

Automatiser la vérification d'expressions algébriques

simplification symbolique et preuves par réécriture

Victor ROBERT

Juin 2025

Abstract

Dans ce TIPE, on s'intéresse aux liens entre le raisonnement en mathématiques et la programmation informatique. En effet, en informatique, on ne fait pas vraiment de manipulations abstraites comme en mathématiques. Chaque variable doit avoir une valeur définie en mémoire pour être interprétée. On pourra alors étudier les moyens possibles pour simuler un raisonnement mathématique faisant intervenir des objets abstraits. Pour ce faire on utilise un interpréteur en OCaml (Trouvable ici) qui implémente un *langage dédié* (ou DSL) le plus simpliste possible. Celui-ci permettant de faire des transformations mathématiques diverses, on peut néanmoins s'appuyer sur des preuves de correction pour être sûr de conserver une certaine rigueur. Une idée serait alors d'arriver à prouver le prouveur dans sa propre syntaxe, de sorte qu'on arrive à un système complètement autonome.

References

- BALLAND, Emilie, and Pierre-Etienne MOREAU. n.d. "Optimizing Pattern Matching by Program Transformation." <https://inria.hal.science/inria-00001127/document>.
- BARRAS, Bruno. 1996. "Coq En Coq." <https://inria.hal.science/inria-00073667/document>.
- CRABBE, Marcel. 1986. <https://www.cahiersdelogique.be/Copies/CCL6/CC L6-crabbe.PDF>.
- KLOP, Jan Willem. 2000. <https://ir.cwi.nl/pub/6129/6129D.pdf>.
- MARANGET, Luc. 2008. <https://www.cs.tufts.edu/~nr/cs257/archive/luc-maranget/jun08.pdf>.
- MESEGUER, José. n.d. <https://maude.cs.illinois.edu/w/images/7/70/Maude-tapas.pdf>.