Benjamin Cho
EE382M-15

Scribing – September 25th

SPEC CPU2006 Sensitivity to Memory Page Sizes

Wendy Korn (IBM), Moon S. Chang (IBM)

- Good thing about this paper: It is possible to build the system based on the info given in the paper and reproduce the results.
- Two of SPEC's criteria
  - It should spend over 95% of the execution time in the submitted code.
    - ✧ System calls & libraries – if used heavily, not picked. Benchmarks will be sensitive to those libraries.
  - It use less than 900MB of memory in 32-bit mode.
- IBM Power 5+ processors
  - It was the latest processor then.
  - Interesting point: Instruction cache > data cache
    - ✧ Data cache won't fit any way.
  - SMT processor – doesn't mean much for SPEC2006
  - IBM POWER5 MMU
    - ✧ Three types of address translation
    - ✧ TLB (translation look-aside buffer)
    - ✧ SLB (segment look-aside buffer)
    - ✧ ERAT (effective to real address table)
- POWER5 PMU (Performance Monitor Unit)

- Reference paper: Maron, B., Chen, T., Vianney, D., Olszewski, B., Kunkel S., Mericas, A. Workload Characterization for the Design of Future Servers. Proceedings of IEEE International Workloads Characterization Symposium (IISWC), 2005.
- 2 dedicated registers that count
  - ✧ instruction completed
  - ✧ cycles
- The way to measure performance
  - ✧ run many times → discard highest/lowest values and use the ones in the middle
- 4 programmable counters, can count 300+ events from CPU
- The meaning of the numbers after L, such as L2, L25, L35, L275…
  - ✧ Whether the cache is near/far by/from processor
  - ✧ Clarifies the Non-uniform cache access latency in this way
  - ✧ Depends on the physical location and the distance

- AIX Support for Multiple Page Sizes
  - 4 different page sizes supported
  - Boot-time: determines the number of 4KB and 64KB pages
  - 3 regions of address space: text, data, stack
  - Kernel uses 64KB pages for shared library
  - 16MB only available for text and data
  - Vmo instruction: enables large pages
- Multiple page size support
  - 3-ways to bind a page size to a particular executable
    - ✧ Linker options
    - ✧ Linker tool
    - ✧ Environment variables
  - Three different page sizes

        - ✧ Small pages (4KB)
        - ✧ Medium pages (64KB)
        - ✧ Large pages (16MB)
- Data collection
    - Used OS commands like svmon, perf-counters
    - Speed-run: running one instance of the benchmark on a single core
    - As opposed to rate-run (spec-rate):
        - ✧ SPEC introduced this concept when multicore came out.
        - ✧ 4 cores 4 copies – then, execution time divided by 4
        - ✧ You can't run different benchmarks because of conflicts in shared caches.
    - Snapshot of text, data, and library regions
- Average and maximum memory usage with various page sizes
    - In appendix A, small different no huge difference
    - gcc: weird drop in avg with 16MB page size, difficult to explain
    - cactusADM: huge difference
- About Figure 1
    - CPI stack: translation miss dominates
    - Not overlapped in time domain
    - 471.omnetpp: good special locality
    - No different between 64KB and 16MB
- About Figure 2
    - No branch misses, i-cache misses

- GM is consistent but consistently wrong.
- What makes good mean?
    - Time-based mean: should be directly proportional to total weighted time

- Rate-based mean: If time doubles, mean value should reduce by half.
- What means satisfy these criteria?
  - ✧ AM: proportional to sum of inverse of times, but not appropriate for summarizeing rate
  - ✧ HM: not directly proportional to sum of time, but good for summarizing rate
  - ✧ GM: maintains consistent relationship when comparing normalized values, but wrong
    - Not good for time nor for rate

- One example that GM works: 2,3,6 → 3,4,7
  - GM(3/2, 4/3, 7/6) = 1.326
- Any situation in which GM is correct?
  - Read the paper: War of the benchmark means: time for a truce (John R. Mashey)
  - Validity of GM in certain situations