



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

SISTEMAS PARALELOS

Carrera: *Licenciatura en Informática*
Plan 2003-2007
Licenciatura en Sistemas
Plan 2003-2007

Año 2010

Año:
Duración: *Semestral*
Profesor: *Ing. Armando De Giusti*
Hs semanales: 6 hs

OBJETIVOS GENERALES:

Caracterizar los problemas de procesamiento paralelo desde dos puntos de vista: la arquitectura física y los lenguajes de programación, poniendo énfasis en la transformación de algoritmos secuenciales en paralelos.
Describir los modelos de cómputo paralelo y los paradigmas de programación paralela.
Estudiar las métricas de performance asociadas al paralelismo, así como modelos de predicción de performance orientados a diferentes arquitecturas multiprocesador.
Plantear casos concretos de procesamiento paralelo, resolubles sobre distintas arquitecturas multiprocesador.

CONTENIDOS MINIMOS:

- Arquitecturas de procesamiento paralelo.
- Modelos de comunicación. Métricas de performance.
- Memoria compartida, Memoria distribuida, esquemas mixtos.
- Lenguajes y sistemas operativos para procesamiento paralelo.
- Paradigmas de resolución de sistemas paralelos.
- Adaptación entre arquitectura y software.
- Aplicaciones.

Programa

Unidad 1: Conceptos básicos

Paralelismo. Objetivos del procesamiento paralelo.
Proceso y Procesador. Interacción, comunicación y sincronización de procesos.
Concurrencia y Paralelismo. Modelos de Concurrencia.
Impacto del procesamiento paralelo sobre los sistemas operativos y lenguajes de programación.
Concepto de Sistema Paralelo.
Speedup y Eficiencia de algoritmos paralelos.
Concepto de asignación de tareas y balance de carga.
Balance de carga estático y dinámico.



Unidad 2: Arquitecturas orientadas a Procesamiento Paralelo

Paralelismo implícito: tendencias en las arquitecturas de microprocesadores.
Optimización de performance en los sistemas de memoria. Manejo de memoria cache.
Estructura de control y modelos de comunicaciones en plataformas de procesamiento paralelo.
Clasificación por mecanismo de control (SISD. SIMD. MISD. MIMD), por la organización del espacio de direcciones, por la granularidad de los procesadores y por la red de Interconexión.
Análisis del impacto del tiempo de comunicación en el speedup alcanzable.
Vector processors, array processors, Arquitecturas cúbicas e hipercúbicas. Supercomputadoras.
Conceptos de procesadores dedicados a la aplicación, tipo DSP.
Clusters de PCs. Multiclusters. Grids.

Unidad 3: Principios de diseño de algoritmos paralelos

Técnicas de descomposición.
Características de los procesos. Interacción.
Técnicas de mapeo de procesos/procesadores. Balance de carga.
Métodos para minimizar el overhead de la interacción entre procesos.
Modelos de algoritmos paralelos.
Problemas paralelizables y no paralelizables.
Paralelismo perfecto. Paralelismo de datos. Paralelismo de control. Paralelismo mixto.

Unidad 4: Operaciones de Comunicación en arquitecturas multiprocesador

One-to-all Broadcast y All-to-one Reduction.
All-to-all Broadcast y Reduction.
Operaciones de All-Reduce y Prefix-Sum.
Comunicaciones de Scatter and Gather.
Comunicaciones All-to-All personalizadas.
Shift circular.
Optimización en las comunicaciones en esquemas multiprocesador.

Unidad 5: Modelos y Paradigmas de Computación Paralela

Parallel Random Access Machine (PRAM)
Bulk Synchronous Parallel (BSP)
LogP. Otras variantes de modelos analíticos.
Paradigma Master/Slave.
Paradigma Divide/Conquer.
Paradigma de Pipelining.
Metodología de diseño de algoritmos paralelos.

Unidad 6: Métricas del paralelismo

Medidas de performance standard.
Fuentes de overhead en procesamiento paralelo.
Speedup. Rango de valores. Speedup superlineal.
Eficiencia. Rango de valores. Grado de paralelismo alcanzable.
Efecto de la granularidad y el mapeo de datos sobre la performance.



Cargas de trabajo y modelos de speedup. Modelo de carga fija (Amdahl). Modelo de tiempo fijo (Gustafson). Modelo de memoria limitada (Sun y Ni). Escalabilidad de sistemas paralelos. Concepto de isoeficiencia. Función de isoeficiencia.

Unidad 7: Programación de algoritmos paralelos con Pasaje de Mensajes

Principios de la comunicación/sincronización por pasaje de mensajes.
Primitivas Send y Receive. La interfaz MPI como modelo.
Cómputo y Comunicaciones
Comunicaciones colectivas y operaciones de procesamiento.
Ejemplos sobre arquitecturas multiprocesador.

Unidad 8: Programación de algoritmos paralelos sobre plataformas con memoria compartida.

Concepto de thread.
Primitivas de sincronización en PThreads.
Control de atributos en threads.
OpenMP como modelo Standard.
Análisis de problemas.

Unidad 9: Conceptos de Arquitecturas GRID.

Clusters, multiclusters y GRID.
Extensión de conceptos de Cluster-processing a Grid-processing.
Funcionalidades del Middleware en arquitecturas GRID.
Algoritmos sobre GRID.
Modelos y paradigmas de Sistemas Paralelos extendidos a GRID.
Análisis de casos de sistemas GRID.

Unidad 10: Algoritmos paralelos clásicos.

Presentación de casos clásicos:
Sorting / Algoritmos sobre grafos /Procesamiento de matrices.
Algoritmos de búsqueda para optimización discreta.
Programación dinámica.
Análisis de soluciones sobre diferentes arquitecturas paralelas.

Metodología de enseñanza

La asignatura se estructura con clases teóricas, explicaciones de práctica y prácticas experimentales.

Las clases teóricas son dictadas por los Profesores de la asignatura y no son obligatorias.

Las explicaciones de práctica son introductorias al trabajo en Laboratorio, para facilitar la utilización del equipamiento y software por los alumnos.

Las clases prácticas consisten en el desarrollo de trabajos con diferentes arquitecturas paralelas y lenguajes de programación. Los trabajos se pueden realizar individualmente o en grupo de 2 personas. Las consultas y correcciones son realizadas en los horarios de



práctica definidos, o por medio de la plataforma de Educación a Distancia de la UNLP en el resto de la semana.

Todos los trabajos son realizados en Laboratorio, sobre diferentes arquitecturas paralelas que dispone la Facultad (Laboratorio de Procesamiento Paralelo, Sala IBM, Cluster Blade).

Propuesta de evaluación

Los alumnos deben aprobar las entregas de los diferentes trabajos experimentales a realizar y su posterior coloquio. Cada trabajo cuenta con más de una entrega donde se evalúa por parte de los auxiliares y se le explica a el/los alumno/s los errores que se deben corregir.

La prueba teórica final puede ser mediante examen o un proyecto específico que el alumno resuelve, presenta una monografía sobre el tema y defiende en un coloquio en una fecha de examen final.

BIBLIOGRAFÍA:

Introduction to Parallel Computing. Grama, Gupta, Karypis, Kumar. Addison Wesley 2003
Foundations of Multithreaded, Parallel and Distributed Programming. Andrews. Addison Wesley 2000.

Parallel Programming. Wilkinson, Allen. Prentice Hall 2005.

Sourcebook of Parallel Computing. Dongarra, Foster, Fox, Gropp, Kennedy, Torczon, White. Morgan Kauffman 2003.

Parallel and Distributed Computing. A Survey of Models, Paradigms and Approaches. Leopold. Wiley, 2001.

GRID Computing: A practical guide to technology and applications. Abbas. Charles River Media 2004.

The GRID 2. Blueprint for a new computing infrastructure. Foster, Kesselman Morgan Kauffman 2004.

IEEE, ACM Digital Library

Bibliografía Adicional

Parallel Computation. Models and Methods. Akl. Prentice Hall 1997.

The Design and Analysis of Parallel Algorithms. Akl. Prentice-Hall, Inc. 1989.

Studies in Computational Science. Parallel Programming Paradigms. Brinch Hansen, P., Prentice Hall, 1995.

Parallel Program Design. A Foundation. Chandy, Misra. Addison Wesley, 1988.

Distributed and Parallel Computing. El-Rewini H., Lewis T., Manning Publications, 1998.

PVM: Parallel Virtual Machine - A Users Guide and Tutorial for Network Parallel Computing. Geist, Beguelin, Dongarra, Jiang, Mancheck, Sunderam, MIT Press, 1994.

Communicating Sequential Processes. Hoare, Englewood Cliffs, Prentice Hall, 1985

Computer Architecture and Parallel Processing. Hwang, Briggs. McGraw-Hill, Inc, 1984

Scalable Parallel Computing. Hwang, Xu, McGraw-Hill, 1998.

Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes. Leighton F., Morgan Kaufmann Publishers, Inc, San Mateo, California, 1992

Introduction to Parallel Computing. Lewis, El-Rewini, Prentice-Hall, Inc., Englewood Cliffs, 1992.



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

In Search of Clusters. Pfister , Prentice Hall, 2nd Edition, 1998.

MPI: The complete Reference. Snir, Otto, Huss-Lederman , Walker , Dongarra , Cambridge, MA: MIT Press, 1996.

Procesamiento Paralelo. Conceptos de Arquitecturas y Algoritmos. Tinetti , De Giusti . Editorial Exacta 1998.