
Optimisations dans le compilateur formellement vérifié CompCert

David Monniaux*¹

¹VERIMAG – Centre National de la Recherche Scientifique : UMR5104, Université Grenoble Alpes,
Institut polytechnique de Grenoble (Grenoble INP) – France

Résumé

CompCert est un compilateur de C vers de multiples architectures cibles, avec l'originalité d'une preuve de préservation de la sémantique des programmes : si un programme C a une exécution bien définie (succession d'appels de fonctions externes, d'accès aux variables volatiles...), alors le programme compilé suit la même exécution. Cette preuve s'obtient par composition de relations de simulation (pas à pas ou plus complexes) entre les différentes représentations intermédiaires, reliées par les passes de transformation et d'optimisation. Elle est vérifiée à l'aide de l'assistant Coq.

La version "officielle" de CompCert (diffusée sur github et achetable auprès de la société Absint) est modérément optimisante. À Verimag, nous avons rajouté diverses optimisations permettant des gains de performance notables, notamment des ordonnancements d'instructions adaptés à divers processeurs, dont des processeurs VLIW et des éliminations de code redondant.

La difficulté est qu'il ne suffit pas d'implanter des méthodes existantes telles que celles présentes dans gcc et LLVM, mais qu'il faut dégager une approche de preuve qui passe à l'échelle et soit facile à maintenir par la suite. Notre preuve de correction de l'ordonnancement est basé sur un vérificateur prouvé, fonctionnant par exécution symbolique.

*Intervenant