

# *Métricas y escalabilidad de algoritmos paralelos*

Marcelo Naiouf<sup>1</sup>, Armando De Giusti<sup>2</sup>, Diego Tarrío<sup>3</sup>, Laura De Giusti<sup>3</sup>  
{mnaiouf,degiusti,dtarrio,ldgiusti}@lidi.info.unlp.edu.ar

LIDI. Laboratorio de Investigación y Desarrollo en Informática  
Facultad de Informática. UNLP  
50 y 115. 1er Piso. La Plata

## Introducción

Resulta innegable la importancia del procesamiento paralelo en el espectro de la Ciencia de la Computación. Existen diversas razones para esta realidad:

- la capacidad de reducir el tiempo de procesamiento en problemas de cómputo intensivo o de grandes volúmenes de datos
- el límite físico alcanzado por las computadoras secuenciales en algunos casos torna inaceptable el tiempo para resolver determinados problemas, y hace que la solución paralela sea la única factible.
- la existencia de sistemas en los que no interesa tanto la velocidad de cómputo sino la necesidad de estar en más de un lugar a la vez, capacidad que puede mapearse a una configuración paralela
- las posibilidades que el paradigma paralelo ofrece en términos de investigación de técnicas para el análisis, diseño y evaluación de algoritmos.

En una de las líneas de investigación aplicada dentro del proyecto *Procesamiento Concurrente y Paralelo* del LIDI se trabaja en temas de especificación y evaluación de algoritmos paralelos. Esto incluye el desarrollo de procesos paralelos, la transformación de algoritmos secuenciales en paralelos y las métricas de evaluación de performance sobre diferentes plataformas de soporte (tanto de hardware como de software). Más allá de la arquitectura de hardware y el software de soporte, en esta línea la mayor importancia está dada en los *algoritmos paralelos*, y en los métodos utilizados para su construcción y análisis de performance.

Un punto importante es que no es adecuado referirse a un algoritmo paralelo sin mencionar el modelo de computación paralela para el que se lo diseñó. En este marco, debe destacarse que no hay un modelo teórico unificador, a diferencia del cómputo secuencial donde el modelo RAM es aceptado como tal. Esto se debe a que los diferentes modelos paralelos (PRAM para memoria compartida, BSP para memoria distribuida, LogP para redes de procesadores asincrónicos, etc), enfatizan determinados aspectos a costa de otros. Una razón más sutil es que no es probable una máquina paralela universal; para cada aplicación existe una máquina óptima.

Por esto, nos referimos a *sistemas paralelos* como la combinación de algoritmo y arquitectura. La diversidad hace complicado el análisis de performance del sistema paralelo. En otras palabras: *qué es lo que interesa medir? Qué indica que un sistema paralelo es mejor que otro? Qué sucede si se utiliza un algoritmo con una mayor cantidad de procesadores? Se sigue comportando adecuadamente?* En el mundo serial podemos realizar la evaluación a través de los requerimientos de tiempo y espacio del programa. En el paralelismo, hay otra cantidad de medidas que pueden ser de interés, esas métricas están ligadas tanto al algoritmo como a la arquitectura paralela de destino.

---

<sup>1</sup> Profesor Adjunto Dedicación Exclusiva.

<sup>2</sup> Investigador Principal CONICET. Profesor Titular Dedicación Exclusiva.

<sup>3</sup> Ayudante Diplomado Dedicación SemiExclusiva. Becario LIDI.

LIDI - Facultad de Informática UNLP - Calle 50 y 115 1er Piso, (1900) La Plata, Argentina.

TE/Fax +(54)(221)422-7707. <http://lidi.info.unlp.edu.ar>

## Temas de Investigación y Desarrollo

Existen una gran cantidad de métricas para evaluar sistemas paralelos. Las más conocidas son el tiempo de ejecución paralelo, el speedup y la eficiencia de utilización del sistema. Pero existen otras medidas que pueden ser útiles en algunas circunstancias: costo, overhead paralelo, grado de concurrencia, escalabilidad, isoeficiencia, isospeed, etc. En particular, en el caso del speedup se han propuesto distintos modelos de performance, entre los que se pueden citar a la Ley de Amdahl (tamaño de problema fijo), la Ley de Gustafson (problemas escalados, donde el tamaño del problema crece con el incremento en el tamaño de la máquina), y el modelo de speedup de Sun y Ni (problemas escalados acotados en la capacidad de memoria). La *escalabilidad* de un sistema paralelo es una medida de su capacidad de incrementar speedup en proporción al número de procesadores. Refleja la capacidad de un sistema paralelo de incrementar los recursos de procesamiento efectivamente.

El tema de la escalabilidad, y su relación con la función de isoeficiencia, es de importancia dado que permiten capturar las características de un algoritmo paralelo así como de la arquitectura en la que se lo implementa. Se puede testear la performance de un programa paralelo sobre unos pocos procesadores y predecir su performance sobre un número mayor de procesadores. También permite caracterizar la cantidad de paralelismo inherente en un algoritmo paralelo, y puede usarse para estudiar el comportamiento de un sistema paralelo con respecto a cambios en parámetros de hardware tales como la velocidad de los procesadores y canales de comunicación.

En este marco, interesa evaluar la performance de distintas clases de aplicaciones sobre diferentes arquitecturas reales para obtener resultados concretos. El énfasis está en adecuar los resultados teóricos ideales en las métricas a la realidad de la implementación de determinada clase de problema sobre distintas plataformas. Este interés se basa en que muchos sistemas paralelos no alcanzan su capacidad teórica, y las causas de esta degradación son muchas y variadas. En parte puede deberse a la poca correspondencia entre aplicaciones, software y hardware, pero existen otros factores que pueden influir y no siempre son tenidos en cuenta al formular una métrica. Entre ellos están: desbalance en la carga de trabajo de procesos y procesadores, demoras debidas a la sincronización, overhead por scheduling, topología, efecto de la granularidad, grado de escalabilidad del sistema, mapeo de procesos y datos a procesadores que puede acarrear mayor o menor comunicación, distintos niveles de memoria involucrados, etc.

El análisis sobre plataformas disponibles permite estudiar el impacto que tienen algunos de estos factores sobre las implementaciones, y adecuar las métricas clásicas a las mismas. En particular se estudia la influencia de las estrategias de distribución de procesos y datos, y la carga (estática o dinámica) asignada a cada procesador sobre el speedup, la eficiencia y la escalabilidad. Se realiza experimentación para el caso de procesadores homogéneos y heterogéneos, en algoritmos numéricos y no numéricos.

### Equipamiento de experimentación

La línea de investigación en procesamiento paralelo existe en el LIDI desde el año 1994, utilizando en ese entonces redes de PC con Netbios, kits de 4 transputers y simuladores de DSPs. Actualmente la Facultad de Informática de la UNLP cuenta con un Laboratorio de Procesamiento Paralelo, en el cual se dispone de una serie de equipos que permiten realizar experiencias en el área. Se ha trabajado con:

- Multiprocesador homogéneo con 32 transputers T805 4Mb de memoria local. La importancia está dada en la posibilidad de reconfiguración, lo que permite la evaluación de performance sobre diferentes topologías de conexión.
- Red heterogénea con PCs y workstations. Permite la emulación de una arquitectura paralela utilizando software de soporte como PVM o MPI.
- MultiDSP, en especial para aplicaciones en procesamiento de señales (DSPs Motorola 56000 y MS340).

Respecto del soporte de software, básicamente se utilizó C (en sus distintas versiones paralelas para cada arquitectura), OCCAM y Ada.

Actualmente se está gestionando ante la SeTCIP la posibilidad de utilización de la supercomputadora Clementina II (SGI Origin 2000, multiprocesador con característica NUMA) incorporada al sistema científico.

### **Algunos resultados obtenidos y temas actuales**

Los resultados obtenidos y temas actuales incluyen publicaciones, tesinas de grado y tesis de postgrado. En los mismos se pone énfasis en las métricas de evaluación y el impacto de los factores mencionados anteriormente. En este sentido pueden mencionarse:

- efecto del desbalance de carga en el procesamiento, por ejemplo en operaciones sobre imágenes con carga no homogénea sobre arquitecturas homogéneas. Esto involucra distribución de tareas, estudio del número óptimo de particiones para el mejor speedup con buena eficiencia, distribución de datos, etc.
- análisis de performance en problemas de reconocimiento y clasificación de objetos, con restricciones de tiempo real
- problemas clásicos de sorting y manejo de matrices sobre diferentes arquitecturas
- trabajos sobre escalabilidad en algoritmos de búsqueda de caminos mínimo en grafos, sobre distintas arquitecturas y variación de topologías.
- paralelización de algoritmos de reconocimiento basado en texturas, con procesamiento intensivo.
- evaluación de performance en algoritmos de seguimiento de trayectorias de objetos en una secuencia de imágenes
- modelización de memoria cache distribuida
- análisis de similitud en imágenes

### **Bibliografía**

Akl S, "Parallel Computation. Models and Methods", Prentice-Hall, Inc., 1997.

Brinch Hansen, P., "Studies in computational science: Parallel Programming Paradigms", Prentice-Hall, Inc., 1995.

Colbrook A., Brewer E., Dellarocas C., Weihl W., "Algorithms for Search Trees on Message-Passing Architectures", IEEE Transactions on Parallel and Distributed Systems, Vol. 7, No. 2: February 1996, pp. 97-108

Day K., Tripathi A., "A Comparative Study of Topological Properties of Hypercubes and Star Graphs" IEEE Trans. Parallel and Distributed Systems, vol. 5, no. 1, pp. 31-38, Jan. 1994

Gupta A., Kumar V., "Performance properties of large scale parallel systems", Journal of Parallel and Distributed Computing, November 1993.

Hwang K., "Advanced Computer Architecture: Paralelism, Scalability, Programability", McGraw, 1993.

Kumar V., Grama A., Gupta A., Karypis G., "Introduction to Parallel Computing. Design and Analysis of Algorithms", Benjamin/Cummings, 1994.

Leighton F. T., "Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes", Morgan Kaufmann Publishers, 1992.

Miller R., Stout Q. F., "Algorithmic Techniques for Networks of Processors", CRC Handbook of Algorithms and Theory of Computation, M. J. Atallah, ed, 1998.

Sima D, Fountain T, Kacsuk P, "Advanced Computer Architectures. A Design Space Approach", Addison Wesley Longman Limited, 1997.

Steenkiste P., "Network-Based Multicomputers: A Practical Supercomputer Architecture", IEEE Transactions on Parallel and Distributed Systems, Vol. 7, No. 8, August 1996, pp. 861-875

Tinetti F., De Giusti A., "Procesamiento Paralelo. Conceptos de Arquitectura y Algoritmos", Editorial Exacta, 1998.

Zomaya A., "Parallel Computing. Paradigms and Applications", Int. Thomson Computer Press, 1996.ç