

TEMA 1:

ARQUITECTURAS PARALELAS: CLASIFICACIÓN Y PRESTACIONES

Lección 3: EVALUACIÓN DE PRESTACIONES DE UNA ARQUITECTURA.

Medidas usuales para evaluar prestaciones.

Tiempo de respuesta de un programa de una arquitectura:

- Real (Wall-clock, elapsed time, real time).
- CPU time = user + sys (no incluye todo el tiempo).
- Con un flujo de instrucciones: elapsed \geq CPU time.
- Con múltiples flujos de instrucciones: elapsed $<$ CPU time, elapsed \geq CPU time/nº flujos de control.
- Tiempo de CPU de usuario: tiempo en ejecución en espacio de usuario.
- Tiempo de CPU de sistema: tiempo en el nivel del SO.

Alternativas para obtener tiempos:

Función	Fuente	Tipo	Resolución aprox. (microsegundos)
time	SO (/usr/bin/time)	<i>elapsed, user, system</i>	10000
clock()/CLOCKS_PER_SEC	SO (time.h)	<i>CPU</i>	10000
gettimeofday()	SO (sys/time.h)	<i>elapsed</i>	1
clock_gettime()/clock_getres()	SO (time.h)	<i>elapsed</i>	0.001
omp_get_wtime()/omp_get_wtick()	OpenMP (omp.h)	<i>elapsed</i>	0.001
SYSTEM_CLOCK()	Fortran	<i>elapsed</i>	1

Tiempo de CPU: **$T_{cpu} = NI \times CPI \times T_{ciclo}$** .

MIPS y MFLOPS:

- Millones de instrucciones por segundo (MIPS)= **NI (Número de instrucciones) / $T_{cpu} \times 10^6$ o $MIPS = F$ (Frecuencia) / $CPI \times 10^6$** .
- Depende del repertorio de instrucciones (difícil la comparación de máquinas con repertorios distintos).
- Puede variar inversamente con las prestaciones (mayor valor de MIPS corresponde a peores prestaciones).
- Millones de operaciones coma flotante por segundo (MFLOPS)= **$n^{\circ}Operaciones / T_{cpu} \times 10^6$** .

Ganancia en prestaciones al realizar una mejora.

Mejora o Ganancia en Prestaciones (Speed-up o ganancia en velocidad).

- Si se incrementan las prestaciones (velocidad) que se consigue en la nueva situación, p, con respecto a la previa (sistema base, b) se expresa mediante la ganancia en presentaciones o speed-up, S.

$$S = V_p / V_b = T_b / T_p$$

$$S = T_{cpu}(b) / T_{cpu}(p) = NI(b) * CPI(b) * T_{ciclo}(b) / NI(p) * CPI(p) * T_{ciclo}(p)$$

(Vb: velocidad de la máquina base, Vp: velocidad de la máquina mejorada (un factor p en uno de sus componentes, Tb: tiempo de ejecución en la máquina base, Tp: tiempo de ejecución en la máquina mejorada).

¿Qué impide que se pueda obtener la ganancia en velocidad pico?

- Riesgos:
 - o Datos.
 - o Control.
 - o Estructurales.
- Accesos a memoria (debido a la jerarquía).

Ley de Amdahl.

- La mejora de velocidad, S, que se puede obtener cuando se mejora un recurso de una máquina en un factor p está limitada por:

$$S = V_p / V_b = T_b / T_p \leq 1 / [f + (1 - f) / p]$$

Si "p" tiende a infinito = 1 / f

Si "f" tiende a cero = p

Donde f es la fracción del tiempo de ejecución del sistema base durante el que no se usa dicha mejora.

Conjunto de programas de prueba (Benchmark).

Benchmarks:

- Propiedades exigidas a medidas de prestaciones:
 - o Fiabilidad: Representativas, evaluar diferentes componentes del sistema y reproducibles.
 - o Permitir comparar diferentes realizaciones de un sistema o diferentes sistemas: Aceptadas por todos los interesados (usuarios, fabricantes, vendedores).
- Interesados:
 - o Vendedores y fabricantes de hardware o software.
 - o Investigadores de hardware o software.
 - o Compradores de hardware o software.

Tipo de Benchmark:

- De bajo nivel o microbenchmark (test ping-pong, evaluación de las operaciones con enteros o con flotantes).
- Kernels (resolución de sistemas de ecuaciones, multiplicación de matrices, FFT, descomposición LU).
- Sintéticos (Dhrystone, Whetstone).
- Programas reales.
- Aplicaciones diseñadas (Predicción de tiempo, dinámica de fluidos, animación etc).