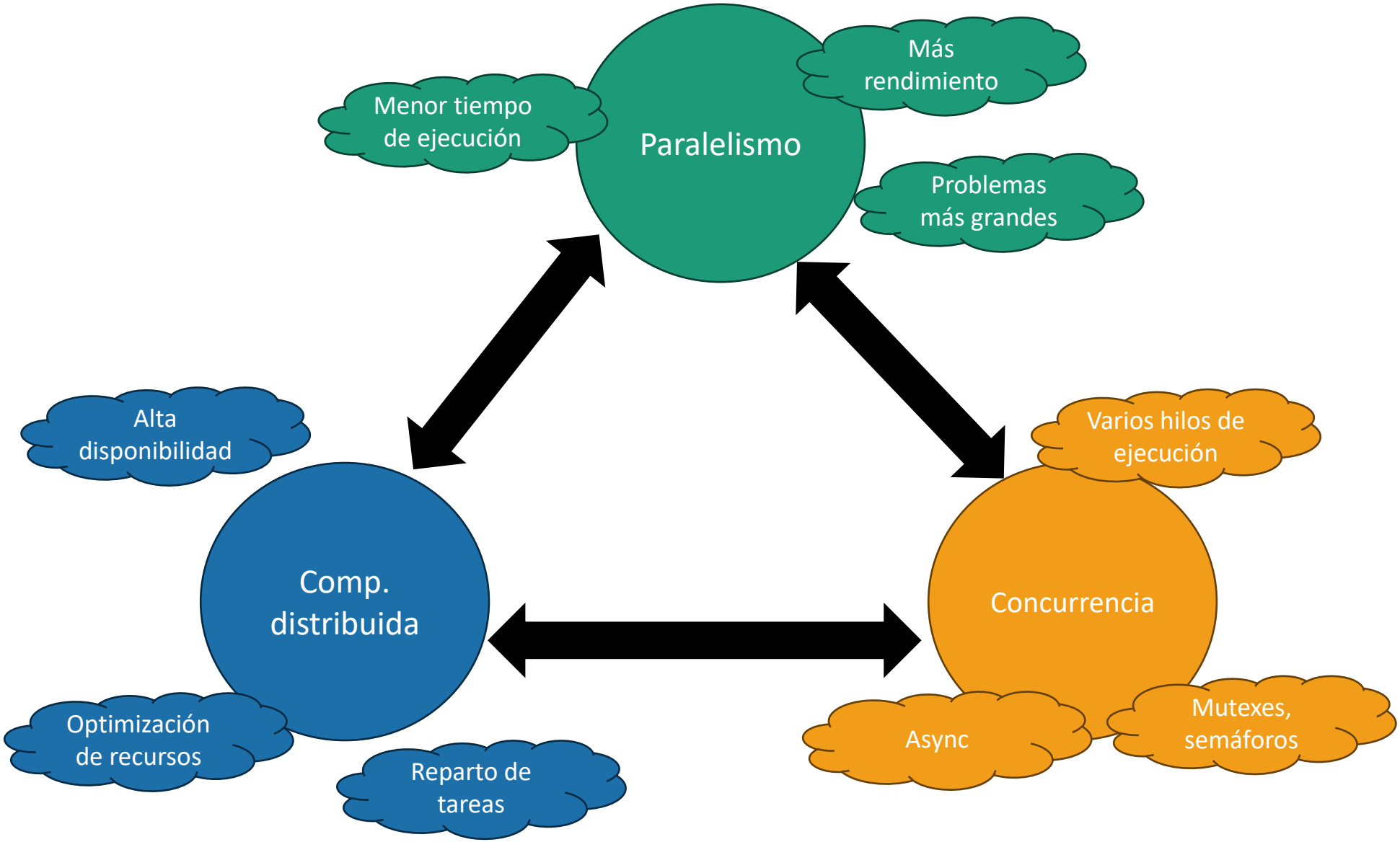
Tema 1: Paralelismo, Concurrencia y Rendimiento

Un resumen rápido…


Paralelismo
Menor tiempo de ejecución
Más rendimiento
Problemas más grandes
Comp. distribuida
Alta disponibilidad
Optimización de recursos
Reparto de tareas
Concurrencia
Varios hilos de ejecución
Async
Mutexes, semáforos

Los computadores según Flynn

Según el flujo de instrucciones y datos.

- SI: una instrucción a la vez.
  - SISD: una instrucción solo afecta a un/os datos.
    - Von Neumann, simple CPUs
  - SIMD: una instrucción afecta a varios datos distintos.
    - MMX, SSE, AVX, GPU, TPUs,

	SD	MD
SI	SISD	SIMD
MI	MISD	MIMD

Los computadores según Flynn

Según el flujo de instrucciones y datos.

- MI: más de una instrucción al mismo tiempo.
  - MISD: todas las instrucciones afectan al mismo dato o datos.
    - Pipeline processing?
  - MIMD: cada instrucción afecta a unos datos distintos.
    - Multiprocesador, multicomputador, superescalar/VLIW, …

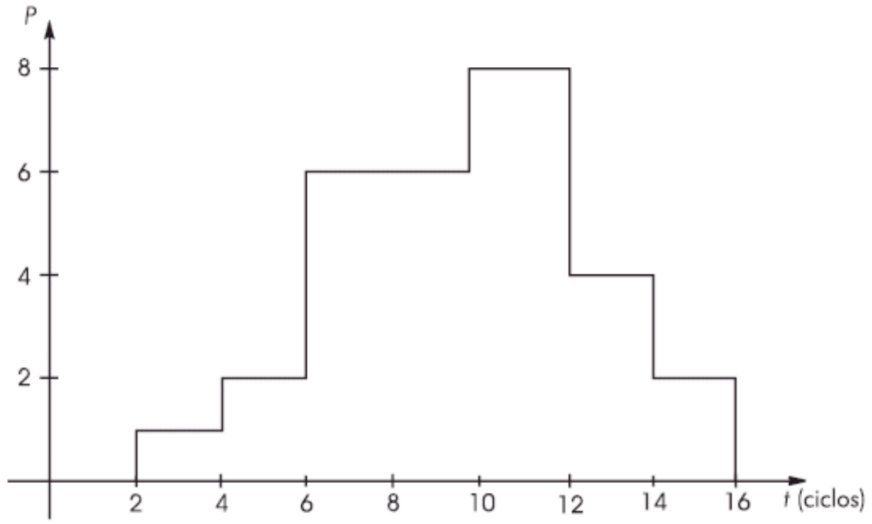
	SD	MD
SI	SISD	SIMD
MI	MISD	MIMD

Resumen métricas de rendimiento

- Número de procesadores: n ó p
- Tiempo de ejecución: $T(n)$
- Speedup: $\mathrm{S(n)} = \frac{T(1)}{T(n)}$
- Eficiencia: $\mathrm{E(n)} = \frac{S(n)}{n}$
- Operaciones: $\mathrm{O(n)}$. Si es necesario, asumimos $\mathrm{O}(1) = T(1)$.
- Redundancia: $\mathrm{R(n)} = \frac{O(n)}{O(1)}$
- Utilización: $\mathrm{U(n)} = R(n)E(n)$
- Calidad del paralelismo: $\mathrm{Q(n)} = \frac{S(n)E(n)}{R(n)}$

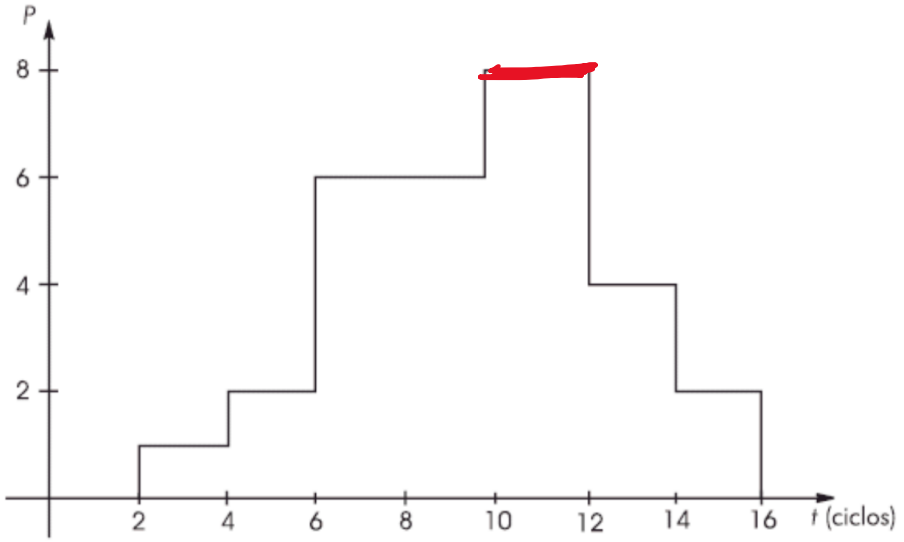
Perfil de paralelismo

- En cada instante de tiempo, el número de procesadores usados (DOP) para ejecutar un programa.
  - Salvo que se diga lo contrario, asumiremos que se ha calculado con infinitos procesadores.



¿Cómo analizarlo?

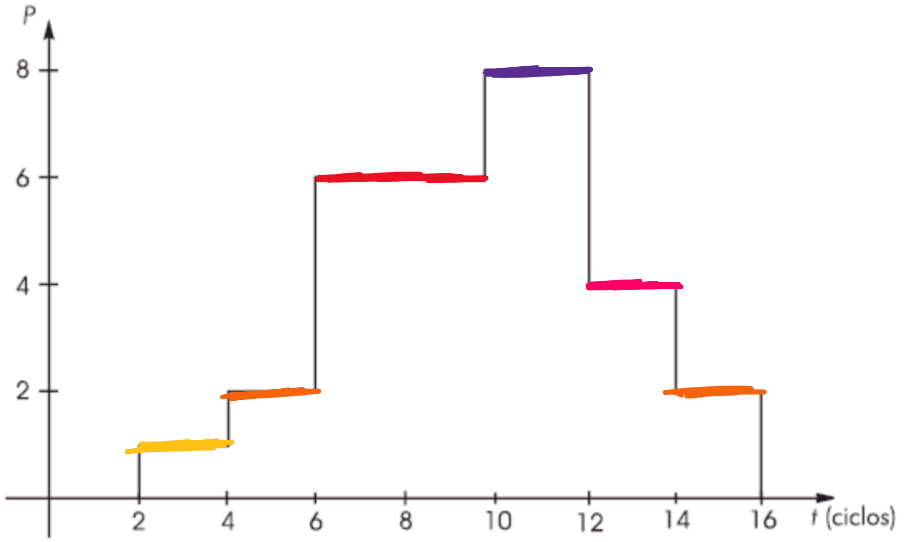
- Máximo DOP
  - m
  - Lo sacamos mirando el gráfico y quedándonos con su **máximo**
  - **OJO**: solo si se ha calculado el gráfico con inf. procesadores (o el observado es menor al usado para calcular el gráfico).



$m = 8$

¿Cómo analizarlo?

- Tiempo a DOP "i":
  - t_i
  - Lo sacamos **<u>mirando el gráfico y contando cuántos instantes temporales</u>** ha estado exactamente en el valor i
- Trabajo a DOP "i":
  - $W_i = (t_i \cdot i)\Delta$
  - Es el trabajo realizado a DOP=i.

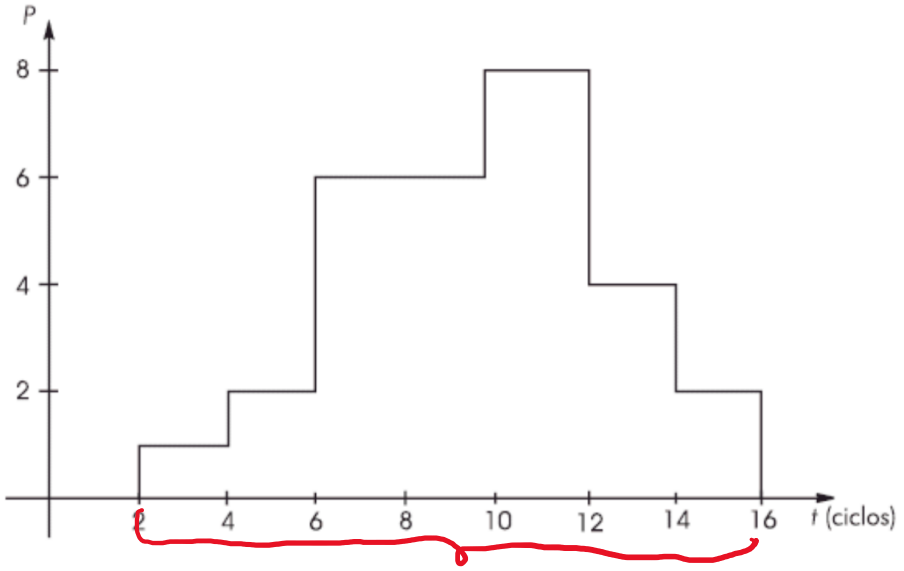


$t_1 = 2 \quad t_2 = 4 \quad t_4 = 2 \quad t_6 = 4 \quad t_8 = 2$

$W_1 = 2 \quad W_2 = 8 \quad W_4 = 8 \quad W_6 = 24 \quad W_8 = 16$

¿Cómo analizarlo?

- Tiempo de ejec. con n ó ∞ procesadores:
  - **Solo si se el pérfil de paralelismo se ha calculado con n procesadores**. Usualmente, $n = \infty$
  - $T(n) = \sum t_i$.
  - Se puede observar también en el gráfico como la anchura o duración del programa.

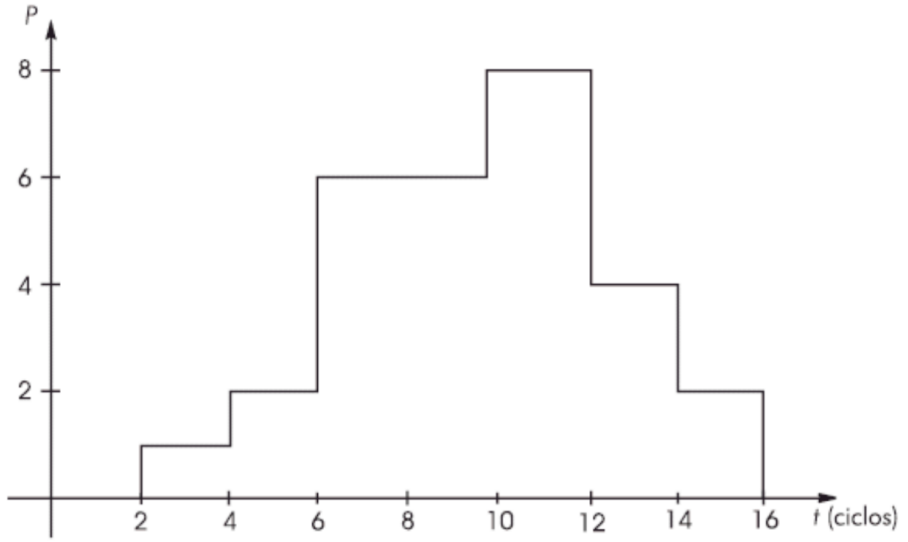


$$T(n) = T(\infty) = 16 - 2 = 14$$

$$\sum t_i = 2 + 4 + 2 + 4 + 2 = 14$$

¿Cómo analizarlo?

- Tiempo de ejec. secuencial
  - $T(1) = \sum t_i \cdot i$
  - Alternativamente, el **<u>número total de cuadraditos del gráfico</u>**.
- Trabajo total:
  - $W = \sum W_i = \Delta \sum t_i \cdot i.$
  - Salvo por Δ, debe coincidir con $T(1)$ (o contar los cuadraditos).

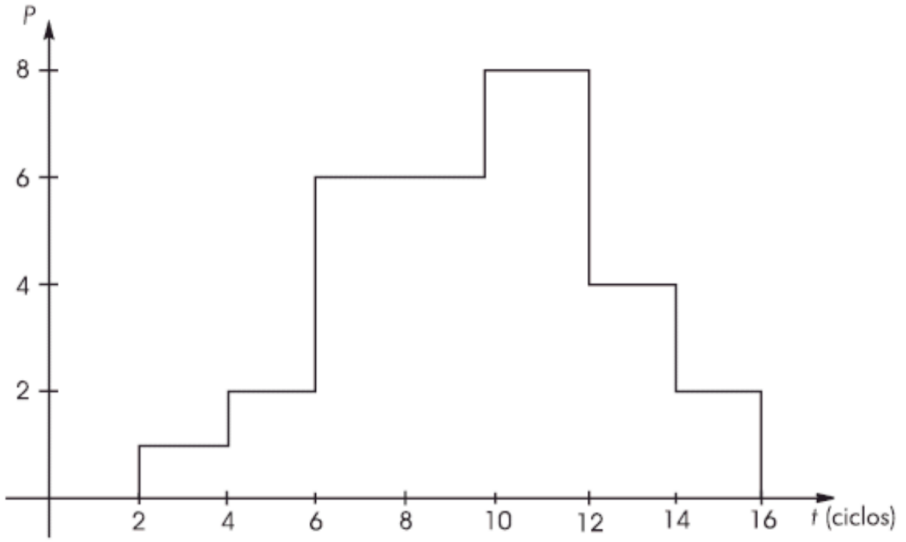


$$T(1) = 2 \cdot 1 + 4 \cdot 2 + 2 \cdot 4 + 4 \cdot 6 + 2 \cdot 8 = 58$$

$$W = 58\,\Delta$$

¿Cómo analizarlo?

- Tiempo de ejec. del trabajo a DOP “i” con n procesadores:
  - $t_i(n)$
  - Si $n \geq i$:
    - Tenemos procesadores para ejecutar todas las tareas en paralelo.
    - $t_i(n) = t_i$
  - Si $n < i$:
    - Me faltan procesadores para ejecutar las tareas.
    - $t_i(n) = t_i \cdot \left\lceil \frac{i}{n} \right\rceil$



$t_1 = 2$ $\quad t_1(4) = 2$

$t_2 = 4$ $\quad t_2(4) = 4$

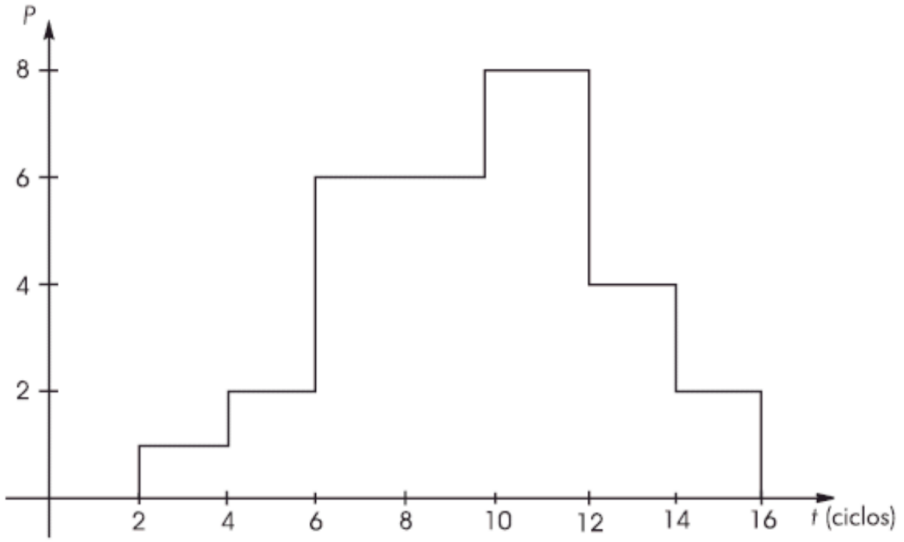
$t_4 = 2$ $\quad t_4(4) = 2$

$t_6 = 4$ $\quad t_6(4) = 4 \cdot 2 = 8$

$t_8 = 2$ $\quad t_8(4) = 2 \cdot 2 = 4$

¿Cómo analizarlo?

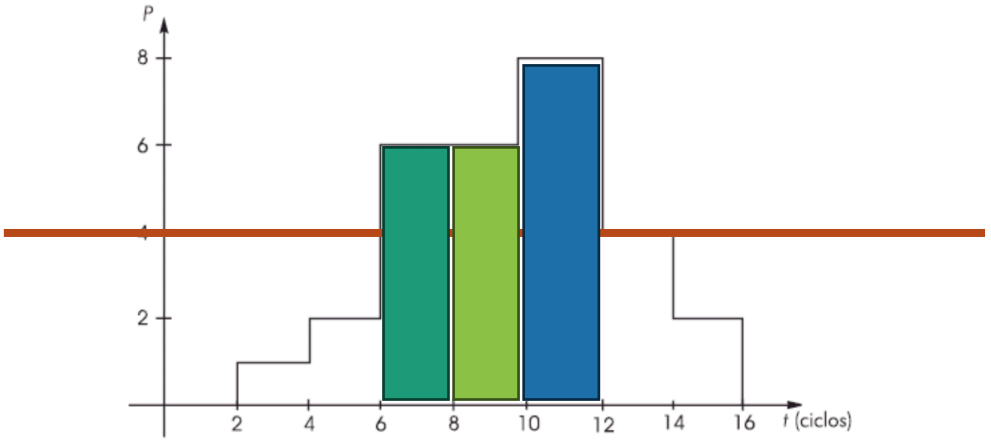
- Tiempo de ejec. con n procesadores:
  - $T(n) = \sum t_i(n)$
  - Útil si me acuerdo de las fórmulas anteriores…

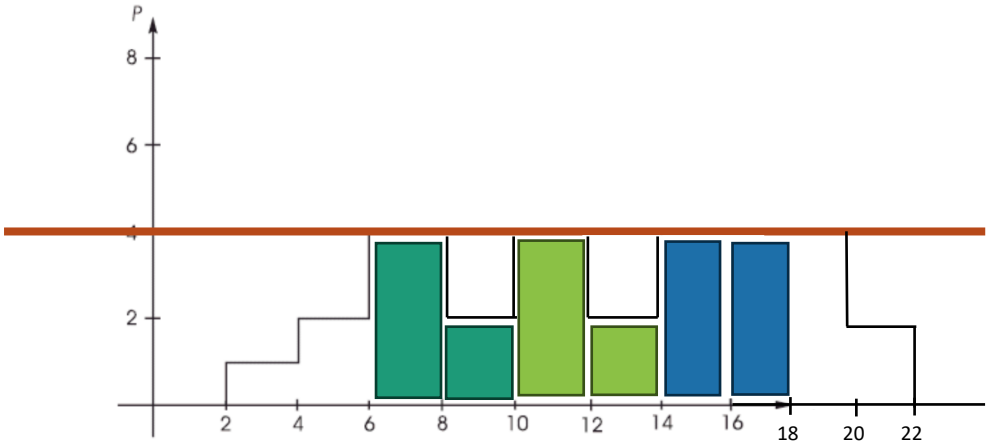


$$T(4) = 2 + 4 + 2 + 8 + 4 = 20$$

¿Cómo analizarlo?

- Tiempo de ejec. con n procesadores:
  - Si $n \geq m$, $\mathrm{T}(n) = T(\infty)$.
    - El ancho del perfil de paralelismo
  - En otro caso, **<u>recalculamos el perfil de paralelismo</u>** asumiendo solo n procesadores.
    - Salvo que sepamos del grafo de dependencias de las tareas, vamos a asumir que las tareas del siguiente instante temporal dependen de la parte pendiente.





$$T(4) = 22 - 2 = 20$$

Modelos de carga

- WC o carga fija: W es fijo.
  - Caso fácil.
  - $S(n) = \frac{T(1)}{T(n)}$

Amdahl: $S(n) = \frac{W_1 + W_n}{W_1 + W_n/n}$;

- TC o tiempo fijo: $T(1) = T'(n)$.
  - El algoritmo ' realiza más trabajo, en el mismo tiempo.
  - $S(n) = \frac{T'(1)}{T'(n)} = \frac{T'(1)}{T(1)} = \frac{W'}{W}$
    - Recordemos: $T(1) = \frac{W}{\Delta}$.

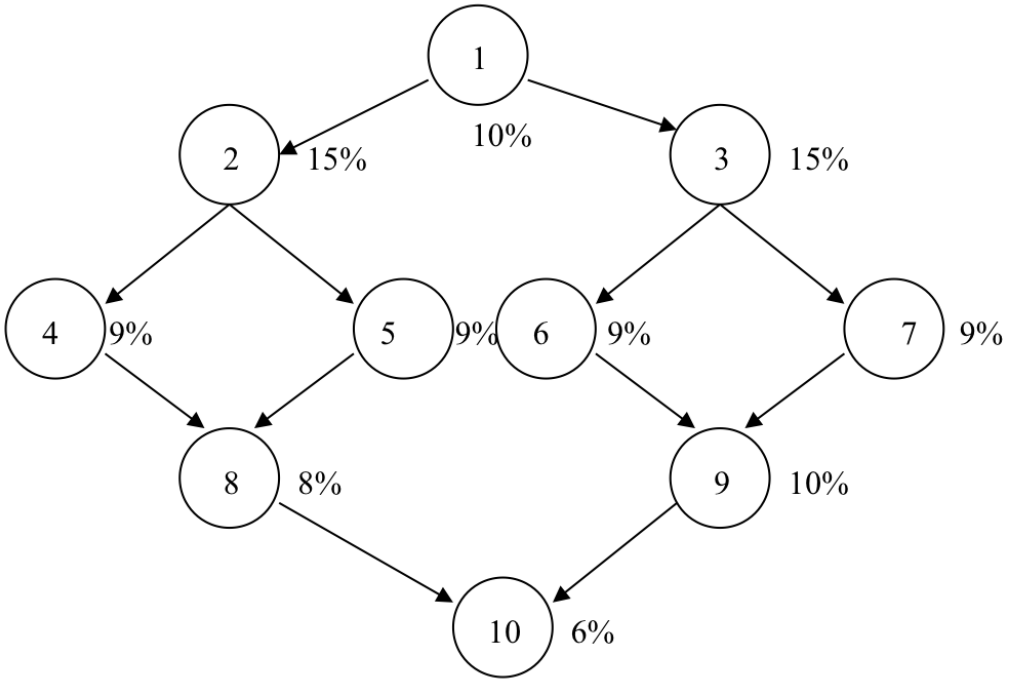
Gustafson: $S(n) = \frac{W_1 + nW_n}{W_1 + W_n}$

Modelos de carga

- MC o memoria fija: el más genérico
  - Una función G nos va a limitar el speedup.
  - $S(n) = \frac{W_1 + G(n)W_n}{W_1 + G(n)W_n/n}$
    - Si $G(n) < 1$: peor que Amdahl.
    - Si $G(n) = 1$: Amdahl.
    - Si $G(n) = n$: Gustafson.
    - Si $G(n) > n$: Mejor que Gustafson.

DAGs

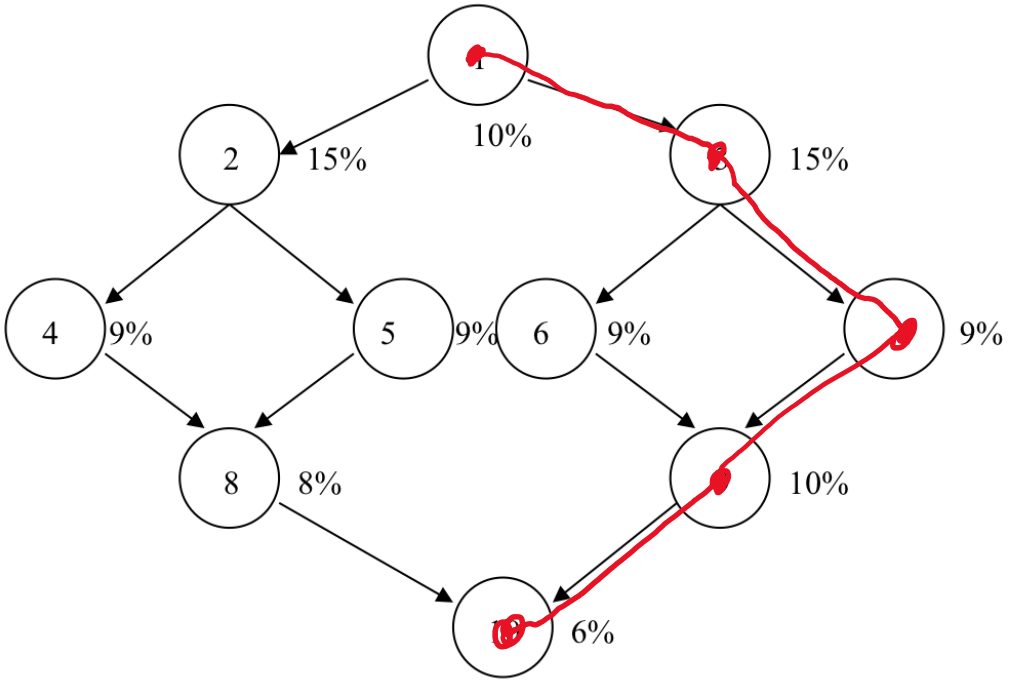
- Cantidad de trabajo: suma de todos los nodos.
  - W
  - Debería coincidir (salvo por Δ) con $T(1)$
  - Nos representa el trabajo secuencial.



$$W = 100$$

DAGs

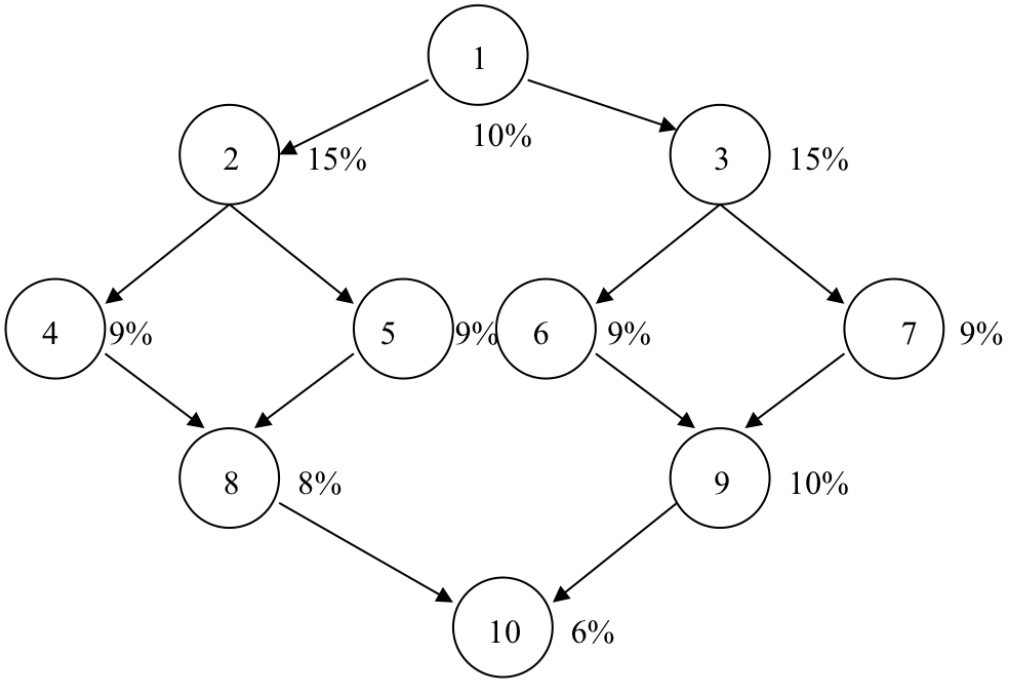
- Profundidad:
  - d
  - Longitud del **camino crítico** (el que suma más).
  - Debería coincidir con el $T(\infty) = T(m)$.



$d = 50$

DAGs

- Paralelismo medio:
  - $AvgP = \frac{W}{d}$
  - Coincide con:
    - $S(\infty) = S(m) = \frac{T(1)}{T(\infty)}$



$$AvgP = S(\infty) = \frac{100}{50} = 2$$

DAGs

- Perfil de paralelismo
  - Ejecutamos todas las tareas posibles a la vez.
    - Si suponemos que hay $\mathrm{p} = \infty$ (ó $\mathrm{p} = m$) procesadores, empezamos todas las tareas **cuanto antes**
    - En caso contrario ($\mathrm{p} < m$), decidimos un **criterio de planificación**.