

Compilation optimisante pour la précision numérique

Lieu : Université de Perpignan Via Domitia.

Equipe d'accueil : Equipe Projet DALI, laboratoire LIRMM (UMR 5506).

Directeur de thèse : Matthieu Martel (matthieu.martel@univ-perp.fr).

Mots clés : Compilation, Transformation de Programmes, Sémantiques formelles, Interprétation Abstraite, Précision Numérique, Nombres flottants.

Remarque : Bourse de thèse disponible sur ce sujet pour septembre 2013 (financement ANR).

Les erreurs d'arrondi sont indissociables des calculs sur ordinateurs dans lesquels le résultat de chaque opération doit être approché pour être représenté en mémoire sur un nombre fini de bits. Ces erreurs sont en général acceptables dans la mesure où le résultat produit par la machine est proche de celui que l'on aurait obtenu en utilisant des nombres réels. Cependant, elles peuvent aussi être amplifiées par propagation, dénuant de sens le résultat d'un calcul. Aussi, est-il souhaitable d'optimiser les traitements numériques présents dans des programmes au moment de la compilation, afin que le code généré produise des résultats aussi proche que possible de ceux que l'on obtiendrait si les calculs étaient effectués avec des nombres réels.

Actuellement, les méthodes permettant d'estimer ou de borner les erreurs d'arrondi survenant au cours d'un calcul réalisé sur ordinateur rencontrent un vif succès, notamment dans le domaine des systèmes embarqués critiques [3] où elles sont utilisées afin de valider des systèmes hautement critiques, en avionique par exemple. Cependant, ces techniques n'indiquent pas comment réduire les erreurs ainsi mesurées. Or, améliorer manuellement la précision numérique d'un programme est tout à la fois fastidieux et difficile, car l'arithmétique utilisée est particulièrement non-intuitive (erreurs d'arrondis, opérations non-associatives, non-inversibles, etc.)

Cette thèse concerne une nouvelle sorte de transformation automatique de programmes permettant d'optimiser la précision des calculs au moment de la compilation : si l'on suppose qu'un programme fonctionnerait correctement s'il était exécuté en utilisant des nombres réels, il est possible de le modifier automatiquement, par des méthodes de transformation de codes, afin de produire un autre programme sémantiquement équivalent et numériquement plus précis [5] (autrement dit, d'améliorer sa précision). Pour cela, on autorise des ré-écritures de formules, normalement fausses dans l'arithmétique de la machine (associativité, factorisation, modification de constantes) de façon à générer automatiquement une nouvelle formule, mathématiquement égale à la première et plus précise en ce qui concerne son évaluation par un ordinateur. La correction de la transformation repose sur la théorie de l'interprétation abstraite [1, 2].

Les premiers travaux dans cette direction ont porté sur l'optimisation des expressions arithmétiques et la notion de Program Equivalence Graph [6, 4]. L'objet de cette thèse sera de se consacrer aux optimisations de blocs de codes : ré-agencement des calculs effectués dans différentes instructions, adaptation de techniques d'optimisations à la compilation au cas de la précision numérique (loop unrolling, spécialisation, etc.) Les résultats théoriques obtenus seront intégrés au logiciel Sardana qui permet d'optimiser la précision de programmes synchrones écrits en Lustre. Cette thèse s'inscrit dans le cadre de l'ANR CAFEIN dans laquelle l'ONERA et Rockwell Collins fourniront des exemples réels de codes à optimiser.

Références

- [1] P. Cousot and R. Cousot. Abstract interpretation : A unified lattice model for static analysis of programs by construction of approximations of fixed points. In *Principles of Programming Languages 4*, pages 238–252. ACM Press, 1977.
- [2] P. Cousot and R. Cousot. Systematic design of program transformation frameworks by abstract interpretation. In *Conference Record of the Twentyninth Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 178–190, Portland, Oregon, 2002. ACM Press, New York, NY.
- [3] D. Delmas, E. Goubault, S. Putot, J. Souyris, K. Tekkal, and F. Vedrine. Towards an industrial use of fluctuat on safety-critical avionics software. In *Formal Methods for Industrial Critical Systems, 14th International Workshop, FMICS, 2009*.
- [4] A. Ioualalen and M. Martel. A new abstract domain for the representation of arithmetic expressions. In *Static Analysis Symposium, SAS, LNCS*. Springer, 2012.
- [5] M. Martel. Enhancing the implementation of mathematical formulas for fixed-point and floating-point arithmetics. *Journal of Formal Methods in System Design*, 35 :265–278, 2009.
- [6] R. Tate, M. Stepp, Z. Tatlock, and S. Lerner. Equality saturation : A new approach to optimization. In *POPL*, pages 264–276. ACM, 2009.