Apéndice B

Modelo: Descripción

En este Apéndice se explican los aspectos de nuestros algoritmos que modelamos para describir su comportamiento en un determinado computador. La descripción se realiza tanto para los aspectos que influyen en los algoritmos secuenciales (modelo secuencial) como los que influyen en los paralelos (modelo paralelo).

En el Apéndice C se describen los modelos para los algoritmos, adaptados a un computador en particular, basado en p630 con procesadores Power4. En el Apéndice D se hace lo mismo para el computador SGI O2000 con procesadores R10K. En ambos casos se desarrollan los modelos secuencial y paralelo.

Las unidades de medida de los modelos son CSE, ciclos por elemento ordenado, en inglés, *Cycles per Sorted Element*; que consiste en el número total de ciclos invertidos en la ordenación dividido entre el número total de elementos a ordenar.

Modelo Secuencial

El modelo secuencial es un modelo de memoria que calcula el número de CSE totales como la suma de:

1. CSE_{mem}: CSE de penalización por fallos de accesos a memoria y,
2. CSE_{cpu}: CSE considerando que no hay fallos de acceso a la jerarquía de memoria.

Aspectos que afectan a CSE_{mem}

Se diferencia entre CSE por fallos de memoria cache y por fallos de TLB.

- **Penalización por fallos de memoria cache.** Sólo se tienen en cuenta los fallos obligatorios y de capacidad. No se contabilizan los fallos por conflictos ya que la experiencia nos demuestra que contribuyen en menor medida al tiempo de los algoritmos analizados. Por lo tanto, se contabilizará la penalización de un

fallo de memoria cache cada vez que se acceda a una nueva línea de memoria cache, es decir, si el acceso fuera secuencial, habría un fallo cada R elementos accedidos, donde R es el número de elementos de una línea de cache. En el documento, las líneas de primer, segundo y tercer nivel memoria cache tienen R_1, R_2 y R_3 elementos de clave y puntero respectivamente. La penalización por fallo de primer, segundo y tercer nivel de memoria cache son C_{f1}, C_{f2} y C_{f3} ciclos respectivamente.

Así, los CSE por fallos obligatorios o de capacidad en el acceso secuencial de un vector de elementos de clave y puntero son:

$$CSE_{mem\_mc} = (C_{f1}\frac{1}{R_1} + C_{f2}\frac{1}{R_2} + C_{f3}\frac{1}{R_3})$$

dónde mc, en CSE_{mem_mc}, quiere decir memoria cache.

- **Penalización por fallo de TLB.** Por un lado, se tiene en cuenta la penalización por fallos obligatorios y de capacidad, es decir, la penalización por cada vez que se accede a una nueva página de memoria. Por consiguiente, los CSE por fallos obligatorios y de capacidad son:

$$C_{fTLB}\frac{1}{R_{page}}$$

dónde C_{fTLB} es la penalización por fallos de TLB y R_{page} es el número de elementos de clave y puntero que caben en una página de memoria.

Además, debido a que los algoritmos modelados, en ocasiones, necesitan mayor número de páginas de memoria concurrentemente utilizadas, que el número de páginas que puede abarcar el TLB (E_{TLB}), se considera que hay una probabilidad no despreciable (p_f) de fallos por conflicto de TLB. Los CSE debido a conflicto son :

$$p_f(C_{fTLB})$$

Así, los CSE totales por fallos de TLB son:

$$CSE_{mem\_TLB} = C_{fTLB}(\frac{1}{R_{page}} + p_f)$$

Aspectos que afectan a CSE_{cpu}

Se distingue entre CSE por accesos al primer nivel de memoria cache y CSE por operaciones que no son de lectura ni escritura.

- **Accesos al primer nivel de memoria cache.** Para contabilizar los CSE debido a accesos de lectura y/o escritura al primer nivel de memoria cache del cuerpo del bucle que se está modelando, se tienen en cuenta:

  - El grafo de dependencias entre las operaciones del cuerpo del bucle a modelar. Esto es importante para saber si dos operaciones se pueden realizar a la vez o no.
  - El número de unidades funcionales de lectura y/o escritura que se tienen y si son segmentadas o no. En esta tesis todas son segmentadas. Así, dos operaciones que se pueden ejecutar a la vez, ya que no son dependientes, se ejecutarán realmente a la vez si se dispone de unidades funcionales suficientes.

  Se considerarán los CSE_{cpu} debido a accesos a memoria como:

  $$CSE_{cpu\_accesos} \quad = \quad m + m'(C_a - 1)$$

  C_a es la latencia de acceso al primer nivel de memoria cache. m' es el número de accesos al primer nivel de memoria cache que no se pueden solapar con otras operaciones (ya sean éstas, accesos o no a memoria). m es el número de ciclos en los que se ha lanzado, en paralelo en el mismo ciclo, una o más operaciones de acceso, al primer nivel de memoria cache. Se podrá lanzar más de una operación (sin dependencias) de acceso al primer nivel de memoria cache a la vez siempre y cuando haya unidades funcionales disponibles.

  Se resta un ciclo a los C_a ciclos de acceso a primer nivel de memoria cache debido a que este ciclo se contabiliza implícitamente en los m ciclos de lanzamiento de operaciones de acceso a memoria.

- **Operaciones que no son de lectura ni escritura.** Este coste se identificará con CSE_{cpu_opers}. Hay casos en los que este coste es importante. Por ejemplo, cuando la complejidad del algoritmo en sí es proporcional a $n \log(n)$ o exponencial. Pero también puede haber aspectos de la arquitectura del computador que pueden afectar al rendimiento del algoritmo. Por ejemplo, el mayor o menor coste de penalización en caso de fallo en la predicción de saltos puede afectar al rendimiento.

Así, según el modelo secuencial, los CSE de un algoritmo de ordenación secuencial son:

$$\begin{aligned} CSE &= CSE_{mem} + CSE_{cpu} \\ CSE_{mem} &= (\frac{C_{f1}}{R_1} + \frac{C_{f2}}{R_2} + \frac{C_{f3}}{R_3}) \\ &+ C_{fTLB}(\frac{1}{R_{page}} + p_f) \\ CSE_{cpu} &= m + m'(C_a - 1) \\ &+ CSE_{cpu\_opers} \end{aligned}$$

En todo momento, y para hacer que el estudio sea más sencillo, si el conjunto de estructuras necesarias para el algoritmo caben en un determinado nivel de memoria se supondrá que esos datos estarán en ese nivel de memoria. En este caso no se contabilizarán fallos de capacidad ni obligatorios en ese nivel de memoria, tampoco de conflicto.

Modelo Paralelo

El modelo paralelo caracteriza los CSE de los algoritmos según dos puntos de vista: los CSE de cálculo (CSE_{cal}) y los CSE de comunicación (CSE_{com}).

- CSE_{cal}: se calcula según el modelo secuencial que se ha explicado arriba
- CSE_{com}: se refiere al total de CSE invertidos en comunicación por el algoritmo. Aquí se calcula como:

  $$CSE_{com} = \tau/n + \alpha T_e(\frac{m}{n})$$

  Donde τ es la latencia en ciclos de procesador para poner un mensaje a los puertos de la red de interconexión y α el número de ciclos de procesador que tarda un elemento de 1 byte en transmitirse por la red. T_e es el tamaño de cada dato enviado en bytes. m es el número de elementos que se envían y n es el número total de elementos a ordenar.

  En esta tesis se utilizan las operaciones de comunicación colectiva *Transpose* , *All to All* y *Broadcast* de la librería MPI. En el *Transpose* cada procesador consigue una copia local de los datos que envían todos los procesadores en paralelo. La cantidad recibida por cada procesador es la misma para todos los procesadores. Cada procesador envía la misma cantidad de datos al resto de procesadores. Para modelarlas se ha incorporado el factor αT_e a los costes propuestos en [21];

Operación	CSE_{cal}	CSE_{comm}
Transpose m datos	$O(m/n)$	$\tau/n + \alpha T_e(mP - m)/n$
All to All n datos x procesador	$O(n/(nP))$	$\tau/n + \alpha T_e(n/P - \frac{n}{P^2})/n$
Broadcast s datos	$O(s/n)$	$2(\tau/n + \alpha T_e(s - s/P)/n)$

Tabla B.1: CSE de las operaciones *transpose*, *AlltoAll* y *broadcast* para P procesadores y n datos a ordenar.

además de expresar los costes en CSE y no en segundos. Con el factor αT_e se quiere tener en cuenta la topología de la red y el ancho de banda que ésta tiene. De esta forma se consigue que los modelos se aproximen más a la realidad.

Los CSE de las operaciones de comunicación colectiva utilizadas se muestran en la Tabla B.1

Nomenclatura

En la tabla B.2 se resume la nomenclatura básica que se utilizará para el modelo.

	Modelos
C_a	Latencia de acceso, en ciclos, al primer nivel de memoria cache
R_1	Núm. de datos (clave+puntero) por línea del primer nivel de memoria cache
R_2	Núm. de datos por línea del segundo nivel de memoria cache
R_3	Núm. de datos por línea del tercer nivel de memoria cache
R_{page}	Núm. de datos que caben en una página de memoria
C_{f1}	Penalización por fallo, en ciclos, del primer nivel de memoria cache
C_{f2}	Penalización por fallo del segundo nivel de memoria cache
C_{f3}	Penalización por fallo del tercer nivel de memoria cache
C_{fTLB}	Penalización en ciclos en caso de fallo de TLB
p_f	Probabilidad de fallo, por conflicto, accediendo al TLB
E_{TLB}	Número de entradas del TLB
τ	Latencia en ciclos de procesador para insertar un mensaje en la red
α	Número de ciclos de procesador que tarda un elemento de 1 byte en transmitirse por la red
T_e	Tamaño en bytes de un elemento a enviar

Tabla B.2: Nomenclatura básica de los modelos.