



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



Especialización en Cómputo de Altas Prestaciones – Modalidad a distancia

Fundamentos de Procesamiento Paralelo

Año 2021

Duración: 70 hs. Totales.

Cantidad de horas presenciales/VC: 25 hs.

Cantidad de horas de actividades en línea y de trabajo final: 45 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. 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 diferentes arquitecturas multiprocesador.

Pre requisitos:

Conocimientos básicos de Concurrencia y Sistemas Operativos.

Manejo de lenguajes de expresión de algoritmos.

COMPETENCIAS A DESARROLLAR EN RELACION CON EL OBJETIVO DE LA CARRERA

C.1- Analizar problemas del mundo real que por su complejidad y/o volumen de datos requieran cómputo paralelo y diseñar soluciones desde el punto de vista del hardware necesario, lo que requiere un conocimiento de las arquitecturas paralelas actuales.

C.2- Conocer los fundamentos para el desarrollo de Sistemas Paralelos (incluyendo la relación entre hardware y software).



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



CONTENIDOS MINIMOS:

- Conceptos básicos de Concurrencia y Paralelismo. Comunicación y Sincronización.
- Arquitecturas orientadas a procesamiento paralelo.
- Modelos y paradigmas para la programación de Algoritmos Paralelos.
- Métricas del paralelismo: rendimiento y eficiencia.
- Programación sobre arquitecturas de memoria compartida / distribuida e híbridas.
- Conceptos de Cloud Computing.
- Algoritmos paralelos sobre diferentes arquitecturas multiprocesador. Aplicaciones.

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.

Placas aceleradoras. Configuraciones multiprocesador de servidores en la nube.

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.



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



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.

Conceptos y expresión 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.

Conceptos y expresión 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 Cloud Computing

Clusters, multiclusters, configuraciones GRID y CLOUD.

Extensión de conceptos de Cluster-computing a Grid y Cloud-computing.

Modelos y paradigmas de Sistemas Paralelos aplicables en Grid y Cloud.

Análisis de casos en Grid y Cloud computing.,

Tópicos de Programación paralela en diferentes arquitecturas

Conceptos generales de aceleradores, en particular GPU.

Utilización de las GPU para programación paralela.

Análisis de casos, comparando algoritmos paralelos sobre arquitecturas multiprocesador.



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



ACTIVIDADES EXPERIMENTALES y DE INVESTIGACION

Tareas en Laboratorio (presencial o remoto)

Tal como se explica en el ítem relacionado con la metodología, ésta se basa en clases sincrónicas (presenciales o remotas) combinadas con actividades demostrativas en el laboratorio para aplicar los conceptos teóricos y que así el alumno adquiera las competencias y habilidades sobre cada uno de los temas que forman parte del contenido de la asignatura.

Además, el alumno debe analizar un proyecto relacionado con los temas dictados en la teoría, cuyo despliegue práctico se realiza sobre una infraestructura virtualizada que los alumnos pueden acceder en forma remota (en el Laboratorio dedicado a Paralelismo en el Postgrado).

Investigación/ Estudios adicionales:

Los alumnos analizarán papers relacionados con Fundamentos del Paralelismo, tanto en cuanto a la evolución de las arquitecturas como los lenguajes y paradigmas de programación adecuadas a las nuevas arquitecturas.

Se les propondrán temas de I+D orientados al estudio comparativo de soluciones de sistemas paralelos basados en diferentes soportes físicos y lógicos.

METODOLOGIA Y MODALIDAD DE EVALUACION

La metodología se basa en clases sincrónicas a través del sistema de videoconferencias adoptado por el Postgrado de Informática combinadas con sesiones en el laboratorio remoto para aplicar los conceptos teóricos y que así el alumno adquiera las competencias y habilidades sobre cada uno de los temas que forman parte del contenido de la asignatura.

Se requiere un 80% de asistencia a los encuentros sincrónicos, incluyendo el encuentro inicial de presentación de la materia, y el encuentro final de integración, ambos de asistencia obligatoria.

El trabajo se complementa con un proyecto experimental que debe desarrollar el alumno para cumplimentar las horas asignadas con soporte tutorizado por el profesor (*on-line*) y seguimiento a través del Entorno Virtual IDEAS contemplado en el SIED de la Facultad de Informática de la UNLP.

El despliegue práctico se realizará sobre una infraestructura virtualizada (parcialmente desplegada) que mostrarán los aspectos funcionales y donde los alumnos deberán ampliar y ejecutar programas paralelos/distribuidos de prueba (*benchmarks*) y realizar su evaluación de performance con las métricas estudiadas en el curso.



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



La evaluación se realizará mediante un examen escrito al final de las sesiones sincrónicas para evaluar el grado de conocimientos del alumno (20%), el proyecto/desarrollo experimental que deberá entregar el alumno al final de las horas programadas (70%) y la participación y aportaciones de calidad/excelencia a las soluciones propuestas (10%).

RECURSOS Y MATERIALES DE ESTUDIO

Como materiales de estudio, se dispone de:

- Presentaciones multimedia desarrolladas ad-hoc para introducir cada uno de los diferentes ejes temáticos.
- Ejemplos donde se aplican los conceptos teóricos
- Píldoras formativas con la explicación de algunos temas
- Ejercicios prácticos que son desarrollados en clase
- Material de lectura para estudiar y profundizar conceptos abordados en las clases
- Enlaces a artículos de actualidad de repositorios reconocidos en el área
- Libros digitales
- Acceso a equipamiento remoto situado en la Facultad de Informática de la UNLP y también en la nube (Cloud) de acuerdo a la disponibilidad del Postgrado de la Facultad de Informática para sus cursos.
- Software específico para determinadas actividades de laboratorio que se detallan en este programa.

ACTIVIDADES EXPERIMENTALES Y APROPIACIÓN DE SABERES

Los trabajos experimentales pueden desarrollarse en cada clase o continuarse en más de una clase. Parten de una especificación/consigna del docente (explicada en la clase) y un trabajo individual o en grupos que interactúan en el que los alumnos resuelven un problema experimental concreto relacionado con la temática.

Los trabajos podrán ser individuales o grupales. Para esto último se configurará el entorno virtual para que los alumnos del mismo grupo se encuentren en un espacio virtual diferente del resto. Durante el desarrollo del trabajo, el docente estará conectado respondiendo dudas y consultas.

Estos trabajos pretenden desarrollar y/o fortalecer las aptitudes de opinión crítica en los temas relativos del curso. Los alumnos deberán sintetizar su comprensión de los temas, al realizar correctamente la tarea experimental propuesta.

También se pretende desarrollar la capacidad de poder comunicar y transmitir los resultados, en presentaciones pautadas a lo largo del curso.

En general, finalizada una actividad, hay una sesión de discusión conjunta donde los participantes comunicarán sus opiniones e intercambiarán los distintos puntos de vista.



FACULTAD DE INFORMATICA



UNIVERSIDAD
NACIONAL
DE LA PLATA



BIBLIOGRAFÍA BASICA

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.

An introduction to parallel programming.

Peter Pacheco. Elsevier (2011).

BIBLIOGRAFÍA COMPLEMENTARIA

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

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

Michael Armbrust, Armando Fox, Rean Griffith, 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>.

Cloud Computing: A Hands-On Approach

Arshdeep Bahga, Vijay Madisetti. CreateSpace Independent Publishing Platform, 2013. ISBN-13: 978-1494435141

Programming Massively Parallel Processors: A Hands-on Approach (Applications of GPU Computing Series)

David B. Kirk, Wen-mei W. Hwu. Morgan Kaufmann, 2010. ISBN-13: 978-0123814722

CUDA Programming: A Developer's Guide to Parallel Computing with GPUs (Applications of Gpu Computing)

Shane Cook. Morgan Kaufmann, 2012. ISBN-13: 978-0124159334