



Doctorado en Ingeniería 2018

FORMULARIO DE PRESENTACIÓN DE CURSO

Cuatrimestre que se dicta:

1er Cuatrimestre 2do Cuatrimestre

Denominación del curso:

Computación de alto rendimiento en mecánica computacional. MPI, PETSC y OPEN MP.

Objetivos del curso:

El objetivo del curso es desarrollar las aptitudes del alumno en cuanto a la programación en C++ con los estándares/librerías MPI, PETSc, OpenMP, y CUDA los cuales son actualmente una combinación efectiva para la resolución de grandes problemas de mecánica computacional en cluster de procesadores multi-core. La orientación del curso no es hacia el aprendizaje general de la computación en paralelo sino al uso intensivo de estas tecnologías.

Programa analítico:

- Conceptos básicos de MPI. Es MPI pequeño o grande? Uso de MPI en programas simples. Ejemplos en Fortran y C.. Tomando tiempos de programas.. Un ejemplo con self-scheduling. I/O en MPI. Estrategia master/slave en SPMD. Formato de llamadas, códigos de error. Funciones básicas.
- Comunicación punto a punto. Envelope de un mensaje, condiciones de recepción. Dead-lock. La función send-recv. Comunicaciones no-bloqueantes. Ejemplos: Cálculo del ancho de banda de una red. Cálculo del ancho de banda de disección de un cluster. Implementación de broadcast lineal y en árbol con comunicación punto a punto.
- Comunicación colectivas. Broadcast. Reducción global. Operaciones asociativas. Definición de nuevas operaciones asociativas. Allreduce. Tiempos de comunicación y sincronización. Escalabilidad. Ejemplo:Cálculo de PI por integración. Llamadas colectivas.
- MPI en ambientes Unix. MPICH. Utilitarios. Ejemplo: Prime Number Theorem. Estrategia compute-on-demand. MPE logging. Escalabilidad. Usod e jumpshot. Rendimiento en clusters heterogéneos. Paralelismo trivial. Ejemplo: El Problema del Agente Viajero(TSP). Adaptación del algoritmo dinámico. Ejemplo:Cálculo de PI por Montecarlo. Uso de communicators. Ejemplo: producto de matrices en paralelo. Algoritmos estático y dinámico. Escalabilidad. Ejemplo: El Juego de la Vida de Conway. Automatas celulares. Algoritmo estático y dinámico. Comunicación encadenada. Escalabilidad. Comunicación no-bloqueante. Ejemplo: El problema de Poisson. Topologías virtuales. Escalabilidad
- Operaciones colectivas avanzadas de MPI. Scatter, gather y all-gather. Versiones vectorizadas. Ejemplo: la función print-par (gather and print). Ejemplo: La función rescatter. Definiendo tipos de datos derivados. Ejemplo: rotar columnas de matrices. Ejemplo: El método iterativo de Richardson.
- PETSc. La librería PETSC de álgebra lineal en paralelo. Objetos PETSc. Estructura de la librería PETSc. Usando PETSc. Escribiendo programas que usan PETSc. Ejemplo simple. Ec. de Laplace 1D. Headers. Bases de datos/Opciones.
- Vectores y datos distribuidos. Creando vectores. Operaciones básicas sobre vectores. Indiciación. Operaciones de scatter y gather. Matrices. Matrices sparse. Matrices densas. Operaciones básicas sobre matrices. Operaciones `matrix-free`.
- SLES: Solvers lineales . Resolviendo secuencias de



sistemas lineales.. Familia de métodos de Krylov . Precondicionamiento. Tests de convergencia. Monitoreo de la convergencia. Espectro del operador. • OpenMP. Parte 1. Conceptos básicos de OpenMP. Memoria compartida. Datos privados y públicos. Race conditions. El estándar de OpenMP. Threads. Regiones paralelas. Pragmas. Cláusulas. • OpenMP. Parte 2: directivas, variables de entorno, entorno en tiempo de ejecución. Compilación condicional. Lazos paralelos. Sincronización. Cláusulas private, shared, first/last private, reduction. Directivas de trabajo compartido: parallel. • OpenMP. Parte 3. Balance de carga: la cláusula `schedule'. Secciones. Directivas huérfanas de trabajo compartido. Sincronización con barreras. Regiones críticas. Directivas single and master. Otras directivas de sincronización. Ejemplo: Productos de muchas matrices. Ejemplo: Producto matriz vector. • OpenMP. Parte 4. Variables de entorno. Funciones de la librería en tiempo de ejecución. Rutinas utilitarias de información y ajuste de parámetros. Rutinas para el manejo de semáforos. Semáforos anidados. Ejemplo: simulando scheduling dinámico con semáforos.

Bibliografía:

• Storti, M., Apuntes del curso. <https://goo.gl/uE5vmz> • Ruud van der Pas, Senior Staff Engineer, Scalable Systems Group, Sun Microsystems, An Introduction Into OpenMP, IWOMP 2005, University of Oregon, Eugene, Oregon, USA, June 1-4, 2005 • NVIDIA. NVIDIA CUDA, Programming Guide 3.0 • M. Papadrakakis (ed.), Parallel Solution Methods in Computational Mechanics , J.Wiley and Sons, Chichester, 1997. • T.L. Sterling, J. Salmon, D.J. Becker and D.F. Savarese, How to build a Beowulf, The MIT Press, Cambridge, 1999. • M.A. Storti, PETSc-FEM: A General Purpose, Parallel, Multi-Physics FEM Program, <http://www.cimec.org.ar/petscfem> • W. Gropp, E. Lusk and A. Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, Cambridge University Press (2003) • G. Karniadakis, R.M. Kirby, Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and their Implementation, Cambridge University Press; ISBN-10: 0521520800, ISBN-13: 978-0521520805. • M. Papadrakakis, Solving Large-Scale Problems in Mechanics, J.Wiley and Sons, 1993. • Satish Balay, Kris Buschelman, Victor Eijkhout, William Gropp, Dinesh Kaushik, Matt Knepley, Lois Curfman McInnes, Barry Smith, Hong Zhang, PETSc, Portable, Extensible Toolkit for Scientific Computation, User manual

Docente responsable:

Mario Storti

Docentes corresponsables:

Victorio Sonzogni, Jorge D'elia

Docentes colaboradores:

Conocimientos previos requeridos:

Álgebra lineal. Conocimiento mediano de programación en C++.

Carga horaria:

- Teoría: 60hs
- Coloquio y/o Práctica en el aula, laboratorio o campo: 15Hs
- Total: 75hs



Instancias de evaluación:

Guías de trabajos prácticos individuales con fecha de entrega Trabajos prácticos de laboratorio a realizar en clase con supervisión

Requisitos de aprobación del curso:

Tres trabajos prácticos de laboratorio (TPLs) Entrega y aprobación de 6 Guías de Trabajos Prácticos.

Cupo mínimo:

20

Cupo máximo:

0

Fecha inicio:

06-04-2018

Duración:

15 semanas

Horarios de dictado:

	Lugar	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Teoría						0900-1300	
Práctica						1430-1530	

Infraestructura y equipamiento necesarios:

Se necesita: Pizarra, cañón, conexión a Internet por SSH. Los alumnos deben contar con computadoras personales, preferiblemente con SO GNU/Linux para poder acceder al cluster Aquiles del CIMEC.

Otros:

Lugar y fecha: Santa Fe, 23 de febrero de 2018