

Midiendo el desempeño

José Galaviz

¿Por qué medir el desempeño?

¿Por qué medir el desempeño?

- Para poder evaluar el impacto que tienen las decisiones que se toman en el diseño de:
 - La Arquitectura del Conjunto de Instrucciones.
 - La Organización de la CPU.
 - El resto del sistema: chipset, periféricos, memoria, buses.
- Para poder evaluar el impacto del Sistema Operativo y su relación con el hardware.
- Para poder evaluar si la relación costo beneficio es adecuada.

Tipos de medidas de desempeño

Hay dos tipos de competencias de ciclismo:



¿Diferencias?



Tiempo de respuesta

- Se mide el tiempo que tarda el sistema en hacer una tarea.
- Mejor mientras más pequeño (Lower Is Better o LIB).
- Es el preferido por los usuarios del sistema.

Rendimiento

- Número de labores realizadas por unidad de tiempo.
- Es mejor mientras mayor sea (Higher Is Better o HIB).
- Es preferida por los administradores de sistemas o el personal técnico.

¿Con qué medimos?

¿Con qué medimos?

Tratándose de un sistema de cómputo que es útil porque ejecuta programas lo medimos...

pues con programas.

Paréntesis de terminología

- Benchmark.
 - Prueba de desempeño o de rendimiento.
 - Comparativa de desempeño o rendimiento.
 - El objetivo es poder **comparar**.
- Suite.
 - Conjunto de programas usado para probar el desempeño.
 - También se usa el término **workload**.

¿Qué programas usamos para probar el desempeño?

Programas reales

Tomamos programas populares y los ejecutamos en los sistemas cuyo desempeño queremos comparar.

Kernels

Trozos de programas reales que por su uso exhaustivo de cierto tipo de operaciones se usan para probar el desempeño en cierto rubro.

Ejemplos:

- Linpack.
- Livermore loops.
- NAS.

Benchmarks de juguete

Programas pequeños que solían usarse para probar el desempeño de microcomputadoras en los 80:

Ejemplos:

- Quicksort.
- La criba de Eratóstenes.

Benchmarks sintéticos

Evalúan el desempeño del sistema procurando sintetizar en un sólo programa las características de muchos, haciéndolo una muestra representativa de la frecuencia de uso de las operaciones. No son programas o fragmentos de programas útiles, se diseñan ad-hoc.

Ejemplos:

- Whetstone.
- Dhrystone.

Una opción más general

Un conjunto (suite) de programas representativos, estandarizado pero adaptable con el tiempo, que permita estimar cómo se comportará el sistema en su uso real.

Ejemplo:

- SPEC CPU.
- Embedded Microprocessor Benchmark Consortium (EEMBC).
- Business Applications Performance Corporation (BAPCo).

Principios de la evaluación

- Relevancia. La prueba debe evaluar aspectos esenciales del sistema, que sean significativos.
- Representatividad. La prueba debe ser un indicador del comportamiento del sistema en condiciones de uso real.
- Equidad. Todos los sistemas deben ser comparados con justicia.
- Repetibilidad. La prueba debe ser repetible, verificable por otros.
- Escalable. Una prueba general debe ser aplicable a sistemas con configuraciones diversas, de recursos modestos o poderosos.
- Transparencia. Las métricas utilizadas deben estar bien definidas y ser comprensibles.
- Costo-efectiva. No debe ser muy oneroso aplicar la prueba.

Reglas del juego

Con frecuencia a lo largo del tiempo los fabricantes han procurado resaltar los sistemas que fabrican por sobre la competencia y a veces francamente han hecho trampa.

Un compilador optimizado puede tirar a la basura el 25% del código de Dhrystone, por ejemplo.

Ejemplo

Tiempos de ejecución en milisegundos de cinco programas para dos sistemas de cómputo

Programa	Computadora A	Computadora B
Prog1	20	1
Prog2	15	2
Prog3	10	15
Prog4	10	27
Prog5	10	20
Total	65	65

¿Qué podemos decir?

- La computadora B es 20 veces mejor que la computadora A para el programa 1 y 7.5 veces mejor para el programa 2.
- La computadora A es 1.5 veces mejor que B para el programa 3, 2.7 veces mejor en el programa 4 y 2 veces mejor que B en el programa 5.
- El tiempo total de ambos conjuntos es el mismo.

Necesitamos un valor sintético

- Que nos permita evaluar integralmente todo el conjunto de prueba. Decirnos alrededor de donde están
- Que escale bien si incorporamos o eliminamos programas del conjunto.
- Que no se deje llevar mucho por *outliers*.
- Que todo valor de la muestra aporte información al valor sintético.

Medida de tendencia central

¿cuáles conocen?

Medias

$$D = \{\delta_1, \delta_2, \dots, \delta_n\}$$

Media aritmética

$$A(D) = \frac{1}{n} \sum_{i=1}^n \delta_i \quad (2.1)$$

Media armónica

$$H(D) = \frac{n}{\sum_{i=1}^n \frac{1}{\delta_i}} \quad (2.2)$$

Media geométrica

$$G(D) = \sqrt[n]{\prod_{i=1}^n \delta_i} \quad (2.3)$$

Curiosidades

- Si los números son positivos la armónica es la menor de las tres y la aritmética la mayor, la geométrica está siempre en medio.
- Si tenemos $D = \{2, 2.7, 5\}$ la media geométrica es 3 (la raíz cúbica de 27). El lado de un cubo cuyo volumen sea igual al del prisma de las dimensiones de D .
- La armónica es el inverso de la aritmética de los inversos.
- La media aritmética se ve afectada por valores extremos. La armónica también.

¿Cuándo usamos cada una?

Velocidad (tareas/min)
70
30
40
60

Tabla 2.2. Las cuatro velocidades a las que puede operar un sistema de cómputo. La media aritmética es 50 tareas/minuto, la media armónica es 44.8 tareas/minuto.

Caso 1

Ponemos a funcionar el sistema durante dos minutos en cada uno de los regímenes de velocidad, haciendo un total de 8 minutos.

Vel (tareadas/min)	Trabajo (min)	#tareadas
70	2	140
30	2	60
40	2	80
60	2	120
Total	8	400

Tabla 2.3. Resultado de ejecutar tareadas durante dos minutos en cada régimen en el sistema de ejemplo. La suma del total de tareadas ejecutadas es 400.

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tarefas/minuto) durante los 8 minutos?

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareass/minuto) durante los 8 minutos?

Unidades: (tareass / minuto) × minuto = tareass

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tarejas/minuto) durante los 8 minutos?

Unidades: (tarejas / minuto) \times minuto = tarejas

- Armónica: $44.8 \times 8 = 358.4 \neq 400$

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante los 8 minutos?



Unidades: (tareas / minuto) \times minuto = tareas

- Armónica: $44.8 \times 8 = 358.4 \neq 400$
- Artitmética: $50 \times 8 = 400$

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante los 8 minutos?

Unidades: (tareas / minuto) \times minuto = tareas

- Armónica: $44.8 \times 8 = 358.4 \neq 400$ 
- Artitmética: $50 \times 8 = 400$ 

Caso 2

Ponemos a funcionar el sistema haciendo 14 tareas en cada uno de los regímenes de velocidad, haciendo un total de 56 tareas.

Vel (tareas/min)	Tareas	Tiempo (min)
70	14	0.2
30	14	0.47
40	14	0.35
60	14	0.23
Total	56	1.25

Tabla 2.4. Resultado de ejecutar 14 tareas en cada régimen en el sistema de ejemplo. La suma del total de tiempo es 1.25 minutos.

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante las 56 tareas?

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante las 56 tareas?

Unidades: tareas / (tareas / minuto) = minutos

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante las 56 tareas?

Unidades: tareas / (tareas / minuto) = minutos

- Armónica: $56 / 44.8 = 1.25$

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tarejas/minuto) durante las 56 tareas?

Unidades: tareas / (tareas / minuto) = minutos

- Armónica: $56 / 44.8 = 1.25$
- Artitmética: $56 / 50 = 1.12 \neq 1.25$

¿qué media usamos?

¿cuál media nos da una medida de la tasa promedio a la que trabajó el sistema (tareas/minuto) durante las 56 tareas?

Unidades: tareas / (tareas / minuto) = minutos

- Armónica: $56 / 44.8 = 1.25$ ✓
- Artitmética: $56 / 50 = 1.12 \neq 1.25$ ✗

¿Qué observaron?

- Si el valor que tratas de estimar usando la media es directamente proporcional a la métrica de la media: media aritmética.
- Si son inversamente proporcionales: media armónica.




Caso 3

Producción (ton)	Factor de crecimiento
100	
130	1.3000
180	1.3846
240	1.3333
305	1.2708
345	1.1311
M. arit.	1.284
M. geom.	1.281
M. armo.	1.278

Tabla 2.5. Producción anual, en toneladas, de una empresa productora de cereales durante los últimos seis años.

¿Cuál es la buena?

¿Cuál de las medias nos permite decir “este es el valor promedio del factor de crecimiento anual durante los últimos 5 años”?

- Aritmética: $100 \times 1.284^5 = 348.98 \neq 345$ 
- Armónica: $100 \times 1.278^5 = 340.91 \neq 345$ 
- Geométrica: $100 \times 1.281^5 = 345$ 

Características del caso 3

- La métrica de la media no tiene unidades.
- Es una tasa de cambio.

En una medición de desempeño

- Se tiene una colección de programas.
- Posiblemente algunos de ellos deban ser considerados más importantes que otros:
 - Porque son más representativos.
 - Porque tienen una importancia relativa mayor que otros.

Media aritmética ponderada

La media aritmética le asocia un peso igual (a $1/n$) a los elementos de la muestra.

Dada una muestra $D = \{\delta_1, \delta_2, \dots, \delta_n\}$ de n datos y un conjunto de pesos $W = \{w_1, w_2, \dots, w_n\}$, tales que $\sum_{i=1}^n w_i = 1$, la media aritmética ponderada se define como:

$$P(D, W) = \sum_{i=1}^n w_i \delta_i \quad (2.4)$$

Uso

- Le podemos asociar un peso relativo mayor o menor a ciertos programas de la muestra.

¿Con base en qué criterio?

Idea

Que la ponderación sea con base en una máquina patrón.

Tiempo normalizado

Los tiempos en una máquina c para cada uno de los n programas de prueba:

$$T_c = \{t_{1,c}, t_{2,c}, \dots, t_{n,c}\}$$

Los tiempos en una máquina p elegida como patrón:

$$T_p = \{t_{1,p}, t_{2,p}, \dots, t_{n,p}\}$$

Los tiempos normalizados:

$$\hat{t}_{i,c} = \frac{t_{i,c}}{t_{i,p}}$$

Ejemplo

Programa	A	B	C
Prog1	5	13	35
Prog2	730	250	45
Prog3	1200	230	50
Total	1935	493	130

¿A quién usamos de patrón?

Programa	A	B	C
Prog1	1	2.6	7
Prog2	1	0.34	0.06
Prog3	1	0.19	0.04
Total	3	3.13	7.1
M. arit.	1	1.04	2.37

Programa	A	B	C
Prog1	0.38	1	2.69
Prog2	2.92	1	0.18
Prog3	5.22	1	0.22
Total	8.52	3	3.09
M. arit.	2.84	1	1.03

Programa	A	B	C
Prog1	0.14	0.37	1
Prog2	16.22	5.56	1
Prog3	24	4.6	1
Total	40.37	10.53	3
M. arit.	13.46	3.51	1

Con la media adecuada

Programa	A	B	C
Prog1	1	2.6	7
Prog2	1	0.34	0.06
Prog3	1	0.19	0.04
Total	3	3.13	7.1
M. arit.	1	1.04	2.37
M. geom.	1	0.55	0.26

Programa	A	B	C
Prog1	0.38	1	2.69
Prog2	2.92	1	0.18
Prog3	5.22	1	0.22
Total	8.52	3	3.09
M. arit.	2.84	1	1.03
M. geom.	1.8	1	0.47

Programa	A	B	C
Prog1	0.14	0.37	1
Prog2	16.22	5.56	1
Prog3	24	4.6	1
Total	40.37	10.53	3
M. arit.	13.46	3.51	1
M. geom.	3.82	2.12	1

Media geométrica

Ventajas

- No se deja sesgar por valores excesivamente grandes o pequeños.
- entrega valores consistentes sin importar a quien se elija para normalizar.

Desventaja

- Deja de ser intuitiva la interpretación. Un programa con desempeño de 10 no necesariamente es dos veces mejor que uno de 5.

SPEC

Standard Performance Evaluation Corporation (SPEC)

Consorcio formado por fabricantes de sistemas de cómputo, universidades, consultores y grupos de investigación de todo el mundo.

Objetivo: definir pruebas de desempeño y protocolos de aplicación de las mismas que garanticen la imparcialidad y veracidad de los resultados reportados.

Pruebas

Originalmente SPEC CPU solamente. Prueba general de desempeño para procesadores.

Hoy en día:

- Consumo.
- Servidores web.
- Virtualización.
- Cómputo gráfico.
- Nube.
- Servidores de correo.

SPEC CPU

Se hacen 4 pruebas obligatorias:

- SPECint Speed Base.
- SPECint Rate Base.
- SPECfp Speed Base.
- SPECfp Rate Base.

Y 4 opcionales

- SPECint Speed Peak.
- SPECint Rate Peak.
- SPECfp Speed Peak.
- SPECfp Rate Peak.

Diferencias

- Speed son pruebas con medida LIB (tiempo de respuesta).
- Rate son pruebas de rendimiento (throughput), HIB (unidades de trabajo por unidad de tiempo).
- int son con aritmética entera
- fp son de punto flotante.
- Base son fijando todos los parámetros de compilación para la prueba.
- Peak son dejándolos libres.

Composición

- Mezcla de programas reales ejecutándose sobre entradas preestablecidas.
- Adaptable a lo largo del tiempo.
- Usando una máquina de referencia y calculando la media geométrica de tiempos normalizados.

La máquina de referencia

Sun Fire v490, Sun
Microsystems, 2006.



SPECint 2017

Programa	Lenguaje	Área de aplicación
500.perlbench_r	C	Perl interpreter
502.gcc_r	C	GNU C compiler
505.mcf_r	C	Route planning
520.omnetpp_r	C++	Discrete Event simulation - computer network
523.xalancbmk_r	C++	XML to HTML conversion via XSLT
525.x264_r	C	Video compression
531.deepsjeng_r	C++	Artificial Intelligence: alpha-beta tree search (Chess)
541.leela_r	C++	Artificial Intelligence: Monte Carlo tree search (Go)
548.exchange2_r	Fortran	Artificial Intelligence: recursive solution generator (Sudoku)
557.xz_r	C	General data compression

SPECfp 2017

Programa	Lenguaje	Área de aplicación
503.bwaves_r	Fortran	Explosion modeling
507.cactuBSSN_r	C++, C, Fortran	Physics: relativity
508.namd_r	C++	Molecular dynamics
510.parest_r	C++	Biomedical imaging: optical tomography with finite elements
511.povray_r	C++, C	Ray tracing
519.lbm_r	C	Fluid dynamics
521.wrf_r	Fortran, C	Weather forecasting
526.blender_r	C++, C	3D rendering and animation
527.cam4_r	Fortran, C	Atmosphere modeling
628.pop2_s	Fortran, C	Wide-scale ocean modeling (climate level)
538.imagick_r	C	Image manipulation
544.nab_r	C	Molecular dynamics
549.fotonik3d_r	Fortran	Computational Electromagnetics
554.roms_r	Fortran	Regional ocean modeling

La ley de Amdahl

Ganancia (speedup)

Al introducir una mejora en un sistema, se espera que tenga un impacto positivo en el desempeño.

$$\frac{Tiempo_{sin}}{Tiempo_{con}}$$

Un menor tiempo en la ejecución de una tarea. El cociente debe ser mayor que uno. Mejor cuanto más grande.

Uso limitado

- Normalmente al introducir una mejora, esta puede usarse sólo en una parte de un proceso más grande.
- Habría que determinar la ganancia cuando se puede usar.
- Y la ganancia total en el proceso completo, considerando esas partes en las que la mejora no se puede usar.

Dos tipos de ganancia

- Ganancia bruta. La que se obtiene considerando solamente la parte del proceso en la que la mejora introducida se puede usar:

g

- Ganancia neta. La que se obtiene considerando el proceso completo y el uso de la mejora cuando es posible hacerlo:

G

Ejemplo

- Recorrido a campo traviesa (*trekking*).
- 30 Km: 26 Km bosque y terreno rocoso + 4 Km nieve blanda.
 - 86.66% terreno sólido, 13.33% terreno blando.
- 12 horas el año pasado: $T_o = 12$.
- Usando sólo botas de trekking.
- Si usamos *snowshoes* podemos ir al doble de la velocidad sobre nieve (o sea recorreremos ese tramo en la mitad del tiempo).

Análisis

- Ganancia bruta:

$$g = \frac{Tiempo_{sin}}{Tiempo_{con}} = 2$$

- Fracción de proceso que usa la mejora:

$$F = 0.1333$$

- Fracción de proceso que no usa la mejora:

$$1 - F = 0.8666$$

Tiempo (total) mejorado

Cuando se puede usar la mejora (0.1333) el tiempo se divide por 2. Cuando no se puede usar 0.8666, el tiempo permanece siendo el mismo:

$$T_m = T_0 \left(0.8666 + \frac{0.1333}{2} \right) = 0.9333 T_0 \approx 11.2$$

Ganancia neta

Ahora estamos en condiciones de evaluar la ganancia neta G :

$$G = \frac{T_0}{T_m} = \frac{12}{11.2} = 1.07$$

Generalizando

$$T_m = T_0 \left[(1 - F) + \frac{F}{g} \right]$$

$$G = \frac{T_0}{T_m} = \frac{1}{(1 - F) + \frac{F}{g}}$$

Intuitivamente

¿Qué nos indica la ley de Amdahl?

- Tiene más impacto en el desempeño aquello que se puede usar más frecuentemente o una mayor porción del proceso.
- Tiene más impacto lo que reporta una ganancia bruta mayor.

Ejemplo

- Para igualar la ganancia neta de una mejora con $g=2.1$ que se puede usar el 30% del tiempo.
- Se necesita por ejemplo, una mejora con $g=5$ que se pueda usar el 20% del tiempo.
- Puede haber mejoras sutiles que se usen mucho.

Evaluación de compromiso

Es también útil siempre hacer evaluaciones relativas. Que permitan observar la relación costo-beneficio:

- Desempeño SPEC / costo.
- (instrucciones / segundo) / (área en el chip)
- (Transacciones / segundo) / ancho de banda consumido
- Frecuencia de reloj / temperatura
- (Tareas / segundo) / Watt

Otras métricas

¿Serán útiles?

- MIPS (Millones de Instrucciones Por Segundo).
- Tiempo de ejecución de un programa.
- BogoMIPS.
- MFLOPS.

Otras métricas

¿Serán útiles?

- MIPS (Millones de Instrucciones Por Segundo). ¿cuáles instrucciones?
- Tiempo de ejecución de un programa. ¿Tiempo de procesador o tiempo total incluyendo E/S y accesos a memoria?
- BogomIPS. Es sólo un truco para parametrizar tiempos de espera en Linux.
- MFLOPS. A lo mejor no está mal en el contexto de *number crunching*.