

Programa Curso de Postgrado

Sistemas Paralelos: Modelos y Paradigmas

OBJETIVOS:

La computación paralela surgió como la alternativa válida y natural para cubrir la insistente demanda de mayor desempeño (performance). Su incorporación como paradigma de computación no fue tan natural. La mayoría de los estudios teóricos y prácticos sobre computación paralela, generalmente están enfocados a analizar diferentes arquitecturas y lenguajes de programación o herramientas paralelas reales. El diseño de programas paralelos y su eficiencia están fuertemente relacionados a la arquitectura paralela subyacente y al modelo de programación aplicado. Para obtener eficiencia, un programa paralelo debe ser adecuado a las características de la máquina paralela real. Esto se logra, si el programa paralelo es: diseñado siguiendo un modelo de computación paralela, implementado a través de un modelo de programación y evaluado según un modelo de costo.

El curso tiene como objetivo establecer el marco teórico adecuado para incorporar el paradigma paralelo en una buena y eficiente solución computacional de problemas comunes. Para ello, se propone realizar un estudio formal de los sistemas paralelo basado en el análisis de distintos modelos de computación paralela, de los modelos de programación, las estructuras típicas y los paradigmas.

CONTENIDOS MINIMOS:

Unidad I: Principios del Diseño de Algoritmos Paralelos.

Unidad II: Modelos de Computación.

Unidad III: Modelo de Programación de Memoria Compartida.

Unidad IV: Modelo de Programación de Memoria Distribuida.

Unidad V: Modelo de Computación PRAM.

Unidad VI: Modelo de Computación BSP.

Unidad VII: Otros Modelos de Computación Paralela.

PROGRAMA:

I- Principios del Diseño de Algoritmos Paralelos.

Características de los Algoritmos Paralelos. Descomposición, técnicas de descomposición: Recursiva, Datos, Exploratorias, entre otras. Granularidad y Concurrencia. Tareas, relación entre tareas. Consecuencias de la inter-relación de tareas y técnicas para reducir su costo. Mapping de tareas. Procesos y Procesadores.

II- Modelos de Computación.

Características de los Modelos de Computación. Necesidad de un modelo de Computación. Modelo de Programación. Propiedades de los modelos de programación. Clasificación de los modelos de programación. Modelo de programación "divide y vencerás", "ramificación y acotación", "master/slave", "farm", entre otras. Modelo de Análisis de Costo. Características y

propiedades de los modelos de costo. Relación entre Modelo de Computación, Modelo de Programación y Modelo de Análisis de Costo. Paradigmas de Computación Paralela: Paralelismo de Datos, de Tareas, Anidado de Datos, Pipeline.

III- Modelo de Programación de Memoria Compartida

Fundamentos. Ventajas y Dificultades. Modelo de Threads. Threads y Tareas. Programación Paralela Estructurada. Memoria Compartida Distribuida. Modelos de Comunicación. Sincronización. Comunicación One-Side. Modelos de Programación establecidos por la biblioteca: Modelo OpenMP, Modelo Pthreads y Modelo Java threads. Análisis de Costo de Programas Paralelos.

IV- Modelo de Programación de Memoria Distribuida.

Fundamentos. Ventajas y Dificultades. Tareas, Procesos y Procesadores. Administración de Tareas. Balance de Carga: Distribución Óptima de Procesadores. Grupos de Tareas. Comunicación Inter-tareas. Interoperabilidad. Modelos de Comunicación. Sincronización. Memoria Distribuida vs. Memoria Compartida. Modelos de Programación establecidos por la biblioteca: Modelo MPI, Modelo PVM, Modelo sockets, entre otros. Ventajas y Desventajas de cada uno. Análisis de Costo de Programas Paralelos.

V- Modelo de Computación PRAM.

Características generales. Computadora PRAM. Resolución de Conflictos de acceso a Memoria. Modelo de Programación PRAM. Activación y Asignación de Procesadores. Políticas de Asignación de Procesadores. Modelo de Costo PRAM. Características. Cuestionamientos al modelo PRAM. Factibilidad del modelo PRAM.

VI- Modelo de Computación BSP

Características generales. Computadora BSP. Computadora teórica vs. Computadoras reales. Modelo de Programación BSP. Conceptos de paso y superpaso. Relación entre comunicación, cómputo y sincronización. Modelo de Costo. Características. Parámetros del Modelo BSP: L, g. Predicción del rendimiento de un programa BSP. Análisis del costos de un superpaso: cómputo y comunicaciones. Concepto de h-relación. Relación entre modelo PRAM y modelo BSP.

VII- Otros Modelos de Computación Paralela:

OTMP, OBSP, OBSP*, LogP.

SISTEMA DE EVALUACION:

Consistirá en el desarrollo de un proyecto teórico-práctico. Su objetivo es desarrollar sistemas paralelos de mayor complejidad aplicando los conceptos vistos durante el curso.

BIBLIOGRAFIA:

- [1] O. Bonorden, N. Huppelshauer, B. Juurlink, I. Rieping - *PUB library, Release 6.0 - User guide and function reference*. University of Paderbon, Germany. 1998.
- [2] O. Bonorden, B. Juurlink, I. von Otte, I. Rieping - *The Puderborn University BSP (PUB) Library - Design, Implementation and performance*. 13th. International Parallel Processing Symposium 10th. Symposium on Parallel and Distributed Processing (IPPs/SPDP). 1999.
- [3] R. Chandra, L. Dagum, D. Kohr, D. Mandan, J. McDonald and R. Menon - *Parallel Programming in OpenMP* - Morgan Kaufmann Publishers. 2001
- [4] I. Foster - *Designing and Building Parallel Programs*. Addison-Wesley. 1994.

- [5] J.A. González, C. León, M.F. Piccoli, M. Printista, J.L. Roda, C. Rodriguez, F. Sande - *Performance Prediction of Oblivious BSP Programs* - Euro-Par 2001. Parallel Processing. 7th. International Euro-Par Conference. Pp. 96-105. Springer. Manchester, UK, August, 2001.
- [6] J.A. González, C. León, M.F. Piccoli, M. Printista, J.L. Roda, C. Rodriguez, F. Sande - *Oblivious BSP* - Expuesto en el 6to. International Euro-Par Conference. Publicado en EURO-PAR 2000, Parallel Processing(Springer). ACM. Pp. 682-685. Munich, Germany. Agosto-Setiembre de 2000.
- [7] M. Goudreau, K. Lang, S. Rao - *Towards Efficiency and Portability: Programming with the BSP model*. SPAA'96: 8th Annual ACM SPAA. PP 1-12. 1996.
- [8] A. Grama, A. Gupta, G. Karypis, and V. Kumar - *Introduction to Parallel Computing*. Addison-Wesley, 2003.
- [9] M. Gou and Layang - *High Performance Computing : Paradigm and Infrastructure*. Wiley. 2006.
- [10] J. Keller, C. Kesler, J. Larsson - *Practical PRAM programming*. John Wiley & Sons inc.. 2001.
- [11] C. Leopold - *Parallel and Distributed Computing: A survey of models, paradigms, and approaches*. John Wiley & Sons inc.. 2001.
- [12] W.F. McColl - *General purpose parallel computing*. Gibbons and Spirakis. Pp 337-391. 1993.
- [13] W.F. Mccoll - *BSP Programming*. DICMACS Series in Discrete Mathematics and Theoretical Computer Science. Vol: 0. May, 1994.
- [14] S. Pelagatti - *Structured Development of Parallel Programs* - Taylor & Francis -1998.
- [15] M.F. Piccoli - *Computación Paralela Anidada a través de Hiper cubos Generalizados*. Tesis Doctoral. Universidad Nacional de San Luis. 2005.
- [16] A.M. Printista - *Análisis y Predicción de Modelos de Computación Paralela*. Tesis Doctoral. Universidad Nacional de San Luis. 2004.
- [17] M. Quinn - *Parallel Computing. Theory and Practice. Second Edition*. McGraw-Hill. Inc. 1994.
- [18] L.G. Valiant - *A Bridging Model for Parallel Computation*. Communications of the ACM. Vol. 33, N° 8. Pp 103-111. 1990.
- [19] L. Valiant - *General Purpose parallel Architectures-* Handbook of Theoretical Computer Science. Edit. In J. van Leeuwen. Cap. 18, pp 943-971. 1990.
- [20] B. Wilkinson, M. Allen - *Parallel Programming: Techniques and Aplications using Networked Workstations and Parallel Computers*. Prentice Hall. 1999.