

2EL1550 - Calcul haute performance

Responsables : **Stephane VIALLE**

Langues d'enseignement : **FRANCAIS**

Type de cours : **Electif 2A**

Campus où le cours est proposé : **CAMPUS DE PARIS - SACLAY**

Nombre d'heures d'études élèves (HEE) : **60**

Nombre d'heures présentielles d'enseignement (HPE) : **30**

Année académique : **2024-2025**

Catégorie d'électif : **Sciences fondamentales**

Niveau avancé : **oui**

Présentation, objectifs généraux du cours :

Ce cours permettra aux étudiants de :

- Comprendre les enjeux et les difficultés de la simulation numérique intensive dans tous les domaines de la recherche et du développement.
- Appréhender l'algorithmique parallèle et maîtriser les étapes de la parallélisation d'un code de calcul.
- Connaître et avoir expérimenté des bibliothèques et environnements de calcul parallèle.

Période(s) du cours (n° de séquence ou hors séquence) :

SG6

Prérequis :

- Cours commun « Systèmes d'Information et Programmation » de SG1 (1CC1000)
- Cours commun « Algorithmique et complexité » de ST2 (1CC2000)
- Ainsi que des connaissances de bases en algèbre linéaire

Plan détaillé du cours (contenu) :

- **Architectures parallèles et distribuées** : composants d'un supercalculateur ; hiérarchie de mémoires ; aspects énergétiques ; besoin de tolérance aux pannes.
- **Optimisation et parallélisation de boucles en mémoire partagée** : optimisation sérielles et vectorisation, algorithmique et programmation multithreads avec OpenMP, analyse et réécriture de boucles.
- **Algorithmique distribuée par envoi de messages** : circulation de données et communications point-à-point en MPI (mpi4py) ; déploiement d'application et exécution distribuée en MPI (OpenMPI + mpi4py) ; distribution de données et communications collectives en MPI (mpi4py) ; algèbre linéaire, méthodes directes et méthodes itératives.
- **Calcul scientifique parallèle** : stratégies pour la résolution de problèmes linéaires de grande taille ; méthodes itératives de sous-structurations ; méthodes de décomposition de domaines.

- **Mesure et analyse de performances** : méthodologie de mesure ; métriques et limites d'accélération et d'efficacité ; métriques et limites de passage à l'échelle.

Déroulement, organisation du cours :

Les approches mathématiques et les algorithmes présentées en cours seront mis en oeuvre lors de TD sur clusters de calculs, et des performances seront mesurées et analysées lors de chaque expérimentation. L'expérimentation sera une part importante du cours, et permettra de bien appréhender les concepts étudiés.

- **Composition du cours en présentiel** : 18h00 de CM, 1h30 de TD, 9h00 de TD sur machines (3 TP), et 2h00 d'examen écrit final
- **Composition du cours en autonomie** : lecture de polycopié, exercices de programmation multithread, TP noté de calcul distribué.
- **Séquencement possible du cours** :
 - 5 x 1h30 de CM ;
 - Autonomie : exercices de parallélisation de boucles et de programmation multithreads en OpenMP
 - 2 x 1h30 de TD sur machines (TP) ;
 - 3 x 1h30 de CM
 - 1x1h30 de TD de programmation distribuée
 - Autonomie : approfondissement du cours de MPI à partir d'un polycopié et de l'analyse numérique à partir d'un livre (distribués aux élèves)
 - 2 x 1h30 de TD sur machines (TP) ;
 - Autonomie : TP noté de programmation distribuée sur clusters de PC, avec enseignants joignables à certaines heures (office hours)
 - 4 x 1h30 de CM
 - Autonomie : exercices sur les méthodes de résolution
 - 2 x 1h30 de TD sur machines (TP) ;
 - 2h00 d'EE final
- **Organisation des TD en présentiels (36% du volume total en présentiel)** :
 - 1 TD + 6 TD sur machines qui seront groupés par 2 (i.e. par blocs de 3h),
 - les codes développés seront exécutés sur les clusters de calculs du Data Center d'Enseignement de CentraleSupélec, accessibles à partir de salles informatiques, ou à partir des ordinateurs des élèves.

Organisation de l'évaluation :

Poids relatifs des différents examens:

- 50% : Comptes rendus de TDs sur machines
- 50% : Examen écrit de 2h00 (en fin de cours)
- En cas d'absence justifiée à l'un des TDs sur machines, la note de ce dernier est remplacée par celle du contrôle final. En cas d'absence injustifiée une note de 0/20 sera appliquée pour ce TD sur machines.
- L'examen de rattrapage consistera entièrement en un examen écrit, similaire à l'examen initial.

Moyens :

- **Equipe pédagogique** : **Frédéric MAGOULES** et **Stéphane VIALLE**
- 64% de cours et 36% de TD, avec des groupes de TD de 25 étudiants travaillant sur des machines à haute performance.
- Accès à différents serveurs et clusters de calculs (Data Center d'Enseignement de CentraleSupélec, et/ou mésocentre CentraleSupélec-ENS Paris Saclay).
- Expérimentation avec des logiciels Open-source standards : C/C++/Python, bibliothèque de

multithreading pour multicoeurs (OpenMP), bibliothèque d'envois de messages pour clusters de calculs (MPI: MPICH2/OpenMPI), bibliothèque scientifiques optimisées (OpenBLAS).

Acquis d'apprentissage visés dans le cours :

A l'issue du cours les étudiants seront capables de :

- **[Acquis d'Apprentissage 1* (AA1*)]** contribuant aux compétences **C1 C2 C6** :
 - adapter des méthodes numériques pour le calcul à haute performance (HPC)
 - concevoir des méthodes numériques pour le calcul à haute performance (HPC) afin de résoudre des problèmes complexes
- **[Acquis d'Apprentissage 2* (AA2*)]** contribuant aux compétences **C1 C2 C6** :
 - concevoir des algorithmes parallèles de simulation intensive selon les règles du calcul à haute performance (HPC)
 - implanter des algorithmes parallèles de simulation intensive sur supercalculateurs ou sur clusters de PC multi-coeurs
 - gérer un quota d'heures de calculs lors de l'exécution d'une simulation intensive
- **[Acquis d'Apprentissage 3* (AA3*)]** contribuant aux compétences **C1 C2 C6** :
 - mettre en œuvre une simulation à haute performance complète et cohérente :
 - choisir des modèles adaptés en précision et extensibles (pour un passage à l'échelle)
 - choisir des stratégies d'implantations parallèles et efficaces
 - réaliser une campagne de simulation en temps et ressources limitées

Description des compétences acquises à l'issue du cours :

- **C1:** Analyze, design and build complex systems with scientific, technological, human and economic components
- **C2:** Develop an in-depth skills in an engineering field and in a family of professions
- **C6:** Be operational, responsible, and innovative in the digital world

Bibliographie :

Supports de cours fournis aux étudiants :

- Frédéric Magoulès, François-Xavier Roux, Guillaume Houzeaux. Parallel Scientific Computing. Wiley & Sons, Inc., 2015. Hardcover 354 pages (in English). This course support is available in other languages: in French (Dunod, 2017), in Spanish (CIMNE, 2014), in Japanese (Morikita Publishing Co Ltd, 2015), in Hungarian (Pollack Press, 2018).
- Numerical Methods: Slides of the lectures
- Parallel and Distributed Computing: Slides of the lectures.

Autres livres suggérés :

- W. Gropp, E. Lusk, A. Skjellum. "Using MPI". MIT Press. 1999.
- R. Chandra, R. Menon, L. Dagum, D. Kohr, D. Maydan, J. McDonald. "Parallel Programming in OpenMP". Morgan Kaufmann Publishers. 2000.
- B. Chapman, G. Jost, R. Van Der Pas.. "Using OpenMP". MIT Press. 2007.