Programmer = démontrer?
La correspondance de Curry-Howard aujourd'hui

Septième cours

Le forcing : une transformation de programme comme une autre?

Xavier Leroy

Collège de France

2019-01-09

COLLÈGE
DE FRANCE
1530

Or voici qu'il y a huit mois Kan, travaillant sur un adjoint à lui (voir D. Kan, Adjoint Functors, *Transactions*, V, 3,18) montra par induction, croit-on, (il raisonnait — a-t-il dit à Jaulin — sur un grand cardinal, par ≪forcing≫ pour part) la

Proposition Soit G soit H soit K ($H \subset G$, $G \supset K$) trois magmas (nous suivons Kurosh) où l'on a $a(bc) = (ab)c$; où pour tout a, $x \to xa$, $x \to ax$ sont ≪sûrs≫, sont monos, alors on a $G \simeq H \times K$ si $G = H \cup K$; si H, si K sont invariants ; si H, K n'ont qu'un individu commun $H \cap K =$

Las ! Kan mourut avant d'avoir fini son job. Donc à la fin, l'on n'a toujours pas la solution (1).

G. Perec, *La disparition*, pp. 62–63 (1969)

I

L’hypothèse du continu

Les cardinaux

Une généralisation (due à Cantor) de la notion de nombre d'éléments aux ensembles infinis.

Deux ensembles X et Y ont le même cardinal si et seulement si il existe une bijection h entre X et Y.

Les cardinaux

L'ordre entre cardinaux :

- $\text{card}(X) = \text{card}(Y)$ s'il existe une bijection $X \to Y$.
- $\text{card}(X) \leq \text{card}(Y)$ s'il existe une injection $X \to Y$.
- $\text{card}(X) < \text{card}(Y)$ s'il existe une injection $X \to Y$ mais pas d'injection $Y \to X$

Théorème (Cantor, 1874, 1891)

$\text{card}(X) < \text{card}(\mathcal{P}(X)) = \text{card}(X \to \{0,1\})$ pour tout ensemble X.

En corollaire : $\text{card}(\mathbb{N}) < \text{card}(\mathbb{R})$.

Deux sortes d'infini

L'infini dénombrable	**L'infini continu**
$\mathbb{N}$	$\mathbb{R}$
$\mathbb{Z}$	$\mathcal{P}(\mathbb{N})$
$\mathbb{Q}$	$\mathbb{C}$
$\mathbb{N} \times \cdots \times \mathbb{N}$	$\mathbb{R} \times \cdots \times \mathbb{R}$
mots finis sur un alphabet fini	$\mathbb{N} \to \{0, 1, \ldots, k\}$
mots finis sur $\mathbb{N}$	$\mathbb{N} \to \mathbb{N}$
formules mathématiques	
programmes informatiques	
machines de Turing	
fonctions calculables	

L'hypothèse du continu (HC)

Il n'y a aucun cardinal entre l'infini dénombrable et l'infini continu.

$$\neg\exists X,\ \ \mathrm{card}(\mathbb{N}) < \mathrm{card}(X) < \mathrm{card}(\mathcal{P}(\mathbb{N}))$$

Autrement dit : tout sous-ensemble de $\mathbb{R}$ est soit fini, soit dénombrable, soit en bijection avec $\mathbb{R}$.

L'hypothèse du continu généralisée (HCG)

L'énumération des cardinaux infinis : (nécessite l'axiome du choix)

$\aleph_0 = \text{card}(\mathbb{N})$ $\quad \aleph_{\alpha+1} = \text{le plus petit cardinal} > \aleph_\alpha$ $\quad \aleph_\lambda = \sup_{\alpha<\lambda} \aleph_\alpha$

Par le théorème de Cantor : $\aleph_{\alpha+1} \leq 2^{\aleph_\alpha}$ pour tout α.

Hypothèse du continu : $\aleph_1 = 2^{\aleph_0}$

Hypothèse du continu généralisée : $\aleph_{\alpha+1} = 2^{\aleph_\alpha}$ pour tout α.

Histoire d'un grand problème

1878 : G. Cantor énonce l'hypothèse du continu. Il échoue toute sa vie à la démontrer.

1900 : D. Hilbert liste HC en premier dans sa liste de 23 grands problèmes ouverts.

1938 : K. Gödel montre que HCG est cohérente avec la théorie des ensembles ZFC.

1964 : P. Cohen montre que la négation de HC est cohérente avec ZFC. Il développe pour cela une technique entièrement nouvelle : le *forcing*. Il reçoit la médaille Fields en 1966.

1970 : W. B. Easton montre la cohérence d'une généralisation de $\neg$HC : pour tout α, $\aleph_{\alpha+1} < 2^{\aleph_\alpha}$.

L'indépendance de l'hypothèse du continu

L'hypothèse du continu (généralisée) est donc indépendante de ZF, la théorie des ensembles de Zermelo-Fraenkel, au sens où :

- On peut supposer que HC est vraie (la prendre comme axiome) et aucune contradiction (incohérence logique) n'en découle.
- On peut supposer que HC est fausse (prendre sa négation comme axiome) et aucune contradiction n'en découle.
- En corollaire, on ne peut démontrer ni HC ni $\neg$HC à partir des axiomes de ZF.

Autre exemple : l'axiome du choix est indépendant de ZF.
(Démontré en même temps que l'indépendance de HC par Gödel et Cohen.)

Modèles de la théorie des ensembles

La théorie des ensembles ZF :
Un symbole $\ll\in\gg$ et 8 axiomes :
Extensionnalité
Paire
Compréhension
Union
Ensemble des parties
Infinité
Remplacement
Fondation

Un modèle de la théorie des ensembles :
Une collection d'objets et un prédicat $\in$ qui satisfont les 8 axiomes.

La structure des groupes :
Trois symboles $\ll 1\gg$, $\ll\cdot\gg$ et $\ll^{-1}\gg$ et 3 identités :

$$(x \cdot y) \cdot z = x \cdot (y \cdot z)$$
$$1 \cdot x = x = x \cdot 1$$
$$x \cdot x^{-1} = 1 = x^{-1} \cdot x$$

Un groupe :
Un ensemble G et des opérations $(1, \cdot, ^{-1})$ qui satisfont les 3 identités.

Modèles de la théorie des ensembles

L'existence d'un modèle de ZF montre la cohérence des axiomes de ZF (on ne peut pas démontrer l'absurdité $\bot$).

Réciproquement : si ZF est cohérent, il admet un modèle (Gödel, 1930).

L'existence d'un modèle de ZF qui satisfait une hypothèse H montre que ZF $+ H$ est cohérent, et donc qu'on ne peut pas démontrer $\neg H$ à partir des axiomes de ZF.

La preuve de Gödel, 1938 : étant donné un modèle M de ZF, construire un modèle intérieur $L \subseteq M$ qui satisfait HC.

La preuve de Cohen, 1964 : étant donné un modèle M de ZF, construire une extension de ce modèle $M[G] \supset M$ qui satisfait $\neg$HC.

Modèles de la théorie des ensembles

L'existence d'un modèle de ZF montre la cohérence des axiomes de ZF (on ne peut pas démontrer l'absurdité $\bot$).

Réciproquement : si ZF est cohérent, il admet un modèle (Gödel, 1930).

L'existence d'un modèle de ZF qui satisfait une hypothèse H montre que ZF $+ H$ est cohérent, et donc qu'on ne peut pas démontrer $\neg H$ à partir des axiomes de ZF.

La preuve de Gödel, 1938 : étant donné un modèle M de ZF, construire un modèle intérieur $L \subseteq M$ qui satisfait HC.

La preuve de Cohen, 1964 : étant donné un modèle M de ZF, construire une extension de ce modèle $M[G] \supset M$ qui satisfait $\neg$HC.

Modèles de la théorie des ensembles

L'existence d'un modèle de ZF montre la cohérence des axiomes de ZF (on ne peut pas démontrer l'absurdité $\bot$).

Réciproquement : si ZF est cohérent, il admet un modèle (Gödel, 1930).

L'existence d'un modèle de ZF qui satisfait une hypothèse H montre que ZF $+ H$ est cohérent, et donc qu'on ne peut pas démontrer $\neg H$ à partir des axiomes de ZF.

La preuve de Gödel, 1938 : étant donné un modèle M de ZF, construire un modèle intérieur $L \subseteq M$ qui satisfait HC.

La preuve de Cohen, 1964 : étant donné un modèle M de ZF, construire une extension de ce modèle $M[G] \supset M$ qui satisfait $\neg$HC.

Modèles de la théorie des ensembles

L'existence d'un modèle de ZF montre la cohérence des axiomes de ZF (on ne peut pas démontrer l'absurdité $\bot$).

Réciproquement : si ZF est cohérent, il admet un modèle (Gödel, 1930).

L'existence d'un modèle de ZF qui satisfait une hypothèse H montre que ZF $+ H$ est cohérent, et donc qu'on ne peut pas démontrer $\neg H$ à partir des axiomes de ZF.

La preuve de Gödel, 1938 : étant donné un modèle M de ZF, construire un modèle intérieur $L \subseteq M$ qui satisfait HC.

La preuve de Cohen, 1964 : étant donné un modèle M de ZF, construire une extension de ce modèle $M[G] \supset M$ qui satisfait $\neg$HC.

Les ensembles constructibles de Gödel

Soit $(M, \in)$ un modèle de ZF.

Si X est un ensemble de ce modèle, on note $Def(X)$ l'ensemble des ensembles définissables par des formules logiques Φ dont toutes les variables (quantifiées ou libres) parcourent X :

$$Def(X) = \{ \{x \in X \mid (X, \in) \models \Phi(x) \} \}$$

On définit par récurrence transfinie :

$$L_0 = \emptyset \qquad L_{\alpha+1} = Def(L_\alpha) \qquad L_\lambda = \bigcup_{\alpha<\lambda} L_\alpha$$

Autrement dit : L_α est tous les ensembles que l'on peut construire en faisant référence uniquement à des éléments de L_β pour $\beta < \alpha$.

Les ensembles constructibles de Gödel

Si $(M, \in)$ est un modèle de ZF, et *Ord* la collection de ses ordinaux, on définit $L = \bigcup_{\alpha \in Ord} L_\alpha$. Alors, $(L, \in)$ est un modèle de ZF. De plus :

- L satisfait l'axiome du choix.
  (Tout ensemble A de L est bien ordonné par un ordre dérivé de celui des ordinaux.)

- L satisfait l'hypothèse du continu généralisée.
  (Pour tout α, $\mathcal{P}(L_\alpha) \cap L \subseteq L_\beta$ pour un β «pas beaucoup plus grand» que α. On en déduit que $2^{\aleph_\gamma} \leq \aleph_{\gamma+1}$ et donc $\aleph_{\gamma+1} = 2^{\aleph_\gamma}$.)

Les extensions génériques de Cohen

Dans l'approche de Gödel, on part d'un modèle M et on ne garde que les ensembles ≪bien élevés≫ de M (ceux qui sont constructibles), éliminant ainsi les ensembles ≪sauvages≫ qui pourraient être de cardinalité intermédiaire et donc invalider HC.

L'approche de Cohen est duale : on part d'un modèle M et on lui adjoint un nouvel ensemble G qui va ≪faire grossir $\mathcal{P}(\mathbb{N})$≫ au point que $\aleph_0 < \aleph_1 < 2^{\aleph_0}$ dans le modèle $M[G]$ ainsi obtenu.

Extension d'une structure algébrique

Un concept familier en mathématiques. Par exemple :

- Si on ajoute un élément X à un anneau A, on ajoute aussi $2X$, $-X$, X^2, X^3, …, et on obtient $A[X]$, l'anneau des polynômes à coefficients dans A.
- Si on ajoute au corps $\mathbb{R}$ un élément i tel que $i^2 = -1$, on ajoute aussi tous les $x + iy$, et on obtient $\mathbb{C}$.

Prudence ! Une extension peut être incohérente ! Par exemple :

- Si on ajoute au corps totalement ordonné $\mathbb{R}$ un élément i tel que $i^2 = -1$, on contredit la propriété $\forall x,\ x^2 \geq 0$ qui était vraie avant l'extension.

Extension d'une structure algébrique

Un concept familier en mathématiques. Par exemple :

- Si on ajoute un élément X à un anneau A, on ajoute aussi $2X$, $-X$, X^2, X^3, …, et on obtient $A[X]$, l'anneau des polynômes à coefficients dans A.
- Si on ajoute au corps $\mathbb{R}$ un élément i tel que $i^2 = -1$, on ajoute aussi tous les $x + iy$, et on obtient $\mathbb{C}$.

Prudence ! Une extension peut être incohérente ! Par exemple :

- Si on ajoute au corps totalement ordonné $\mathbb{R}$ un élément i tel que $i^2 = -1$, on contredit la propriété $\forall x,\ x^2 \geq 0$ qui était vraie avant l'extension.

La démonstration de Cohen

- Soit M un modèle transitif dénombrable de ZF.
- Soit k un ensemble de M tel que $M \models \text{card}(k) = \aleph_2$.
- Ajouter à M un nouvel élément G qui est une fonction «générique» de k dans $\mathcal{P}(\mathbb{N})$, obtenant $M[G]$.
- Montrer que $M[G]$ est un modèle de ZF.
- Montrer que $M[G] \models$ «la fonction G est injective», et donc que $M[G] \models \text{card}(k) \leq \text{card}(\mathcal{P}(\mathbb{N})) = 2^{\aleph_0}$.
- Montrer que les cardinaux sont préservés par l'extension, et donc que $M[G] \models \text{card}(k) = \aleph_2$.
- Conclure que $M[G] \models \aleph_0 < \aleph_1 < \aleph_2 \leq 2^{\aleph_0}$, et donc $M[G] \models \neg\text{HC}$.

II

Le forcing

Conditions de forcing

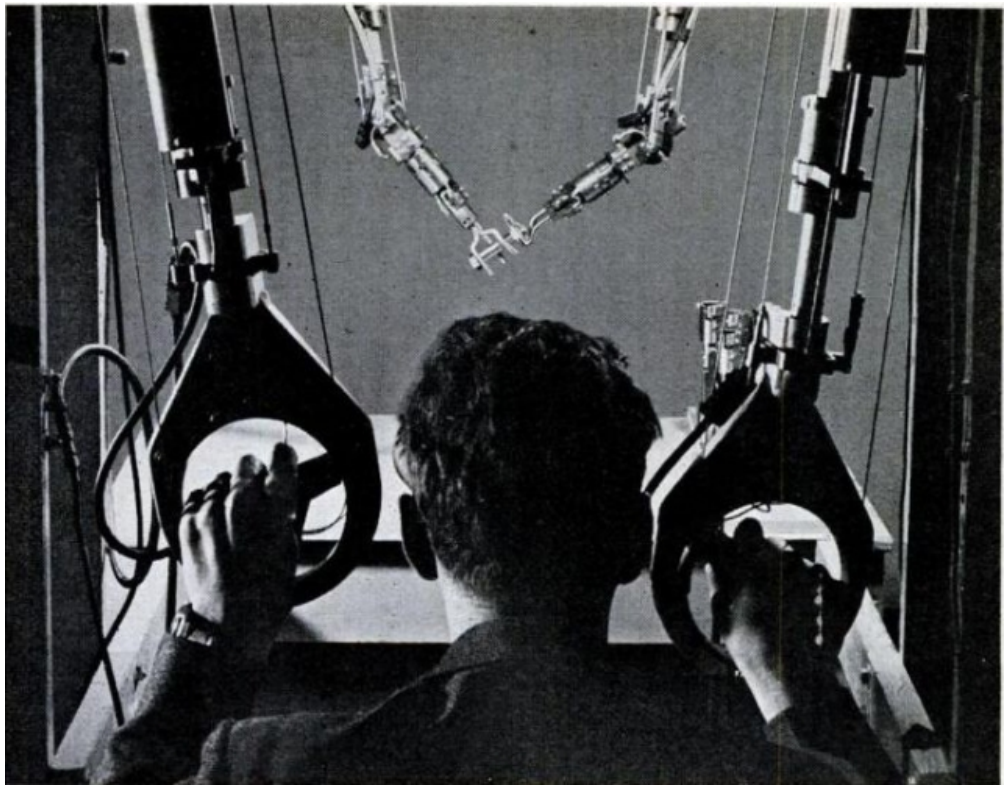
Construire le modèle étendu $M[G]$ n'est pas très difficile ; mais comment raisonner dans ce modèle ?

Quelles sont les propriétés de G ?

Comment démontrer qu'une formule logique est vraie dans $M[G]$?

Idée de Cohen : on peut décrire G et ses propriétés via des approximations finies (mais aussi précises qu'on le souhaite) qui vivent dans M et qu'on appelle conditions de forcing.

Conditions de forcing



Objet dangereux manipulé : $M[G]$.
Poignées du manipulateur : conditions de forcing.

Conditions de forcing

Définition

Un ensemble de conditions de forcing est un ensemble partiellement ordonné $(\mathcal{C}, \preccurlyeq)$.
$q \preccurlyeq p$ signifie que la condition q est plus «fine» que la condition p, ou encore que q implique p.

Exemple

Si l'élément générique G est un ensemble d'entiers, on prend comme conditions de forcing p des fonctions finies des entiers dans $\{0, 1\}$, p.ex. $\{4 \mapsto 1,\ 13 \mapsto 0\}$.

- $p(n) = 1$ signifie «n appartient à G»
- $p(n) = 0$ signifie «n n'appartient pas à G»

On ordonne les conditions par inclusion inverse : $q \preccurlyeq p \overset{def}{=} p \subseteq q$.

Prédicat de forcing

Étant donnée une formule logique A qui parle d'éléments de $M[G]$, on dit que A est forcée par la condition p, et on écrit $p \Vdash_W A$, si :

$$\begin{aligned} p \Vdash_W n \in \bar{G} &\text{ ssi } p(n) = 1 \\ p \Vdash_W A \wedge B &\text{ ssi } p \Vdash_W A \text{ et } p \Vdash_W B \\ p \Vdash_W \neg A &\text{ ssi } \forall q \preccurlyeq p,\ \neg(q \Vdash_W A) \\ p \Vdash_W \forall x \in X.\, A(x) &\text{ ssi } p \Vdash_W A(x) \text{ pour tout } x \in X \end{aligned}$$

Remarque : si $p \Vdash_W A$ alors $q \Vdash_W A$ pour tout $q \preccurlyeq p$.

Théorème

1- Pour toute extension $M[G]$ et pour toute formule A,
$M[G] \models A$ si et seulement s'il existe $p \in G$ t.q. $M \models (p \Vdash_W A)$.

2- Pour tout p, il existe une extension $M[G]$ telle que $p \in G$.

Exemple d'application

Lemme

L'ensemble générique d'entiers G contient une infinité de nombres premiers.

Démonstration.

Il faut montrer $M[G] \models \forall m, \exists n, n \in \bar{G} \wedge n \geq m \wedge n$ premier.
Par le théorème de forcing, il suffit de montrer (dans M)

$$\emptyset \Vdash_W \forall m, \exists n, n \in \bar{G} \wedge n \geq m \wedge n \text{ premier}$$

c.à.d. $\emptyset \Vdash_W \forall m, \neg(\forall n, \neg(n \in \bar{G} \wedge n \geq m \wedge n \text{ premier}))$

c.à.d. $\forall m, \forall p, \exists q \preccurlyeq p, \exists n,\ q(n) = 1 \wedge n \geq m \wedge n$ premier

La fonction p étant finie et l'ensemble des nombres premiers infinis, on peut toujours trouver $n \geq m$ premier et hors du domaine de p. On prend alors $q = p \cup \{n \mapsto 1\}$ et on a $q \preccurlyeq p$ ainsi que $q(n) = 1$.

□

Exemple d'application

Si G est la fonction $k \to \mathcal{P}(\mathbb{N})$ de la démonstration de Cohen, on prend comme conditions de forcing les fonctions finies $k \times \mathbb{N} \to_{fin} \{0,1\}$, ordonnées par inclusion inverse.

On définit $p \Vdash_W n \in \bar{G}(x)$ ssi $p(x,n) = 1$.

Exercice : montrer que G est injective, c.à.d.
$M[G] \models \forall x_1, x_2,\ x_1 \neq x_2 \Rightarrow G(x_1) \neq G(x_2)$.

Des idées qui entrent en résonance

- Le forcing (Cohen, 1963–1964)
  Théorie des ensembles ; logique classique.

- Les modèles de Kripke (Kripke, 1959–1965)
  Logiques modales, logique intuitionniste.

- Les constructions de (pré-)faisceaux (Lawvere et Tierney, 1971–1972)
  Théorie des catégories, topos.

Les modèles de Kripke

Une relation $p \Vdash_K A$, «la formule A est vraie dans le monde p».

Un monde $p \approx$ un ensemble de faits (propositions atomiques).

Les mondes sont ordonnés : $q \preccurlyeq p$,
se lit «le monde q est accessible à partir du monde p»
et implique que q contient tous les faits de p.

Modèles de Kripke intuitionnistes

$$
\begin{aligned}
p \Vdash_K F(a_1, \ldots, a_n) &\text{ ssi } F(a_1, \ldots, a_n) \in \mathit{Faits}(p) \\
p \Vdash_K A \wedge B &\text{ ssi } p \Vdash_K A \text{ et } p \Vdash_K B \\
p \Vdash_K A \vee B &\text{ ssi } p \Vdash_K A \text{ ou } p \Vdash_K B \\
p \Vdash_K A \Rightarrow B &\text{ ssi } \text{pour tout } q \preccurlyeq p,\ q \Vdash_K A \text{ implique } q \Vdash_K B \\
p \Vdash_K \neg A &\text{ ssi } \forall q \preccurlyeq p,\ \neg(q \Vdash_K A) \\
p \Vdash_K \forall x.\, A(x) &\text{ ssi pour tout } x,\ p \Vdash_K A(x) \\
p \Vdash_K \exists x.\, A(x) &\text{ ssi il existe } x \text{ t.q. } p \Vdash_K A(x)
\end{aligned}
$$

Propriété de monotonicité :

$$p \Vdash_K A \wedge q \preccurlyeq p \Rightarrow q \Vdash_K A$$

(En rouge, la «modification minimale» qui garantit la monotonicité.)

Modèles de Kripke intuitionnistes

$$
\begin{aligned}
p \Vdash_K F(a_1, \ldots, a_n) &\text{ ssi } F(a_1, \ldots, a_n) \in \mathit{Faits}(p) \\
p \Vdash_K A \wedge B &\text{ ssi } p \Vdash_K A \text{ et } p \Vdash_K B \\
p \Vdash_K A \vee B &\text{ ssi } p \Vdash_K A \text{ ou } p \Vdash_K B \\
p \Vdash_K A \Rightarrow B &\text{ ssi pour tout } q \preccurlyeq p,\ q \Vdash_K A \text{ implique } q \Vdash_K B \\
p \Vdash_K \neg A &\text{ ssi } \forall q \preccurlyeq p,\ \neg(q \Vdash_K A) \\
p \Vdash_K \forall x.\, A(x) &\text{ ssi pour tout } x,\ p \Vdash_K A(x) \\
p \Vdash_K \exists x.\, A(x) &\text{ ssi il existe } x \text{ t.q. } p \Vdash_K A(x)
\end{aligned}
$$

Propriété de monotonicité :

$$p \Vdash_K A \wedge q \preccurlyeq p \Rightarrow q \Vdash_K A$$

(En rouge, la ≪modification minimale≫ qui garantit la monotonicité.)

Modèles de Kripke et logique modale

Kripke a introduit ces modèles (classiques ou intuitionnistes) pour étudier les logiques modales. En effet, les modalités s'interprètent naturellement en termes de quantification sur les mondes accessibles :

$$p \Vdash_K \Box A \text{ ssi } \forall q \preccurlyeq p,\ q \Vdash_K A$$

$$p \Vdash_K \Diamond A \text{ ssi } \exists q \preccurlyeq p,\ q \Vdash_K A$$

Modèles de Kripke intuitionnistes

Les modèles de Kripke intuitionnistes sont aussi ≪le bon modèle≫ pour la logique intuitionniste, en ce sens que :

- Toute formule A démontrable en logique intuitionniste est vraie dans tous les mondes de tous les modèles : $p \Vdash_K A$.
- Les lois classiques (tiers exclu, élimination de la double négation) sont invalides dans certains mondes de certains modèles.

Exemple

Si F est une formule atomique, on se donne deux mondes p_0, p_1

$$p_1 \preccurlyeq p_0 \qquad \mathit{Faits}(p_0) = \emptyset \qquad \mathit{Faits}(p_1) = \{F\}$$

et on a

$$\begin{array}{ll} p_0 \nVdash_K F & \\ p_0 \nVdash_K \neg F & \text{(parce que } p_1 \Vdash_K F\text{)} \\ p_0 \nVdash_K F \vee \neg F & \end{array}$$

Modèles de Kripke et forcing

Il y a une grande ressemblance entre

- conditions de forcing et mondes ;
- la relation $p \Vdash_W A$, $\ll$la condition p force la formule $A\gg$
  et la relation $p \Vdash_K A$, $\ll$le monde p satisfait la formule $A\gg$.
  (Au point que certains lisent $p \Vdash_K A$ comme $\ll p$ force $A\gg$.)

Cela débouche sur une théorie du forcing intuitionniste à base de modèles de Kripke qui montre les résultats d'indépendance de Cohen pour la théorie des ensembles intuitionniste.

(M. Fitting, *Intuitionistic logic model theory and forcing*, 1969)

Modèles de Kripke et forcing

Exemple

On prend comme mondes p les fonctions finies $\mathbb{N} \to_{fin} \{0,1\}$, avec comme interprétation $\mathit{Faits}(p) = \{\ll n \in G \gg \mid p(n) = 1\}$.

On ne peut pas montrer directement
$\emptyset \Vdash_K \ll G$ contient une infinité de nombres premiers$\gg$,
mais on peut montrer une de ses double négations,

$$\emptyset \Vdash_K \forall m, \neg\neg(\exists n, n \in G \wedge n \geq m \wedge n \text{ premier})$$

c.à.d. $\forall m, \forall p, \exists q \preccurlyeq p,\ q \Vdash_K \exists n, n \in G \wedge n \geq m \wedge n$ premier

c.à.d. $\forall m, \forall p, \exists q \preccurlyeq p, \exists n,\ q(n) = 1 \wedge n \geq m \wedge n$ premier

Double négation et forcing

Plus généralement, on retrouve les lois du prédicat de forcing $\Vdash_W$ en composant $\Vdash_K$ avec la traduction doublement négative de Gödel-Gentzen (cf. cours du 5 décembre 2018) :

$$[\![A \Rightarrow B]\!] = [\![A]\!] \Rightarrow [\![B]\!]$$

$$[\![A \wedge B]\!] = [\![A]\!] \wedge [\![B]\!] \qquad [\![A \vee B]\!] = \neg\neg([\![A]\!] \vee [\![B]\!])$$

$$[\![\forall x.\, A]\!] = \forall x.\, [\![A]\!] \qquad [\![\exists x.\, A]\!] = \neg\neg\exists x.\, [\![A]\!]$$

En définissant $p \Vdash_W A$ comme $p \Vdash_K [\![A]\!]$, on a bien p.ex.

$$p \Vdash_W A \wedge B \text{ ssi } p \Vdash_W A \text{ et } p \Vdash_W B$$

$$p \Vdash_W A \vee B \text{ ssi } \forall q \preccurlyeq p, \exists r \preccurlyeq q,\ r \Vdash_W A \text{ ou } p \Vdash_W B$$

De plus, $[\![A]\!] \Leftrightarrow \neg\neg A$, et donc

$$\emptyset \Vdash_K \neg\neg A \text{ si et seulement s'il existe } p \text{ t.q. } p \Vdash_W A$$

III

Internaliser le forcing dans une théorie des types

Forcing et théorie des types

Ce que le forcing / les modèles de Kripke / la construction des pré-faisceaux apportent à la théorie des types :

- Des résultats d'indépendance.
  (P.ex. de l'axiome d'univalence de Voevodsky.)
- Des outils pour la logique catégorique.
  (P.ex. le modèle ≪cubique≫ pour l'univalence de Coquand et al.)
- Des outils pour la programmation et la sémantique.
  (P.ex. types récursifs généraux, ou encore *step-indexing*.)

Forcing et théorie des types

Ce que la théorie des types et autres approches à la Curry-Howard apportent au forcing :

- Une présentation possible sous forme de transformations (codage) des propositions de la théorie des types étendue $TT[G]$ dans la théorie des types initiales TT.
  (Cf. les transformations négatives pour coder la logique classique en logique intuitionniste, cours du 5 décembre 2018.)
- La transformation s'étend aux termes de preuves, ce qui garantit la cohérence logique de l'approche.
  (Cf. le codage de la paramétricité par Bernardy et al, cours du 19 décembre 2018.)

Forcing et théorie des types

Travaux récents : (références à la fin du support de cours)

- A. Miquel (2011) et L. Rieg (2014), inspirés par J.-L. Krivine :
  forcing classique pour la logique PAω ($\approx$ Fω + `call/cc`).
  $\Rightarrow$ séminaire du 16 janvier 2019

- G. Jaber, N. Tabareau et M. Sozeau (2012) :
  forcing intuitioniste pour CC + univers + Σ,
  internalisation de la construction des pré-faisceaux.

- G. Jaber, G. Lewertowski, P.-M. Pédrot, N. Tabareau, M. Sozeau (2016) :
  forcing intuitioniste pour Coq,
  transformation quasi-monadique, en appel par nom.

Les grandes lignes de la transformation

(D'après Jaber, Tabareau, Sozeau, LICS 2012)

On se donne un type $\mathbb{P}$ des mondes (alias conditions de forcing) et un préordre $\preccurlyeq$.

- À une proposition A on associe une proposition $[\![A]\!]_p$ indexée par un monde p, similaire à $p \Vdash_K A$ ($\ll A$ est vraie dans le monde $p \gg$).
- À une preuve $\vdash a : A$ on associe une preuve $p : \mathbb{P} \vdash [a]_p : [\![A]\!]_p$.

La traduction est guidée par la propriété usuelle de l'implication :

$$p \Vdash_K A \Rightarrow B \text{ ssi } \forall q \preccurlyeq p, q \Vdash_K A \Rightarrow q \Vdash_K B$$

En termes de produits dépendants : (avec $P_p \stackrel{def}{=} \{q : \mathbb{P} \mid q \preccurlyeq p\}$)

$$[\![\Pi x : A.\, B]\!]_p \;=\; \Pi(q : P_p).\, \Pi(x : [\![A]\!]_q).\, [\![B]\!]_q$$

La monade de forcing

Essayons de mettre cela sous la forme d'une transformation monadique dans une monade T d'ordre supérieur. On peut écrire

$$[\![A \to B]\!]_p = T\,(\lambda q.\, [\![A]\!]_q \to [\![B]\!]_q)\,p$$

avec

$$T\,A = \lambda(p : \mathbb{P}).\, \Pi(q : P_p).\, A\,q$$

On peut voir cette ≪monade de forcing≫ comme une monade d'entrées asynchrones :

- p est le journal des entrées déjà reçues ;
- $q \preccurlyeq p$, signifie qu'on a reçu 0, 1 ou plusieurs entrées nouvelles ;
- tout calcul dans cette monade doit être prêt à recevoir de nouvelles entrées, d'où $\Pi(q : P_p) \ldots$

La monade de forcing

$$T\,A = \lambda(p : \mathbb{P}).\ \Pi(q : P_p).\ A\ q$$

Ce n'est pas la monade d'environnement

$$T\,A = \mathbb{P} \to A$$

car l'environnement p change pendant le calcul, de manière non déterministe mais monotone.

Ce n'est pas la monade d'état monotone

$$T\,A = \Pi(p : \mathbb{P}).\ \{(a, q) : A \times \mathbb{P} \mid q \preccurlyeq p\}$$

car dans la monade d'état le changement d'état p est décidé par le calcul, alors que dans la monade de forcing ce changement est imposé par l'extérieur.

Un début de traduction

$$[\lambda(x : A).\, B]_p = \lambda(q : P_p).\, \lambda(x : [\![A]\!]_q).\, [B]_q$$
$$[A\ B]_p = [A]_p\ p\ [B]_p$$

Les types étant des termes, il faut définir $[\![\cdot]\!]$ en fonction de $[\cdot]$:

$$[\![A]\!]_p = [A]_p\ p$$
$$[\Pi(x : A).\, B]_p = \lambda(q : P_p).\, \Pi(r : P_q).\, \Pi(x : [\![A]\!]_r).\, [\![B]\!]_r$$
$$[U]_p = \lambda(q : P_p).\, U$$

Quid des variables $[x]_q$?

Une variable peut être utilisée dans un monde q différent de celui p où elle a été liée !

Morphismes

L'interprétation $[A]_p$ d'un type n'est pas juste une fonction $f : P_p \to \square$ mais aussi un morphisme $\theta\ q\ r : f\ q \to f\ r$ pour passer de l'interprétation au monde q à l'interprétation au monde $r \preccurlyeq q$.

$$\begin{array}{ccc} q & \xrightarrow{f} & f\ q \\ \preccurlyeq \downarrow & & \downarrow \theta\ q\ r \\ r & \xrightarrow{f} & f\ r \end{array}$$

Dans le cas où A est une proposition, θ est la preuve de la monotonicité du forcing : $p \Vdash_K A \wedge q \preccurlyeq p \Rightarrow q \Vdash_K A$.

Dans le cas où A est un «type qui calcule», on veut en plus des propriétés de fonctorialité de θ, à savoir : $\theta\ q\ q = id$ et $\theta\ q\ s = \theta\ r\ s \circ \theta\ q\ r$.

Morphismes

On définit simultanément la traduction des types $[\![A]\!]_p$ et les morphismes $\theta(A)_{p\to q}$ de $[\![A]\!]_p$ dans $[\![A]\!]_q$.

$$[A]_p : \Sigma f : P_p \to \square.$$
$$\{\theta : \Pi(q : P_p).\, \Pi(r : P_q).\, f\ q \to f\ r \mid \texttt{fonctoriel}_p(\theta)\}$$

$$[\![A]\!]_p = \pi_1([A]_p)$$
$$\theta(A)_{p\to q} = \pi_2([A]_p)\ p\ q$$

Finalement, la traduction d'une variable est

$$[x]^\sigma_p = \theta(\texttt{type}(\sigma, x))_{\texttt{monde}(\sigma,x)\to p}(x)$$

dans un environnement σ : variable $\to$ type $\times$ monde.

Problèmes techniques

Ces morphismes sont évidents en théorie des catégories mais posent des problèmes liés à l'égalité en théorie des types.

En particulier : si deux types sont convertibles $A =_{\beta\eta} B$,
leurs traductions ne sont généralement pas convertibles.

$$\frac{\Gamma \vdash M : A \qquad A =_{\beta\eta} B}{\Gamma \vdash M : B}$$

Traduction version 2

(Jaber, Lewertowski, Pédrot, Tabareau, Sozeau, LICS 2016)

On peut se passer de ces morphismes en traduisant les types Π de fonctions «en appel par nom», c.à.d. en laissant flexible le monde de l'argument.

$$\begin{aligned} \text{par valeur} \quad & [\![\Pi x : A.\, B]\!]_p = \Pi q : P_p.\, \Pi x : [\![A]\!]_q.\, [\![B]\!]_q \\ \text{par nom} \quad & [\![\Pi x : A.\, B]\!]_p = \Pi x : (\Pi q : P_p.\, [\![A]\!]_q).\, [\![B]\!]_p \end{aligned}$$

La traduction des variables :

$$\begin{aligned} \text{par valeur} \quad & [x]^\sigma_p = \theta(\mathtt{type}(\sigma, x))_{\mathtt{monde}(\sigma,x)\to p}(x) \\ \text{par nom} \quad & [x]^\sigma_p = x\ p \end{aligned}$$

Plus besoin des morphismes θ ; il suffit que l'environnement σ prouve que $p \preccurlyeq \mathtt{monde}(\sigma, x)$.

Bénéfice supplémentaire : si $A =_{\beta\eta} B$ alors $[\![A]\!]_p =_{\beta\eta} [\![B]\!]_p$.

Utilisation pour le forcing

Les traductions $[\cdot]$ permettent de transporter mécaniquement les définitions et les théorèmes de *TT* (la théorie des types de départ, p.ex. Coq) dans TT[*G*] (son extension).

(Des *plug-ins Coq* ont été développés pour automatiser cela.)

Pour déclarer un élément générique G de type A dans l'extension, il suffit de définir manuellement dans TT un terme G^f de type $\forall p,\ [\![A]\!]_p$

Exemple

Pour avoir un ensemble d'entiers générique $G : \texttt{nat} \to \texttt{Prop}$, on prend $\mathbb{P} = \texttt{Finfun.t nat bool}$ et on définit

$$G^f = \lambda(p : \mathbb{P}).\ \lambda(q : P_p).\ \lambda(n : \texttt{nat}).\ \texttt{Finfun.app}\ q\ n = \texttt{Some true}$$

IV

Forcing sur les entiers naturels

Forcing sur les entiers naturels

(Aussi appelé ≪logique interne du topos des arbres≫ par Birkedal et al)

Un exemple simple de conditions de forcing / de mondes de Kripke est

$$\mathbb{P} \overset{def}{=} \mathbb{N} \quad \text{ordonné naturellement par } q \preccurlyeq p \overset{def}{=} q \leq p$$

Une interprétation intuitive en termes temporels :
$p \Vdash_K A$ se lit comme ≪A est vraie maintenant et pendant p jours≫.

Modalité $\triangleright$ et règle de Löb

La modalité $\triangleright A$ se lit «plus tard A» et se définit par

$$0 \Vdash_K \triangleright A \qquad p+1 \Vdash_K \triangleright A \text{ si } p \Vdash_K A$$

Autrement dit : $\triangleright A$ est vraie aujourd'hui pour p jours si A est vraie demain pour $p-1$ jours.

Dans une telle logique modale, la règle de Löb est valide :

$$\frac{\triangleright A \Rightarrow A}{A}$$

Démonstration.

Supposons $p \Vdash_K \triangleright A \Rightarrow A$. On a $(q \Vdash_K \triangleright A) \Rightarrow (q \Vdash_K A)$ pour tout $q \leq p$.
On montre $q \Vdash_K A$ pour tout $q \leq p$ par récurrence sur q :
$0 \Vdash_K A$ puisque $0 \Vdash_K \triangleright A$.
Si $q < p$ et $(q \Vdash_K A)$, alors $(q+1 \Vdash_K \triangleright A)$ et donc $(q+1 \Vdash_K A)$. □

Généralisation : un opérateur de point fixe

On peut déclarer les termes suivants dans l'extension par forcing, simplement en donnant des termes qui habitent la traduction de leur type :

$$\begin{aligned}
\triangleright &: \mathtt{Type} \to \mathtt{Type} \\
\mathtt{fix} &: \forall (A : \mathtt{Type}),\ (\triangleright A \to A) \to A \\
\mathtt{next} &: \forall (A : \mathtt{Type}),\ A \to \triangleright A \\
\mathtt{fix_eq} &: \forall (A : \mathtt{Type}). \forall (f : \triangleright A \to A).\ \mathtt{fix}\ A\ f = f\ (\mathtt{next}\ A\ (\mathtt{fix}\ A\ f))
\end{aligned}$$

(Constructions : par récurrence sur p.)

`fix` est donc le terme de preuve pour la règle de Löb, mais il nous donne aussi un opérateur de point fixe intéressant.

Types récursifs généraux

En spécialisant `fix` sur un univers, disons $A = \texttt{Set}$, on arrive à construire

$$\mu : (\texttt{Set} \to \texttt{Set}) \to \texttt{Set}$$
$$\texttt{unfold} : \forall (F : \texttt{Set} \to \texttt{Set}),\ \mu\, F \to F\,(\triangleright\, \mu\, F)$$
$$\texttt{fold} : \forall (F : \texttt{Set} \to \texttt{Set}), F\,(\triangleright\, \mu\, F) \to \mu\, F$$

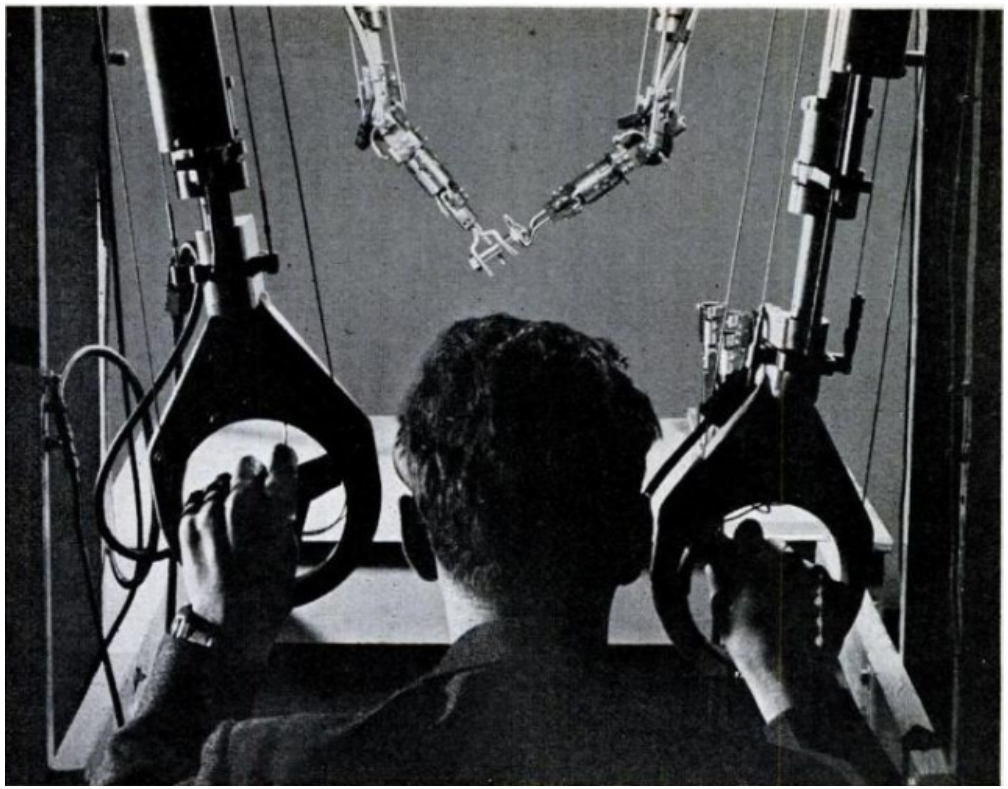
ainsi que des démonstrations de $\mu\, F = F(\triangleright\, \mu\, F)$ et
$\texttt{fold}\ F \circ \texttt{unfold}\ F = \texttt{id}$ et $\texttt{unfold}\ F \circ \texttt{fold}\ F = \texttt{id}$.

Le type $\mu\, F$ est donc l'équivalent du type récursif Caml `t`

```
type t = C of t F      unfold (C x) = x      fold x = C x
```

On ne fait aucune hypothèse sur le constructeur de types F : il n'est pas nécessairement croissant, ni contractif.

Types récursifs généraux



Objet dangereux manipulé : un type récursif comme $T = T \to T$, qui met en danger la terminaison.

Poignées du manipulateur : les termes produits par la traduction $[\cdot]$.

Pour aller plus loin

Les types récursifs généraux obtenus par forcing permettent de donner des sémantiques dénotationnelles simples à des langages Turing-complets (sans normalisation forte). Par exemple :

- $D = D \to D$ pour le λ-calcul pur ;
- $D = (Loc \to D) \to \mathcal{P}(Val)$ pour les références mutables.

Plus généralement, l'idée naïve de ≪compter les jours≫ et l'idée moins naïve de la modalité ≪plus tard≫ ($\triangleright$) rejoignent une puissante technique de sémantique : le *step-indexing*, objet du prochain cours.

V

Bibliographie

Bibliographie

Présentations accessibles du forcing en théorie des ensembles :

- Timothy Y. Chow, *A beginner's guide to forcing*, Contemporary Mathematics (479), 2008. `https://arxiv.org/abs/0712.1320`
- Robert S. Wolf, *A tour through mathematical logic*, chapitre 6. Carus Mathematical Monographs, 2005.

Le forcing comme traduction de propositions et de preuves :

- A. Miquel, *Forcing as a Program Transformation*, LICS 2011. `https://www.fing.edu.uy/~amiquel/publis/lics11.pdf`
- G. Jaber, N. Tabareau, M. Sozeau, *Extending Type Theory with Forcing*, LICS 2012. `https://hal.inria.fr/hal-00685150/`
- G. Jaber, G. Lewertowski, P.-M. Pédrot, N. Tabareau, M. Sozeau, *The Definitional Side of the Forcing*, LICS 2016. `https://hal.inria.fr/hal-01319066`