

Computación de Altas Prestaciones

Ejercicios de repaso Tema 1 y 2

1. Explique bajo qué condiciones de funcionamiento de un sistema con procesamiento en paralelo se dan los comportamientos de aceleración que se corresponden con la ley de Amdahl y con la ley de Gustafson.

Speed-up de tiempo fijo, carga de trabajo aumenta con el número de procesadores(Ley de Gustafson)

Speed-up de carga de trabajo fija, El tiempo de ejecución disminuye con el número de procesadores pero esta limitado por el trabajo serie (Ley de Amdahl)

b) Formule ambas situaciones y explique qué representa cada parámetro en cada una de las leyes.

Amdahl $S = p/(1+fs*(p-1))$ si p muy grande $S = 1/fs$

Gustafson $S = p + (1-p)*fs$

S = Speed-up o aceleración = tiempo serie/ tiempo paralelo.

Número de procesadores es p

La fracción de trabajo realizada en serie es fs

En Amdahl el tiempo de trabajo en paralelo en un procesados es (1-fs) y si se realiza con p procesadores es (1-fs)/p

En Gustafson el tiempo de trabajo en paralelo con p procesadores es (1-fs) y si se hace en un procesador es p*(1-fs)

c) Suponga que un sistema con P=60 procesadores cuando funciona bajo las condiciones de la ley de Amdahl tiene una aceleración de 15. Se consigue cambiar su modelo de funcionamiento a las condiciones de Gustafson y se obtiene una aceleración de 58. Calcule el comportamiento de la fracción serie en ambos comportamientos y la aceleración obtenida cuando P sea muy grande para ambos casos.

$$\text{Amdahl } S = p/(1+fs*(p-1)) \quad 15 = 60/(1+59fs) \quad 1+59fs=4 \quad fs = 3/59 = 0,05$$

fs = 5% en Amdahl fs permanece constante

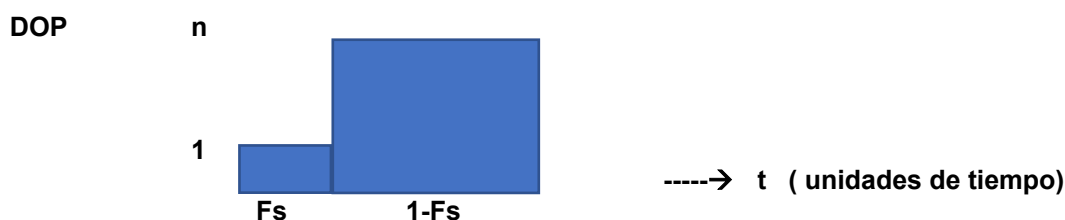
Gustafson

$$S = p + (1-p)*fs \quad 58 = 60 - 59fs \quad fs = 2/59 = 0,034$$

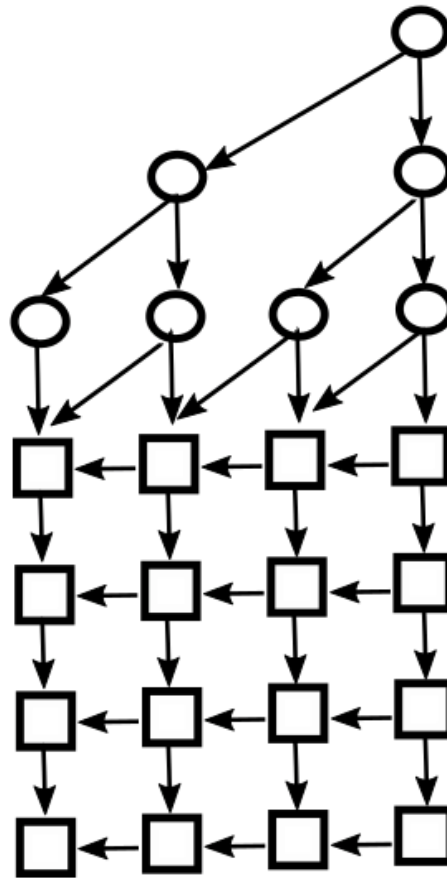
En gustafson conforme p aumente fs tiende a cero

d) Represente el perfil de paralelismo que corresponde exactamente a la descripción de la aceleración de acuerdo a la ley de Amdahl.

Por ejemplo para un programa con trabajo serie W1 durante un tiempo Fs y con trabajo Wn totalmente paralelizable entre n procesadores durante el resto de tiempo. Sin que exista grado de paralelismo entre 2 y n-1, se da la situación que corresponde a la ley de Amdahl.



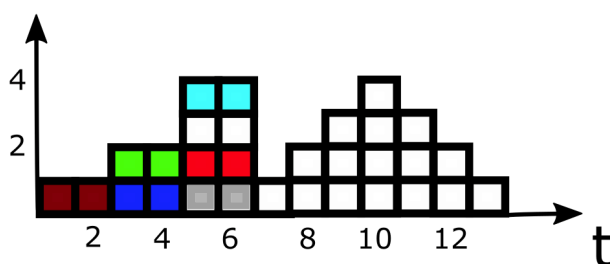
2. La figura representa el grafo de dependencia de una aplicación que se va a ejecutar en un sistema multiprocesador con p procesadores. Se quiere generalizar denominando N al número de nodos en el grafo y n una potencia de 2, de tal manera que hay $N = (2^{(n-1)} - 1) + n^2$ nodos. En la figura se representa para n=4.



Se supone que no hay penalización por las comunicaciones y que los **nodos circulares** representan una tarea que tarda **2 unidades de tiempo** y los nodos **cuadrados** tardan **1 unidad** de tiempo.

- a. Represente el perfil de paralelismo durante la ejecución de la aplicación para el número de nodos representados (n=4).

SOL



- b. El grado de paralelismo máximo y el trabajo realizado con grado de paralelismo máximo.

DOP max =4

W8 = 12 Δ

- c. Calcule el speedup (S) y la eficiencia (E) en función del número de procesadores $S(p)$ y compare para este caso los valores $S(\infty)$, $S(p=2)$ y $S(p=4)$

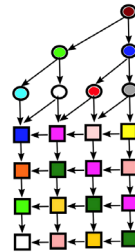
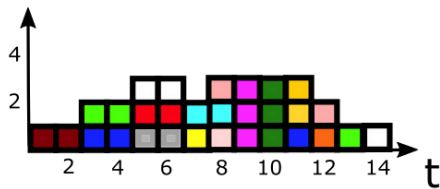
$$S(\infty) = S(4) = (4+8+6+12) / (4+4+2+3) = 30/13 = 2,3 \mid$$

$$E(4) = 30/4 \cdot 13 = 0,57$$

$$S(2) = 30 / (4+4+4+6) = 30/18 = 1,66 \quad | \quad E(2) = 30/36 = 0,83$$

- d. Represente la planificación de tareas mas efectiva posible para $p = 3$ procesadores y calcule la eficiencia de esta planificación.

$$S(3) = 30/14 = 30/14 = 2,14 \quad | \quad E(3) = 30/42 = 0,71$$



3. Dado el siguiente bucle, indique cómo ejecutarlo en paralelo de la manera más eficiente posible. Escriba el código resultante.

```
do i = 2, 1001
  (1) A(i) = C(i) + E(i-1) - 10
  (2) B(i+1) = C(i-2) + alfa
  (3) C(i-1) = D(i+1) * 3
  (4) E(i) = E(i-1) + A(i-1)
enddo
```

Nota: alfa es un valor constante

a. Se pide analizar las dependencias entre las instrucciones y generar el grafo de dependencias y el espacio de iteraciones.

RAW de (1) a (4) en A de distancia 1.
 WAR de (1) a (3) en C de distancia 1.
 RAW de (3) a (2) en C de distancia 1.
 RAW de (4) a (1) en E de distancia 1.
 RAW de (4) a (4) en E de distancia 1.

b. Escriba el código (pseudocódigo) resultante de la paralelización más eficiente posible.

Sin peeling

```
do i = 2, 1001
  (1) A(i) = C(i) + E(i-1) - 10
  (4) E(i) = E(i-1) + A(i-1)
enddo
doall i = 2, 1001
  (3) C(i-1) = D(i+1) * 3
enddoall
barrier
doall i = 2, 1001
  (2) B(i+1) = C(i-2) + alfa
enddoall
```

Con peeling

```
do i = 2, 1001
  (1) A(i) = C(i) + E(i-1) - 10
  (4) E(i) = E(i-1) + A(i-1)
enddo
B(3)=C(0)+alfa
doall i = 2, 1000
  (3) C(i-1) = D(i+1) * 3
  (2) B(i+2) = C(i-1) + alfa
enddoall
C(1000)=D(1002)*3
```

- c. Si el tiempo de ejecución de cada instrucción es $T_{inst} = T$, el tiempo de lanzamiento de hilos es $T_{fork} = T$, el tiempo de destruir los hilos es $T_{join} = T$ y el tiempo de sincronización es $T_{barr} = 1.98T$, haga una estimación del factor de aceleración y de la eficiencia que se conseguirá al utilizar 100 procesadores.

$$T_s = 4 \cdot 1000T = 4000T$$

Sin peeling:

$$\begin{aligned} T_p(100) &= 2 \cdot 1000T_{inst} + T_{fork} + 2 \cdot \frac{1000}{100}T_{inst} + T_{join} + T_{barr} = \\ &= 2000T + T + 20T + T + 1.98T \\ &= 2023.98T \end{aligned}$$

Con peeling

$$\begin{aligned} T_p(100) &= 2 \cdot 1000T_{inst} + T_{fork} + T_{inst} + 2 \cdot \frac{999}{100}T_{inst} + T_{inst} + T_{join} = \\ &= 2000T + T + T + 2 \cdot \frac{999}{100}T + T + T \\ &= 2023.98T \end{aligned}$$

$$S(100) = \frac{T_s}{T_p(100)} = \frac{4000}{2023.98} = 1.976304; E = \frac{S}{100} = 1.976304\%$$

- d. ¿Cuál es la máxima aceleración alcanzable con tantos procesadores como se necesite? Puede asumir para este apartado $T_{barr} = T_{join} = T_{fork} = 0$. Justifique los resultados.

$$\begin{aligned} T_p(\infty) &\approx 2 \cdot 1000T_{inst} \\ S(\infty) &= 2 \end{aligned}$$