

Coloration avec préférences dans les graphes triangulés ou : vérification formelle d'un algorithme de coloration

Sandrine Blazy Benoît Robillard Éric Soutif
blazy@ensiie.fr robillard@ensiie.fr soutif@cnam.fr
Laboratoire CEDRIC, 292 rue Saint-Martin, 75 141 Paris cedex 03

L'utilisation de méthodes formelles permet d'obtenir des garanties fortes sur le code source des logiciels critiques. Cependant, des bugs dans le compilateur utilisé pour produire un exécutable à partir de ces sources peuvent invalider ces garanties. Ce risque est écarté si l'on vérifie formellement le compilateur : on prouve que le compilateur produit du code machine qui se comporte comme le code source qu'on lui fournit. Le travail réalisé s'inscrit dans le cadre du projet Compcert ayant pour but le développement et la vérification formelle, utilisant l'assistant de preuve Coq, d'un compilateur réaliste potentiellement utilisable pour le logiciel embarqué critique.

Un compilateur effectue une succession de transformations pour générer un code machine à partir d'un code source. L'allocation de registres par coloriage de graphes est une de ces transformations ; c'est une des plus difficiles à mettre en œuvre. Elle a pour but de proposer une utilisation optimale des registres, c'est-à-dire minimisant l'occupation de la mémoire durant l'exécution d'un programme. Le compilateur Compcert utilise une des meilleures heuristiques d'allocation de registres, mais celle-ci n'a pas été spécifiée formellement en Coq.

Aussi, nous avons proposé un algorithme de coloration avec préférences, qui a la particularité de généraliser deux problèmes d'optimisation connus : le problème de coloration et le problème de multicoupe minimale. La résolution de ce nouveau problème dans les graphes triangulés peut être réalisée grâce à un test d'existence de solution admissible (que nous réalisons grâce à l'algorithme de coloration gourmande) suivi d'un algorithme de coupes.

Notre algorithme a été formellement spécifié en Coq, ce qui a permis de vérifier formellement certaines de ses propriétés. En particulier, nous avons formellement spécifié un algorithme de coloration gourmande, ainsi qu'un algorithme de recherche d'ordre d'élimination simplicial. Nous avons également prouvé en Coq la validité de la solution calculée par notre algorithme, ainsi que son optimalité dans le cas de graphes triangulés.

Ce travail est un premier pas vers la vérification formelle d'algorithmes opérant sur des graphes.