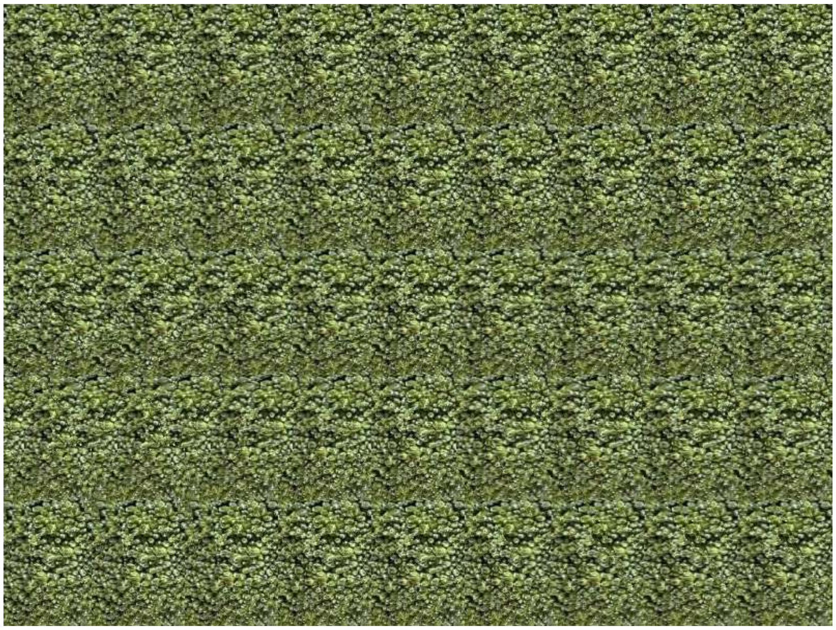


Topic 4: Performance

COS / ELE 375

Computer Architecture and Organization

Princeton University
Fall 2015

Prof. David August

Which Aircraft Is Best?

Aircraft	Passengers	Range (Miles)	Speed (mph)
Boeing 737-100	101	1540	598
Boeing 747	470	4150	610
Concorde	132	4000	1350
Douglas DC-8-50	146	8720	544

Longest Range?

Aircraft	Passengers	Range (Miles)	Speed (mph)
Boeing 737-100	101	1540	598
Boeing 747	470	4150	610
Concorde	132	4000	1350
Douglas DC-8-50	146	8720	544

- Suitability to task

Fastest?

Aircraft	Passengers	Range (Miles)	Speed (mph)
Boeing 737-100	101	1540	598
Boeing 747	470	4150	610
Concorde	132	4000	1350
Douglas DC-8-50	146	8720	544

- Suitability to task
- Customer Latency (time of a trip)

Biggest Capacity?

Aircraft	Passengers	Range (Miles)	Speed (mph)
Boeing 737-100	101	1540	598
Boeing 747	470	4150	610
Concorde	132	4000	1350
Douglas DC-8-50	146	8720	544

- Suitability to task
- Customer Latency
- Customer Bandwidth (number of passengers in a trip)

Largest Throughput?

Aircraft	Passengers	Speed (mph)	Passenger-mph
Boeing 737-100	101	598	60,398
Boeing 747	470	610	286,700
Concorde	132	1350	178,200
Douglas DC-8-50	146	544	79,424

- Suitability to task
- Customer Latency
- Customer Bandwidth
- Customer Throughput (passenger trips per unit time)

Which Aircraft Is Best?

Aircraft	Passengers	Speed (mph)	Passenger-mph
Boeing 737-100	101	598	60,398
Boeing 747	470	610	286,700
Concorde	132	1350	178,200
Douglas DC-8-50	146	544	79,424

- Suitability to task
- Customer Latency
- Customer Bandwidth
- Customer Throughput
- Cost to purchase? Operation cost? Safety?

Defining Performance

What is important to whom?

Computer system user:

- response time - related to: program elapsed time
- elapsed_time = time_end – time_start
- Lower elapsed time for program is better

Computing center manager:

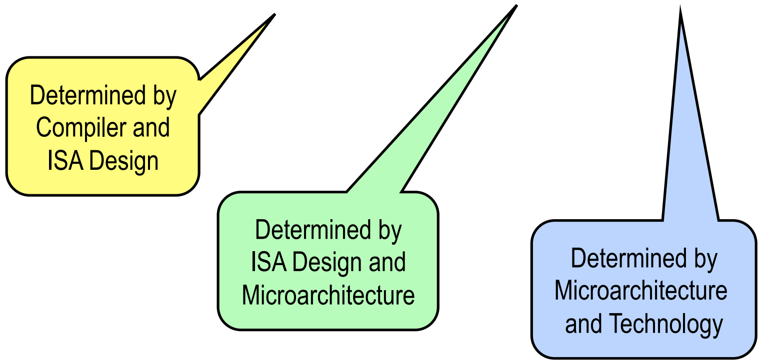
- throughput - job completion rate
- job completion rate (#jobs/second)
- Larger job completion rate (throughput) is better

Improving Performance

- Response Time, Throughput, or Both?
- If we upgrade a machine with a new processor what do we increase?
- If we add a new machine to the lab what do we increase?

Response Time Measurement

$$CPU\ time = \frac{Instructions}{Program} X \frac{Cycles}{Instruction} X \frac{Seconds}{Cycle}$$



Response Time Measurement

$$CPU\ time = \frac{Instructions}{Program} X \frac{Cycles}{Instruction} X \frac{Seconds}{Cycle}$$

- Performance is inverse of CPU time
- Dynamic Instructions
- Instruction-Level Parallelism
  - CPI - Cycles per Instruction
  - IPC - Instructions per Cycle

Throughput Measurement

- Rates: Units of work per unit time
- Examples:
  - millions of instructions / second (MIPS)
  - millions of floating point instructions / second (MFLOPS)
  - millions of bytes / second (MBytes/sec)
  - millions of bits / second (Mbits/sec)
  - images / second
  - samples / second
  - transactions / second (TPS)

Beware: MIPS and MFLOPS

- MIPS = instruction count / (execution time x 10^6)
- MIPS = clock rate / (CPI x 10^6)
- MFLOPS - MIPS for floating point operations
- But MIPS has serious shortcomings…
- When is MIPS OK?
- What about clock rate?

Meaningful Rates

Use rates that measure something useful!

Example: Video Image Processing

- Bad: MFLOPS
  - Number of FLOPS depends on algorithm
  - O(n^2) matrix-vector product vs. O(n log n) FFT
- Better: frames/sec
  - A faster running program will process more frames per second
  - Frames/sec measures speed of target application

Processor Performance

Aircraft have many applications
Computer systems have many applications

- Scientific computing
- Transaction processing
- Real-time systems
- Multimedia applications
- Commercial workloads
- Software development

Systems will perform differently in each domain

Use Benchmark Suites

Benchmark suites are designed to standardize the evaluation of machines

Suites just from **Standard Performance Evaluation Corporation:**

SPECapc, SPECviewperf, SPEC HPC2002, SPEC OMP
SPEC CPU2000, SPECjAppServer2001,
SPECjAppServer2002, SPEC JBB2000 ,
SPEC JVM98, SPEC MAIL2001, SPEC SFS97_R1,
SPEC WEB99, SPEC WEB99_SSL

Choose the suite to match a particular domain

Beware: Kernels

Kernels are extracted from programs
Meant to be the essence of an application

Example: Olden

- Something is often lost in the kernel-ization
  - Monolithic task
  - Small is more regular
- Some programs in Olden produce no output!
  - Compiler Optimization

Why is Olden called Olden?

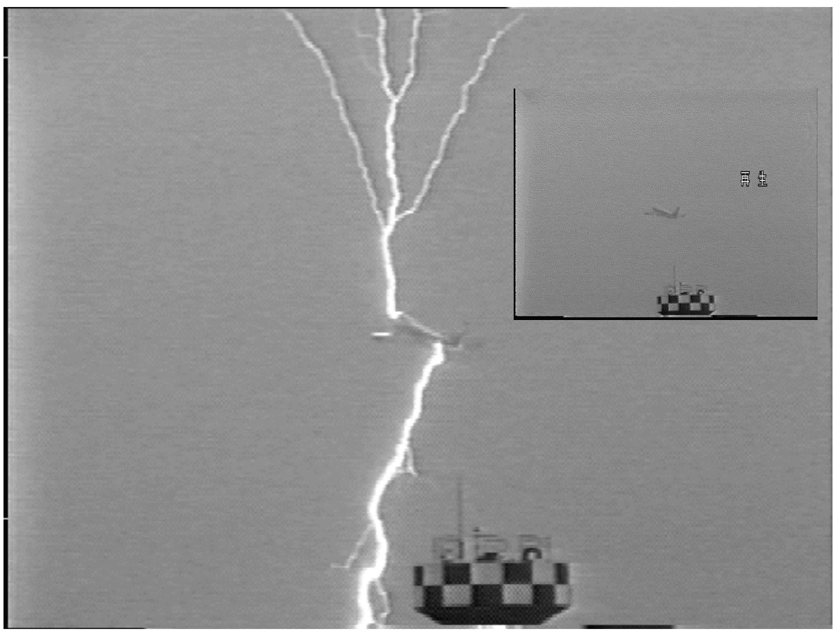
Beware: Peak Rates

- Example:
  The i860 is *advertised* as having a peak rate of 80 MFLOPS (40 MHz with 2 flops per cycle).

- Measured MFLOPS tell a different story:

Kernel	1D FFT	SASUM	SAXPY	SDOT	SGEMM	SGEMV	SPVMA
MFLOPS	8.2	3.2	6.1	10.3	6.2	15.0	8.1
% Peak	11%	4%	7%	13%	8%	19%	10%

- Peak MFLOPS: MFLOPS obtained for some contrived (and mostly likely useless) scenario.
- Peak rates are useless!



Emerging Issue: Transient faults

- Randomly change bits of state element or computation
- Caused by external energetic particle striking processor
- Cannot test for fault before hardware use

0x 675309

*   0x42

0x32AA36852

Severity of Transient Faults

- IBM historically adds 20-30% additional logic for mainframe processors for fault tolerance [Slegel 1999]
- In 2000, Sun server systems deployed to America Online, eBay, and others crashed due to cosmic rays [Baumann 2002]
- In 2003, Fujitsu released SPARC64 with 80% of 200,000 latches covered by transient fault protection [Ando 2003]

- Processors are becoming more susceptible…
  - lower voltage thresholds
  - increased transistor count
  - faster clock speeds

Reliability Metrics?

- Mean Time To Failure (MTTF)
- Mean Instructions To Failure (MITF) (Weaver et al. ISCA 04)
- Mean Work To Failure (ISCA 05)
  - Generalization of Mean Instructions To Failure Instructions not constant unit of work in hybrid systems

- What do you think?

Relative Performance

<u>Absolute time measure:</u>
- Straightforward measurement of time a task takes
- AKA: running time, elapsed time, response time, latency, completion time, execution time

<u>Relative (normalized) time measures:</u>
- Running time normalized to some reference time
- task_time / reference_time (time = 1 / performance)
- Used to compare machines:
  - Machine A finishes task in 10 seconds
  - Machine B finishes task in 15 seconds
  - Machine A is (15 seconds / 10 seconds) 1.5x faster than B
  - Machine A is 50% faster than B

Travel Time

- You plan to visit a friend in Turkey
- Concorde to Paris + 737 to Istanbul = $3500
- 747 to Paris + 737 to Istanbul = $1200

Equipment	New York to Paris	Paris to Istanbul	Total
747 + 737	8 Hours	4 Hours	12 Hours
SST + 737	3 Hours	4 Hours	7 Hours

- Taking the SST (which is 2.7 times faster) speeds up the overall trip by only a factor of 1.7!
- Teleporter to Paris? (Teleporter is 10^6 times faster)

Amdahl's Law

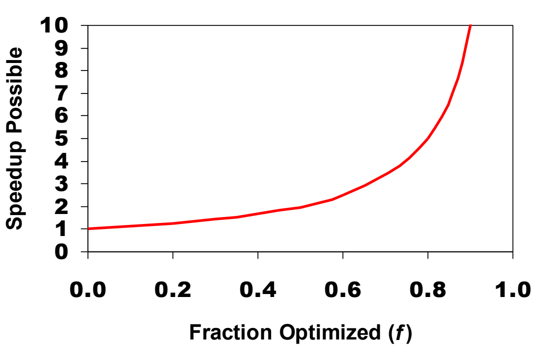
- Fraction optimized limits overall speedup
- Amdahl's Law:

$$Speedup = \frac{1}{1 - f + \frac{f}{s}}$$

where f is fraction optimized,
s is speedup of that fraction

Amdahl's Law

Speed Enhancement is limited by fraction optimized:



$$\lim_{s \to \infty} \frac{1}{1 - f + \frac{f}{s}} = \frac{1}{1 - f}$$

where f is fraction optimized,
s is speedup of that fraction

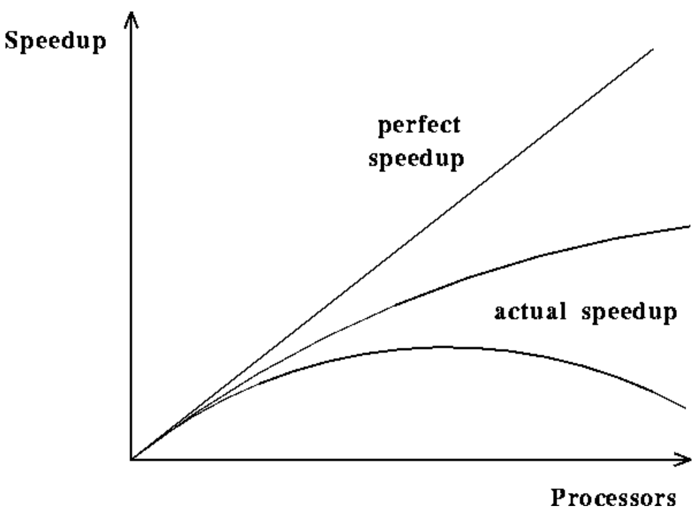
Parallelism

Parallel Processing - throw more processors at problem
- 1024 parallel processors - LOTS OF MONEY!
- 90% of code is parallel (f = 0.9)
- Parallel portion speeds up by 1024 (s = 1024)
- Serial portion of code (1-f) limits speedup

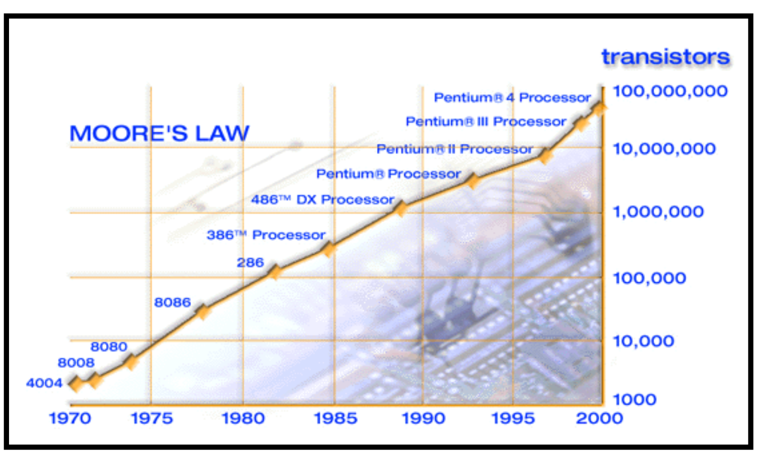
$$\lim_{s\to\infty} \frac{1}{1-f+\frac{f}{s}} = \frac{1}{1-f}$$

Serial portion limits to 10x speedup!

Reality

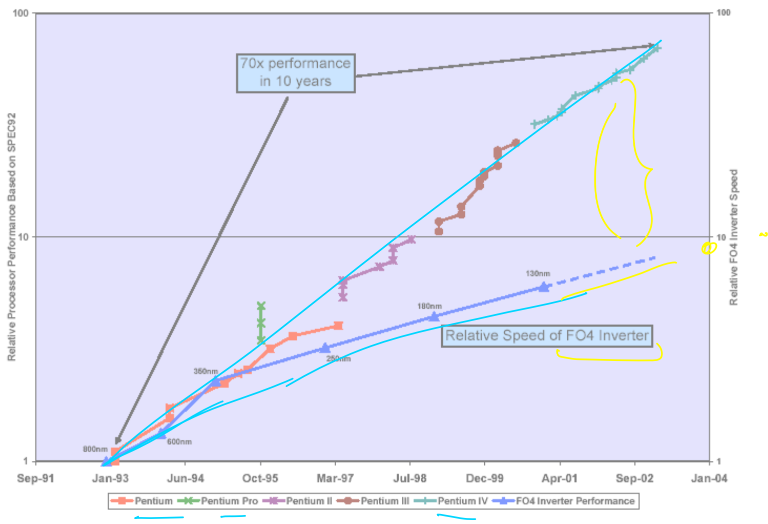


Moore's Law

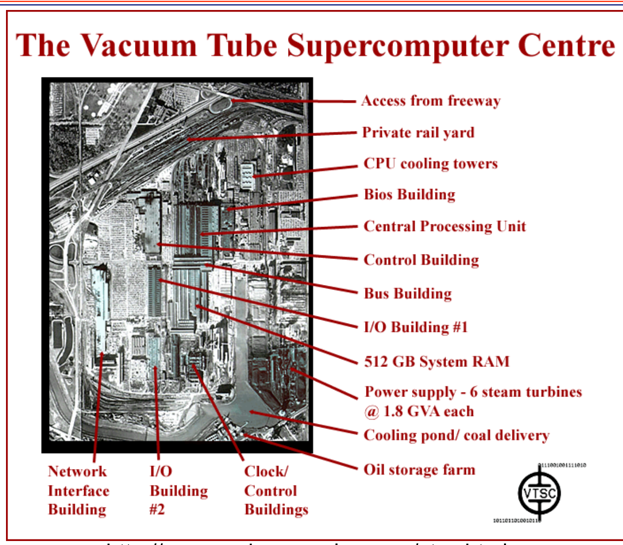


"Grove giveth and Gates taketh away." - Bob Metcalfe (an inventor of Ethernet)

The Importance of Computer Architecture



Without the Transistor:



http://www.ominous-valve.com/vtsc.html

Technology Trends

- Processor
  - Logic Capacity: ~30% increase per year
  - Clock Rate: ~20% increase per year
- Memory
  - DRAM Capacity: ~60% increase per year
  - Memory Speed: ~10% increase per year
  - Cost per Bit: ~25% decrease per year
- Disk
  - Capacity: ~60% increase per year

Summary

- Beware of metrics in general (MFLOP, MIPS)
- Beware of peak measurements
- Beware of kernels
- Relative and Absolute Performance
- Moore's law
- Amdahl's law
- IPC/CPI
- The Memory Wall

$$Speedup = \frac{1}{1 - f + \frac{f}{s}}$$

$$CPU\,time = \frac{Instructions}{Program} X \frac{Cycles}{Instruction} X \frac{Seconds}{Cycle}$$