# Obtention de témoins en arithmétique réelle non-linéaire

**Encadrants :** Sujet proposé par David Monniaux, directeur de recherche au CNRS, et Pierre Corbineau, maître de conférence à l'Université Joseph Fourier.

**Lieu :** laboratoire VERIMAG, Grenoble.

## 1 Sujet

Des outils informatiques tels que Coq, Isabelle etc. permettent de prouver des théorèmes de mathématiques, ou la conformité d'un logiciel à une spécification. Plus exactement, ces assistants de preuve vérifient les étapes d'une preuve fournie par un humain : ce dernier fournit les grandes étapes, et l'assistant fait de lui-même certaines étapes bien circonscrites. Bien entendu, plus l'assistant est puissant, plus il peut effectuer d'étapes de preuve sans aide humaine, et moins la tâche de l'humain est difficile et fastidieuse.

L'outil Coq a ceci de particulier parmi les assistants de preuve qu'il construit explicitement une preuve dans un petit langage élémentaire, et qu'ensuite il vérifie cette preuve à l'aide d'un algorithme de typage relative- ment simple. Ainsi, il n'y a pas besoin de faire confiance aux procédures assez complexes qui permettent d'automatiser les étapes de preuves, car celles-ci fournissent non seulement un résultat « prouvé, OK », mais également un témoin de la correction de leur réponse.

Lorsque le théorème que l'on veut montrer est de la forme $P_1(X_1, \ldots, X_n) \geq 0 \wedge \cdots \wedge P_m(X_1, \ldots, X_n) \geq 0 \implies P(X_1, \ldots, X_n) > 0$, une méthode possible est d'exprimer $P$ comme une combinaison de produits des $P_i$ et de sommes de carrés de polynômes ; cette combinaison constitue un témoin facile à vérifier (il suffit d'appliquer des propriétés comme « la somme de deux réels positifs est positive », « le carré d'un réel positif est positif »). La difficulté est d'obtenir cette décomposition.

Il a été proposé de réduire ce problème à de la programmation semidéfinie, c'est-à-dire à la recherche de $F_0 + \sum_{i=1}^{n} \lambda_i F_i$ semidéfinie positive, étant données $F_0, \ldots, F_n$ des matrices symétriques réelles, problème pour lequel il existe des outils numériques. Malheureusement, ces outils ne conviennent pas si l'ensemble des solutions de ce problème est d'intérieur vide, ce qui est en pratique souvent le cas. Nous avons proposé une méthode contournant cette difficulté, s'appuyant notamment sur l'usage de la réduction de réseau par l'algorithme LLL (Monniaux et Corbineau 2011).

L'objet du stage est de l'étudier, de travailler certaines preuves, et/ou de l'améliorer. On pourra par exemple par introduire une détection automatique de symétries dans l'implication polynomiale, ou l'utilisation dans certains cas de programmation linéaire (et non plus semidéfinie).

## 2 Compétences requises

Il faut les compétences de mathématiques que l'on attend d'un normalien.

Par souci de commodité, nous implémentons en Sage, un logiciel de calcul formel basé sur le langage Python. Ce logiciel est d'abord facile et il n'est pas demandé de connaissances sophistiquées en Python ; le stagiaire pourra apprendre la programmation en Sage en quelques jours.

## Références

Monniaux, David et Pierre Corbineau (août 2011). "On the Generation of Positivstellensatz Witnesses in Degenerate Cases". Dans : *Interactive Theorem Proving (ITP)*. Sous la dir. de Marko Van Eekelen et al. T. 6898. Lecture Notes in Computer Science. Springer Verlag, p. 249–264. isbn : 978-3-642-22862-9. arXiv : 1105.4421.