Scribe notes 9-30-14

HINT: A New Way to Measure Computer Performance

- SPEC directly relates to actual applications (a.k.a. application based benchmarking)
- HINT is a new way of looking at benchmarks
- There are different types of benchmarks:
  - Fixed-Computation Benchmarks
    - Example: How long does it take you to walk 5 miles?
    - Compare different designs based on time it takes to do the task.
    - Who does the computation faster?
    - Measures speed (Mflops, MIPS, etc.)
      - Problems with MIPS: millions of instructions per second, what are the instructions? Fluff? Same job can be done with different instruction counts.
      - Problems with MFLOPS: Instructions aren't standardized. They have different speeds
  - Fixed-Time Benchmarks
    - Example: walk for an hour, how far do you get?
    - Measure the amount of computation a design is able to do in fixed time
    - Example: Count number of prime numbers found in fixed time
    - SLALOM made by Gustafson earlier
      - First fixed-time variable computation strategy
      - Compute radiosity accuracy in 1 minute
      - Benchmark didn't specify a specific algorithm. Can have different algorithms for radiosity and this then measures the algorithm, not the system.
      - Number of “patches” or areas which were subdivided in the 1-minute interval
      - Weaknesses:
        - Problem statement was loosely defined
        - With modifications, the complexity changed from O(n^3) → O(n^2) → eventually O(nlogn)
        - Certain assumptions can hurt your benchmarks, like whetstone which assumed no dead code elimination
        - Non-linear algorithm, therefore the performance metric is non-linear. Computation should be linear to get similar differences.
        - Unrealistically forgiving of machines with inadequate memory bandwidth.
        - Trying to make a parallel version of this is very difficult.
  - Variable Computation & Variable Time
    - Error keeps getting smaller and smaller as time goes on (measuring quality of the answer)
    - HINT: Hierarchical INTegration
      - “keeps going until it crashes”
        - Every machine has its limits. HINT never fully fits in any cache.
      - QUIPS: QUality Improvement Per Second
      - Trying to do integral subdivision for y = (1-x)/(1+x). See pictures on slides.
        - Initial:
          - Upper bound is the whole area (256 squares), lower bound is zero.
          - Quality = (max area)/(U-L) = 256/(256) = 1
        - First subdivision is at x = ½

        - Refining :
            - The top right (white) area is removed from upper bound calculation.
            - The bottom left (purple) area is added to lower bound calculation
            - U = 256 – 80 (white boxes)
            - L = 0 + 40 (purple boxes)
            - Q = 256/(U-L) = 256/136
    - Hierarchically subdivide top-left blue area, and bottom-right blue area.
        - For each blue area, add the guaranteed under-the-curve boxes (new purple boxes) to the lower bound. These come from the old blue area (now purple) that are found to be definitely under the curve in that subdivision. Subtract the guaranteed boxes to be OVER the curve (white boxes) from the upper bound. This will produce more “blue” areas each time for further inspection.
- Results: QUIPS curve, and net QUIPS
    - QUIPS curve:
        - QUIPS curve (KQUIPS or MQUIPS), the rate of quality improvement over time, is compared versus the amount of HINT runtime.
        - As data keeps being added to the structure with each subdivision, the data stops fitting in the L1 cache. Then L2, and then main memory follow the same trend, and QUIPS decreases.
        - SPEC always worries about fitting in the cache. HINT doesn't need to worry since cache usage grows over time.
        - SPEC benchmarks can measure machines at specific points along the QUIPS curve, and this can limit your view of the machine. You can find the specific point on the graph for your specific application.
    - Net QUIPS:
        - This is area under the QUIPS curve.
        - This gives a single number metric that many people want.
- Complaints:
    - Some say the function itself has no meaning. Humans don't have the ability to relate their workloads to QUIPS or to convince other people it is a good machine when the measurement doesn't really relate to a specific application. Gustafson: Just need to convince people.