Implémentation par tâches utilisant le support d'exécution StarPU d'un schéma de volumes finis pour les systèmes hyperboliques de lois de conservation.

Jonathan Jung, Université de Pau et des Pays de l'Adour & Cagire, Inria Bordeaux Sud-Ouest

Mohamed Essadki, CentraleSuplec, Châtenay-Malabry & IFP nergies nouvelles, Rueil-Malmaison

Adam Larat, CentraleSuplec, Châtenay-Malabry & IFP nergies nouvelles, Rueil-Malmaison

Milan Pelletier, CentraleSupélec, Châtenay-Malabry

Vincent Perrier, Cagire, Inria Bordeaux Sud-Ouest & LMAP, UPPA

Nous nous intéressons à la résolution numérique de systèmes hyperboliques de lois de conservation du type

$$\partial_t W + \nabla \cdot \mathbf{F}(W) = 0, \tag{1}$$

où W est le vecteur de variables conservatives et $\mathbf{F}$ est le flux. Le système (1) est discrétisé par une méthode de volumes finis sur maillage cartésien. Du fait des phénomènes physiques multi-échelles (écoulement à bas nombre de Mach, écoulement turbulent, ...), il est nécessaire de résoudre très précisement le système (1). Pour cela, nous souhaitons obtenir un code précis, robuste et massivement parallèle.
Si la nature exacte des futurs supercalculateurs n'est pas encore claire, il est fortement possible qu'elle soit reliée aux architectures hétérogènes, telles que les processeurs graphiques (Graphics Processing Unit) ou les architectures multiprocesseurs Intel Xeon-Phi. Adapter les algorithmes pour différentes architectures est possible [2], mais risque d'être très dépendant du code et de l'architecture. Afin de contourner le problème de dépendances liées aux architectures, les supports d'exécution (runtime) ont été développés [1]. La stratégie de base consiste à reformuler l'algorithme avec un graphe des tâches.
Nous avons pour cela choisi d'utiliser un support d'exécution développés au centre Inria de Bordeaux [1]. Avec l'utilisation de tels supports d'exécution nous obtenons un code qui est compatible avec les différentes architectures indépendamment de leur hétérogénéité.
Je présenterai tout d'abord la manière dont l'algorithme a été décomposé en tâches pouvant être exécutées sur CPU ou sur GPU puis les premiers résultats de performance obtenus sur la plateforme PlaFRIM.
Ces travaux ont été initiés lors du projet Cemrac's 2016 Hodin.

Références

[1] C. Augonnet, S. Thibault, R. Namyst, and P.-A. Wacrenier, *Starpu : A unified platform for task scheduling on heterogeneous multicore architectures*, In Euro-Par 2009 Parallel Processing, pages 863874. Springer, 2009.

[2] A. Klckner, T. Warburton, J. Bridge, and J. S. Hesthaven, *Nodal discontinuous Galerkin methods on graphics processors.* Journal of Computational Physics, 228(21) :78637882, 2009.

Jonathan Jung, LMAP UMR 5142, UPPA & Cagire Team, INRIA Bordeaux Sud-Ouest
jonathan.jung@inria.fr