Performance Measurement

Krerk Piromsopa, Ph.D.
Dept. of Computer Engineering
Chulalongkorn University

Definition: Performance

- Performance is in units of things per sec
  - bigger is better
- If we are primarily concerned with response time

$$\text{performance}(x) = \frac{1}{\text{execution\_time}(x)}$$

" X is n times faster than Y" means

$$n = \frac{\text{Performance}(X)}{\text{Performance}(Y)} = \frac{\text{Execution\_time}(Y)}{\text{Execution\_time}(X)}$$

Execution Time

- Wall-clock time, Response Time, Elapsed Time
- CPU Time? (no I/O, other programs)

Arithmetic Mean

- Same unit & Starting point

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{1}{n}(x_1 + \cdots + x_n).$$

- e.g. Average time to complete a program (10 secs, 20 secs, 15 secs) = 15 secs
- Consider the scores (1, 2, 2, 2, 3, 9). The arithmetic mean is 3.17, but five out of six scores are below this!

Harmonic Mean

- appropriate for situations when the average of rates is desired

$$H = \frac{n}{\frac{1}{a_1} + \frac{1}{a_2} + \cdots + \frac{1}{a_n}}.$$

- if for half the *distance* of a trip you travel at 40 kilometres per hour and for the other half of the *distance* you travel at 60 kilometres per hour, then your average speed for the trip is given by the harmonic mean of 40 and 60, which is 48; that is, the total amount of time for the trip is the same as if you travelled the entire trip at 48 kilometres per hour

Geometric Mean

- Good for Ratio

$$\left(\prod_{i=1}^{n} a_i\right)^{1/n} = \sqrt[n]{a_1 \cdot a_2 \cdots a_n}$$

- If one experiment yields a ratio of 10,000 and the next yields a ratio of 0.0001, an arithmetic mean would misleadingly report that the average ratio was near 5000. Taking a geometric mean will more honestly represent the fact that the average ratio was 1.

SPECRatios

Benchmarks	Ultra5 Time (sec)	Opteron Time (sec)	SPECRatio	Itanium 2 Time (sec)	SPECRatio
Program 1	1600	51.5	31.06	56.1	28.53
Program 2	3100	125.0	24.73	70.7	43.85
Geometric mean			27.71		35.37

SPECRatios

Benchmarks	Ultra5 Time (sec)	Opteron Time (sec)	SPECRatio	Itanium2 Time (sec)	SPECRatio	Opteron/ Itanium (sec)	SPECRatio
Program 1	1600	51.5	31.06	56.1	28.53	0.92	0.92
Program 2	3100	125.0	24.73	70.7	43.85	1.77	1.77
Geometric mean			27.71		35.37	1.27	1.27

What does SPEC CPU2006 measure?

SPEC CPU2006 focuses on compute intensive performance, emphasizing:
- the computer processor (CPU),
- the memory architecture, and
- the compilers.

SPEC CPU2006 contains two components that focus on two different types of compute intensive performance:
- **CINT2006** suite measures compute-intensive integer performance,
- **CFP2006** suite measures compute-intensive floating point performance.

SPEC CPU2006 is not intended to stress other computer components such as networking, the operating system, graphics, or the I/O system.
For single-CPU tests, the effects from such components on SPEC CPU2006 performance are usually minor. For large rate runs, operating system services may affect performance, and the I/O system - number of disks, speed, striping - can have an effect.

CINT 2006

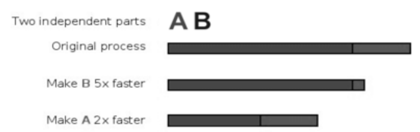
400.perlbench	C	PERL Programming Lang
401.bzip2	C	Compression
403.Gcc	C	C Compiler
429.Mcf	C	Combinatorial Optimization
445.Gobmk	C	Artificial Intelligence: go
456.Hmmer	C	Search Gene Sequence
458.Sjeng	C	Artificial Intelligence: chess
462.Libquantum	C	Physics: Quantum Computing
464.h264ref	C	Video Compression
471.Omnetpp	C++	Discrete Event Simulation
473.Astar	C++	Path-finding Algorithms
483.Xalancbmk	C++	XML Processing

CFP 2006

410.Bwaves	Fortran	Fluid Dynamics
416.Gamess	Fortran	Quantum Chemistry
433.Milc	C	Physics: Quantum Chromodynamics
434.Zeusmp	Fortran	Physics/CFD
435.Gromacs	C/Fortran	Biochemistry/Molecular Dynamics
436.cactusADM	C/Fortran	Physics/General Relativity
437.leslie3d	Fortran	Fluid Dynamics
444.Namd	C++	Biology/Molecular Dynamics
447.dealII	C++	Finite Element Analysis
450.Soplex	C++	Linear Programming, Optimization
453.Povray	C++	Image Ray-tracing
454.Calculix	C/Fortran	Structural Mechanics
459.GemsFDTD	Fortran	Computational Electromagnetics
465.Tonto	Fortran	Quantum Chemistry
470.Lbm	C	Fluid Dynamics
481.Wrf	C/Fortran	Weather Prediction
482.sphinx3	C	Speech recognition

Amdahl's Law

- Law of diminishing returns
- Overall effect of an enhancement is weighted by proportion of time that the enhancement is used



Example

- Suppose that we want to enhance the processor used for Mail Server. The new server is 10 times faster on computation in the mail server application. Assuming that the original processor is busy with computation 40% of the time and is waiting for I/O 60% of the time, what is the overall speed up gained by incorporating the enhancement?

Example

- If we can make all FP instructions in the graphics processor run faster by a factor of 1.6; FP instructions are responsible for half of the execution time for the application, calculate the speedup.

Processor Performance

- CPU time= CPU cycles for a program x cycle time
- Clock rate (Hz)= 1/ (cycle time)
- CPI= CPU cycles for a program / IC
  - IC - Instruction Count
  - CPI - Cycle per instruction

Example

- Given the following measurements:
  - Frequency of FP operations = 25%
  - Average CPI of FP operations = 4.0
  - Average CPI of other instructions = 1.33

  If we can decrease the average CPI of all FP operations to 2.5, calculate the speedup.