

Élimination des horloges des programmes X10 polyédriques

Eric Violard (Eric.Violard@inria.fr)

Sujet de TER



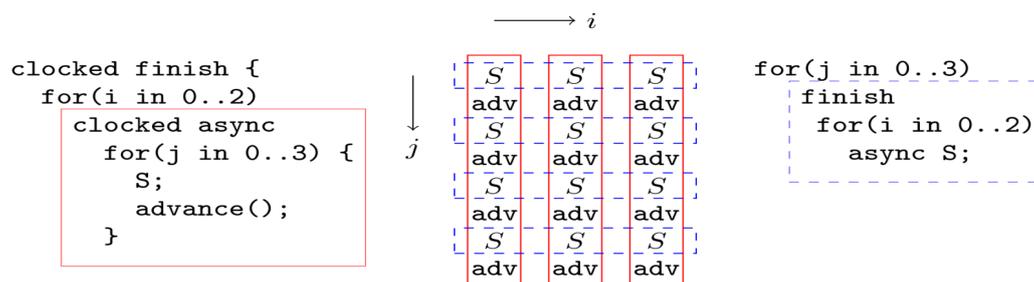
Ce TER concerne la parallélisation automatique de programmes et il s'inscrit dans les activités de recherche de l'équipe INRIA CAMUS qui se concentrent sur l'exploitation efficace des processeurs multi-cœurs.

Le travail porte particulièrement sur l'optimisation de programmes X10. Le langage X10 [3] est un langage parallèle récent et prometteur. Il est développé par IBM Research et conçu pour augmenter la productivité des programmeurs et relever le défi de programmer une large variété de plateformes d'exécution incluant des architectures complexes comme les clusters de processeurs multi-cœurs et les accélérateurs.

La partie séquentielle de X10 est un langage orienté objet de la famille de Java. Cette partie est complétée par un petit nombre de constructions parallèles qui permettent de créer des "activités" et de les synchroniser. Comme beaucoup d'autres langages parallèles, ces constructions sont redondantes et peuvent être utilisées de façon interchangeable, ce qui autorise le programmeur à choisir différentes formes de programmes ou à adopter différents points de vue. En particulier, la synchronisation des activités peut être réalisée en utilisant des horloges qui peuvent être vues comme une généralisation des barrières de synchronisation classiques. La synchronisation avec une horloge est spécifiée par un appel à la méthode `Clock.advanceAll()`. Les activités qui exécutent cette méthode attendent que toutes les autres activités aient fait un appel à cette même méthode, puis reprennent leur exécution à la même date (logique).

Nous avons récemment présenté une transformation [2] de programmes qui permet, pour une large classe de programmes X10 (les programmes dits "polyédriques"), de passer automatiquement d'une forme de programmes à une autre : d'une composition parallèle d'activités séquentielles qui se synchronisent en utilisant une horloge, à une séquence

d'activités parallèles sans horloge.



Cette transformation ouvre la voie à de nouvelles optimisations, élargissant la portée de la parallélisation automatique. Elle est basée sur le modèle polyédrique [1] qui est ici utilisé pour modéliser le contrôle des instructions. Les manipulations syntaxiques du programme reviennent à des opérations sur des polyèdres.

Objectif

L'objectif de ce TER est d'implémenter cette transformation source-à-source pour le langage X10 dans le cas le plus simple où toutes les dates sont affines. (Un objectif plus lointain est d'implémenter cette transformation dans le cas général et de l'intégrer à un compilateur du langage X10). Le travail consistera à :

- Étudier l'algorithme pour transformer un programme X10 en utilisant le modèle polyédrique.
- Utiliser les outils existants (ISL [4], CLoog) pour modéliser le programme par un polyèdre et appliquer les opérations polyédriques.
- Manipuler l'AST (Abstract Syntax Tree) correspondant au programme X10 original et construire les éléments syntaxique du nouveau programme.
- Coder l'algorithme : calculer les dates, construire un nouvel AST et produire un nouveau programme X10.

References

- [1] Paul Feautrier and Christian Lengauer. The polyhedral model. In David Padua, editor, *Encyclopedia of Parallel Programming*. Springer, 2011.
- [2] Paul Feautrier, Eric Violard, and Alain Ketterlin. Improving X10 Program Performances by Clock Removal. In *Compiler Construction 2014*, Grenoble, France, January 2014. <https://hal.inria.fr/hal-00924206>.
- [3] Vijay Saraswat, Bard Bloom, Igor Peshansky, Olivier Tardieu, and David Grove. X10 language specification version 2.2, March 2012. Available at <http://x10.sourceforge.net/documentation/languagespec/x10-latest.pdf>.

- [4] Sven Verdoolaege. isl: An integer set library for the polyhedral model. In Komei Fukuda, Joris van der Hoeven, Michael Joswig, and Nobuki Takayama, editors, *Lecture Notes in Computer Science, International Congress on Mathematical Software (ICMS 2010), Kobe, Japan, 13-17 September 2010*, pages 299–302. Springer, September 2010. <http://www.cs.kuleuven.be/cgi-bin/dtai/barvinok.cgi>.