

Annexe

Contenu du stage

Cette formation se décompose en deux modules.

Module 1: Analyse de performance, optimisation de code et outils de développement
(Françoise Roch, Laurence Viry - 2 journées)

L'exposé s'appuie sur des principes généraux facilement applicables à différentes architectures matérielles.

La présentation se décompose principalement en cinq parties:

Principes généraux d'optimisation

- Rappels sur l'architecture des processeurs
- Méthodologie d'optimisation
- Software pipelining
- Analyse Interprocédurale
- Inlining
- Optimisation des accès mémoire
- Transformation de boucle
- Transformation conforme avec la norme IEEE 754
- Utilisation de bibliothèques optimisées (Blas, Lapack,...)
- ...

Optimisations effectuées par le compilateur - Applications à différents compilateurs: Intel, Compaq, IBM, GNU...

Analyse de performance

- Principes généraux et applications
- Outils: prog, gprof, pixie, uprofile...
- Autres outils (PAPI, Vampir,...)

Outils d'aide au développement

- utilisation de make
- débogueur: dbx, totalview,...
- cvs: outil de développement collaboratif
- ...

Langages de programmation

- Fortran 95

- Concepts objets: Fortran90 / C++
- Exemple d'application scientifique utilisant les concepts objets
- Utilisation de bibliothèques: blas, lapack, nag, imsl, bibliothèques constructeur...

Module 2: Parallélisme

Généralités (Laurence Viry - 1/2 journée)

- Introduction Générale au parallélisme
 - Terminologie et concepts de base
 - Evaluation d'un programme parallèle (efficacité, speedup, scalabilité, Loi d'Amdhal ,...)
- Taxonomie des architectures parallèles et modèle de programmation
 - Flynn's Hardware Taxonomy
 - Système à mémoire distribuée
 - Système à mémoire partagée
 - réseau de communication (bus, switch...)
 - Mémoire hiérarchique et calcul parallèle
- Modèle de programmation parallèle (MIMD, SIMD,...)
- Programmation en mémoire distribuée (échange de messages - MPI)
- Programmation en mémoire partagée (directives de compilation - OpenMP)
- Support des langages au parallélisme
- Bibliothèques mathématiques parallèles

Programmation parallèle par échange de messages - MPI (Céline Robert, Laurence Viry - 1j 1/2)

- Concepts de base - Environnement
- Communications point à point bloquantes ou non bloquantes
- Communications collectives
- Communicateurs
- Topologies de processus
- Types dérivés
- Apports MPI-2
 - gestion dynamique des processus
 - communications de mémoire à mémoire
 - interfaçage avec C++ et Fortran 95.
 - Entrées-sorties parallèles : MPI-IO
 - Extensions concernant les intracommunicateurs et intercommunica-teurs
- SCALAPACK (Scalable LAPACK) modèle MIMD

Programmation parallèle par directives de compilation: OpenMP (Françoise Roch, Françoise Berthoud - 1 journée)

- Modèle d'exécution
- Structure d'OpenMP
- Partage des données
- Partage du travail
- Ordonnancement
- Synchronisation

Besoins spécifiques et généraux des personnels techniques

L'évolution des matériels informatiques, des logiciels (augmentation considérable de la puissance de calcul et de la taille des problèmes pouvant être traités localement) ainsi que la multiplicité des domaines d'application (le calcul scientifique est "sorti" des domaines physique/mathématique pour les domaines liés à la biologie ou les besoins sont "explosifs") font apparaître la nécessité de former très rapidement les personnels techniques à ces outils afin que les chercheurs et étudiants puissent en bénéficier le plus vite possible.

Pour les techniciens et ingénieurs systèmes/réseau

- une meilleure appréhension des besoins des utilisateurs,
- une connaissance générale des techniques et des outils utilisés dans le développement des applications de calcul scientifique.

Pour les techniciens et ingénieurs participant au développement d'application

- le partage d'expérience sur les nouveaux outils de développement,
- une utilisation optimale des outils à disposition,
- l'acquisition de nouvelles techniques de développement (parallélisme, langage objets,...)