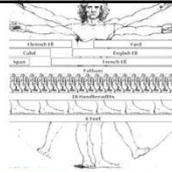


Early Adopters Awarded
Fall 2011 (NSF/IEEE)



Programación Paralela y Distribuida

Evaluación y Rendimiento de las Aplicaciones Paralelas

Trabajo Práctico N° 3

Ejercicio 1



Laboratorio de Investigación en
Cómputo Paralelo/Distribuido



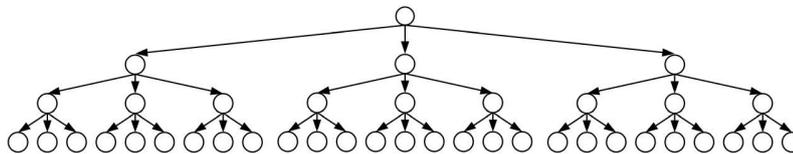
UNCUYO
UNIVERSIDAD
NACIONAL DE CUYO



FACULTAD
DE INGENIERÍA

TP3: Programación y Rendimiento

Ejercicio N°1 – En la siguiente figura se presenta un grafo de dependencias. Se asume que la ejecución de cada tarea insume una unidad de tiempo y que la comunicación entre procesos es instantánea (es decir que no consume tiempo).

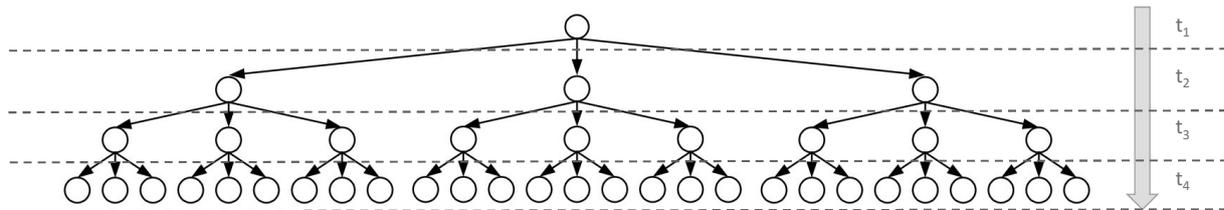


Se proponen las siguientes actividades:

- Calcular el grado de concurrencia.
- Calcular el máximo Speedup posible si se dispone de una cantidad ilimitada de nodos computacionales.
- Calcular Speedup y Eficiencia si el número de nodos computacionales es igual al grado de concurrencia.
- Calcular Speedup y Eficiencia si el número de nodos computacionales es igual a la mitad del grado de concurrencia.
- Calcular Speedup y Eficiencia si el número de nodos computacionales es igual a la tercera parte del grado de concurrencia.
- Graficar el factor de Speedup y explicar los resultados obtenidos.

TP3 – Ejercicio 1

a) Calcular el grado de concurrencia

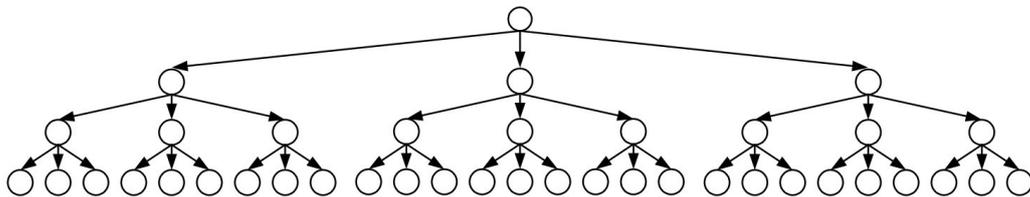


Para pensar: Cantidad de tareas concurrentes en cada instante:

- t_1 : 1
- t_2 : 3
- t_3 : 9
- t_4 : 27

TP3 – Ejercicio 1

b) Calcular el máximo Speedup posible si se dispone de una cantidad ilimitada de nodos computacionales

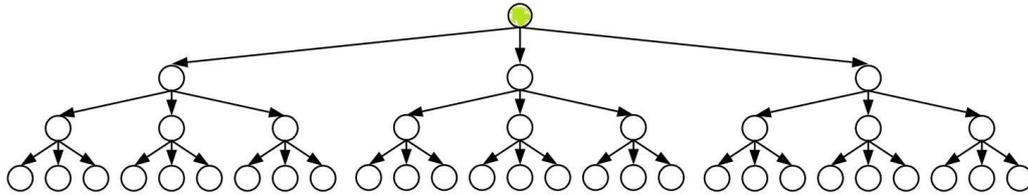


Para pensar: Granularidad, acomplamiento, dependencias, grado máximo de concurrencia, ...

- Límite de speedup: $\lim_{n \rightarrow \infty} \text{Speedup}(n) = 1/f$ donde f representa...
 - La sección no paralelizable
- ¿Cómo calculamos f?
 - ...

TP3 – Ejercicio 1

b) Calcular el máximo Speedup posible si se dispone de una cantidad ilimitada de nodos computacionales

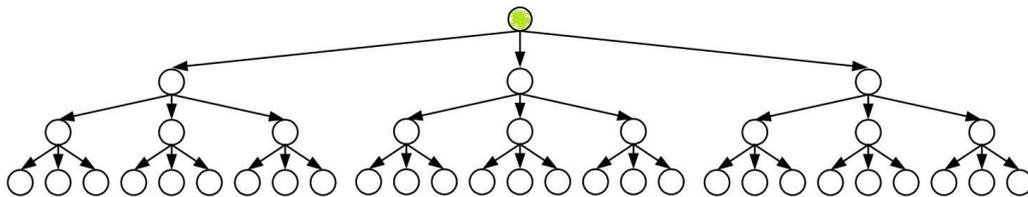


Para pensar: Granularidad, acomplamiento, dependencias, grado máximo de concurrencia, ...

- Límite de speedup: $\lim_{n \rightarrow \infty} \text{Speedup}(n) = 1/f$ donde f representa...
 - La sección no paralelizable
- ¿Cómo calculamos f?
 - Para este caso $f=1/40$

TP3 – Ejercicio 1

b) Calcular el máximo Speedup posible si se dispone de una cantidad ilimitada de nodos computacionales



Para pensar: Granularidad, acomplamiento, dependencias, grado máximo de concurrencia, ...

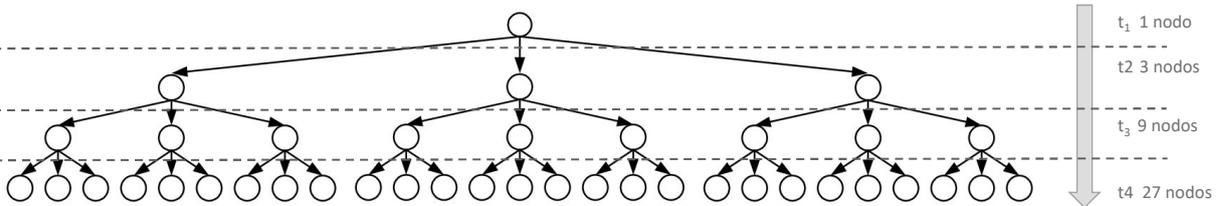
- Límite de speedup: $\lim_{n \rightarrow \infty} \text{Speedup}(n) = 1/f$ donde f representa...
 - La sección no paralelizable
- ¿Cómo calculamos f?
 - Para este caso $f=1/40$ y por lo tanto el límite será $1/(1/40)=40$
 - ¿Qué nos sugiere este valor?

La máxima aceleración que podamos alcanzar con cierta cantidad de nodos, será a lo sumo 40

$$Speedup(n) = \frac{T(1)}{T(n)}$$

TP3 – Ejercicio 1

c) Calcular Speedup y Eficiencia si n es el grado de concurrencia ($n=27$)



Para pensar: Cantidad de nodos ociosos en cada instante... ¡y su impacto!

- Para calcular el Speedup(27) debemos calcular primero sus parámetros...
 - $T(1)=40$
 - $T(27)=4$
- $Speedup(27)=40/4=10$
 - ¡Está lejos del máximo teórico!

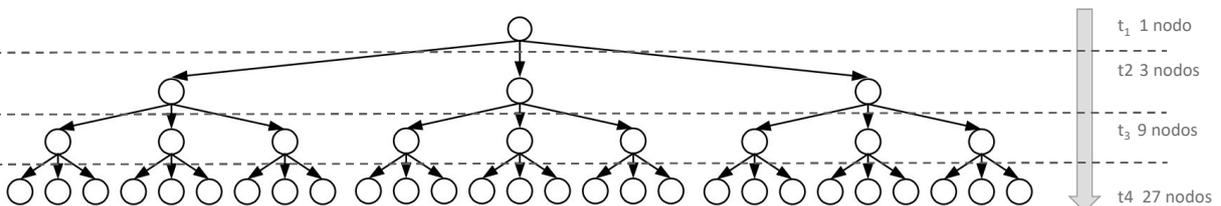
$$Speedup(n) = \frac{T(1)}{T(n)}$$

TP3 – Ejercicio 1

$$Eficiencia = \frac{T(1)}{T(n) \times n}$$

$$Eficiencia = \frac{Speedup(n)}{n} \times 100\%$$

c) Calcular Speedup y Eficiencia si n es el grado de concurrencia ($n=27$)



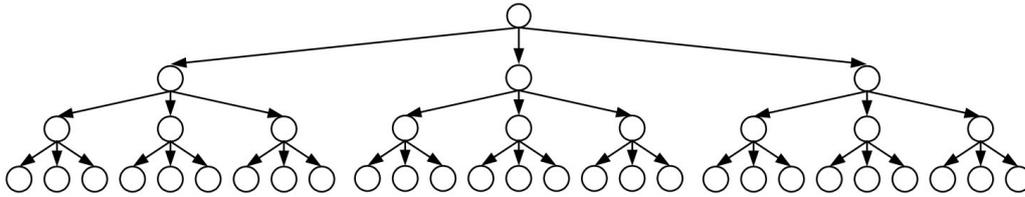
Para pensar: Cantidad de nodos ociosos en cada instante... ¡y su impacto!

- Sabemos que $T(1)=40$ y $T(27)=4$
- $Eficiencia(27)=40/(4 \times 27)=40/108=0,37$
- $Eficiencia(27)=(10/27) \times 100=0,37 \times 100=37\%$
 - ¡El aprovechamiento de los recursos disponibles es bastante pobre!



TP3 – Ejercicio 1

d) Calcular Speedup y Eficiencia si n es la mitad del grado de concurrencia ($n=14$)

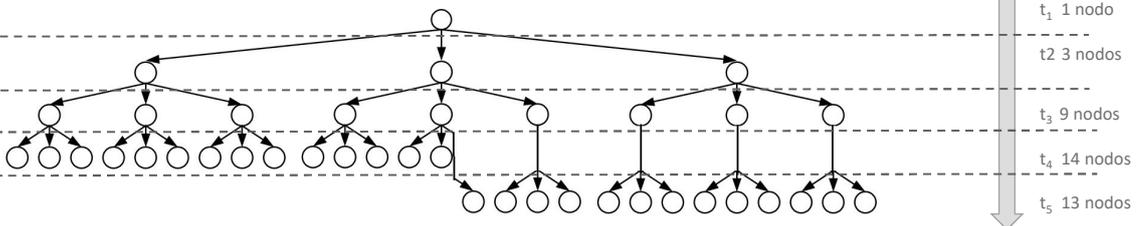


Para pensar: ¿Podemos ejecutar en paralelo todas las tareas en todos los niveles?

$$\text{Speedup}(n) = \frac{T(1)}{T(n)}$$

TP3 – Ejercicio 1

d) Calcular Speedup y Eficiencia si n es la mitad del grado de concurrencia ($n=14$)



- Para calcular el Speedup(14) debemos calcular primero sus parámetros... $T(1)=40$
 - $T(14)=5$
- $\text{Speedup}(14)=40/5=8$
 - ¡Está lejos del máximo teórico! Y no hay mayor diferencia cuando $n=27$

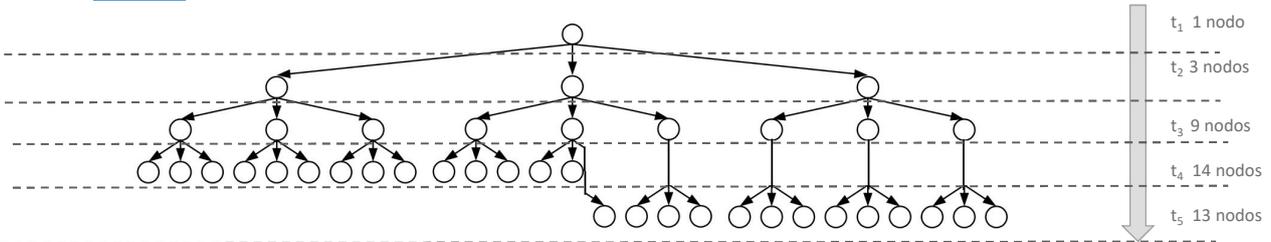
TP3 – Ejercicio 1

$$Speedup(n) = \frac{T(1)}{T(n)}$$

$$Eficiencia = \frac{T(1)}{T(n) \times n}$$

$$Eficiencia = \frac{Speedup(n)}{n} \times 100\%$$

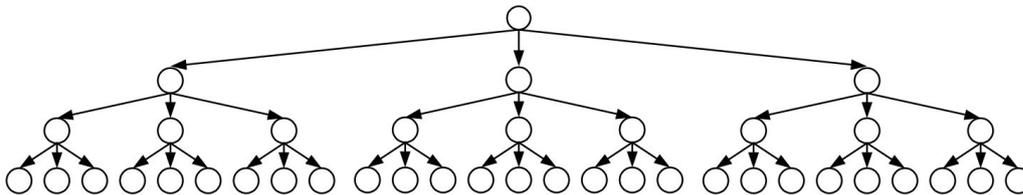
d) Calcular Speedup y Eficiencia si n es la mitad del grado de concurrencia ($n=14$)



- Sabemos que $T(1)=40$ y $T(14)=5$
- $Eficiencia(14)=40/(5*14)=40/70=0,57$
- $Eficiencia(14)=(8/14)*100=0,57*100=57\%$
 - El aprovechamiento de los recursos disponibles mejora respecto de $n=27$

TP3 – Ejercicio 1

e) Calcular Speedup y Eficiencia si n es 1/3 del grado de concurrencia ($n=9$)

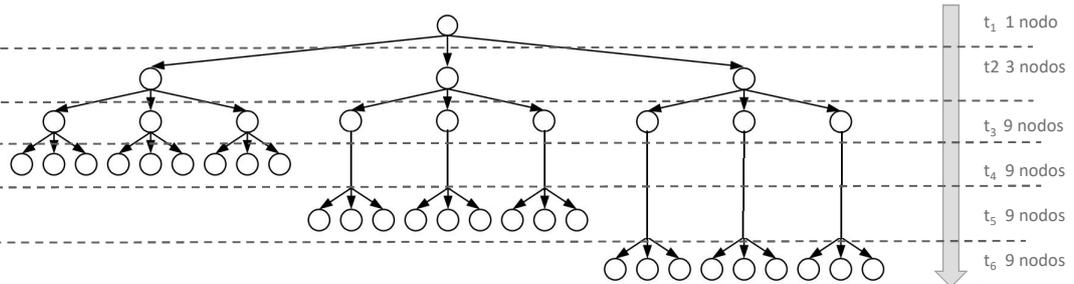


Para pensar: ¿Podemos ejecutar en paralelo todas las tareas en todos los niveles?

$$Speedup(n) = \frac{T(1)}{T(n)}$$

TP3 – Ejercicio 1

e) Calcular Speedup y Eficiencia si n es 1/3 del grado de concurrencia ($n=9$)



- Para calcular el Speedup(9) debemos calcular primero sus parámetros... $T(1)=40$
 - $T(9)=6$
- $Speedup(9)=40/6=6,66$

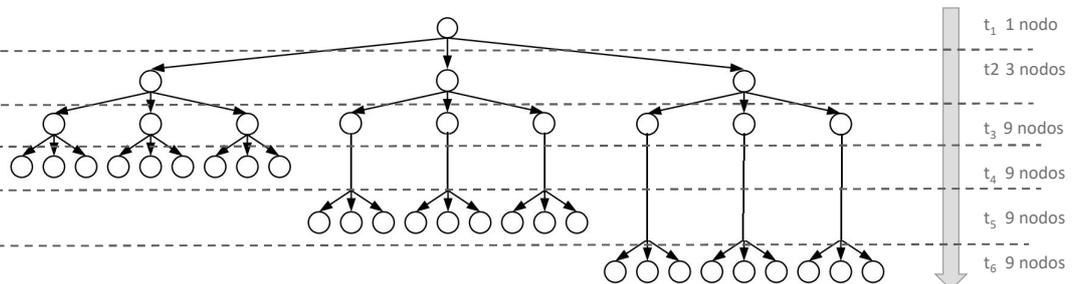
$$Speedup(n) = \frac{T(1)}{T(n)}$$

TP3 – Ejercicio 1

$$Eficiencia = \frac{T(1)}{T(n) \times n}$$

$$Eficiencia = \frac{Speedup(n)}{n} \times 100\%$$

e) Calcular Speedup y Eficiencia si n es 1/3 del grado de concurrencia ($n=9$)



- Sabemos que $T(1)=40$ y $T(9)=6$
- $Eficiencia(9)=40/(6*9)=40/54=0,74$
- $Eficiencia(9)=(6,66/9)*100=0,74*100=74\%$
 - El aprovechamiento de los recursos mejora **bastante** respecto de $n=27$ y $n=14$

TP3 – Ejercicio 1

e) Graficar el factor de Speedup y analizar los resultados

N	Speedup(n)	SpeedupIdeal(n)	Eficiencia(n)
1	1	1	100%
9	6,66	9	74%
14	8	14	57%
27	10	27	37%

