



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

## **Fundamento de Procesamiento Paralelo**

**Año 2010**

Carrera: Magister en Computo de altas  
Prestaciones y Tecnología GRID  
Prestaciones.

Año: 2010

Duración: 70 Hs.

Profesor a Cargo: **Marcelo Naiouf**  
**Marcela Printista, Laura De Giusti, M.**  
**Fabiana Píccoli**

### **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.

Analizar 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.

Analizar las extensiones del procesamiento paralelo sobre arquitecturas GRID y CLOUD.

#### Pre requisitos:

Conocimientos básicos de Concurrencia y Sistemas Operativos.

Manejo de lenguajes de expresión de algoritmos.

### **PARA LOS ALUMNOS DE MAGISTER**

- 1) Extender los temas teóricos de multicores, cluster de multicores y cloud.
- 2) Se pondrá énfasis en la investigación y optimización de algoritmos paralelos sobre arquitecturas de cluster de multicores y GPUs.

### **MODALIDAD DE EVALUACION**

Proyectos de trabajo de investigación y desarrollo individual con 3/6 meses para presentarlos, incluyendo investigación bibliográfica actualizada de los temas de multicores, cluster de multicores y cloud.



## PROGRAMA

### Conceptos básicos

Paralelismo. Procesos y Procesadores. Interacción, comunicación y sincronización de procesos. Concurrencia y Paralelismo.

Impacto del procesamiento paralelo sobre los sistemas operativos y lenguajes de programación. Speedup y Eficiencia de algoritmos paralelos.

Concepto de asignación de tareas y balance de carga. Balance de carga estático y dinámico.

### Arquitecturas orientadas a Procesamiento Paralelo

Clasificación por mecanismo de control. Clasificación por la organización del espacio de direcciones. Memoria distribuida y memoria compartida.

Clasificación por la granularidad de los procesadores.

Clasificación por la red de Interconexión. Redes estáticas y dinámicas.

Análisis del impacto en el speedup alcanzable.

Vector processors, array processors, Arquitecturas cúbicas e hipercúbicas.

Supercomputadoras.Clusters de PCs. Multiclusters. Grids.

### Diseño de algoritmos paralelos. Modelos y Paradigmas.

Técnicas de descomposición. Características de los procesos. Interacción.

Técnicas de mapeo de procesos/procesadores. Balance de carga.

Modelos de algoritmos paralelos. Problemas paralelizables y no paralelizables.

Paralelismo de datos. Paralelismo de control.

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.

### Métricas del paralelismo

Medidas de performance standard.

Fuentes de overhead en procesamiento paralelo.

Speedup. Rango de valores. Speedup superlineal.

Overhead paralelo. Grado de paralelismo.

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.

### 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.

**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.

**Conceptos de Grid Computing y Cloud Computing**

Clusters, multiclusters y GRID.  
Extensión de conceptos de Cluster-computing a Grid-computing.  
Funcionalidades básicas del middleware en arquitecturas GRID.  
Conceptos de Cloud Computing.  
Modelos y paradigmas de Sistemas Paralelos aplicables en Grid y Cloud.  
Análisis de casos en Grid y Cloud computing.,

**BIBLIOGRAFIA**

**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.

**The GRID 2. Blueprint for a new computing infrastructure.**

Foster, Kesselman Morgan Kauffman 2004.

**MPI: The complete Reference**

Snir, Otto, Huss-Lederman, Walker, Dongarra, Cambridge, MA: MIT Press, 1996.

**IEEE, ACM Digital Library**



UNIVERSIDAD NACIONAL DE LA PLATA  
FACULTAD DE INFORMÁTICA

---

**"Above the clouds: A berkeley view of cloud computing". Technical report.**

Michael Armbrust, Armando Fox, Rean Gri\_th, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia. February 2009

**A break in the clouds: towards a cloud definition. SIGCOMM Comput. Commun.**

Vaquero, L. M., Rodero-Merino, L., Caceres, J., and Lindner, M. 2008. Rev. 39, 1 (Dec. 2008), 50-55. DOI= <http://doi.acm.org/10.1145/1496091.1496100>

**"Cloud Computing and Grid Computing 360-Degree Compared"**

Foster, I.; Yong Zhao; Raicu, I.; Lu, S.; , Grid Computing Environments Workshop, 2008. GCE '08 , vol., no., pp.1-10, 12-16 Nov. 2008 doi: 10.1109/GCE.2008.4738445  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4738445&isnumber=4738437>.