Computer Science 703

Advance Computer Architecture

2010 Semester 1

Lecture Notes

23Apr10

Methods: Tools & Benchmarks

James Goodman

Department of Computer Science

Evaluation

- Why?
  - Quantitative information is important
- What level?
  - Appropriate for the question
- How?
  1. Mean-value analysis/Analytical modelling
  2. Simulation
  3. Build it
- What?
  - Computer performance modelling
  - Benchmark applications

A Simple Example: Checkout

- Grocery store has one or more lanes, multiple customers
  - Too many lanes open requires more servers (costs money)
  - Too few lanes open drives away customers (costs money)
- Queuing model
  - How many lanes?
  - How many customers?
  - How does one checkout take?
  - How long does a customer have to wait?
  - What is probability customer has to wait?
  - How much of the time is a server idle?
  - When would a single queue work better?

Simple Queueing Model

- Assume
  - Fixed number of servers
  - Constant (but random) arrival rate of customers
- May also want to assume
  - Single queue, single server
  - Constant service time
- Can derive analytically
  - Mean service time
  - Mean length of queue
  - Mean idle time of server
  - Relative costs for different models

Analytical Model vs. Simulation

- Analytical model
  - Must be simple to be solvable!
  - If solvable, can be done very quickly using a computer
- Simulation model
  - Can be arbitrarily detailed (i.e., complex)
    - Input data can be actual data, or output from another simulator
  - Can be arbitrarily slow
  - Simulator can be purpose-built to make it faster, but
  - Flexibility is important
    - Often want to ask questions not anticipated

Event-driven Simulation

- A change of state is called an event.
  - Might be regular & predictable (fixed interval) or variable
  - Might be unpredictable (e.g., interrupt)
- An event triggers an activity - a unit of work - in the simulator
- An activity typically causes the creation of further events.
- A logically-related set of activities constitutes a process.
- As simulation proceeds, simulation time advances in steps (number of events), and reflecting simulated activities

Time-based Simulation

- Time steps are regular
  - Constant interval: a clock
  - Variable interval: instruction time
- Size of step is critical for accuracy & efficiency
  - too course: loss of accuracy
  - too fine: simulation may be very slow

Simulating a Computer

- Two parts to simulate
  - Sequence of instructions executed
    - Programme may be deterministic
      - Can use a log (trace)
  - Time to execute programme
    - May need to execute many times for combination of parameters
- Timing model
  - Simple for simple processor
  - Hard for pipelined processor
  - Really hard for out-of-order processor
  - Cache simulation?
- Multiprocessor simulation?

Results

- Results are counts of occurrences
  - How many clock cycles
  - How many arrivals
  - How many failures
  - How many ???
- Results may be repeatable
  - If inputs are repeatable
  - Have to introduce randomness!
  - Multiprocessors often introduce randomness
  - Small changes in input can produce large changes in results
- Multiple simulations may be required
  - Multiple parameters, multiple combinations
- Would like to simulate multiple cases simultaneously

Validation

- Can you believe the results?
  - Do you know how to interpret the results?
  - Do you understand what is being modelled?
  - Do you understand how it is being modelled?
- Results can mislead regarding accuracy
  - Exact, repeatable counts are precise
  - Accuracy depends on assumptions
- Studying results: sanity checks
  - Compare against real system (or similar real system)
  - Compare small variations in parameter(s)
  - Compare against other methods
    - Other simulators
    - Analytical models
  - Experience is important

How to Report Performance

- Clock rate?
- MIPS? MFLOPS?
- Peak performance?
- Time to execute a programme?
  - What programme?

Quotations from my Father

"Figures don't lie, but liars do figure!"

Programme Benchmarks

- What is an appropriate programme?
  - Choice of programme can dramatically affect results
  - Changes in some parameters can dramatically affect results
- What is a "good" benchmark?
- All benchmarks have problems
  - Should accurately reflect real-world programmes
  - Should be portable
  - Should be repeatable
  - Should capture all aspects of performance
- A suite of benchmarks is valuable
  - More widely distributed, the better
  - But computers change, benchmarks get old…

Combining Multiple Benchmarks

- J.E. Smith, "Characterizing computer performance with a single number," *CACM*, v. 31, #10 (October 1988), pp. 1202-1206.
  - Arithmetic mean (of MFLOPS)
  - Geometric mean
  - Harmonic mean

"… the time required to perform a specified amount of computation is the ultimate measure of computer performance."

Property 1. A single-number performance measure for a set of benchmarks expressed in units of time should be directly proportional to the total (weighted) time consumed by the benchmarks.

Property 2. A single-number performance measure for benchmarks expressed as a rate should be inversely proportional to the total (weighted) time consumed by the benchmarks.