

Utiliser Coq dans l'enseignement des Mathématiques

Yves Bertot
yves.bertot@inria.fr

Février 2022

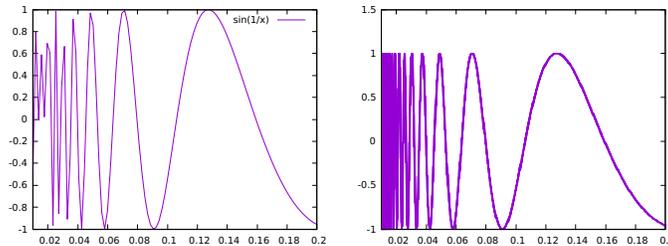
Un groupe de chercheurs enthousiastes développe le système Coq depuis des dizaines d'années. Ce logiciel a été utilisé pour décrire des langages de programmation, des algorithmes dans une grande variété de domaines (automates, géométrie, calcul numérique, etc) et des preuves mathématiques. Depuis longtemps, des chercheurs et enseignants se sont intéressés à l'utilisation de ce logiciel dans l'enseignement, mais faute de moyens, le système n'a pas évolué pour être adapté à cet usage, restant essentiellement un outil pour experts, à la fois en informatique et en mathématiques.

Du point de vue des mathématiques, le système Coq est maintenant assez puissant pour représenter les preuves de théorèmes mathématiques assez avancés : l'irrationalité de $\zeta(3)$, la transcendance de π et e et le théorème de Lindemann-Weierstrass sur la transcendance, l'insolvabilité par radicaux de polynômes de degré 5. En analyse, il y a également beaucoup de résultats : Taylor-Lagrange, quelques intégrales remarquables, le théorème de Lax-Milgram. Tout ce savoir représentable dans un système de preuve permet d'envisager de nouvelles manières d'enseigner aux étudiants, en leur donnant de nouveaux moyens de d'appréhender le savoir mathématique.

Dans ce document, nous allons donner un exemple de connaissances mathématiques représentées en Coq et de leur utilisation pour faire prendre conscience aux étudiants de certains problèmes et de leur résolution par des techniques à apprendre.

1 Précision dans les calculs et courbes mathématiques

Si l'on demande à un outil de calcul de formule mathématiques de tracer la courbe de la fonction $\sin \frac{1}{x}$ près de 0, on peut obtenir le résultat de gauche dans la figure suivante (`gnuplot`, Nov. 2021) :



Cette courbe est clairement erronée. La raison est que `gnuplot` est prévu pour afficher les courbes de fonctions « bien élevées ». Ce logiciel n'est pas prévu pour afficher des courbes dont la valeur change trop brutalement à l'intérieur d'un intervalle. Il ne sait même comment exprimer ou vérifier qu'il pourrait y avoir un problème.

Dans `Coq`, il existe maintenant une fonction d'affichage de courbes qui donne le résultat de droite¹. Il est possible de traiter de façon logique le comportement de la fonction et d'exprimer la propriété minimale que doit satisfaire un point de l'image pour être hors de la courbe. Ceci se fait par un calcul d'intervalle, dont la correction s'exprime par des propriétés logiques.

Le calcul par intervalle peut être fait automatiquement, mais il a aussi des limites. Si l'on s'intéresse à la fonction `sin`, le système permet de démontrer que cette fonction a une valeur toujours inférieure à son argument, lorsque celui-ci est positif. L'outil de calcul par intervalle naïf ne peut pas démontrer ce résultat automatiquement, mais l'utilisateur peut en faire la preuve et faire vérifier cette preuve par `Coq`.

- pour $\frac{\pi}{2} < x$, on a aisément $\sin x \leq 1 < x$
- pour $0 < x \leq \frac{\pi}{2}$, on a $x - \sin x = (1 - \cos c) * (x - 0)$ avec $0 < c < x$, par un simple calcul de dérivée et le théorème des accroissements finis. La valeur de $1 - \cos c$ est elle-même strictement positive dans cet intervalle, etc.

L'ensemble de cette preuve demande que l'on décompose la demi-droite des réels positifs en 2 parties, que l'on fasse appel au théorème des accroissements finis, que l'on calcule la dérivée d'une fonction composite, etc. Tous ces exercices peuvent être faits par les étudiants, et le système de preuve vérifiera que toutes les conditions sont réunies. Faire tous ces raisonnements avec une vérification par ordinateur permet de travailler à son rythme et de comprendre à quels endroits un raisonnement rigoureux est attendu.

Aujourd'hui, il est possible d'exprimer cette propriété et ce raisonnement en `Coq`, avec le texte suivant² :

```
Lemma sinx_ltx x : 0 < x -> sin x < x.
Proof.
intros xgt0.
```

1. L'échelle est légèrement différente
2. Extrait d'un fichier disponible en https://liberabaci.gitlabpages.inria.fr/sin_properties.v

```

enough (0 < x - sin x) by check.
assert (1 < PI / 2) by exact PI2_1.
(* Cut the proof in two parts: after PI/2 and before. *)
assert (right_part : PI / 2 < x -> 0 < x - sin x).
  assert (sin x <= 1) by (assert (tmp := SIN_bound x); check).
  check.
assert (left_part : ~ (PI / 2 < x) -> 0 < x - sin x).
  intros xsmall.
(* Show the value of the derivative. *)
assert (der : forall c, 0 <= c <= x ->
  derivable_pt_lim (fun x => x - sin x) c
    (1 - cos c)).
  intros c _.
  auto_derive; check.
assert (exists c, x - sin x - (0 - sin 0) =
  (1 - cos c) * (x - 0) /\ 0 < c < x)
  as [c [feq cint]].
  apply (MVT_cor2 (fun x => x - sin x)
    (fun x => 1 - cos x) _ _ xgt0 der).
rewrite sin_0, !Rminus_0_r in feq.
rewrite feq.
apply Rmult_lt_0_compat; cycle 1.
  easy.
enough (cos c < 1) by check.
rewrite <- cos_0.
apply cos_decreasing_1.
  check.
  check.
  check.
  check.
either_or_not (PI / 2 < x); check.
Qed.

```

2 Les défis de l'utilisation d'un système de preuve

Dans l'état actuel, le système Coq n'est pas conçu pour un objectif d'enseignement des mathématiques. Il y a une longue expérience d'enseignement autour de Coq, mais l'objectif est alors « apprendre à se servir de Coq », ou plus précisément « apprendre à se servir de Coq pour construire du logiciel sûr » et « apprendre à se servir de Coq pour faire des mathématiques avancées ».

Si l'on veut améliorer le système Coq pour un objectif d'enseignement, il faut réfléchir à de nouveaux aspects :

- Quels étapes de l'apprentissage des mathématiques peuvent être facilitées par l'utilisation de Coq ?

- Dans quel ordre faut-il introduire de nouveaux concepts? Un exemple remarquable est celui des nombres (naturels, entiers, réels, complexes).
- Quelles aspects de la théorie de la preuve entrent en conflit avec l'objectif d'enseignement? Un exemple remarquable dans le cas de Coq est l'utilisation des types à la place des ensembles.
- Comment permettre une prise en main rapide, fluide vis-à-vis des conditions d'enseignement traditionnel? Par exemple, comment aider les étudiants à trouver les théorèmes et outils de preuve applicables.
- Comment configurer l'automatisation pour améliorer la mémorisation des techniques par les étudiants?

3 Liber Abaci : les premières années universitaires

En l'état actuel des bibliothèques de Coq, une grande partie des contenus utilisés dans les premières années universitaires sont déjà présents dans la base de connaissance. Nous choisissons donc l'objectif d'améliorer le système pour ce public précis et nous espérons impliquer de nombreux enseignants à ce niveau d'étude pour comprendre les besoins et suivre les progrès de nos travaux.

L'Inria, où le système Coq a été développé depuis des décennies, organise un projet coordonné appelé *Liber Abaci*, réunissant plusieurs équipes de recherche engagées soit sur les fondements et les modifications à apporter au langage, soit sur l'outil interactif, soit sur la création de contenus adaptés sur l'enseignement. Yves Bertot joue le rôle de coordinateur pour ce travail collaboratif et peut être contacté pour plus de renseignements.