Proposition de sujet de stage de M2R

Composition d'ordonnancements parallèles

Encadrants: Christophe Alias et Matthieu Moy

Lieu: Laboratoire de l'Informatique du Parallélisme (LIP)
École Normale Supérieure de Lyon

Depuis le début des années 2000, la limite de miniaturisation des transistors force à multiplier les unités de calcul (processeurs, processeurs spécialisés) des superordinateurs pour améliorer les performances [2]. De nombreux verrous doivent être levés, comme la distribution efficace d'une application sur les unités de calcul. Pour cela, l'application doit être divisée en unités à exécuter en parallèle et les communications entre unités doivent être réglées. Il existe des algorithmes capables de paralléliser automatiquement des noyaux de calcul intensifs comme les opérations matricielles [1]. Malheureusement, ces algorithmes ne passent pas à l'échelle: quand le code est trop gros, la parallélisation ne marche plus.

L'équipe CASH nouvellement créée travaille sur des approches innovantes d'extraction du parallélisme vers une représentation intermédiaire avec pour objectif la production de code efficace pour accélérateurs matériels de type FPGA. Nos outils sont des compilateurs dits « source-to-source », qui lisent du code C séquentiel, en extraient le parallélisme, puis génèrent du code qui explicite le parallélisme.

L'objectif de ce stage est d'expérimenter les stratégies de passage à l'échelle d'un algorithme de parallélisation. On suppose le noyau divisé en plusieurs sous-noyaux pour lesquels la parallélisation est possible. Le but du stage est de trouver une façon de composer les parallélisations des sous-noyaux pour produire une parallélisation globale à la fois correcte et efficace. Plus précisément, une parallélisation est une réorganisation des calculs décrite par une fonction d'ordonnancement θ qui associe à chaque calcul c une date logique $\theta(c)$. Il s'agit donc de trouver une parallélisation globale θ_{noyau} à partir de chaque parallélisation locale $\theta_{\text{sous-noyau}}$. Autrement dit, comment composer chaque parallélisation locale $\theta_{\text{sous-noyau}}$ pour obtenir une parallélisation globale θ_{noyau} correcte et efficace? En particulier, comment synchroniser correctement les communications entre sous-noyaux? Une solution naïve serait d'attendre que la dernière opération d'un sous-noyau soit terminée pour démarrer la première opération du sous-noyau suivant, mais cette solution est loin d'être optimale : elle n'exploite pas correctement le parallélisme disponible, et peut demander de stocker plus de données que nécessaire entre les sous-noyaux. La notion d'efficacité est évidemment multicritère. Dans ce stage, on cherchera à minimiser la taille des canaux de communication. Autrement dit, comment ordonnancer l'écriture et la lecture dans un canal de communication pour limiter la bufferisation?

Plan indicatif

Voici une description plus précise des tâches attendues.

- **1.** Etude bibliographique des techniques de parallélisation automatique dans le modèle polyédrique [1].
- **2.** Définir les notions de composition correcte et efficace dans le formalisme polyédrique. Définir un algorithme de composition.
- **3.** Implémenter et valider l'algorithme sur les benchmarks Polybench/C [3].

Encadrement

Ce stage sera co-encadrée par Christophe Alias (CR1 Inria, ENS-Lyon) et Matthieu Moy (MCF HDR UCBL).

Christophe Alias (`http://perso.ens-lyon.fr/christophe.alias/`) s'intéresse à la synthèse de circuit haut niveau dans le modèle polyédrique depuis plus de 8 ans. Il a co-encadré deux thèses (Alexandru Plesco avec Alain Darte et Tanguy Risset, et Guillaume Iooss avec Sanjay Rajopadhye). Dans le même temps, il a écrit un compilateur de réseaux de processus, transféré dans la startup Xtremlogic qu'il a co-fondé en 2014 avec Alexandru Plesco. Actuellement Christophe Alias est en concours scientifique à 20% dans XtremLogic et n'a plus de charge d'encadrement depuis la soutenance de Guillaume Iooss en juillet 2016.

Matthieu Moy (`https://matthieu-moy.fr`) travaille sur la simulation en SystemC depuis une quinzaine d'années (en partenariat avec STMicroelectronics et en particulier dans le cadre de la HLS), et a déjà encadré plusieurs thèses et post-doctorants sur le sujet. Plus récemment, il s'est intéressé aux calculs de pire temps d'exécution de logiciel et de pire temps de traversée de réseaux sur puces dans le cadre de systèmes temps-réel critiques. Il est titulaire de l'habilitation à diriger des recherches depuis 2014. Anciennement responsable de l'équipe Synchrone du laboratoire Verimag, il a intégré le LIP en septembre 2017. Il co-encadre aujourd'hui 4 thèses.

Compétences souhaitées

Notions en parallélisme. Maîtrise du langage C, bases solides en C++.

Candidatures

Envoyez vos candidatures par email à : Christophe.Alias@ens-lyon.fr et Matthieu.Moy@univ-lyon1.fr.

Nous proposons plusieurs sujets sur des thèmes similaires, n'hésitez pas à en discuter avec nous. Une poursuite en thèse est envisageable.

Références

[1] Paul Feautrier and Christian Lengauer. Polyhedron model. In *Encyclopedia of Parallel Computing*, pages 1581–1592. 2011.

[2] Nor Zaidi Haron and Said Hamdioui. Why is cmos scaling coming to an end? In *Design and Test Workshop, 2008. IDT 2008. 3rd International*, pages 98–103. IEEE, 2008.

[3] Louis-Noël Pouchet. Polybench: The polyhedral benchmark suite. *URL: http://www. cs. ucla. edu/˜ pouchet/software/polybench/[cited July,]*, 2012.