

# Parallélisation de la méthode des multipôles rapide en mémoire distribuée à l'aide d'un moteur d'exécution

Encadrant s: E. Agullo, O. Coulaud, L. Stanisic

Emails : emmanuel.agullo@inria.fr, olivier.coulaud@inria.fr

Lieu : équipe HiePACS - Inria Bordeaux Sud-Ouest

## Présentation du sujet :

Les logiciels HPC ont longtemps été parallélisés avec des paradigmes « fork-join » (tels qu' « OpenMP for ») ou avec des échanges de messages (MPI). Avec l'arrivée des machines manycores composées d'unités de calcul hétérogènes (CPU et accélérateurs (GPU, MIC)) ces paradigmes ne sont pas adaptés pour utiliser pleinement et efficacement toutes les ressources. La programmation à base de tâches essaie de palier à ces limites. Ce paradigme est une façon d'abstraire la couche matérielle pour déléguer la gestion bas-niveau de la parallélisation à un moteur d'exécution qui dispatchera les tâches sur les unités de calcul disponibles sur la machine. Il existe de nombreux outils qui implémentent cette approche (StarPU, ParSEC, ...) La bibliothèque ScalFMM [1] implémentant une méthode des multipôles rapide (FMM) haute-performance a été portée sur StarPU pour exploiter efficacement dans un premier temps un nœud de calcul multicœur [2] éventuellement accéléré par des GPUs [3] et étendu à un cluster de multicœurs [4].

## Travail

Le code actuel a été testé jusqu'à 30 nœuds soit environ 960 cœurs en dupliquant l'octree. L'objectif du stage est de passer à l'échelle sur des dizaines de milliers de cœurs. Pour cela il faudra développer des structures données scalable pour éviter notamment que l'octree soit dupliqué. Une étude de performance à grande échelle sera effectuée pour valider les développements effectués. Après s'être familiarisé avec la programmation en tâches et l'algorithme de la FMM, le stagiaire proposera une construction en parallèle de l'octree extension parallèle distribuée. L'implémentation sera réalisée en C++11 et devra respecter une architecture logicielle modulaire et maintenable. L'étude de performance sera réalisée sur la plate-forme PlaFRIM (2000 cœurs) et la machine Occigen du Cines (50544 cœurs).

**Mots-clés :** FMM, programmation en tâche, MPI, C++11, moteur d'exécution

Compétences requises : parallélisme, algorithmique, MPI, OpenMP, C++

**Commentaires :** Le stage se déroulera dans l'équipe-projet HiePACS au sein du centre Inria Bordeaux – Sud-ouest.

## Références :

[1] ScalFMM, <http://scalfmm-public.gforge.inria.fr>

[2] Task-based Fmm for Multicore Architectures, Emmanuel Agullo, Berenger Bramas, Olivier Coulaud, Eric Darve, Matthias Messner, Toru Takahashi, SIAM Journal on Scientific Computing. [pdf](#)

[3] Emmanuel Agullo, Berenger Bramas, Olivier Coulaud, Eric Darve, Matthias Messner, Toru Takahashi ; Task-based FMM for heterogeneous architectures. Concurrency and Computation: Practice and Experience, 2016, 28 (9),

[4] Emmanuel Agullo, Bérenger Bramas, Olivier Coulaud, Martin Khannouz, Luka Stanisic. Task-based fast multipole method for clusters of multicore processors. RR-8970, Inria Bordeaux Sud-Ouest. 2016, pp.15. [pdf](#)