Benchmarking CPU Performance

- Many benchmarks available
- MHz (cycle speed of processor)
- MIPS (million instructions per second)
- Peak FLOPS
- Whetstone
  - Stresses unoptimized scalar performance, since it is designed to defeat any effort to find concurrency.
  - Popular way to estimate MIPSMFLOPS (million floating point operations per second)

Benchmarking CPU Performance

- SPEC benchmarks (SPECint, SPECfloat, SPECmark)
  - Maintained by a consortium of workstation vendors
  - Frequently-changing collection of programs one might run on a workstation, plus kernels like matrix multiplication
  - It is virtually impossible to track SPEC performance from one year to the next since the definition of the problem set is always changing
- LINPACK - dense linear solver with partial pivoting
  - 100x100, 1000x1000 or larger

Benchmarking CPU Performance

- NAS parallel benchmark (NPB) - a small set of programs designed to help evaluate the performance of parallel supercomputers.
- Derived from computational fluid dynamics (CFD) applications
- Consist of five kernels and three pseudo-applications in three sizes called Sample Code, Class A and Class B
  - Embarrassingly parallel EP
  - Multigrid MG
  - Conjugate gradient CG
  - D FFT PDE FT
  - Integer sort IS
  - LU solver LU
  - Pentadiagonal solver SP
  - Block tridiagonal solver BT

Benchmarking CPU Performance

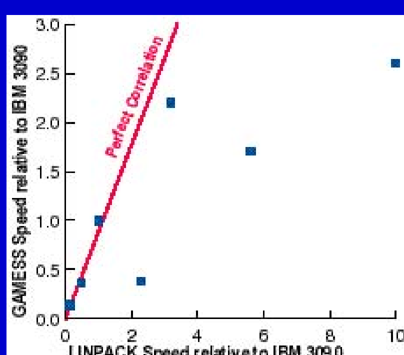
- STREAM, peak memory bandwidth
  - small collection of very simple loop operations
  - tries to estimate the total rate at which all addressable memory spaces can deliver data to respective processors
- **Fhourstones, Dhrystone, nsieve, heapsort, Hanoi, queens, flops, fft, mm**
  - assorted integer and floating-point benchmarks for small problems

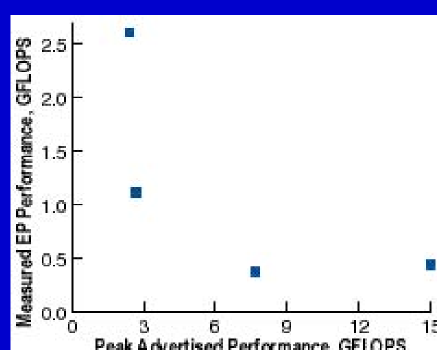
Problems with Benchmarks

- Benchmark performance does not necessarily correlate with application performance
- Performance on two benchmarks may not correlate
- Benchmarks problems tend to be small, easily portable and easy to explain
- As speed increases the benchmarks run too quickly and must be redefined
- Benchmarks tend to measure performance for a particular size of problem

Correlation Examples

- LINPACK v GAMESS computational chemistry application
- Peak FLOPS v FLOPS from NAS benchmark 1
- Correlation is -0.692



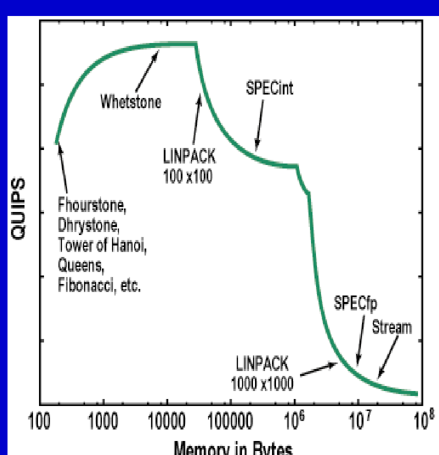


HINT - an attempted synthesis

- HINT benchmark created in 1995 at Ames DOE Laboratory by John L. Gustafson and Quinn Snell
- Problem with previous benchmarks - tended to emphasize one part of performance curve
- HINT - Hierarchical INTegration tries to produce curve rather than number
- Aim: to provide a scalable benchmark that reflects the type of work done in iterative refinement

DiSCoV Fall 2008 Paul A. Farrell Cluster Computing 7

HINT -relation to previous benchmarks



- Hint shows the effects of cache size, and memory size
- Corresponds to
  - Fhourstone etc initially
  - LINPACK 100x100
  - SPECint around 100K
- Eventually corresponds to Stream - a benchmark for memory performance when end up using Virtual Memory

DiSCoV Fall 2008 Paul A. Farrell Cluster Computing 8

HINT

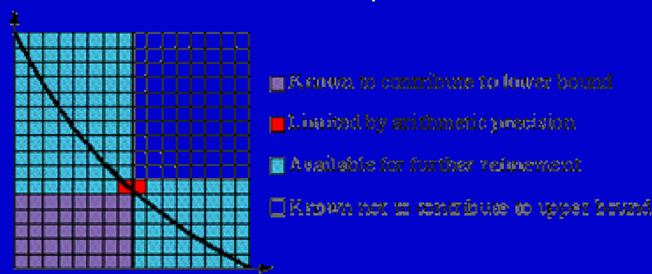
- Infinitely scalable
- Speed defined by quality improvement per second (QUIPS). "Quality" is the reciprocal of the error, which combines precision loss and discretization error.
- The problem can be run with any data type: floating point (any precision), integer (any precision), extended-precision arithmetic, etc
- HINT provides a graph of performance, it also has a "single number" measure (the area under the graph) that summarizes performance
- As the size of the HINT task grows, the memory access pattern becomes more complicated in a way that defeats caches.

Hint Algorithm

- Use interval subdivision to find rational bounds on the area in the xy plane for which x ranges from 0 to 1 and y ranges from 0 to (1- x) / (1+ x).
- Subdivide x and y ranges into 2^k equal subintervals and count the squares thus defined that are completely inside the area (lower bound) or completely contain the area (upper bound).
- The function (1- x) / (1+ x) is monotone decreasing, so the upper bound comes from the left function value and the lower bound from the right function value on any subinterval.
- No other knowledge about the function may be used.

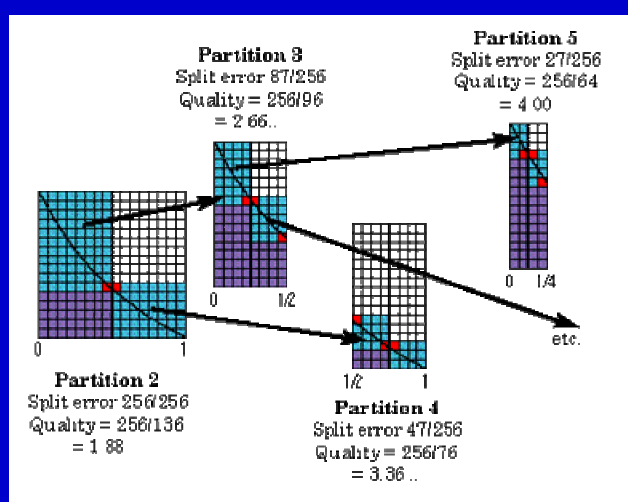
HINT - first subdivision

- Bounds after subdivision into two intervals
- Upper left and lower right contain 87 and 47 squares
- 87-square region should be subdivided
- 47-square error will then move to the front of the queue of subintervals to be split



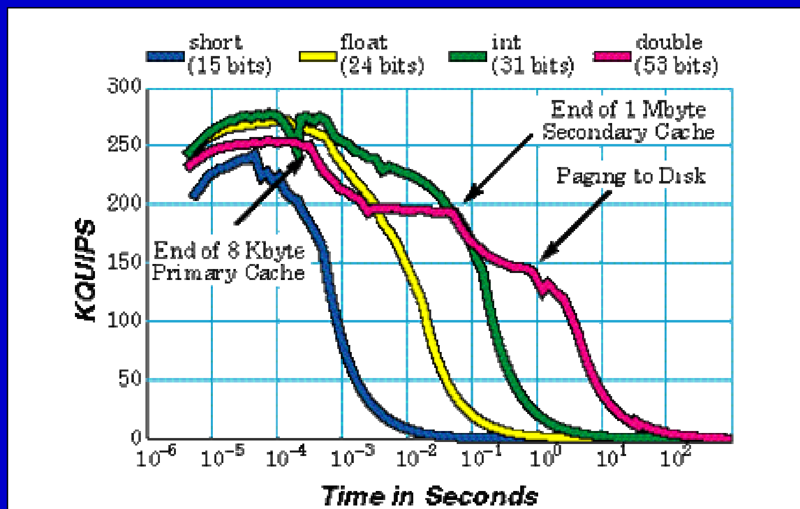
DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 11

HINT Illustration



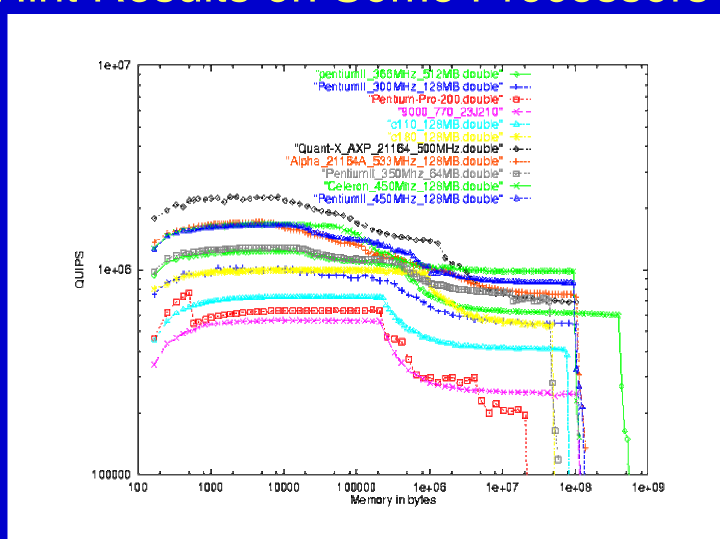
DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 12

Different Precisions



DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 13

Hint Results on Some Processors



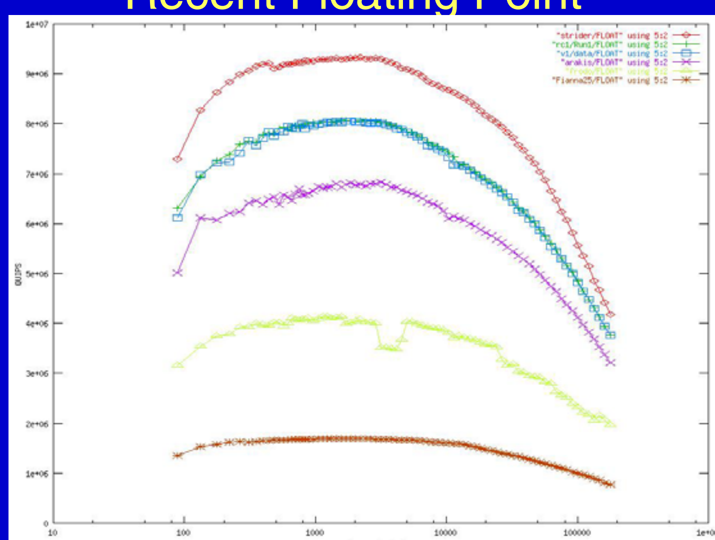
DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 14

Recent Results - Machines

- Strider
  - AMD Athlon(tm) MP 2100+ (1733.41 Mhz)
  - 256KB cache each, 3 GB of memory. **Memory bus speed**
- Rc1, v1 (RocketCalc nodes)
  - Intel Pentium Xeon 2.4 GHz, 512 KB cache
  - 8 GB Registered ECC DDR SDRAM per processor
- Arakis
  - Intel Pentium 4 2.4Ghz. 512KB cache
  - Asus P4T533-C motherboard, 850E BIOS, 533/400 MHz FSB, 1GB RDRAM
- Frodo
  - Intel Pentium 4 1.5GHz, 256KB cache
  - RDRAM
- Fianna25
  - Intel Pentium III/450 MHz, 512 KB cache, 256MB PC100 Compliant SDRAM

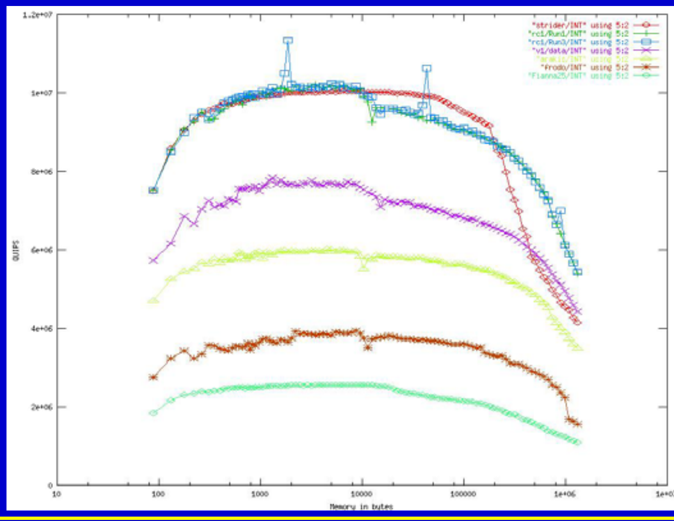
DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 15

Recent Floating Point



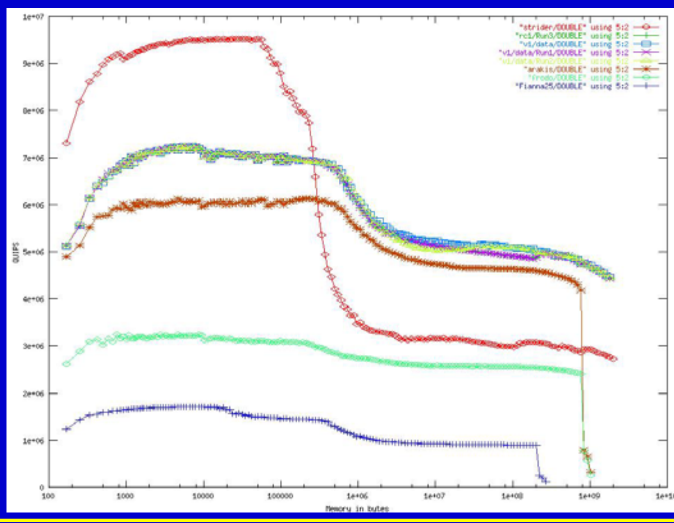
DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 16

Recent Integer



DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 17

Recent Double Precision



DiSCoV KENT STATE UNIVERSITY Fall 2008 Paul A. Farrell Cluster Computing 18

References

- http://discov.cs.kent.edu/resources/perf/hint/Publications
- http://discov.cs.kent.edu/resources/perf/hint/