«DÉCOMPOSITION AUTOMATIQUE DES PROGRAMMES PARALLÈLES POUR L’ OPTIMISATION ET LA PRÉDICTION DE PERFORMANCE» PAR MIHAIL POPOV

Présentée par : Mihail Popov Discipline : informatique Laboratoire : LI-PaRAD

Résumé :
Dans le domaine du calcul haute performance, de nombreux programmes étalons ou benchmarks sont utilisés pour mesurer l'efficacité des calculateurs, des compilateurs et des optimisations de performance. Les benchmarks de référence regroupent souvent des programmes de calcul issus de l'industrie et peuvent être très longs. Le processus d’ étalonnage d'une nouvelle architecture de calcul ou d'une optimisation est donc coûteux. La plupart des benchmarks sont constitués d'un ensemble de noyaux de calcul indépendants. Souvent l'étalonneur n'est intéressé que par un sous-ensemble de ces noyaux, il serait donc intéressant de pouvoir les exécuter séparément. Ainsi, il devient plus facile et rapide d'appliquer des optimisations locales sur les benchmarks. De plus, les benchmarks contiennent de nombreux noyaux de calcul redondants. Certaines opérations, bien que mesurées plusieurs fois, n'apportent pas d'informations supplémentaires sur le système à étudier. En détectant les similarités entre eux et en éliminant les noyaux redondants, on diminue le coût des benchmarks sans perte

d'information. Cette thèse propose une méthode permettant de décomposer automatiquement une application en un ensemble de noyaux de performance, que nous appellons codelets. La méthode proposée permet de rejouer les codelets, de manière isolée, dans différentes conditions expérimentales pour pouvoir étalonner leur performance. Cette thèse étudie dans quelle mesure la décomposition en noyaux permet de diminuer le coût du processus de benchmarking et d'optimisation. Elle évalue aussi l'avantage d'optimisations locales par rapport à une approche globale. De nombreux travaux ont été réalisés afin d'améliorer le processus de benchmarking. Dans ce domaine, on remarquera l'utilisation de techniques d'apprentissage machine ou d' échantillonnage. L'idée de décomposer les benchmarks en morceaux indépendants n'est pas nouvelle. Ce concept a été appliqué avec succès sur les programmes séquentiels et nous le portons à maturité sur les programmes parallèles.
Évaluer des nouvelles micro-architectures ou la scalabilité est 25x fois plus rapide avec des codelets que avec des exécutions d'applications.
Les codelets prédisent le temps d'exécution avec une précision de 94x et permettent de trouver des optimisations locales jusqu'à 1.06x fois plus efficaces que la meilleur approche globale.

Abstract :
In high performance computing, benchmarks evaluate architectures, compilers and optimizations. Standard benchmarks are mostly issued from the industrial world and may have a very long execution time. So, evaluating a new architecture or an optimization is costly. Most of the benchmarks are composed of independent kernels. Usually, users are only interested by a small subset of these kernels. To get faster and easier local optimizations, we could find ways to extract kernels as standalone executables. Also, benchmarks have redundant computational kernels. Some calculations do not bring new informations about the system that we want to study, despite that we measure them many times. By detecting similar operations and removing redundant kernels, we can reduce the benchmarking cost without loosing information about the system. This thesis proposes a method to automatically decompose applications into small kernels called codelets. Each codelet is a standalone executable that can be replayed in different execution contexts to evaluate them. This thesis quantifies how much the decomposition method accelerates optimization and benchmarking processes. It also quantify the benefits of local optimizations over global optimizations. There are many related works which aim to enhance the benchmarking process. We particularly note machine learning approaches and sampling techniques. Decomposing applications into independent pieces is not a new idea. It has been successfully applied on sequential codes and we extend it to parallel programs.

Evaluating scalability or new micro-architectures is 25x faster with codelets than with full application executions. Codelets predict the execution time with an accuracy of 94x and find local optimizations that outperform the best global optimization up to 1.06x.

INFORMATIONS COMPLÉMENTAIRES

M. William JALBY, Professeur des universités, Université de Versailles Saint-Quentin-en-Yvelines - Laboratoire LI-PaRAD - Directeur de these
M. Michael O'BOYLE, Professeur des universités, Université d'Edimbourg (Royaume-Uni) - Rapporteur
M. François BODIN, Professeur des universités, Institut de Recherche en Informatique et Systèmes Aléatoires - Rapporteur
Mme Christine EISENBEIS, Directeur de recherche, Inria et Université Paris Sud 11 - Examinateur
Mme Alexandra JIMBOREAN, Professeur associé, Université d'Uppsala (Suède) - Examinateur
M. Pablo DE OLIVEIRA CASTRO, Maître de conférences, Université de Versailles Saint-Quentin-en-Yvelines - Laboratoire LI-PaRAD - Co-encadrant de these

Contact : dredval service FED : theses@uvsq.fr