



SISTEMAS PARALELOS

Año 2017

Carrera/ Plan:

Licenciatura en Informática Plan 2015

Licenciatura en Sistemas Plan 2015

Licenciatura en Informática Plan 2003-07/Plan 2012

Licenciatura en Sistemas Plan 2003-07/Plan 2012

Año: 4^{to}

Régimen de Cursada: Semestral

Carácter (Obligatoria/Optativa): Obligatorio (LI)
Optativa (LS)

Correlativas: Programación concurrente

Profesor/es: Franco Chichizola

Hs. semanales: 6 hs.

FUNDAMENTACIÓN

La evolución tecnológica de los procesadores ha impuesto el procesamiento paralelo. Formar al alumno (que ya tiene conocimientos previos de Concurrencia y sus aplicaciones) en los fundamentos de los sistemas paralelos, los paradigmas de programación paralela y las métricas de performance asociadas resulta un aporte fundamental para el futuro profesional.

Esta tarea de formación se combina con trabajo experimental sobre sistemas paralelos concretos, disponibles en la Facultad.

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 (de acuerdo al Plan de Estudios)

- 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.
- Arquitecturas de almacenamiento.
- Green computing.



PROGRAMA ANALÍTICO

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.
Conceptos de Green Computing.*

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.
Multicores. Clusters de PCs. Multiclusters.*

Unidad 3: Principios de diseño de algoritmos paralelos

*Metodología 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: 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.*

Unidad 5: 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 heterogeneidad.
Efecto de la granularidad y el mapeo de datos sobre la performance.
Escalabilidad de sistemas paralelos.
Métricas relacionadas con el consumo energético.*



Unidad 6: 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 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.

Combinación de memoria compartida y pasaje de mensajes.

Unidad 8: 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.

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.*

Bibliografía Complementaria

- *"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. 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.*



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

- "Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes". Leighton. Morgan Kaufmann Publishers Inc, San Mateo, California, 1992
- "Introduction to Parallel Computing". Lewis, El-Rewini. Prentice-Hall, Inc., Englewood Cliffs, 1992.
- "In Search of Clusters". Pfister. Prentice Hall, 2nd Edition, 1998.
- "MPI: The complete Reference". Snir, Otto, Huss-Lederman, Walker, Dongarra, Cambridge. MIT Press, 1996.
- "Procesamiento Paralelo. Conceptos de Arquitecturas y Algoritmos". Tinetti, De Giusti. Editorial Exacta 1998.
- "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

METODOLOGÍA DE ENSEÑANZA

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

- *Las clases teóricas son dictadas por el profesor 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. Si bien no son obligatorias, es recomendado que los alumnos asistan.*
- *Las clases prácticas consisten en el desarrollo de trabajos con diferentes arquitecturas paralelas y lenguajes de programación. De acuerdo a la cantidad de alumnos que cursen la materia, los docentes definen si los trabajos se realizar individualmente o en grupo.*

Las consultas y correcciones son realizadas en forma presencial en los horarios de práctica definidos. También se pueden hacer consultas por medio de la plataforma de Educación a Distancia Ideas en el resto de la semana.

Los trabajos realizados en Laboratorio son sobre diferentes arquitecturas paralelas que dispone la Facultad.

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 él/los alumno/s los errores que se deben corregir.

La aprobación de estos trabajos experimentales otorga la aprobación de la cursada de la asignatura.

La aprobación final de la materia puede ser mediante:

- *examen teórico/práctico en la mesa de final en que se inscriba el alumno.*
- *examen conceptual (al terminar la cursada) más proyecto específico que el alumno resuelve, presenta una monografía sobre el tema y defiende en un coloquio en una fecha de examen final.*



CRONOGRAMA DE CLASES Y EVALUACIONES

El esquema de las clases teóricas es el siguiente:

Clase	Fecha	Contenidos/Actividades
1	Semana del 13/03	Introducción al Procesamiento Paralelo. Conceptos. Fundamentos Sistemas Paralelos. Aplicaciones. Sistemas Distribuidos y Paralelos.
2	Semana del 20/03	Plataformas de procesamiento para Sistemas Paralelos. Evolución de las arquitecturas y las comunicaciones. Relación con los Sistemas Distribuidos.
3	Semana del 27/03	Conceptos de programación distribuida y paralela. Aplicaciones de los sistemas paralelos. Programación en Memoria Compartida, modelos basados en threads (Pthreads).
4	Semana del 03/04	Programación en Memoria Compartida, modelos de programación basados en directivas – OpenMP. Programación en Memoria Compartida y Rendezvous (ADA).
5	Semana del 10/04	Programación de Algoritmos Paralelos utilizando el paradigma de Pasaje de Mensajes. MPI. PVM.
6	Semana del 17/04	Métricas de Performance. Granularidad. Escalabilidad. Grado de concurrencia.
7	Semana del 24/04	Principios de diseño de Algoritmos Paralelos. Descomposición en tareas, granularidad de aplicaciones, mapeo de tareas. Técnicas de descomposición de aplicaciones. Características de las tareas y las interacciones generadas.
8	Semana del 08/05	Mapping de tareas a procesadores. Algoritmos de mapping para modelos basados en grafos. Técnicas de mapping para lograr balance de carga en modelos basados en descomposición de datos.
9	Semana del 15/05	Modelos y paradigmas de programación paralela. Aplicaciones características y formas de resolverlas.
10	Semana del 22/05	Tendencias en paralelismo: Cloud Computing; GPUs; Consumo Energético; FPGA's.

Evaluaciones previstas	Fecha
Coloquio trabajo OpenMP y Pthreads	Semana del 24/04
Coloquio trabajo MPI y ADA	Semana del 15/05
Entrega/coloquio del trabajo de la cursada	Semana del 26/06
Reentrega/coloquio del trabajo de la cursada	Semana del 31/07



UNIVERSIDAD NACIONAL DE LA PLATA
FACULTAD DE INFORMÁTICA

Contacto de la cátedra (mail, sitio WEB, plataforma virtual de gestión de cursos):

Mail: sparalelo@lidi.info.unlp.edu.ar

Web: <http://weblidi.info.unlp.edu.ar/catedras/paralela/>

Plataforma virtual: ideas.info.unlp.edu.ar

Firma del/los profesor/es