MPRI 2.7.1
2017-18

Projet

Formalisation et étude de l'arithmétique de Peano en Coq

1 Objectif

L'objectif de ce projet est d'étudier comment certaines preuves effectuées dans l'arithmétique de Peano classique (c'est-à-dire avec le tiers-exclu) peuvent être rendues constructives et ainsi interprétées en Coq.

2 L'arithmétique de Peano

Il s'agit d'un système formel ancien (1889) ; on trouvera une traduction, du latin vers l'anglais, de l'article originel dans [6]. On trouvera également, une présentation plus récente dans [2] ou dans les notes de cours. Il s'agit surtout d'un système formel à la fois particulièrement concis, et néanmoins expressif, ce qui en fait un objet d'étude apprécié et appréciable.

Ce système vise a formaliser l'arithmétique des entiers naturels, jusqu'aux preuves par récurrence. Formellement, il s'agit d'un système de logique du premier ordre, avec quatre constantes de fonctions :

— 0, d'arité zéro,
— S, d'arité un,
— $+$ et $*$, d'arité deux.

On utilisera la notation infixe habituelle pour les fonctions $+$ et $*$. On dispose par ailleurs d'un seul symbole de prédicat, l'égalité. C'est bien sûr un prédicat binaire et là encore on utilisera la notation infixe habituelle.

Les régles de déduction sont le calcul des prédicats habituel. On note $\neg P$ pour $P \Rightarrow \bot$. Les axiomes, connus sous le nom "d'axiomes de Peano" sont les suivants :

1. $\forall x.\neg S(x) = 0$
2. $\forall x.\exists y.(\neg x = 0 \Rightarrow S(y) = x)$
3. $\forall x.\forall y.(S(x) = S(y) \Rightarrow x = y)$
4. $\forall x.x + 0 = x$
5. $\forall x.\forall y.S(x) + y = S(x + y)$
6. $\forall x.(0 * x = 0)$
7. $\forall x.\forall y.S(x) * y = (x * y) + y$

auxquels s'ajoute le schéma d'axiome suivant (récurrence) :

$$P(0) \Rightarrow (\forall x.P(x) \Rightarrow P(S(x))) \Rightarrow \forall x.P(x)$$

paramètré par un prédicat quelconque P. Finalement, on dispose des propriétés habituelles de l'égalité (reflexivité, élimination).

Même si Peano s'interessait originellement à la logique classique, le calcul des prédicats utilisé peut être respectivement minimal, intuitionniste ou classique. On parlera donc d'*arithmétique de Peano* pour la version classique du formalisme, et d'*arithmétique de Heyting* pour sa contre-partie intuitionniste ; ou simplement d'arithmétique classique, respectivement intuitionniste.

3 Formalisation de l'Arithmétique de Peano en Coq

La première étape du travail consiste à formaliser en Coq ce qu'est une démonstration dans l'arithmétique de Peano. Pour cela, on devra, au moins, définir deux types correspondant respectivement aux représentations des objets et des propositions de l'arithmétique.

Voici quelques indications et conseils.

3.1 Représentation des variables

La présence de quantificateurs nécessite la définition d'une operation de substitution. Il faut donc traiter les problèmes de capture de variables, d'α-conversion, etc. Une bonne possibilité pour cela d'utiliser une notation à base d'indices de de Bruijn, par exemple en utilisant la bibliothèque *Autosubst* [5].

Si jamais on choisit de ne pas utiliser cette bibliothèque, on doit définir soi-même les fonctions de *lifting* des indices et on peut pour cela s'inspirer des travaux de Gérard Huet [4], repris par Thorsten Altenkirch puis Bruno Barras [1], pour formaliser le λ-calcul.

Dans les deux cas, quand on définit en Coq les dérivations arithmétiques, seules les règles portant sur les quantificateurs font intervenir les opérations sur les indices de de Bruijn. Une possibilité est d'indiquer explicitement dans le contexte les liaisons de variables. Par exemple :

```
Inductive Ctxt : Set :=
  nilc : Ctxt
| intc : Ctxt -> Ctxt
| assume : Pprop -> Ctxt -> Ctxt.
```

L'idée est, comme dans Coq, de lier les variables de termes explicitement dans le contexte. Par exemple, le contexte

$$[x : \texttt{nat}; x = 0; y : \texttt{nat}; y = y]$$

est représenté par

```
(assume (eqp (var O)(var O)) (intc (assume (eqp (var O) Ot) (intc nilc))))
```

où `var` est le constructeur de terme correspondant aux variables, `Ot` la constante 0 et `eqp` le constructeur du type des formules d'égalité.

La règle d'élimination du quantificateur existentiel devient alors :

$$\frac{\Gamma \vdash \exists.A \qquad (\text{intc}(\Gamma)) :: A \vdash C \uparrow^1}{\Gamma \vdash C}$$

où $\uparrow^1$ est l'opération consistant à augementer de 1 tous les de Bruijn correspondants à des variables libres dans A.

3.2 Formalisation du calcul des prédicats

L'un des buts étant de traduire par la suite les représentations des démonstrations dans l'arithmétique en de vraies démonstrations Coq, il est plus rationnel d'utiliser la déduction naturelle. Nous rappellons les règles de la déduction naturelle en figure 1.

3.3 Logiques intuitionniste et classique

On demande en fait de coder *deux* formalismes. Les arithmétiques de Peano et de Heyting. Plutôt que de dupliquer la définition des règles d'inférence, il parait judicieux de paramètrer la définition de la déduction naturelle par un ensemble d'axiomes (et schéma d'axiomes). Cet ensemble pouvant alors par la suite être instancié

$$\frac{}{\Gamma :: A \vdash A}\ (\textsc{Assume}) \qquad \frac{\Gamma \vdash A}{\Gamma :: B \vdash A}\ (\textsc{Weak})$$

$$\frac{\Gamma :: A \vdash B}{\Gamma \vdash A \Rightarrow B}\ (\textsc{imp-i}) \qquad \frac{\Gamma \vdash A \Rightarrow B \qquad \Gamma \vdash A}{\Gamma \vdash B}\ (\textsc{imp-e})$$

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ (\textsc{and-i}) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash A}\ (\textsc{and-l-e}) \qquad \frac{\Gamma \vdash A \wedge B}{\Gamma \vdash B}\ (\textsc{and-r-e})$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash A \vee B}\ (\textsc{or-l-i}) \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \vee B}\ (\textsc{or-r-i}) \qquad \frac{\Gamma \vdash A \vee B \qquad \Gamma :: A \vdash C \qquad \Gamma :: B \vdash C}{\Gamma \vdash C}\ (\textsc{or-e})$$

$$\frac{\Gamma \vdash \bot}{\Gamma \vdash A}\ (\textsc{bot-e})$$

$$\frac{\Gamma \vdash A \qquad x \notin \Gamma}{\Gamma \vdash \forall x.A}\ (\textsc{forall-i}) \qquad \frac{\Gamma \vdash \forall x.A}{\Gamma \vdash A[x \setminus t]}\ (\textsc{forall-e})$$

$$\frac{\Gamma \vdash A[x \setminus t]}{\Gamma \vdash \exists x.A}\ (\textsc{exist-i}) \qquad \frac{\Gamma \vdash \exists x.A \qquad \Gamma :: A \vdash B \qquad x \notin \Gamma :: B}{\Gamma \vdash B}$$

FIGURE 1 – La déduction naturelle

— soit par les axiomes de Peano,
— soit par les axiomes de Peano enrichis de la règle du tiers exclu.

Ces deux intanciations correspondant, bien sûr, aux arithmétiques de Peano et de Heyting.

4 Réflection

4.1 Principe

Le principe de la réflection est de traduire dans le méta-formalisme (ici Coq), les constructions faites dans le formalisme représenté (ici l'arithmétique de Peano). D'un point de vue externe, il est bien clair que Coq est une extension de l'arithmétique de Peano intuitionniste ; l'on cherche donc à materialiser cela.

Pour illustrer cela, notons $\forall x.\exists y.x = S(y) \vee x = 0$ la proposition de l'arithmétique de Peano ; cad. qu'il s'agit d'un objet d'un type nommé, par exemple, `Pformula` et qui aura été défini dans la première partie du travail. On définira donc une fonction, notée par example `tr`, tel que `(tr` $\forall x.\exists y.x = S(y) \vee x = 0$`)` se réduise vers l'objet de type `Prop` suivant : `forall x, exists y, x=(S y)\/x=0` (ou `forall x, {y:nat|x=(S y)\/x=0}` si vous avez travaillé dans `Set`).

4.2 Questions : implémentation

On demande, dans un premier temps d'écrire en Coq les fonctions de réflection. Ensuite, de prouver que s'il existe une dérivation dans l'arithmétique de Peano intuitionniste d'une formule P dans un contexte Γ, alors, il existe un terme Coq de type $\texttt{tr}(\Gamma) \to \texttt{tr}(P)$. C'est cette dernière étape qui constitue la réflection proprement dite.

4.3 Questions : théorie

Indiquez informellement, ce que l'existence de cette fonction implique logiquement par rapport aux formalismes logiques que sont l'arithmétique intuitionniste et Coq. Pourrait-on définir l'inverse de cette fonction ?

5 Arithmétique classique

Il n'est pas possible de réflechir l'arithmétique classique dans Coq de la même manière. En revanche, l'on dispose d'un certain nombre de résultats partiels. On se propose d'étudier cela dans cette partie.

On sait depuis longtemps qu'il est possible de plonger un formalisme classique dans sa contre-partie intuitionniste. Pour cela, l'on affaiblit les propositions prouvées classiquement. Voici quelques principes qui sous-tendent ces constructions :

— On obtient un système équivalent au système classique en ajoutant l'axiom du tiers-exclu au système intuitionniste. Mais l'on peut également utiliser le schéma d'axiome suivant : pour toute proposition P, $\neg\neg P \Rightarrow P$.

— Cet axiome n'est donc, bien sûr, pas prouvable intuitionnistiquement. En revanche, il est vrai en logique intuitionniste que $\neg\neg\neg P \Rightarrow \neg P$.

L'idée est donc de rajouter, à tous les niveaux d'une proposition, un certain nombre de double-négations. L'on obtient alors, à partir d'une proposition P une proposition affaiblie $\overline{P}$, telle que :

1. $P \Rightarrow \overline{P}$ est vrai de manière intuitionniste,
2. la transformée du tiers-exclu, ou le schéma de propositions $\overline{\neg\neg P \Rightarrow P}$ sont également prouvables de manière intuitionniste.

En conclusion, si P est prouvable dans l'arithmétique classique, alors $\overline{P}$ est prouvable dans l'arithmétique intuitionniste.

On se propose de formaliser ces résultats dans Coq. Le paragraphe suivant présente une adaptation, au cadre de Coq, d'un résultat du logicien américain H. Friedman [3].

5.1 Une présentation de la traduction de Friedman

Soit A une proposition de l'arithmétique. On note $\neg^A P$ la proposition $P \Rightarrow A$. À toute proposition P de l'arithmétique, on associe la proposition P^A, défini par les équations suivantes :

$$\begin{aligned}
\bot^A &\equiv \bot \vee A \\
t_1 = t_2^A &\equiv t_1 = t_2 \vee A \\
(P_1 \wedge P_2)^A &\equiv \neg^A\neg^A(P_1^A) \wedge \neg^A\neg^A(P_2^A) \\
(P_1 \vee P_2)^A &\equiv \neg^A\neg^A(P_1^A) \vee \neg^A\neg^A(P_2^A) \\
(P_1 \Rightarrow P_2)^A &\equiv \neg^A\neg^A(P_1^A) \Rightarrow \neg^A\neg^A(P_2^A) \\
(\forall x.P)^A &\equiv \forall x.\neg^A\neg^A(P^A) \\
(\exists x.P)^A &\equiv \neg^A\neg^A\exists x.(P^A)
\end{aligned}$$

Lemme 1 *On étend, de manière triviale, la transformée par double négation aux contextes. S'il existe une dérivation de $\Gamma \vdash P$ dans l'arithmétique de Peano, il existe une dérivation de $\Gamma^A \vdash \neg^A\neg^A P^A$ dans l'arithmétique de Heyting.*

Lemme 2 *Les formules sans quantificateurs sont décidables (en particulier dans Coq).*

Lemme 3 *Soit P une formule sans quantificateurs. On a, dans Coq, $P^A \iff P \vee A$.*

Soit alors une preuve classique de $\exists x.P(x)$, où P est une formule sans quantificateurs ; une telle proposition est dite Σ_0^1. D'après le lemme 1, on a donc une preuve intuitionniste de $\neg^A\neg^A\exists x.P^A(x)$, c'est-à-dire

$$((\exists x.P^A(x)) \Rightarrow A) \Rightarrow A.$$

En instanciant A par la proposition originelle $\exists x.P(x)$, et après reflection, on obtient donc :

$$((\exists x : \texttt{nat}.P^{\exists x:\texttt{nat}.P(x)}(x)) \Rightarrow \exists x : nat.P(x)) \Rightarrow \exists x : nat.P(x)$$

ce qui d'après le lemme 3 est équivalent à $\exists x.P(x)$.

On a donc ainsi obtenu une preuve intuitionniste de $\exists x : nat.P(x)$ à partir d'une preuve classique de la même proposition.

En considérant le même développement avec des variables libres, on étend ce résultat aux formules Π_0^2, c'est-à-dire de la forme

$$\forall x.\exists y.P(x, y)$$

où P est sans quantificateurs.

5.2 Question

1. Définir la non-non-traduction dans Coq.
2. Prouver que si $\Gamma \vdash P$ est prouvable dans l'arithmétique classique, alors, pour tout A, $\Gamma^A \vdash \neg^A\neg^A P^A$ est prouvable dans l'arithmétique intuitionniste.
3. Soit P une proposition sans quantificateurs. Prouvez que pour toute interpretation I, toute proposition A,
$$P^A \iff P \vee A$$
4. On appelle formule Σ_1^0 une proposition de la forme $\exists x.P(x)$, où P est sans quantificateur. Monter que si une telle formule est prouvable dans l'arithmétique classique, elle l'est aussi dans Coq.

Références

[1] B. Barras. Coq en coq. Rapport de Recherche 3026, INRIA, October 1996. disponible sur `https://hal.inria.fr/inria-00073667`

[2] R. Cori et D. Lascar. *Logique Mathématique, Cours et exercices.* Axiomes. Masson, 1993.

[3] H. Friedman. Classically and intuitionistically provably recursive function. In G.H. Müller and D.S. Scott, editors, *Higher Set Theory*, Oberwolfach, Germany 77, 1978. Springer-Verlag.

[4] G. Huet. Residual theory in λ-calculus : A complete gallina development. Technical Report 2002, INRIA, 1993. disponible sur `https://hal.inria.fr/inria-00074663`

[5] Steven Schäfer, Tobias Tebbi, and Gert Smolka. Autosubst : Reasoning with de bruijn terms and parallel substitutions. In Xingyuan Zhang and Christian Urban, editors, *Interactive Theorem Proving - 6th International Conference, ITP 2015, Nanjing, China, August 24-27, 2015*, LNAI. Springer-Verlag, Aug 2015.

[6] J. van Heijenoort. *From Frege to Gödel, A Source Book in Mathematical Logic, 1879–1931.* Source Books in the History of Science. Harvard University Press, 1967.