

## **Capítulo 6: EVALUACIÓN Y COMPARACIÓN DE MÉTODOS**

*En este capítulo, se realiza la evaluación y comparación de los métodos de paralelización de recurrencias descritos y propuestos en este trabajo. Los resultados obtenidos se basan en la aplicación de los algoritmos descritos a grafos aleatoriamente generados.*

*En la sección 1 se describen las características de los grafos generados de forma aleatoria.*

*En la sección 2 se estudia la variación del paralelismo y del tiempo de ejecución del código vectorizada en función del tipo de grafo, del número de nodos y de las distancias asociadas a los arcos de dependencias. Estas medidas son características del bucle aparalelizar o vectorizar.*

*En la sección 3 se evalúa el tiempo de ejecución que puede obtenerse con la generación de código propuesta para máquinas vectoriales y se compara con el tiempo evaluado en la sección anterior.*

*En la sección 4 se comparan los resultados obtenidos por los distintos métodos utilizados para paralelizar recurrencias.*

### **6.1 GENERACIÓN DE GRAFOS ALEATORIOS**

En esta sección se describen las características de los grafos aleatoriamente generados y que son base de las evaluaciones y comparaciones realizadas a lo largo de todo este capítulo.

Los grafos que se generan aleatoriamente pueden ser de 3 tipos:

- (a) con una única recurrencia hamiltoniana, en cuyo caso se cumple que

$$S_i \delta S_{i+1} \quad (1 \leq i < n) \quad \wedge \quad S_n \delta S_1.$$

- (b) hamiltonianos con arcos adicionales, siendo arcos adicionales aquellos no incluidos en la recurrencia hamiltoniana. Los arcos adicionales se generan obteniendo de forma equiprobable los nodos fuente y destino correspondientes.

Se denota con *aa* el número de arcos adicionales del grafo hamiltoniano.

- (c) no hamiltonianos, en cuyo caso todos los arcos se generan obteniendo de forma equiprobable los nodos fuente y destino correspondientes.

Se denota con *a* al número de arcos del grafo.

El número de nodos del grafo, que representa el número de sentencias del bucle, se denota con *n*. Los arcos, que representan las relaciones de dependencia, vienen caracterizados por:

- (a) La distancia asociada a los arcos  $d_{ij}$  tales que

$$S_i \delta S_j \quad \wedge \quad i < j$$

varía uniformemente entre 0 y *d*. Por lo tanto, el valor medio de la distancia asociada a dichos arcos es  $d / 2$ .

- (b) La distancia asociada a los arcos  $d_{ij}$  tales que

$$S_i \delta S_j \quad \wedge \quad i \geq j$$

varía uniformemente entre 1 y *d*. Por lo tanto, el valor medio de la distancia asociada a dichos arcos es  $(d+1) / 2$ .

- (c) Entre cada par de nodos del grafo sólo puede existir un arco de dependencias.

En las gráficas de este capítulo se representa el valor medio de determinadas características del bucle, obtenido al aplicar los algoritmos que

permiten evaluar dichas características sobre un número de grafos aleatoriamente generados (del orden de 1000).

## **6.2 VARIACIÓN DEL PARALELISMO Y TIEMPO DE EJECUCIÓN VECTORIAL**

En esta sección se estudia la variación del paralelismo y del tiempo de ejecución vectorial en función del tipo de grafo, número de nodos y arcos del grafo, así como de la distancia máxima asociada a los arcos de dependencias.

### **Variación del paralelismo de un GDC hamiltoniano en función del número de arcos adicionales**

Las figuras 6.1(a) y (b) muestran la variación del paralelismo de un GDC hamiltoniano en función del número de arcos adicionales. La figura 6.2 muestra la variación del paralelismo de un GDC hamiltoniano en función del número de nodos del grafo y de la distancia máxima asociada a los arcos del grafo.

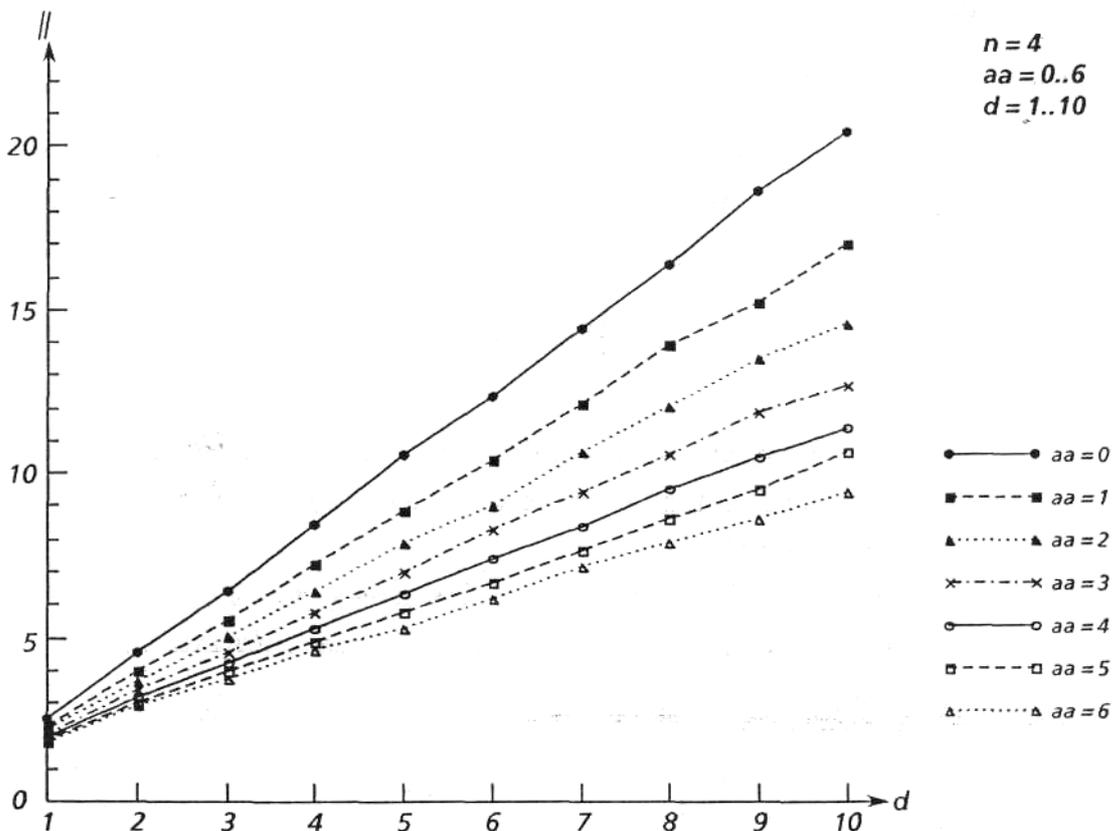


Figura 6.1: Paralelismo de GDC hamiltonianos en función del número de arcos adicionales y de la distancia máxima de los arcos para  $n = 4$  y  $n = 8$ .

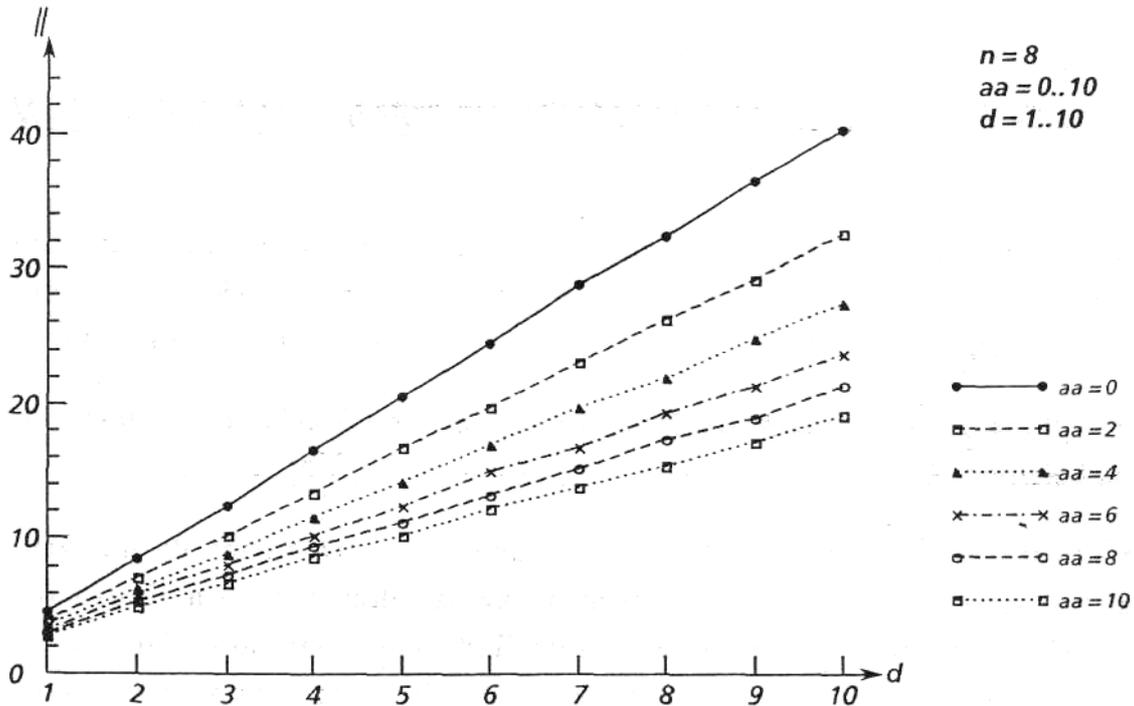


Figura 6.1 (cont): paralelismo de GDC hamiltonianos en función del número de arcos adicionales y de la distancia máxima de los arcos para  $n = 4$  y  $n = 8$ .

Las características básicas a resaltar son:

- (a) Si el grafo posee una única recurrencia hamiltoniana, es decir  $aa = 0$ , la variación del paralelismo en función de la distancia y del número de nodos puede caracterizarse por

$$\| = \frac{d}{2} \cdot n + 0.5$$

dado que

- el paralelismo viene dado por la suma de las distancias asociadas a los arcos que constituyen la recurrencia hamiltoniana y
- las distancias asociadas a los arcos del grafo han sido generadas siguiendo las hipótesis del apartado anterior.

Por lo tanto,

$$\| = \left( \sum_{i=1}^{n-1} \frac{d}{2} + \frac{d+1}{2} \right) = \frac{d}{2} \cdot n + 0.5 .$$

(b) El paralelismo del bucle disminuye a medida que aumenta el número de arcos adicionales  $aa$  en el grafo. Observar que la reducción del paralelismo es pequeña y que es necesario un número de arcos adicionales elevado (prácticamente superior al número de nodos del grafo) para reducir el paralelismo a la mitad.

La reducción del paralelismo con el número de arcos adicionales es debida a la aparición de nuevas recurrencias  $R_i$ , cada una con un paralelismo por sentencia  $\|_s(R_i)$  aleatorio, aunque todas ellas con una distribución aproximadamente igual. Dado que el paralelismo viene dado por

$$\| = n \cdot \min_{R_i} (\|_s(R_i))$$

se tiene que el valor medio del paralelismo cumple

$$\bar{\|} < n \cdot \min_{R_i} (\bar{\|}_s(R_i))$$

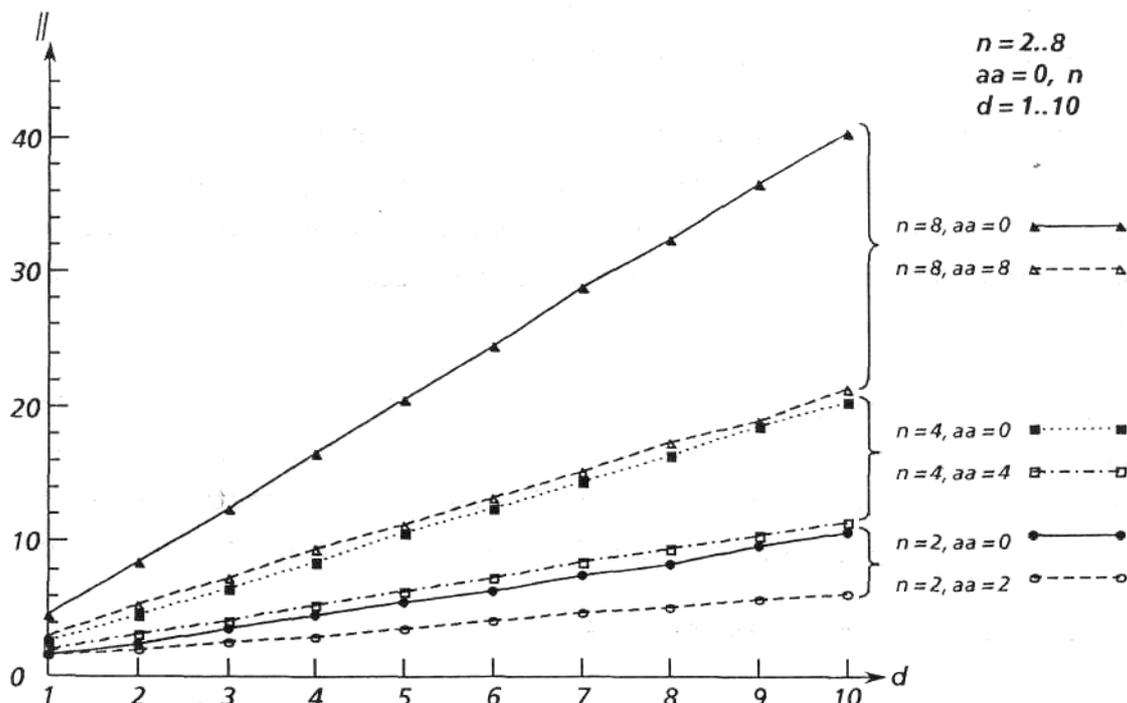


Figura 6.2: paralelismo de GDC hamiltonianos en función del número de nodos.

### **Tiempo de ejecución de bucles vectoriales con GDC hamiltonianos**

Como característica vectorial de un bucle se considera el tiempo de ejecución. Esta medida es más representativa de la eficiencia de la vectorización que la longitud vectorial media de las sentencias que componen el bucle.

La longitud vectorial media es un parámetro controlado por aquellas sentencias con longitud vectorial grande, mientras que el tiempo de ejecución viene es controlado por las sentencias de longitud vectorial menor.

Si se considera  $lv(S_i)$  la longitud vectorial asociada a la sentencia  $S_i$  se puede aproximar el tiempo de ejecución del bucle en forma vectorial por

$$t = \sum_{S_i \in V} \frac{N}{lv(S_i)} \cdot (t_c + t_e \cdot lv(S_i)) = n \cdot N \cdot t_e + t_c \cdot N \cdot \sum_{S_i \in V} \frac{1}{lv(S_i)}$$

siendo  $t_c$  el tiempo de carga de la unidad segmentada y  $t_e$  el tiempo de ciclo de una etapa.

Esta expresión es tanto más correcta cuanto mayor sea  $N$  ya que se hacen despreciables los efectos del prólogo y del epílogo.

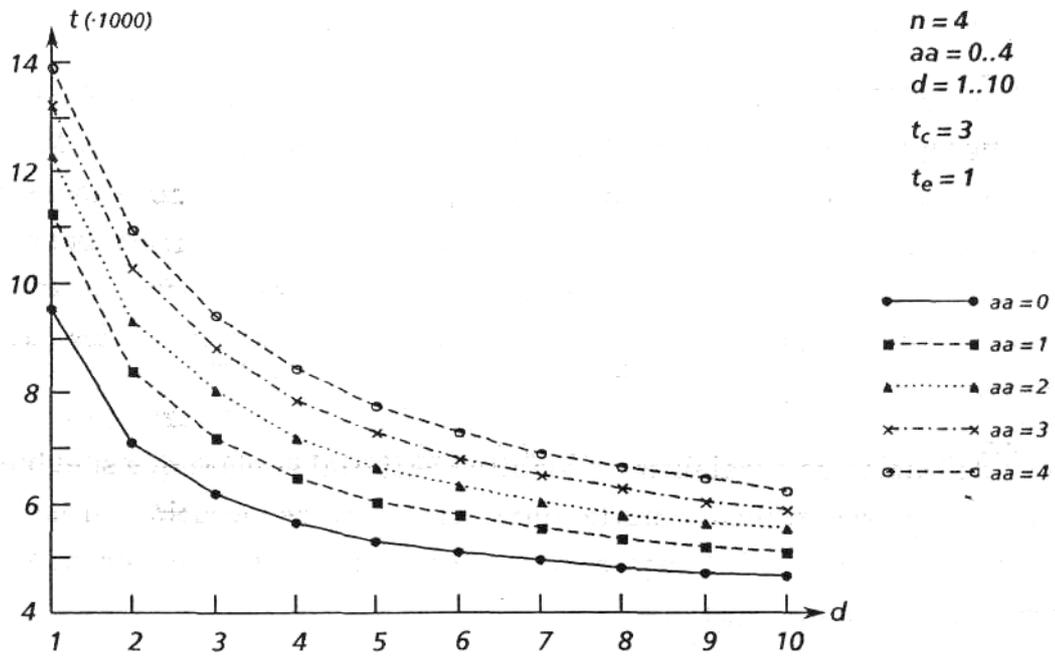
La expresión anterior es una cota inferior del tiempo de ejecución dado que no considera los posibles prólogos y epílogos secuenciales que se deben ejecutar antes de cualquiera de los bucles vectoriales internos generados. En algunos casos estos prólogos y epílogos internos no son necesarios y por lo tanto la expresión es correcta.

La figura 6.3(a) muestra la variación del tiempo de ejecución en función del número de arcos adicionales y de la máxima distancia asociada a los arcos del grafo para GDC hamiltonianos. La figura 6.3(b) muestra la misma característica para GDC en general en función del número de arcos del grafo.

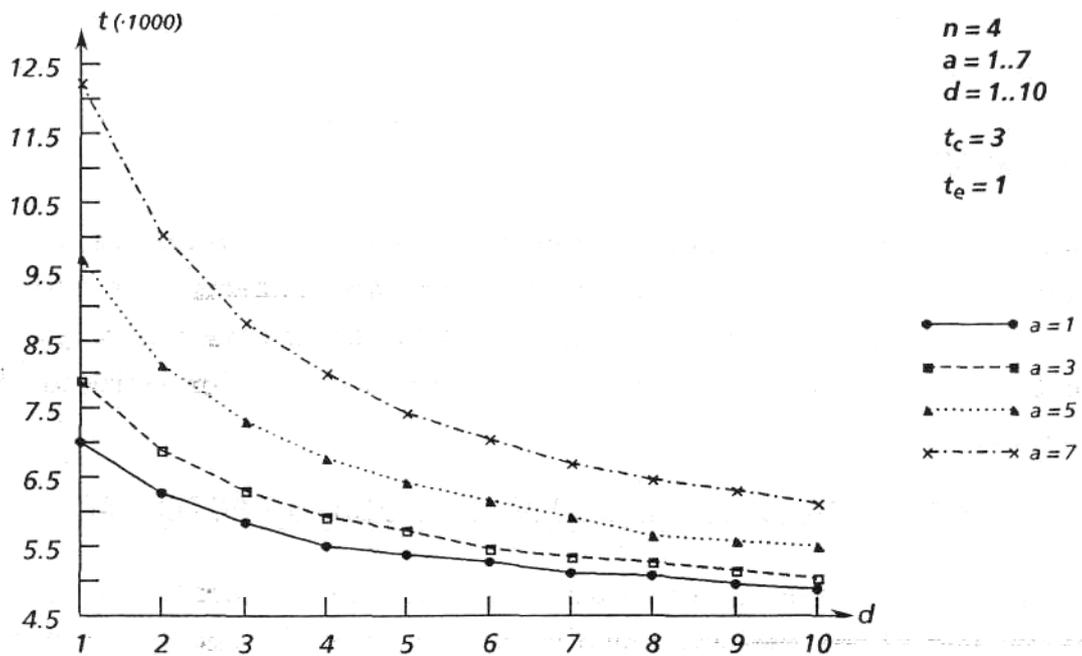
### **Multiprocesador vs. Máquina Vectorial**

Por último, nos interesa comparar la paralelización alcanzable con un multiprocesador y la que puede obtenerse con una máquina vectorial, realizando la evaluación cuantitativa de la flexibilidad de los dos tipos de máquinas.

Se compara para ello el tiempo requerido para ejecutar un bucle en una máquina MIMD (Múltiple Instruction Múltiple Data) y en una SIMD (Single



(a)



(b)

Figura 6.3: variación del tiempo de ejecución en función del número de arcos, (a) GDC hamiltonianos y (b) GDC en general.

Instruction Múltiple Data) [Fly72], suponiendo un número de procesadores infinito y no considerando los problemas de acceso a los datos. También se supone que el ciclo de procesador es el mismo en ambas.

La máquina SIMD puede ejecutar en un ciclo una única operación sobre vectores de cualquier número de elementos, mientras que la máquina MIMD puede ejecutar en el mismo tiempo cualquier número de operaciones no necesariamente idénticas. Esta flexibilidad adicional permite, en general, reducir el tiempo de ejecución de un bucle. Se pretende en este apartado comprobar hasta qué punto es apreciable esta diferencia para los grafos que se consideran.

De la definición de paralelismo y longitud Vectorial es claro que si el bucle tiene una única recurrencia hamiltoniana, el tiempo de ejecución en las dos máquinas es el mismo. En caso de un grafo general, y considerando las hipótesis anteriores, se tiene

$$t_{\text{MIMD}} = N \cdot \frac{n}{\parallel} \quad t_{\text{SIMD}} = \sum_{S_i} \frac{N}{lv(S_i)}$$

La figura 6.4 muestra los tiempos medios de ejecución considerando grafos con una recurrencia hamiltoniana y uno o dos arcos adicionales aleatoriamente generados. Puede observarse que el tiempo de ejecución en la máquina vectorial es sensiblemente superior al tiempo de ejecución en la máquina MIMD.

Aunque las hipótesis utilizadas no tienen en cuenta otros problemas reales que aumentarían el tiempo de ejecución en ambas máquinas, esta comparación da una idea cuantitativa de la diferencia de velocidad debida a la mayor o menor flexibilidad para adaptarse a un algoritmo de las dos estructuras consideradas.

### **6.3 EVALUACIÓN DE LA GENERACIÓN DE CÓDIGO VECTORIAL**

En esta sección se evalúan las alternativas propuestas en la sección 5.4.3 para la generación de código vectorial que no requieren prólogos y epílogos antes de la ejecución de los bucles vectoriales internos. Estas alternativas son:

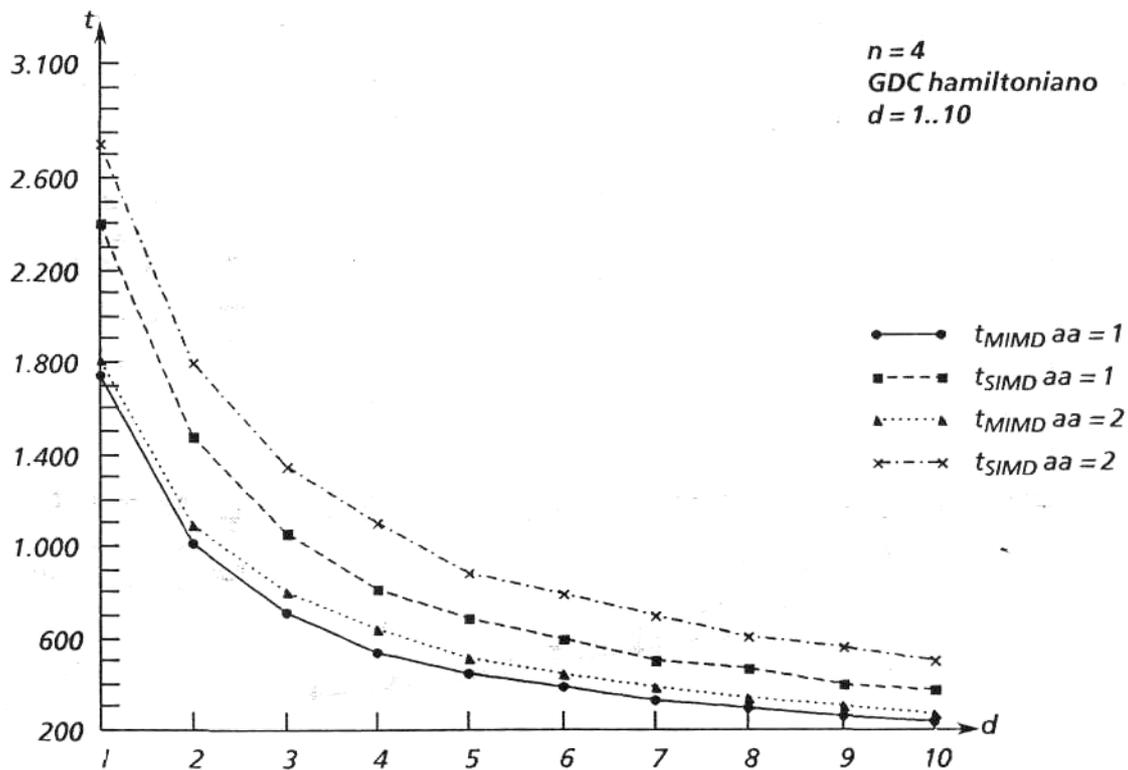


Figura 6.6: Comparación de los tiempos de ejecución entre MIMD y SIMD para GDC hamiltonianos con aa = 1 y aa = 2.

(a) asociar a cada sentencia  $S_i$  la máxima longitud vectorial posible  $lv_i'$  que cumpla la condición

$$lv_i' = K \cdot \min_{S_j \in V} lv(S_j) \quad \wedge \quad lv_i' \leq lv(S_i)$$

En este caso el tiempo de ejecución viene dado por

$$t' = n \cdot N \cdot t_e + t_c \cdot N \cdot \sum_{S_i \in V} \frac{1}{lv_i'}$$

(b) asociar a todas las sentencias del bucle la mínima de las longitudes vectoriales asociadas a las sentencias es decir,

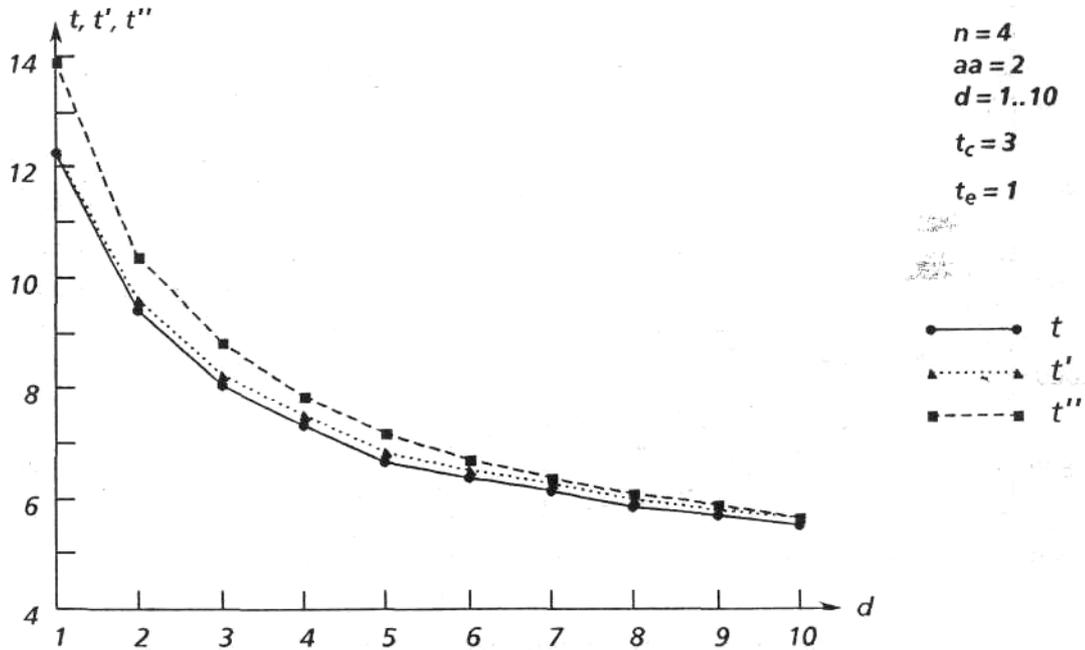
$$lv_{\min} = \min_{S_i \in V} lv(S_i).$$

En este caso, el tiempo de ejecución viene dado por

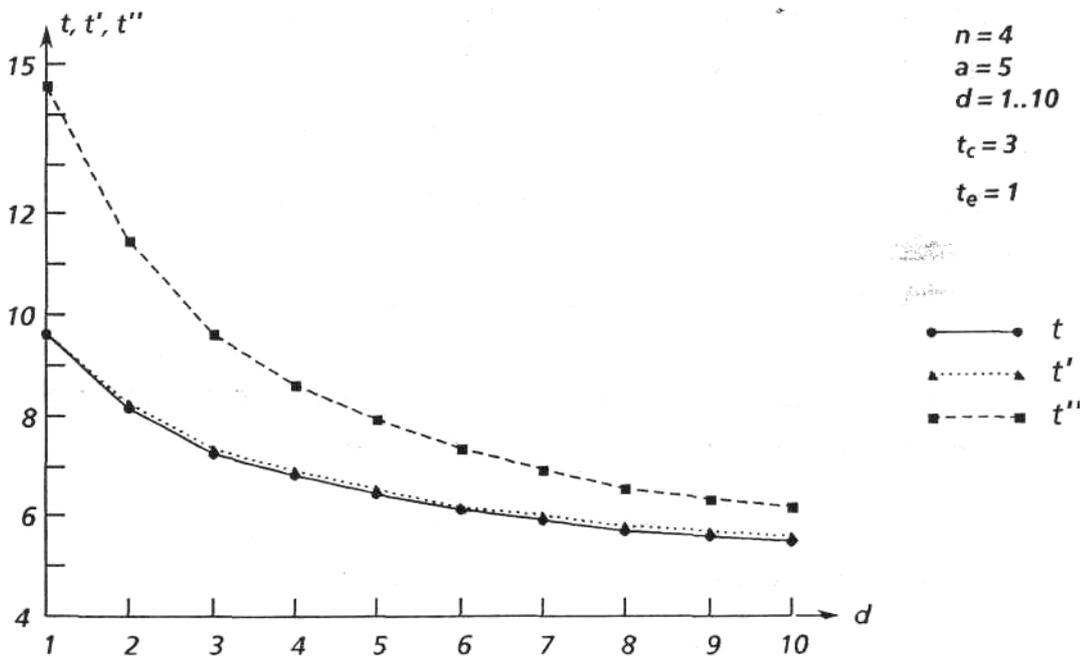
$$t'' = n \cdot N \cdot \left( t_e + \frac{t_c}{lv_{\min}} \right),$$

La figura 6.5(a) muestra los tiempos de ejecución  $t$  (cota descrita en el la sección anterior),  $t'$  y  $t''$  para GDC hamiltonianos con 4 nodos en función de la

distancia máxima asociada a los arcos de dependencias. Observar que el incremento en el tiempo de ejecución en cualquier caso es pequeño. Para el caso de GDC en general, tal y como muestra la figura 6.5(b) se tiene que la diferencia entre  $t''$  y las otras dos características es mayor.



(a)



(b)

Figura 6.5: tiempos de ejecución  $t$ ,  $t'$ ,  $t''$  considerando las tres posibles formas de generar código vectorial.

## **6.4 COMPARACIÓN DE MÉTODOS DE PARALELIZACIÓN DE RECURRENCIAS**

En esta sección se compara GTS con los métodos descritos en el capítulo 2 dedicados a paralelizar recurrencias (Particionado Parcial y Contracción de Ciclos). Esta comparación se realiza sobre grafos generados de forma aleatoria, tal como se describe en la sección 6.1.

En primer lugar se estudia el comportamiento de los métodos en cuanto al paralelismo que se obtiene en función de los parámetros que caracterizan a los grafos de dependencias.

En segundo lugar se realiza un estudio comparativo de la eficiencia obtenida al aplicar estos métodos a la paralelización de recurrencias en bucles con unas características concretas.

En esta sección también se realiza la evaluación del compromiso entre paralelismo que GTS es capaz de obtener y la sincronización necesaria a introducir.

### ***Particionado Parcial***

El paralelismo obtenido al aplicar Particionado Parcial viene determinado por el máximo común divisor  $g$  de las distancias distintas de cero asociadas a los arcos del grafo de dependencias.

La figura 6.6 muestra el paralelismo y la eficiencia del proceso de paralelización obtenidos, en función del número de nodos y de la máxima distancia asociada a los arcos, al aplicar Particionado Parcial a grafos con una única recurrencia hamiltoniana.

Las características básicas a resaltar son:

- el paralelismo obtenido es cercano a 1 y por lo tanto la eficiencia muy baja, dado que al generar las distancias de forma aleatoria es fácil que estas posean un máximo común divisor igual a la unidad.
- el paralelismo decrece al aumentar el número de nodos del grafo en cuanto a que existen más arcos y se hace más probable que el máximo común divisor de las distancias asociadas sea la unidad.

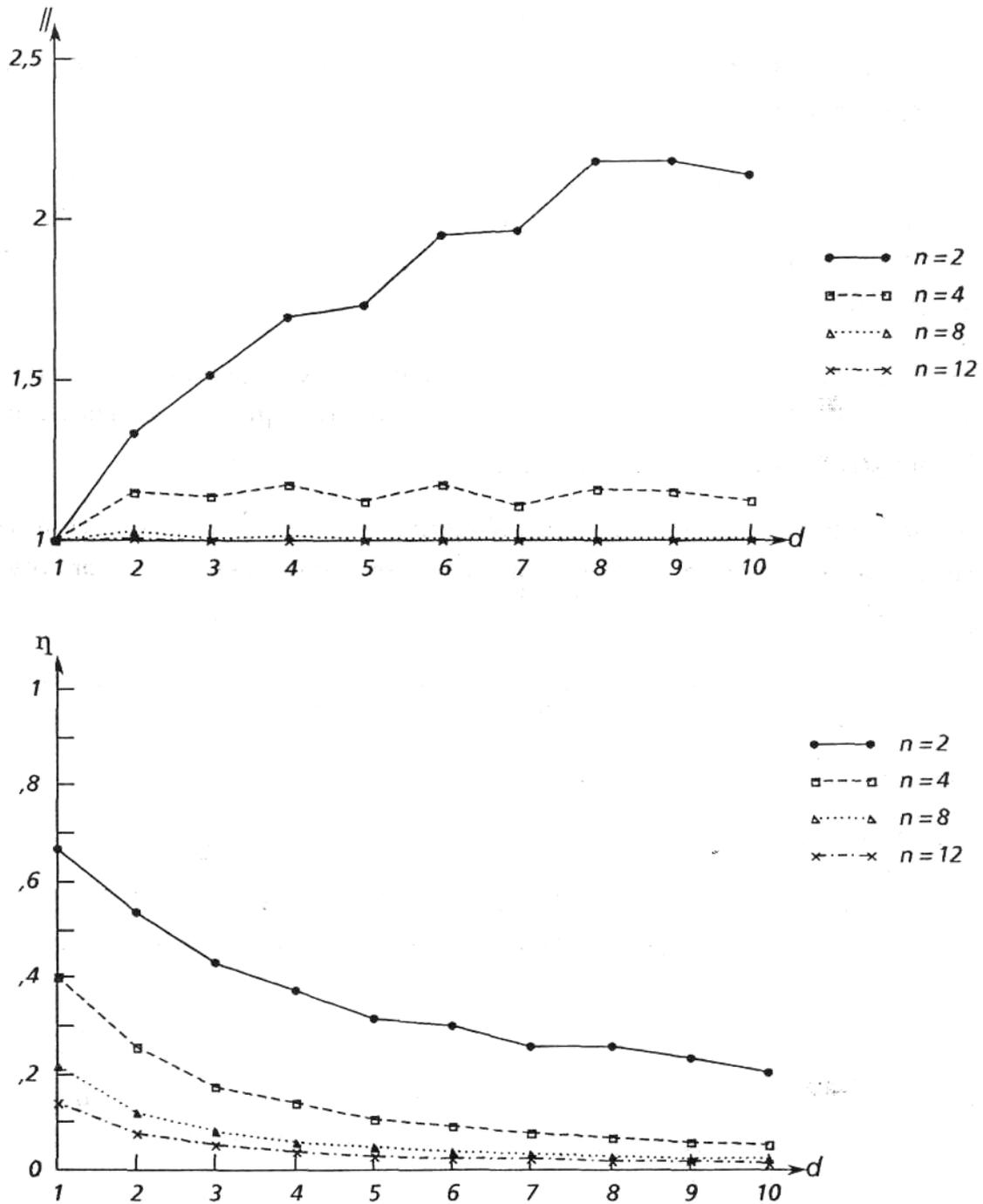


Figura 6.6: Paralelismo y eficiencia obtenidos al aplicar Particionado Parcial a GDC con una única recurrencia hamiltoniana.

### Contracción de Ciclos

El paralelismo obtenido al aplicar Contracción de Ciclos viene determinado por:

- el mínimo  $\lambda$  de las distancias distintas de cero asociadas a los arcos del grafo de dependencias.

- la longitud  $k$  de la cadena de dependencias del grafo más larga con peso cero.

En general, el paralelismo viene dado por

$$\parallel = \frac{m \cdot n}{k}$$

La figura 6.7 muestra el paralelismo y la eficiencia del proceso de paralelización obtenidos, en función del número de nodos y la máxima distancia asociada a los arcos, al aplicar Contracción de Ciclos a grafos con una única recurrencia hamiltoniana.

Las características básicas a resaltar son:

- el paralelismo crece con la distancia máxima de los arcos dado que es más probable obtener mínimos mayores cuanto mayor sea la distancia máxima.
- el paralelismo crece con el número de nodos, pero no de forma proporcional. En principio, al haber más nodos y arcos en el grafo es más probable obtener, por un lado, mínimos inferiores y por otro cadenas de dependencia con peso asociado cero más largas.
- la eficiencia del proceso de paralelización decrece al aumentar el número de nodos y la distancia máxima asociada a los arcos del grafo.

### **Graph Traverse Scheduling**

El paralelismo obtenido por GTS para grafos con una única recurrencia hamiltoniana viene determinado por el peso de la recurrencia.

La figura 6.8 muestra el paralelismo obtenido por GTS para este tipo de grafos, en función del número de nodos y la distancia máxima asociada a los arcos del grafo.

Las características básicas a resaltar son:

- La eficiencia obtenida en el proceso de paralelización al aplicar GTS es la unidad, dado que es capaz de extraer todo el paralelismo del bucle.

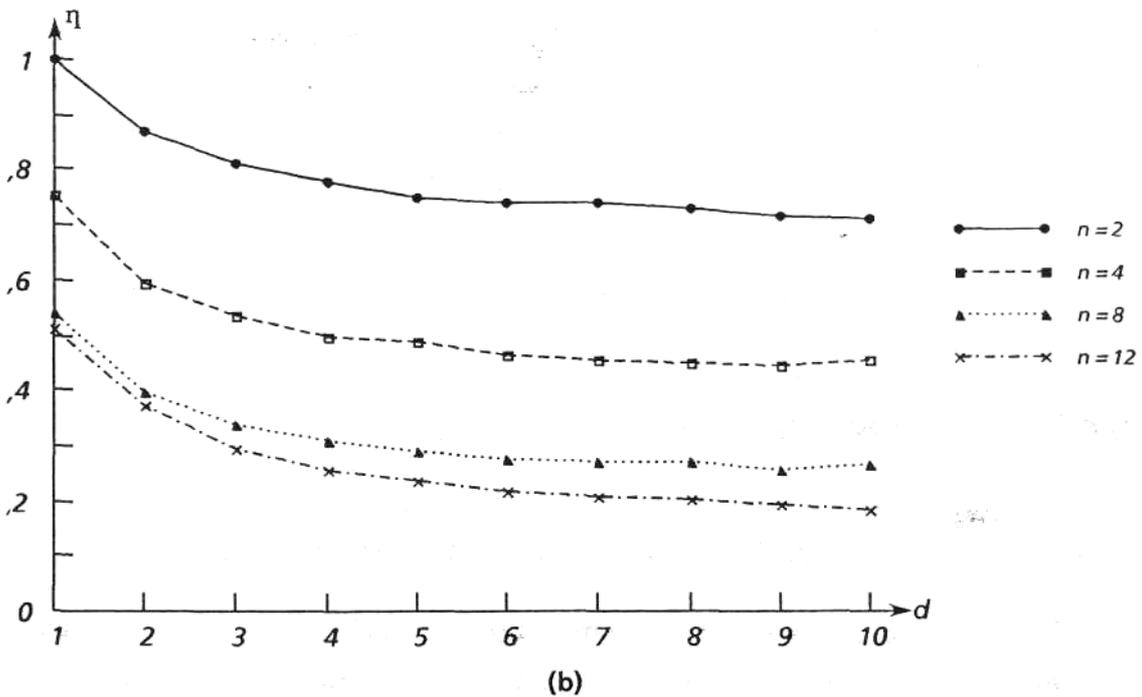
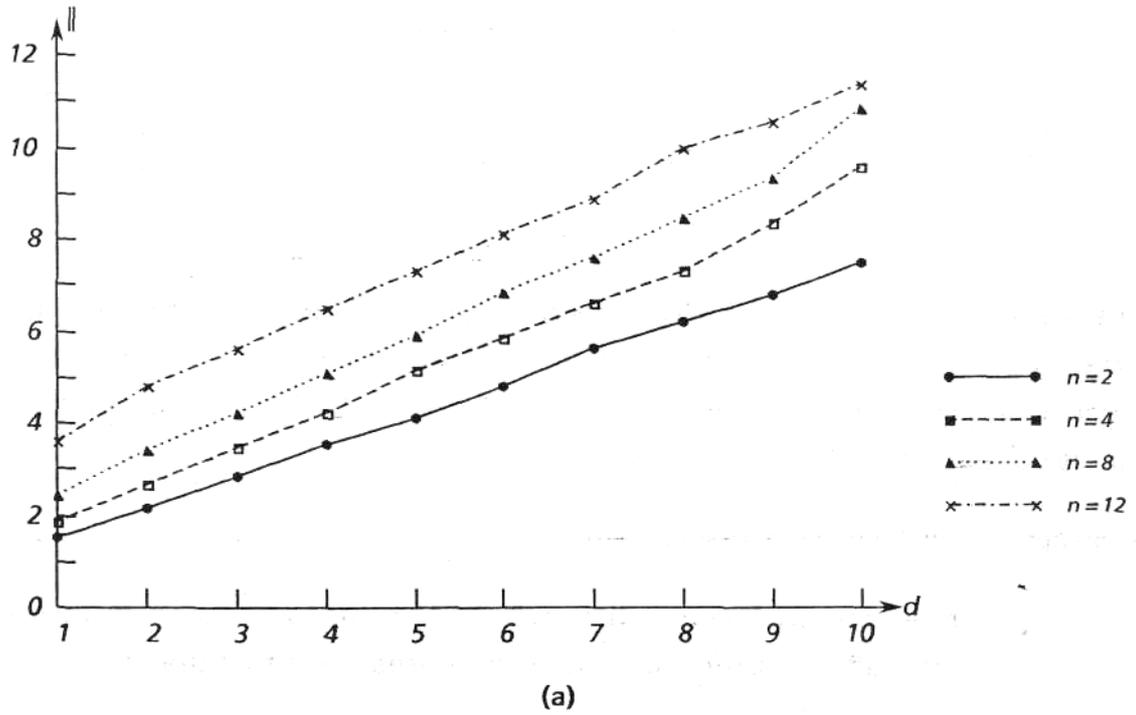


Figura 6.7: Paralelismo y eficiencia obtenidos por Contracción de Ciclos para GDC hamiltonianos con  $aa = 0$ .

- El paralelismo es proporcional a la máxima distancia asociada a los nodos del grafo.
- El paralelismo es proporcional al número de nodos del grafo de dependencias.

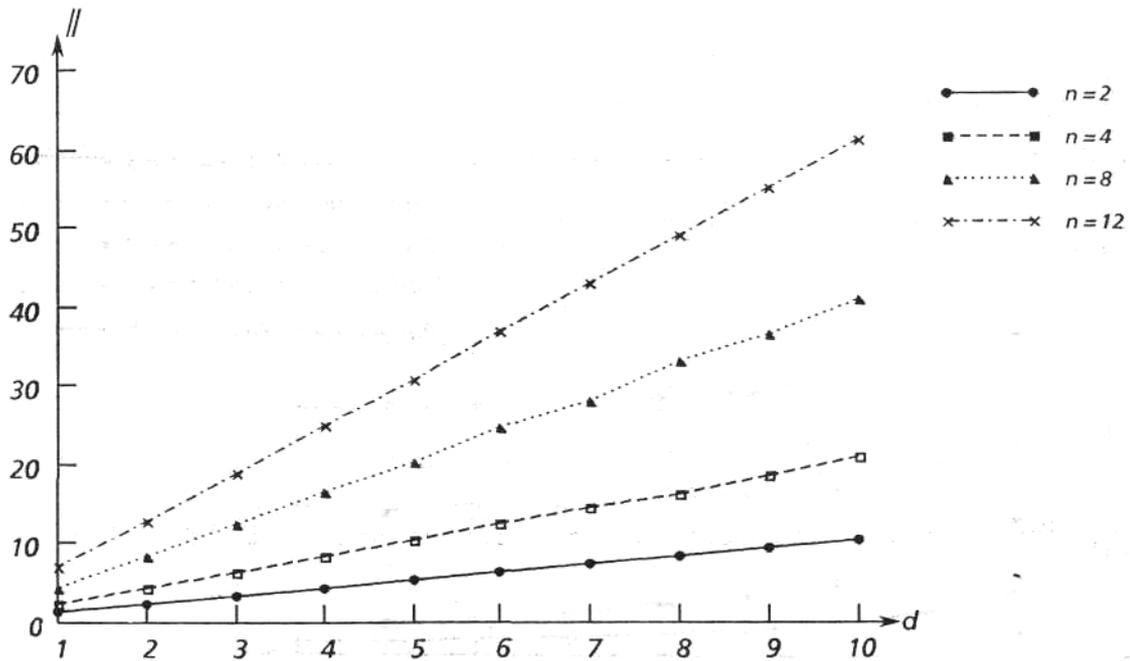


Figura 6.8: Paralelismo obtenido por GTS para GDC hamiltonianos con aa = 0.

### Comparación de métodos

La figura 6.9 muestra una gráfica comparativa de la eficiencia de los procesos de paralelización indicados para grafos con una única recurrencia hamiltoniana, en función de la distancia y para los casos particulares de 4 y 8 nodos. Observar que Contracción de Ciclos obtiene más paralelismo que Particionado Parcial, aunque GTS obtiene mejores resultados que los dos anteriores.

La figura 6.10 muestra el paralelismo obtenido por los distintos métodos considerados en el caso de grafos cíclicos hamiltonianos en función de la distancia máxima. En este caso se han considerado grafos con 8 nodos y 2 arcos adicionales. Observar que GTS es capaz de obtener prácticamente el doble de paralelismo que Contracción de Ciclos. En general, Particionado Parcial no es capaz de obtener paralelismo del bucle.

En la figura 6.11 también se muestra el paralelismo sin sincronización obtenido por GTS. En general es superior al paralelismo obtenido por Particionado Parcial aunque en cualquier caso es muy bajo. Este método puede ser válido para determinados grafos de dependencia aunque en general no es así.

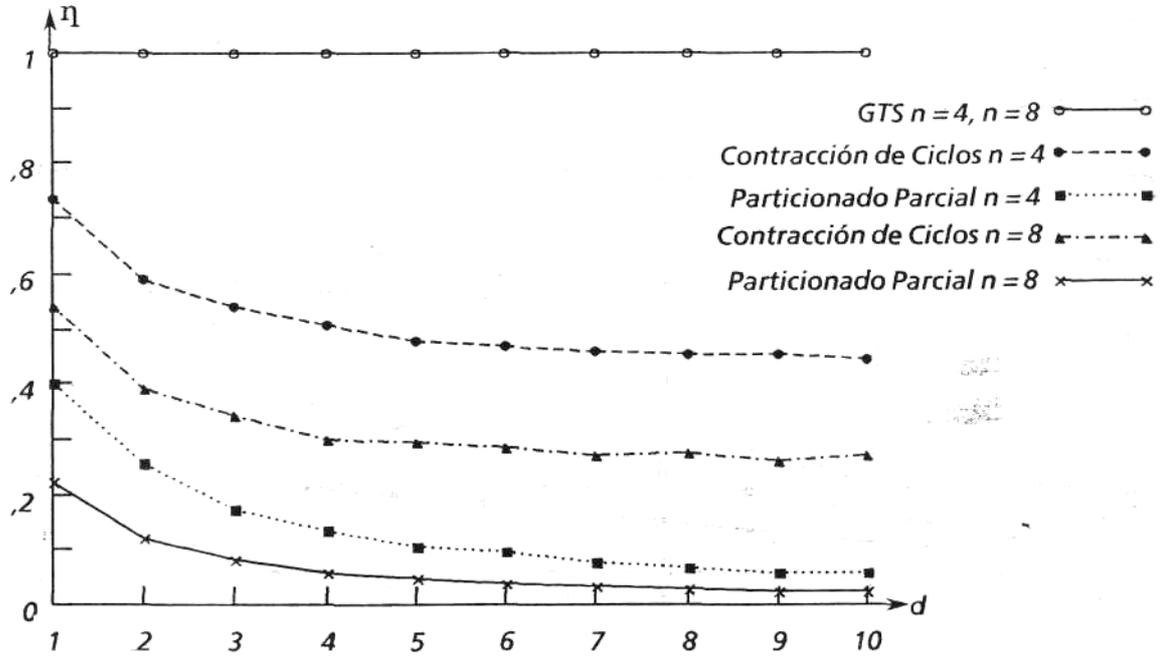


Figura 6.10: Comparación de la eficiencia obtenida por Particionado Parcial, Contracción de Ciclos y GTS ( $n = 4$  y  $n = 8$ ).

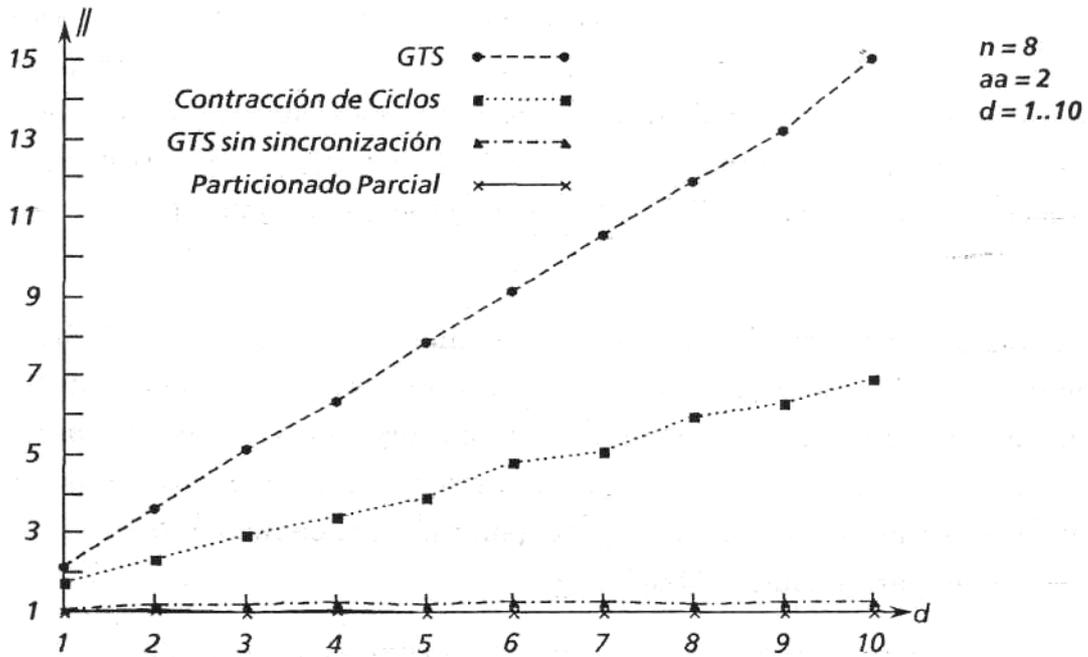


Figura 6.11: Paralelismo obtenido por Particionado Parcial, Contracción de Ciclos y GTS para GDChamiltonianos.

**Paralelismo vs. Sincronización**

En este apartado se estudia el compromiso entre paralelismo obtenido y sincronización requerida al aplicar GTS. La reducción del número de sincronizaciones explícitas provoca una disminución del paralelismo obtenido, dado que determinadas operaciones que se pueden ejecutar de forma paralela sincronizada se ejecutan de forma secuencial.

En la figura 6.12 se muestra el porcentaje de paralelismo obtenido al variar el número de arcos explícitamente sincronizados respecto al paralelismo del grafo. Se consideran GDC hamiltonianos con 4 nodos.

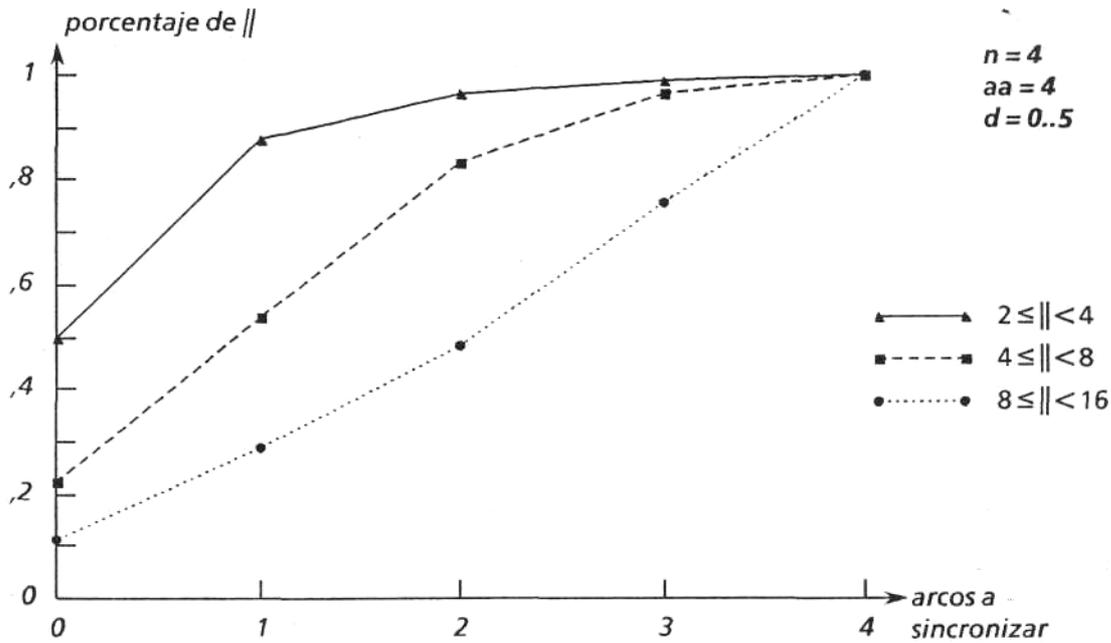


Figura 6.12: Compromiso entre paralelismo y sincronización.

Las características a resaltar son:

- el paralelismo disminuye al eliminar sincronizaciones del grafo; la penalización que provoca la reducción del número de sincronizaciones explícitas depende también del paralelismo original, siendo ésta más acusada cuando el paralelismo es grande.
- en muchos casos sincronizar uno o dos arcos es suficiente para conseguir un porcentaje de paralelismo aceptable.