Cómputo y Comunicación: Definición y Rendimiento en Redes de Estaciones de Trabajo

Fernando G. Tinetti, Andrés Barbieri
Centro de Técnicas Analógico-Digitales (CeTAD)[1]
Laboratorio de Investigación y Desarrollo en Informática (LIDI)[2]
{fernando, barbieri}@lidi.info.unlp.edu.ar

1. Introducción

La relación costo-beneficio de las estaciones de trabajo las hacen claramente ventajosas para las aplicaciones de cómputo intensivo. A medida que crece el la cantidad de instalaciones de redes locales (básicamente estaciones de trabajo con capacidad de comunicación), se incrementa aún más la posibilidad de utilizarlas en cómputo científico, dado que se pueden configurar y utilizar como una máquina paralela. Desde este punto de vista, las redes de estaciones de trabajo (NOW: Networks of Workstations) instaladas tienen la capacidad de ser computadoras paralelas con "costo cero" al menos en términos de hardware.

La paralelización de aplicaciones no ha sido ni es algo inmediato, aún en el campo de las aplicaciones de cómputo intensivo, que tienen una marcada regularidad en cuanto a las operaciones que se resuelven. Las redes de estaciones de trabajo instaladas que se utilizan para cómputo paralelo, por otro lado, aportan sus propias características que normalmente complican la resolución de un problema en paralelo. En este contexto, es necesario investigar y explicitar cuáles son los problemas y las soluciones posibles para las distintas clases de problemas que se pueden encontrar [8].

Dentro de las aplicaciones numéricas, las provenientes del álgebra lineal, en particular operaciones matriciales, han sido y son de gran interés por el amplio rango de aplicaciones en las cuales se utilizan. De hecho, existe un estándar *de facto* que clasifica, define y determina una interfase común para la resolución de este tipo de problemas [1]. Esta interfase se ha extendido a la resolución en paralelo de las mismas operaciones, mayormente para arquitecturas paralelas de memoria distribuida [3].

Si bien existe un gran consenso en cuanto a la factibilidad y utilidad de cómputo paralelo sobre redes de estaciones de trabajo heterogéneas, también se puede notar una marcada necesidad en cuanto a la publicación de resultados satisfactorios respecto de rendimiento. Aún en el contexto de la caracterización de una red de computadoras de esta clase, sigue siendo muy dependiente de la experiencia la capacidad de definir con precisión el tipo de heterogeneidad que se tiene, y por lo tanto el tipo de balance de carga, por ejemplo, que se debe implementar.

2. Balance de Carga y Experimentación

El modelo de programación paralela *más natural* en el contexto de las redes de estaciones de trabajo es el de pasaje de mensajes, derivado de CSP (Communicating Sequential Processes) [6]. Esto se debe al muy bajo acoplamiento entre los procesadores (de hecho, computadoras completas), que se tiene en una red local, que se podría clasificar como una arquitectura MIMD de memoria distribuida [5]. De esta manera, una aplicación paralela es un conjunto de procesos que llevan a cabo cómputo secuencial y además poseen primitivas de comunicación (tales como `send-receive`) por medio de las cuales intercambian información y eventualmente también se sincronizan. Por lo tanto queda claro que todo lo referente a balance de carga computacional es definido por los segmentos de código secuencial y la sobrecarga (*overhead*) de comunicaciones es definida por la frecuencia y cantidad de datos de los mensajes entre procesos/procesadores.

---

[1]Facultad de Ingeniería, Universidad Nacional de La Plata
[2]Facultad de Informática, Universidad Nacional de La Plata

En términos de investigación en esta área se deben tomar un conjunto de recaudos específicos para evitar al máximo la posibilidad de error. Existen dos fuentes de errores muy importantes que no se encuentran con facilidad en otras arquitecturas paralelas/distribuidas homogéneas. Tanto la optimización de código secuencial como la dificultad de reproducir con exactitud los experimentos pueden llevar a conclusiones erróneas.

Es claro que dependiendo del nivel/tipo de optimización que se lleva a cabo en el código se obtiene un determinado rendimiento. En el contexto de las aplicaciones paralelas sobre hardware heterogéneo se tiene, además, la necesidad de balancear el procesamiento para que el rendimiento sea el óptimo (al menos en el contexto de las aplicaciones científicas). En las redes de estaciones de trabajo el objetivo de implementar balance de carga ya no es que todos los procesadores realicen la misma cantidad de operaciones sino que todos los procesadores utilicen aproximadamente el mismo tiempo de cálculo para el código secuencial.

Las múltiples posibilidades de heterogeneidad que se pueden encontrar en las redes de estaciones de trabajo instaladas son un factor importante en cuanto a la dificultad de replicación de los experimentos. Una de las redes sobre las que se trabaja, por ejemplo, está compuesta por cinco modelos diferentes de PCs, una estación de trabajo IBM y tres modelos diferentes de estaciones de trabajo Sun. Solamente como ejemplo, los tamaños de memoria de las computadoras varían entre 16 y 96 MB.

3. Comunicaciones

Así como el grado de heterogeneidad en cuanto a capacidad de cálculo posible de encontrar en una red local es muy grande, es notable la homogeneidad de las redes locales instaladas en cuanto a la interconexión de las computadoras. La gran mayoría de las redes locales están interconectadas bajo la norma Ethernet de 10 Mb/s. En las redes instaladas más recientemente es más probable encontrar Fast Ethernet (100 Mb/s) y Gigabit Ethernet parece ser el siguiente paso en cuanto a mayor capacidad de comunicación. En un porcentaje muy reducido de las redes locales instaladas se encuentran otro tipo de redes de interconexión de estaciones de trabajo.

Dado que la gran mayoría de las redes locales tiene hardware de comunicaciones similar (*Ethernet*), es importante lograr el máximo aprovechamiento en cuanto a la capacidad de comunicaciones que este tipo de redes proporciona. Desde el punto de vista de las aplicaciones paralelas, las características más importantes son:

- Medio único de transmisión de información: bus.
- Tiempo de inicialización de las comunicaciones muy grande.

El hecho de poseer un único medio de comunicaciones tiene ventajas y desventajas. La Principal desventaja es la serialización de las comunicaciones punto a punto: si se están comunicando dos computadoras, todas las demás deben esperar para acceder al medio (bus) de comunicaciones. La principal ventaja reside en su capacidad de *broadcast* físico: desde una computadora se tiene la capacidad de enviar datos a todas las demás simultáneamente. Es interesante cómo las librerías de cómputo paralelo sobre redes de estaciones de trabajo disponibles tales como PVM [4] y las implementaciones de MPI [7] no hacen uso de la capacidad de broadcast físico de las redes Ethernet para las comunicaciones colectivas, y por lo tanto existe cierto vacío (en cuanto a optimización de rendimiento) en esta área.

Aumentar el tiempo de inicialización (startup) de las comunicaciones entre procesadores de una máquina paralela implica necesariamente aumentar la granularidad de la aplicación paralela (disminuir la frecuencia y aumentar la cantidad de datos de las comunicaciones). Esto a su vez impone restricciones sobre la paralelización de las aplicaciones a resolver, aunque el ámbito numérico quizás sea uno de los más apropiados para resolver este problema dada su regularidad.

4. Areas de Investigación

Las áreas de investigación que se pueden identificar con claridad se pueden clasificar en términos de cómputo secuencial, paralelización de aplicaciones y comunicaciones, dado que son suficientemente específicas y separables como para ser llevadas a cabo en paralelo si es necesario.

En cuanto a cómputo, es claramente necesario definir con precisión cómo desarrollar software optimizado *independientemente* o dependiendo *paramétricamente* de la arquitectura de cómputo secuencial [2] [9].

Respecto de la paralelización de aplicaciones, es necesario definir con rigurosidad una metodología para el cálculo de la granularidad mínima de las aplicaciones a resolver sobre redes de estaciones de trabajo y, de alguna manera, definir el máximo número de computadoras que se pueden utilizar eficientemente en la resolución de un problema. Por otro lado, es ampliamente ventajoso contar con algoritmos paralelos definidos en términos de comunicaciones broadcast, ya que de esta manera se puede aprovechar al máximo la capacidad de comunicación de las redes Ethernet a la vez que se evita el inconveniente de la serialización de las comunicaciones punto a punto entre computadoras.

En cuanto a comunicaciones, es importante definir una metodología de experimentación para identificar si la librería utilizada (PVM o las implementaciones de MPI disponibles) para el intercambio de mensajes es apropiada desde el punto de vista del rendimiento obtenido. Asimismo, es necesario verificar si las comunicaciones colectivas se implementan de forma apropiada utilizando la capacidad de broadcast físico de las redes locales, como ya se ha puntualizado. Tal como lo indican algunos experimentos preliminares, ninguna de las librerías de cómputo paralelo implementan las rutinas de comunicaciones colectivas utilizando broadcast, y "transforman" todas las rutinas de comunicaciones en múltiples transferencias de datos punto a punto. Además, si el objetivo es utilizar redes de estaciones de trabajo instaladas, se hace cada vez más importante la posibilidad de utilización de más de una red local para cómputo paralelo. En este sentido, la mayoría de las redes locales tienen definido todo lo necesario para utilizar Internet y por lo tanto se deben definir toda una serie de parámetros para aprovechar estas redes junto con sus interconexiones para cómputo paralelo intensivo.

Referencias

[1] Anderson E., Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. DuCroz, A. Greenbaum, S. Hammarling, A. McKenney, D. Sorensen, LAPACK: A Portable Linear Algebra Library for High-Performance Computers, Proceedings of Supercomputing '90, pages 1-10, IEEE Press, 1990.

[2] Bilmes J., K. Asanovi□, C. Chin, J. Demmel, Optimizing matrix multiply using phipac: a portable, high-performance, ansi c coding methodology, Proceedings of the International Conference on Supercomputing, Vienna, Austria, July 1997, ACM SIGARC.

[3 Blackford L., J. Choi, A. Cleary, E. D'Azevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, R. Whaley, ScaLAPACK Users' Guide, SIAM, Philadelphia, 1997.

[4] Dongarra J., A. Geist, R. Manchek, V. Sunderam, Integrated pvm framework supports heterogeneous network computing, Computers in Physics, (7)2, pp. 166-175, April 1993.

[5] Flynn M, "Very High Speed Computing Systems", Proc. IEEE, Vol. 54, 1966.

[6] Hoare C, "Communicating Sequential Processes", Englewood Cliffs, Prentice-Hall, 1986.

[7] Message Passing Interface Forum, MPI: A Message Passing Interface standard, International Journal of Supercomputer Applications, Volume 8 (3/4), 1994.

[8] Tinetti F., A. Quijano, A. De Giusti, Heterogeneous Networks of Workstations and SPMD Scientific Computing, 1999 International Conference on Parallel Processing, The University of Aizu, Aizu-Wakamatsu, Fukushima, Japan, September 21 - 24, 1999.

[9] Whaley R., J. Dongarra, Automatically Tuned Linear Algebra Software, Proceedings of the SC98 Conference, Orlando, FL, IEEE Publications, November, 1998.