi ::= \diamond \mid u \cdot \pi \quad (u, \pi \text{ clos})$ |
| Processus | $p, q ::= t \star \pi \quad (t, \pi \text{ clos})$ |

- \mathfrak{c} = call-cc
- k_π = continuation (*throw*) = **para-preuve**
- **Quasi-preuve** = terme sans k_π (= « interdit de tricher »)

Évaluation (normalisation de tête faible)

| | |
|-----------|-------------------------------------------------------------------------------|
| (Push) | $tu \star \pi \quad \gamma \quad t \star u \cdot \pi$ |
| (Grab) | $\lambda \xi. t \star u \cdot \pi \quad \gamma \quad t\{\xi := u\} \star \pi$ |
| (Save) | $\mathfrak{c} \star t \cdot \pi \quad \gamma \quad t \star k_\pi \cdot \pi$ |
| (Restore) | $k_\pi \star t \cdot \pi' \quad \gamma \quad t \star \pi$ |

Le rôle des parapreuves (métaphore)



Georges de la Tour. *Le tricheur à l'as de carreau*

Arithmétique du second ordre : le langage

Syntaxe

Individus $e, e' ::= x \mid f(e_1, \dots, e_n)$

Formules $A, B ::= X(e_1, \dots, e_n) \mid A \Rightarrow B \mid \forall x A \mid \forall X A$

- Variables du 1er ordre (= variables d'individu) : x, y, z , etc.
- Variables du 2nd ordre (= variables de prédicat) : X, Y, Z , etc.
... pour toutes les arités
- Un symbole de fonction f pour chaque fonction prim. réc. (dont $0, s, +, \times$)

Codages au 2nd ordre

$\perp \equiv \forall Z Z$

$A \wedge B \equiv \forall Z ((A \Rightarrow B \Rightarrow Z) \Rightarrow Z)$

$A \vee B \equiv \forall Z ((A \Rightarrow Z) \Rightarrow (B \Rightarrow Z) \Rightarrow Z)$

$\exists x A(x) \equiv \forall Z (\forall x (A(x) \Rightarrow Z) \Rightarrow Z)$

$\exists X A(X) \equiv \forall Z (\forall X (A(X) \Rightarrow Z) \Rightarrow Z)$

$e_1 = e_2 \equiv \forall Z (Z(e_1) \Rightarrow Z(e_2))$

$\text{nat}(x) \equiv \forall Z (Z(0) \Rightarrow \forall y (Z(y) \Rightarrow Z(s(y)))) \Rightarrow Z(x)$

Réalisabilité classique : les principes

- Intuition : terme = “preuve” / pile = “contre-preuve”
- Construction paramétrée par un ensemble $\perp \subseteq \Lambda \star \Pi$ clos par anti-réduction ($p \succ p'$, $p' \in \perp$ implique $p \in \perp$)
- Chaque formule A est interprétée par deux ensembles :

| | | |
|--------------------|-------------------------|----------------------------|
| Valeur de vérité | $ A \subseteq \Lambda$ | (“preuves” de A) |
| Valeur de fausseté | $\ A\ \subseteq \Pi$ | (“contre-preuves” de A) |

- Valeur de vérité $|A|$ définie (uniformément) par orthogonalité :

$$|A| = \|A\|^\perp = \{t \in \Lambda \mid \forall \pi \in \|A\| \quad t \star \pi \in \perp\}$$

Cas particulier : $\perp = \emptyset \Rightarrow$ Modèle standard (vrai/faux)

Réalisabilité classique en arithmétique du 2nd ordre

Formules de PA2 (rappel)

$$A, B ::= X(e_1, \dots, e_n) \mid A \Rightarrow B \mid \forall x A \mid \forall X A$$

- Var. du 2nd ordre interprétées par des fonctions $F : \mathbb{N}^n \rightarrow \mathfrak{P}(\Pi)$
- Valeur de fausseté $\|A\|$ définie par induction sur A :

$$\|A \Rightarrow B\| = |A| \cdot \|B\| = \{t \cdot \pi \mid t \in |A|, \pi \in \|B\|\}$$

$$\|\forall x A(x)\| = \bigcup_{n \in \mathbb{N}} \|A(n)\| \qquad \|\forall X A(X)\| = \bigcup_{F: \mathbb{N}^n \rightarrow \mathfrak{P}(\Pi)} \|A(F^\perp)\|$$

Le langage est enrichi avec une constante F^\perp pour chaque fonction $F : \mathbb{N}^n \rightarrow \mathfrak{P}(\Pi)$

- Valeur de vérité $|A|$ définie par

$$|A| = \|A\|^\perp = \{t \in \Lambda \mid \forall \pi \in \|A\| \quad t \star \pi \in \perp\}$$

Notation et remarques

Notation : $t \Vdash A \equiv t \in |A|$

- Dépend du choix de $\perp\!\!\!\perp$: $t \Vdash_{\perp\!\!\!\perp} A$
- $t \Vdash A$ signifie parfois : « $t \Vdash_{\perp\!\!\!\perp} A$ pour tout choix de $\perp\!\!\!\perp$ »

Cas où $\perp\!\!\!\perp = \emptyset$ (réalisabilité dégénérée)

- Le modèle de réalisabilité « calcule » la valeur de vérité au sens du modèle standard

$$\|A\| = \begin{cases} \Lambda & \text{si } \llbracket A \rrbracket = 1 \\ \emptyset & \text{si } \llbracket A \rrbracket = 0 \end{cases}$$

- **Réalisable** \Leftrightarrow **Vrai dans le modèle standard**

Cas où $\perp\!\!\!\perp \neq \emptyset$ (réalisabilité ordinaire)

- Existence de réalisateurs universels : $k_{\pi_0} t_0$, où $t_0 \star \pi_0 \in \perp\!\!\!\perp$
- On recherche des réalisateurs qui sont des quasi-preuves

Typage et adéquation

$$\overline{\Gamma \vdash x : A} \quad (x:A) \in \Gamma$$

$$\frac{\Gamma, x : A \vdash t : B}{\Gamma \vdash \lambda x. t : A \Rightarrow B}$$

$$\frac{\Gamma \vdash t : A \Rightarrow B \quad \Gamma \vdash u : A}{\Gamma \vdash tu : B}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall x A} \quad x \notin FV(\Gamma)$$

$$\frac{\Gamma \vdash t : \forall x A}{\Gamma \vdash t : A\{x := e\}}$$

$$\frac{\Gamma \vdash t : A}{\Gamma \vdash t : \forall X A} \quad x \notin FV(\Gamma)$$

$$\frac{\Gamma \vdash t : \forall X A}{\Gamma \vdash t : A\{X := \lambda x_1 \dots x_n. F\}}$$

Lemme d'adéquation (pour tout choix de \perp)

Si $x_1 : A_1, \dots, x_k : A_k \vdash t : B$, alors pour tous u_1, \dots, u_k ,
 $u_1 \Vdash A_1[\rho], \dots, u_k \Vdash A_k[\rho]$ entraîne $t\{x_1 := u_1; \dots; x_k := u_k\} \Vdash B[\rho]$

(Où les formules sont closes par une valuation ρ arbitraire.)

Réalisation des preuves de PA2

- Logique classique : **call-cc**

1 Si $\pi \in \llbracket A \rrbracket$, alors $k_\pi \Vdash A \Rightarrow B$ (B arbitraire)

2 $\kappa \Vdash (((A \Rightarrow B) \Rightarrow A) \Rightarrow A)$ (loi de Peirce)

- Logique intuitionniste + récurrence : typage (+ adéquation)

- Injectivité + non confusion

• $\lambda z.z \Vdash \forall x \forall y (s(x) = s(y) \Rightarrow x = y)$

• $\lambda z.z (\lambda w.w) \Vdash \forall x s(x) \neq 0$

- Équations définissantes des fonctions prim. rec :

Lemme

Si $\forall \vec{x} e_1(\vec{x}) = e_2(\vec{x})$ est vraie, alors $\lambda z.z \Vdash \forall \vec{x} e_1(\vec{x}) = e_2(\vec{x})$

Opérateurs de mise en mémoire (1/2)

Problème : Comment travailler avec des entiers classiques ?

Réalisateur de $\forall^{\mathbb{N}} x A(x) \equiv \forall x (\text{nat}(x) \Rightarrow A(x))$
= fonction prenant en argument un **entier classique**

- **Notation :** $\bar{e} = \bar{s}^n \bar{0}$ (**entier de Church** associé à e)
avec $n = \downarrow e$, $\bar{0} = \lambda x f . x$, $\bar{s} = \lambda n x f . f(n x f)$

- On étend le langage des formules

Formules $A, B ::= \dots \mid \{e\} \Rightarrow A$

... et la définition de la réalisabilité

$$\|\{e\} \Rightarrow A\| = \{\bar{e} \cdot \pi \mid \pi \in \|A\|\}$$

Rappel : $\|\text{nat}(e) \Rightarrow A\| = \{t \cdot \pi \mid \pi \in \|A\|, t \in |\text{nat}(e)|\}$

Opérateurs de mise en mémoire (2/2)

- Spécification relativisée aux entiers explicites :

$$\forall x_1 \cdots \forall x_n (\{x_1\} \Rightarrow \cdots \Rightarrow \{x_n\} \Rightarrow A(x_1, \dots, x_n))$$

- Spécification relativisée aux entiers (classiques) :

$$\begin{aligned} &\forall x_1 \cdots \forall x_n (\text{nat}(x_1) \Rightarrow \cdots \Rightarrow \text{nat}(x_n) \Rightarrow A(x_1, \dots, x_n)) \\ &\approx \quad \forall^{\mathbb{N}} x_1 \cdots \forall^{\mathbb{N}} x_n A(x_1, \dots, x_n) \end{aligned}$$

Opérateur de mise en mémoire d'arité n

Une quasi-preuve M_n telle que pour toute formule $A(x_1, \dots, x_n)$

$$\begin{aligned} M_n \Vdash \forall x_1 \cdots \forall x_n (\{x_1\} \Rightarrow \cdots \Rightarrow \{x_n\} \Rightarrow A(x_1, \dots, x_n)) \\ \Rightarrow \quad \forall^{\mathbb{N}} x_1 \cdots \forall^{\mathbb{N}} x_n A(x_1, \dots, x_n) \end{aligned}$$

Par exemple : $M_1 = \lambda f n . n f (\lambda h x . h (\bar{s} x)) \bar{0}$

Réalisation de l'axiome du choix dénombrable (1/4)

Axiome du choix dénombrable (en math)

$$\forall n \in \mathbb{N} \exists y A(n, y) \Rightarrow \exists (u_n)_{n \in \mathbb{N}} \forall n \in \mathbb{N} A(n, u_n)$$

Exemple (topologie) :

- $x_0 \in \mathbb{R}^p$, $U \subseteq \mathbb{R}^p$ sous-ensemble quelconque
- **Déf :** x_0 adhérent à U si $\forall \varepsilon > 0 \exists y \in U d(x_0, y) < \varepsilon$

Théorème (Existence d'une suite convergente)

Si x_0 est adhérent à U , alors il existe une suite de points de U $(u_n)_{n \in \mathbb{N}} \in U^{\mathbb{N}}$ qui converge vers x_0

Réalisation de l'axiome du choix dénombrable (2/4)

Axiome du choix dénombrable (dans PA2)

$A[x, Y]$ formule dépendant de x, Y , mais pas de U

$$\forall x \exists Y A[x, Y] \Rightarrow \exists U \forall x A[x, U(x, \cdot)]$$

- On ajoute une nouvelle instruction χ (« quote ») :

$$\chi \star t \cdot \pi \succ t \star \bar{n}_t \cdot \pi$$

Déf. : $n_t =$ indice du terme t dans une énumération fixée $(t_n)_{n \in \mathbb{N}}$

Lemme (« typage » de χ)

Pour chaque formule $A[x, Y]$, il existe $\Phi : \mathbb{N}^3 \rightarrow \mathfrak{P}(\Pi)$ t.q. :

$$\chi \Vdash \forall x \left(\forall n (\text{nat}(n) \Rightarrow A[x, \Phi^\perp(x, n, \cdot)]) \Rightarrow \forall Y A[x, Y] \right)$$

Preuve : $\Phi(x, n) :=$ une fonction $R \in \mathfrak{P}(\Pi)^{\mathbb{N}}$ t.q. $P_n(\perp) \cap \|A[x, R^\perp]\| \neq \emptyset$ s'il en existe, n'importe laquelle sinon. (Avec $P_n(\perp) := \{\pi \in \Pi; t_n \star \bar{n} \cdot \pi \notin \perp\}$)

Réalisation de l'axiome du choix dénombrable (3/4)

D'après le lemme appliqué à $\neg A$, il existe $\Phi : \mathbb{N}^3 \rightarrow \mathfrak{P}(\Pi)$ t.q. :

- $\forall x \left(\forall n (\text{nat}(n) \Rightarrow \neg A[x, \Phi^\perp(x, n, \cdot)]) \Rightarrow \forall Y \neg A[x, Y] \right)$
- $\forall x \left(\exists Y A[x, Y] \Rightarrow \exists n (\text{nat}(n) \wedge A[x, \Phi^\perp(x, n, \cdot)]) \right)$
- $\forall x \exists Y A[x, Y] \Rightarrow \forall x \exists n (\text{nat}(n) \wedge A[x, \Phi^\perp(x, n, \cdot)])$

Principe du minimum

HA $\vdash \exists^{\mathbb{N}} n P(n) \Rightarrow \exists^{\mathbb{N}} n_0 (P(n_0) \wedge \forall^{\mathbb{N}} n (P(n) \Rightarrow n_0 \leq n))$

- $\forall x \exists Y A[x, Y] \Rightarrow$
 $\forall x \exists^{\mathbb{N}} n_0 (A[x, \Phi^\perp(x, n_0, \cdot)] \wedge \forall^{\mathbb{N}} n (A[x, \Phi^\perp(x, n, \cdot)] \Rightarrow n_0 \leq n))$

Réalisation de l'axiome du choix dénombrable (4/4)

- $\forall x \exists Y A[x, Y] \Rightarrow \exists^{\mathbb{N}} n_0 \text{ minimal } (A[x, \Phi^{\perp}(x, n_0, \cdot)])$
- On définit « moralement » $U[x, y] := \Phi(x, n_0, y)$
où n_0 est le plus petit entier t.q. $A[x, \Phi(x, n_0, \cdot)]$

Définition formelle

$$U[x, y] := \exists^{\mathbb{N}} n_0 \left(A[x, \Phi^{\perp}(x, n_0, \cdot)] \wedge \forall^{\mathbb{N}} n (A[x, \Phi^{\perp}(x, n, \cdot)] \Rightarrow n_0 \leq n) \wedge \Phi^{\perp}(x, n_0, y) \right)$$

- Par construction : $\forall y (\Phi^{\perp}(x, n_0, y) \Leftrightarrow U[x, y])$
où n_0 est le plus petit entier t.q. $A[x, \Phi(x, n_0, \cdot)]$

Lemme (extensionnalité)

$$\forall y (\Phi^{\perp}(x, n, y) \Leftrightarrow U[x, y]) \Rightarrow (A[x, \Phi^{\perp}(x, n, \cdot)] \Leftrightarrow A[x, U[x, \cdot]])$$

Preuve : Par induction sur la structure de A

- D'où $\forall x \exists Y A[x, Y] \Rightarrow \forall x A[x, U[x, \cdot]]$ (C.Q.F.D.)