

Séminaire

Parallel Evolutionary Computation for Data Mining on GPGPU

8h30-9h – Accueil et Présentation des intervenants

9h-10h – Michael Krajecki – Université de Reims – Crestic – « Programmation multiGPU – OpenMP versus MPI »

Les GPU (Graphics Processing Units : processeurs graphiques), originellement dédiés à l'accélération de traitements graphiques, ont une structure hautement parallèle. Cela a permis d'envisager de les utiliser comme des accélérateurs matériels, afin d'externaliser certains traitements numériques non graphiques hors des processeurs (CPU, Central Processing Units). Les innovations matérielles et de langage de programmation ont alors permis le développement des domaines du GPGPU puis du GPU computing. L'utilisation de plusieurs GPU, suite logique, est nécessaire pour résoudre des problèmes plus exigeants en temps et en puissance de calcul. Pour considérer les multiGPU d'une manière générale, nous présentons une approche où chaque carte graphique est dirigée par un processus MPI ou un thread OpenMP. Nous comparons les deux schémas en termes de performance et de complexité de développement, en précisant les particularités d'implémentation ainsi que le maillage du code. Il apparaît que l'utilisation d'OpenMP donne les meilleurs résultats, et que l'utilisation de « pinned memory » permet d'améliorer les temps d'exécution. La continuation naturelle de ce travail est de considérer un environnement multiGPU à trois niveaux : communications entre les nœuds (processus lourds, mémoire distribuée) ; gestion des GPU à l'intérieur des nœuds (processus légers, mémoire partagée) ; calcul au sein des cartes GPU.

10h-11h – Herve Deleau – Université de Reims – Crestic – « Max-Min Ant System on Graphics Processing Units »

Ant Colony Optimization (ACO) was introduced in 1991 by Dorigo to solve the well-known Traveling Salesman Problem (TSP). This metaheuristic may require long computation times for large problems but offers an interesting potential for parallelization that can be exploited by computing its tour construction phase on many processing elements. Typical implementations following this scheme are CPU-based. However, with the recent availability of Graphics Processing Units (GPU) for high-performance computing, it becomes interesting to use the massively parallel architecture they provide to accelerate the search process of metaheuristics. Among the rare available literature in this field we find the works of Wong and Wong on Genetic Algorithms and Janiak et al. on Tabu Search. The purpose of this paper is to propose a parallel algorithmic approach for an ACO algorithm, the Max-Min Ant System (MMAS), that is suitable for implementation in a GPU computing environment, as well as to compare the solutions and speedups obtained with an Ant System (AS) implementation in a similar environment.

11h-12h – Hélène Renard – Université de Nice – I3S – « Répartition de charge, propagation de la chaleur et les stratégies de transition »

Nous nous sommes intéressés à la redistribution de données sur ces mêmes plates-formes, lorsque que les caractéristiques de ces dernières changent. Sous l'hypothèse de redistribution légère, nous sommes capable de résoudre le problème de manière optimale. Nous nous proposons donc de confronter nos algorithmes à une application réelle : l'équation de la chaleur. Cette étude a pour objectif de comparer le comportement de nos algorithmes sur de deux plates-formes différentes : la première est simulée - SimGrid - tandis que la seconde - Grid'5000 - est réelle. Nous en verrons les bienfaits ! Par la suite, nous nous sommes intéressée aux mises-à-jour locales avec ou sans redistribution de données sur des grappes hétérogènes, toujours sur SimGrid et sur Grid'5000. Ultérieurement, nous nous intéresserons aux variations de performances globales sur plates-formes hétérogènes, là encore sur SimGrid et sur Grid'5000. Il serait intéressant de simuler ces différentes approches afin de les comparer entre elles. Une possibilité serait d'écrire ces différentes méthodes et, suivant le cas dans lequel nous nous trouvons au moment de la prise de décision (par exemple, une variation de performances importante s'est produite), choisir la meilleure en fonction de son coût et du nombre d'itérations restantes.

14h-15h – Denis Pallez – Université de Nice – I3S – « Algorithmes évolutionnaires (interactifs) et Data Mining »

Dans cette présentation, nous aborderons dans un premier temps les algorithmes évolutionnaires qui représentent des techniques d'optimisation stochastiques et en particulier l'évolution différentielle proposée par Storn & Price en 1997. Ils permettent d'aborder des problèmes d'optimisation très irréguliers et difficiles, souvent hors de portée des méthodes d'optimisation classiques. Comme aucune propriété de continuité ou de régularité n'est imposée aux quantités optimisées, il est très facile de prendre en compte des jugements subjectifs issus d'interactions avec un ou plusieurs utilisateurs ou experts (évolution interactive). Nous présenterons ensuite comment ces techniques d'optimisation peuvent être combinées avec les techniques de fouille de données afin soit d'améliorer la convergence des méthodes d'optimisation soit d'améliorer la pertinence des données extraites par les techniques de fouille.

15h-16h – Andréa Tettamanzi – Université de Milan – Département de Technologies de l'Information – « Extraction de connaissances avec algorithmes évolutionnaires et logique floue »

L'extraction automatique de connaissance à partir de volumes importants de données fait l'objet d'un grand intérêt à niveau scientifique et applicatif. Ce séminaire présente des techniques mises au point par l'intervenant et ses collaborateurs pour l'extraction de modèles basés sur des règles floues à l'aide d'algorithmes évolutionnaires, ainsi que des exemples d'applications dans des domaines divers. La logique floue est une extension de la logique classique qui admet un continu de degrés de vérité. Elle permet d'exprimer des modèles basés sur des règles linguistiques qui sont capables de décrire des relations non linéaires entre les variables et dont l'interprétation se révèle particulièrement intuitive pour l'expert humain. Les algorithmes évolutionnaires sont des méthodes stochastiques d'optimisation inspirées par la théorie de l'évolution de Darwin.