

# Aspectos prácticos de la asignatura Sistemas Multiprocesadores

Julio Pons, José Flich, Miguel Mateo, Alejandro Lucero

Departamento de Informática de Sistemas y Computadores  
Universidad Politécnica de Valencia

46071 Valencia

e-mail: {jpons,jflich,mimateo}@disca.upv.es, alucero@gap.upv.es

## Resumen

En este artículo se realiza una descripción de las prácticas de la asignatura Sistemas Multiprocesadores que se imparte en la Facultad de Informática de Valencia. Esta asignatura es una asignatura optativa de 5º curso con un 50% de contenido práctico. Dado que la temática de la asignatura se orienta hacia los computadores de última generación, es necesario buscar herramientas de simulación que permitan estudiar el comportamiento de dichos computadores y el desarrollo de programas paralelos para ellos. Por otro lado, también se realizan prácticas sobre clusters que permiten que el alumno desarrolle aplicaciones que aprovechen la potencia conjunta de los puestos de trabajo del laboratorio.

## 1. Introducción

La asignatura Sistemas Multiprocesadores es una asignatura optativa que se cursa en el noveno cuatrimestre de los estudios conducentes a la obtención del título de Ingeniero en Informática en la Facultad de Informática de Valencia. La asignatura se enmarca en la intensificación Ingeniería de los Computadores.

La asignatura tiene una carga lectiva de 6 créditos, dividida en tres créditos de teoría y problemas de aula, y tres créditos de prácticas de laboratorio. Por lo tanto, los alumnos realizan dos horas de prácticas por semana.

El objetivo principal de la asignatura es que el alumno conozca las alternativas de diseño más usuales de los sistemas multiprocesadores y de los modelos de programación asociados.

Para ello, en primer lugar, se presentan los distintos tipos de multiprocesadores. A continuación se estudian los modelos de

programación basados en variables compartidas y en paso de mensajes. Posteriormente se profundiza en el estudio de los multiprocesadores simétricos (SMP), los multiprocesadores escalables de memoria compartida (SSMP), los multicomputadores y los clusters.

En el apartado práctico se realizan prácticas orientadas a los modelos de programación de variables compartidas sobre SMP y clusters, y de paso de mensajes en clusters. También se utilizan simuladores de redes de interconexión de multicomputadores para la evaluación de algoritmos de encaminamiento y diversos mecanismos de conmutación. Este artículo se centra en los aspectos relacionados con las prácticas de la asignatura.

En la sección 2 se presenta el temario de la asignatura y se enumeran las prácticas, indicándose la planificación temporal. En la sección 3 se describen los recursos que se utilizan para la realización de las prácticas. En la sección 4 se describen las prácticas y, por último, en la sección 5 se presentan las conclusiones.

## 2. Temario y prácticas

El temario de la asignatura se compone de los siguientes temas:

Tema	Horas
Arquitectura de los sistemas multiprocesadores	4
Programación de los sistemas multiprocesadores	4
Multiprocesadores simétricos de memoria compartida (SMP)	8
Multiprocesadores escalables de memoria compartida (SSMP)	4
Multicomputadores y redes de estaciones de trabajo (clusters)	10

Las prácticas propuestas y su duración en sesiones de dos horas se muestra a continuación:

Práctica	Sesiones
Introducción a los simuladores de redes de interconexión: <i>wormhole</i> y <i>virtual cut-through</i>	1
Los simuladores de redes de interconexión: <i>wormhole</i> y <i>virtual cut-through</i>	3
Introducción al simulador de sistemas multiprocesadores: LIMES	1
Paralelización y evaluación de un sistema secuencial con LIMES	3
PVM en una red de estaciones de trabajo	2
DSMs software en Clusters	2
Recuperación	2

Considerando cuatrimestres de quince semanas, la correspondencia temporal entre teoría y práctica se muestra en la figura 1.

En la fila de prácticas se ha colocado como subíndice para cada práctica, los temas de teoría relacionados con la misma.

La primera sesión de prácticas (numerada en la tabla como práctica 0) sirve para describir al alumno el entorno de trabajo y presentarle la planificación del curso.

En las prácticas 1 y 3 se incluyen breves seminarios en los que se introducen conceptos necesarios para el desarrollo de dichas prácticas y las siguientes.

Las dos últimas sesiones (etiquetadas como *Trabajos*) son sesiones de prácticas en las que los alumnos pueden terminar los ejercicios propuestos en las diferentes prácticas. Los alumnos que deseen realizar ejercicios complementarios sobre las prácticas pueden usar estas sesiones ya que la asistencia a las mismas es voluntaria.

La evaluación de la asignatura se realiza

considerando tanto la teoría como las prácticas. Para ello, el alumno debe realizar una memoria de cada una de las prácticas a realizar. En cada boletín de prácticas se le plantean al alumno cuestiones que deberá contestar.

Las memorias de las prácticas se evalúan y representan el 40% de la nota final de la asignatura. El 60% restante se obtiene con un examen escrito en el que, aparte de los ejercicios y cuestiones derivados de las clases teóricas, se incluyen cuestiones relacionadas con las prácticas.

### 3. El laboratorio de prácticas

El lugar de la realización de las prácticas es el laboratorio de prácticas de la Unidad Docente de Arquitectura y Modelado de Computadores. Este laboratorio cuenta con 27 puestos de prácticas con las siguientes características:

- 11 puestos con AMD K6-2 a 450 MHz.
- 3 puestos con Pentium II a 350 MHz.
- 6 puestos con biprocesadores Pentium II a 350 MHz.
- 4 puestos con biprocesadores Pentium MMX a 200 MHz.
- 3 puestos con Pentium MMX a 166 MHz.

Todos los puestos cuentan con al menos 64 MB de RAM, 8 GB de disco duro y monitor de 17". Todos los puestos están conectados mediante una red Ethernet a 100 Mb a través de conmutadores.

El laboratorio dispone, además de los puestos para los alumnos, de un servidor LINUX para docencia. En este servidor existe una cuenta para cada uno de los alumnos matriculados en las asignaturas que utilizan LINUX en el laboratorio.

El servidor está conectado 24 horas al día, 7 días a la semana, para que los alumnos se puedan conectar de forma remota.

La utilización de NFS (Network File System) en todos los equipos permite que un alumno siempre acceda a la misma información, sea cual sea el puesto de trabajo que ocupe durante una

Teoría	1		2		3				4		5				
Semanas	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Prácticas	0	1 <sub>(1)</sub>	2 <sub>(1)</sub>		3 <sub>(2,3)</sub>		4 <sub>(2,3)</sub>		5 <sub>(1,2,5)</sub>		6 <sub>(5)</sub>		<i>Trabajos</i>		

Figura 1. Planificación temporal

sesión de prácticas. La información de cada usuario se encuentra almacenada en el servidor Linux.

Además, se ha instalado un pequeño cluster Linux formado por 7 computadores con procesador Pentium a 133 MHz interconectados con un concentrador Ethernet a 100 Mb.

Todas las prácticas se realizan bajo el sistema operativo Linux. No obstante, para la confección de las memorias, el alumno también tiene disponible el sistema operativo Windows.

#### 4. Descripción de las prácticas

Las prácticas de la asignatura se dividen en tres bloques claramente diferenciados, constando cada uno de ellos de dos prácticas. El contenido de estos se muestra a continuación:

- **Redes de multicomputadores.** Se evalúan diferentes mecanismos de conmutación, topologías, etc.
- **SMP basados en bus.** Se evalúan diferentes protocolos de coherencia de antememorias y el efecto de diferentes aspectos de implementación.
- **Clusters.** Utilización de diferentes modelos de programación para la ejecución de diversas aplicaciones paralelas.

Debido a que cada bloque trata temas distintos e igualmente importantes, se ha asignado a cada bloque (figura 1) el mismo número de sesiones (cuatro sesiones). La distribución de cada bloque a lo largo del cuatrimestre se ha realizado en función del temario de teoría, de forma que los conceptos teóricos necesarios para la realización de una práctica hayan sido previamente impartidos. En las siguientes secciones se describe cada bloque.

##### 4.1. Redes de multicomputadores

En las dos primeras prácticas se utilizan simuladores de redes de interconexión para multicomputadores con el fin de que el alumno evalúe diferentes aspectos de diseño como mecanismos de conmutación, algoritmos de encaminamiento, topologías, etc.

En la primera práctica se realiza una toma de contacto con diversos simuladores. En concreto se utiliza un simulador del mecanismo de conmutación *wormhole* y un simulador del

mecanismo de conmutación *virtual cut-through* (VCT). Estos simuladores han sido desarrollados por el grupo de arquitecturas paralelas (GAP [6]) y han sido utilizados en diferentes trabajos de investigación [2][3][4][5].

El alumno empieza realizando todo un proceso de simulación guiado por el profesor, desde la definición de los parámetros de entrada hasta la interpretación de los resultados obtenidos. Para la representación de los resultados se le ofrecen las herramientas gráficas necesarias.

Una vez el alumno conoce los simuladores, el proceso de simulación y las herramientas gráficas, se le pide como evaluación de la práctica la simulación de los mecanismos *wormhole* y *virtual cut-through* bajo unas condiciones diferentes del ejemplo guiado y la interpretación de los resultados obtenidos.

La siguiente práctica se centra en el estudio de diferentes características de las redes con el objeto de determinar qué configuraciones resultan más eficientes. Para ello, la práctica se divide en varios apartados, realizándose en cada uno de ellos varias simulaciones.

El objetivo es que en cada apartado se estudie una determinada característica de la red de interconexión. En concreto, se estudia el efecto en las prestaciones de las siguientes características: organización del *crossbar* en VCT, organización de las colas en VCT, y tamaño de las colas de entrada y salida en *wormhole*.

Para la evaluación de la práctica, el alumno debe entregar los resultados de los ejercicios prácticos, en formato gráfico, comentándolos y justificándolos.

##### 4.2. SMP basados en bus

En las dos prácticas de este bloque se hace especial hincapié en los protocolos de coherencia de cache de invalidación (Berkeley) y los protocolos de coherencia de cache de actualización (Dragon) utilizados en los sistemas SMP basados en bus, analizándose las ventajas e inconvenientes de cada uno de ellos.

El objetivo del bloque es que el alumno conozca los problemas potenciales que pueden aparecer en dichos sistemas. Estos problemas se derivan, por una parte, de la escasa localidad de los datos y, por otra, de la posible saturación del sistema debido a la compartición del bus.

Para ello, la herramienta que se utiliza es el simulador LIMES [9]. LIMES simula en un ordenador de un solo procesador la ejecución de una aplicación paralela en un SMP de N procesadores basado en bus. Los procesos de dicha aplicación paralela se ejecutan en los diferentes procesadores virtuales y se tratan de forma que los resultados que obtienen coinciden con los que se obtendrían en un SMP real. Las aplicaciones que pueden ejecutarse son programas escritos en C, que usan las macros ANL [8] para implementar el paralelismo.

En la primera práctica, el alumno realiza la instalación del simulador. Seguidamente, obtiene los resultados al ejecutar diferentes aplicaciones del conjunto de aplicaciones SPLASH-2 [12]. La práctica se centra en la obtención del *speed-up* de las aplicaciones utilizando diferentes protocolos de coherencia cache. Posteriormente, se evalúa el efecto del número de procesadores y el tamaño de la cache sobre las prestaciones para los distintos protocolos de coherencia.

Con la segunda práctica se pretenden cumplir dos objetivos. En primer lugar, que el alumno utilice las macros ANL para la implementación de una aplicación paralela en un SMP a partir de una versión secuencial. A partir de dos vectores previamente ordenados, la aplicación a desarrollar deberá obtener un vector ordenado que incluya todos los elementos de ambos vectores.

El segundo objetivo es que el alumno evalúe la interacción del sistema SMP con la aplicación paralela que ha desarrollado. Para ello utiliza el simulador LIMES. En concreto, se modifican diferentes parámetros de diseño viendo cómo afecta al tiempo de ejecución de la aplicación.

## 4.2. Clusters

Como se ha visto, en los dos bloques anteriores se han utilizado herramientas de simulación. No obstante, en los últimos años, están apareciendo múltiples aplicaciones y sistemas que permiten aplicar el paralelismo a un conjunto de computadores personales y estaciones de trabajo. Un ejemplo de ello son los sistemas software de libre distribución que permiten que un conjunto de computadores sea visto como un cluster.

Es de constatar la importancia que tiene a nivel docente la utilización de dichos entornos ya que motiva al alumno al utilizar herramientas

reales. Por lo tanto, en este último bloque se utilizan diferentes entornos reales de clusters. Para ello, se dispone de un conjunto de computadores dedicado y de diferentes sistemas software para formar clusters.

En una primera práctica el alumno utiliza el sistema PVM [11]. Para ello, se le facilitan dos versiones de una aplicación que genera figuras de Mandelbrot. Una primera versión secuencial y una versión paralela (utilizando PVM) no optimizada. El alumno ejecuta sobre diferentes configuraciones del cluster (diferente número de computadores) la versión paralela y la compara con la aplicación secuencial, obteniendo el correspondiente *speed-up*. A continuación, al alumno se le motiva para que busque una versión paralela que ofrezca un mejor *speed-up*. Para ello, se le instruye para que centre sus esfuerzos en mejorar la distribución y asignación de tareas a cada uno de los procesos involucrados en la versión paralela.

Por último, en la segunda práctica, se ejercita el uso de diferentes sistemas software de memoria compartida distribuida (DSM software), en concreto: JIAJIA [7] y DIPC [1].

DIPC es un sistema que permite utilizar los IPC System V de una forma distribuida entre un conjunto de computadores, por lo que ofrece una visión de sistema único a las aplicaciones. Para ello, DIPC se ejecuta a nivel de kernel y de usuario. No obstante, DIPC prima la facilidad de uso antes que el obtener un buen rendimiento.

Por otro lado, JIAJIA es un sistema que se ejecuta a nivel de usuario y se basa en creación de un espacio de memoria virtual creado a partir de fragmentos de memoria de cada uno de los computadores. Asimismo, ejecuta remotamente cada uno de los procesos de la aplicación. Con todo esto, se crea la visión de un DSM.

Estos sistemas DSM software se encuentran instalados en el grupo de ordenadores que el alumno tiene disponible.

Antes de la práctica, se le introducen los dos sistemas, haciendo hincapié en sus diferencias para que así pueda ver las repercusiones de las técnicas empleadas (tipo de consistencia, diseño de la implementación, *multiple/single writer*) en el rendimiento global del sistema.

Nuevamente, en esta práctica se trabaja sobre dos aplicaciones que generan figuras de Mandelbrot utilizando las técnicas de

programación que facilitan ambos sistemas. En concreto, se implementa una versión optimizada de Mandelbrot utilizando JIAJIA y DIPC.

El alumno debe ejecutar la aplicación variando la granularidad (que afecta de forma directa a la localidad de los datos de la aplicación, lo que es vital en los sistemas DSM) y viendo como afecta al tiempo de ejecución. El alumno lanza diferentes ejecuciones con el fin de encontrar la granularidad óptima. Cabe recordar que el objetivo de la práctica es que el alumno utilice herramientas reales de clusters.

## 5. Conclusión

En el presente artículo, se han descrito las prácticas que se realizan en la asignatura de sistemas multiprocesadores. Dicha asignatura se imparte en la Facultad de Informática de la Universidad Politécnica de Valencia. Se ha presentado el temario de la asignatura y la planificación temporal de los temas de teoría y las sesiones de prácticas.

Las prácticas se han dividido en tres bloques. En el primer bloque se estudian las redes de interconexión de multicomputadores, en el segundo los sistemas SMP basados en bus, mientras que en el tercero se estudian los clusters.

En las prácticas se han utilizado herramientas de simulación. Estas herramientas permiten la evaluación de diferentes aspectos de diseño de las redes de multicomputadores y de los protocolos de coherencia cache de multiprocesadores. Su utilización permite al alumno aplicar los conceptos teóricos estudiados en clase, mejorando su aprendizaje.

Por otro lado, también se utilizan sistemas reales de cluster. El alumno evalúa de forma directa el efecto de diferentes aspectos del cluster sobre las prestaciones globales. De esta forma, el alumno se motiva más al utilizar y evaluar sistemas reales. Los sistemas utilizados han sido PVM, JIAJIA y DIPDC. Todos estos sistemas son de libre distribución y permiten que el alumno pueda crear sus propios clusters.

Al ser una asignatura de último curso, es importante que los alumnos reciban conocimientos derivados de la investigación reciente desarrollada sobre la materia y, por otra parte, que utilicen herramientas disponibles,

similares a las herramientas que se pueden encontrar en su futura labor profesional. Los simuladores utilizados en prácticas son ampliamente utilizados en investigación.

## Referencias

- [1] Web DIPC, <http://wallybox.cei.net/dipc>.
- [2] J. Duato. *Deadlock-free adaptive routing algorithms for multicomputers: evaluation of a new algorithm*. Proceedings of the 3rd IEEE Symposium on Parallel and Distributed Processing, IEEE Computer Society Press. pp 840-847. 1991.
- [3] J. Duato. *A Theory of Fault-Tolerant Routing in Wormhole Networks*. Proceedings of the 1994 International Conference on Parallel and Distributed Systems, IEEE Computer Society Press, pp 600-607. 1994.
- [4] J. Duato and P. López. *Performance Evaluation of Adaptive Routing Algorithms for k-ary n-cubes*. Proceedings Parallel Computer Routing and Communication Workshop, pp 45-59. May 1994.
- [5] J. Duato and P. López. *Deadlock-free Fully-adaptive Minimal Routing Algorithms: Limitations and Solutions*. Computers and Artificial Intelligence Journal, Vol 14, pp 105-125. 1995.
- [6] Web GAP, <http://www.gap.upv.es>.
- [7] Web JIAJIA, <http://www.ict.ac.cn/chpc/dsm/index.html>.
- [8] E. Lusk et al. *Portable Programs for Parallel Processors*. Holt, Rinehart and Winston, Orlando, FL, 1987.
- [9] D. Magdic. *Limes: A Multiprocessor Simulation Environment*. TCCA Newsletter, pp 68-71, March 1997.
- [10] M.A. Mateo, J. Pons. *Experiencias de prácticas con PVM en la asignatura de Arquitectura de Computadores y Sistemas Operativos 2*. JENUI 2000, 2000.
- [11] Web PVM, [http://www.epm.ornl.gov/pvm/pvm\\_home.html](http://www.epm.ornl.gov/pvm/pvm_home.html).
- [12] S.C. Woo et al. *The SPLASH-2 Programs: Characterization and Methodological Considerations*. Int. Symp. on Computer Architecture, 1995.