HPC Challenge Benchmark

Piotr Łuszczek

University of Tennessee Knoxville

SC2004, November 6-12, 2004, Pittsburgh, PA

Contents

Motivation and Sponsors for HPC Challenge

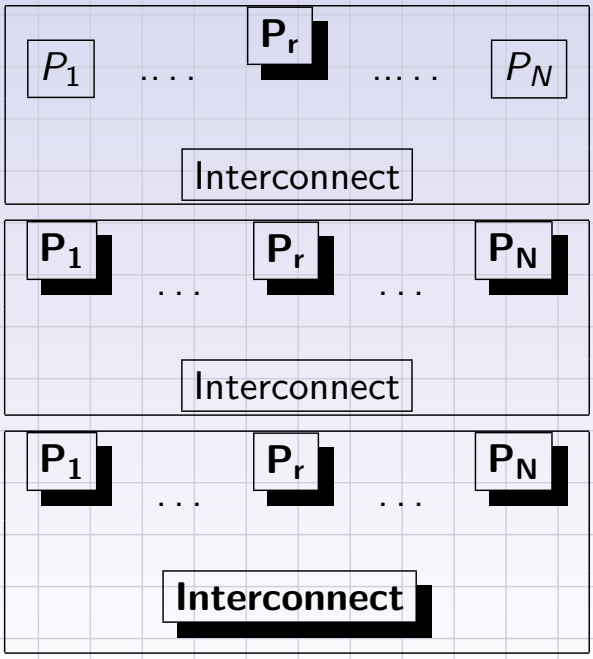
- Uniform benchmarking framework for performance tests
- Measure performance of various memory access patterns
- Testing Peta-scale systems
  - Has to challenge all hardware aspects
- Analyzing productivity
  - Implementation in various programming languages
  - Architecture support
- Rules for running and verification
  - Base run required for submission
  - Optimized run possible
  - Verification
  - Reporting all aspects of run:
    compiler, libraries, runtime environment
- Sponsors
  - High Productivity Computing Systems (HPCS)
  - DARPA, DOE, NSF

Active Collaborators

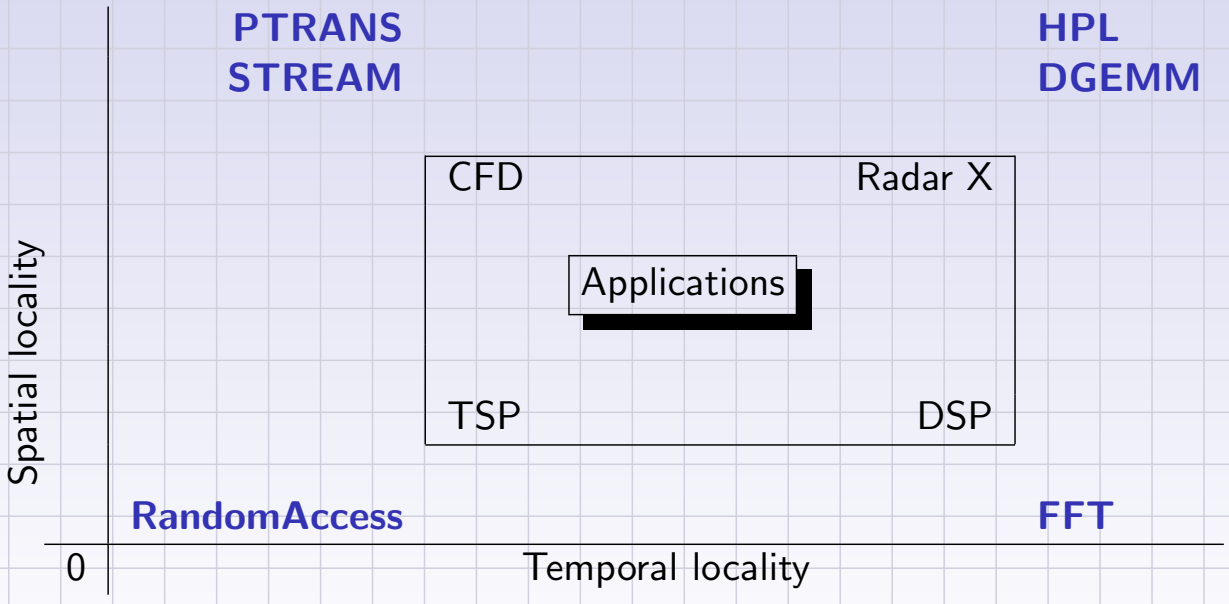
David Bailey	NERSC/LBL
Jack Dongarra	UTK/ORNL
Jeremy Kepner	MIT Lincoln Lab
David Koester	MITRE
Bob Lucas	ISI/USC
John McCalpin	IBM Austin
Rolf Rabenseifner	HLRS Stuttgart
Daisuke Takahashi	Tsukuba

Testing Scenarios

- Local
- Embarrassingly Parallel
- Global



Performance Bounds: Memory Access Patterns



Effective performace peak: HPL and DGEMM

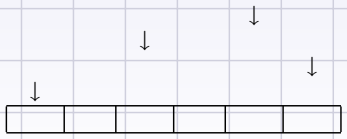
- Effective performance peak (unit: TFlop/s and GFlop/s)
  - Global (entire system): High Performance Linpack (HPL)
  - Local (single node): DGEMM
- Top500 November 2004: 16%-99% of peak
  - Entries #99 and #309
- HPL – High Performance Linpack
  - Written by Antoine Petitet (while at ICL)
  - Non-trivial configuration
    - Global matrix size ($\approx$ total memory)
    - Process grid ($\approx$ square)
    - Blocking factor (for BLAS and BLACS)
    - Described at `http://www.netlib.org/benchmark/hpl/`
  - Runs well on CISC, RISC, VLIW, and vector computers
- DGEMM is matrix-matrix multiply with double precision reals.

Application Bandwidth: PTRANS and STREAM

- Measures sustainable bandwidth for stride one access
  - Global: PTRANS
  - Local: STREAM
- PTRANS – parallel matrix transpose
  - Repeated exchanges of large amounts of data
  - Depends on global bisection bandwidth
- STREAM – simple linear algebra vector kernels
  - Well known and understood
  - Known optimizations
    - No cache allocation on Crays
    - Threading on IBMs

Irregular Memory Updates: RandomAccess (GUPS)

- Measures ability to hide latencies (local and global)
  - Bandwidth (almost) irrelevant
  - Important: capacity for simultaneous message
  - Irregularity in data access kills common hardware tricks
- Many implementations
  - MPI-1: non-blocking `Send()`/`Recv()`
  - MPI-2: uses `Put()`/`Get()`
  - UPC: much faster than all above
- Verification procedure
  - Up to 1% updates may not be performed
  - Allows loosening shared memory consistency



Fast Fourier Transform with FFTE

- Complex 1D, double precision DFT
  - 64-bit input vector size
  - No mixed-stride memory accesses (as in multi-dimensional FFTs)
- Scalability problems
  - "Corner turns"
    - Global transpose with `MPI_Alltoall()`
    - Three transposes (data is never scrambled)
  - But time is not an issue – it runs fast

Rules for Running and Reporting

- Base run is required to submit to the database
  - Reference MPI-1 implementation publicly available
  - Each test is checked for correctness
- Optimzed runs may follow the base run
  - Performance critical (timed) portion of code can be changed
  - Changes are to be described upon submission
    - Records effort (productivity) and architecture optimization techniques
  - Correctness check doesn't change
- Results submitted via web form
  - Output file from the run
  - Hardware information
  - Programming environment: compilers, libraries
  - Submission must be confirmed via email
  - Data immediately available (no restrictions)
    - HTML
    - XML
    - Microsoft Excel

Submission Statistics

- Army computing centers: ARL, ERDC, NAVO, ...
- Government labs: ORNL
- Hardware vendors/integrators
  - Chip makers: Cray, IBM, NEC
  - Integrators: Dalco, Scali
- Universities
  - Europe: Aachen/RWTH, Manchester
  - Asia: Tohoku (Sendai, Japan)
  - North America: Tennessee
- Supercomputing centers
  - DKRZ (Hamburg)
  - HLRS (Stuttgart)
  - OSC (Ohio)
  - PSC (Pittsburgh)
- Countries Germany, Japan, Norway, Switzerland, U.K., U.S.A.
- Interconnects
  - Crossbar
  - Fat tree
  - Omega
  - Tori: 1D, 2D
- Processors
  - CISC
  - RISC
  - Vector
  - VLIW

Planned Activities

- Code improvements
  - New languages: Fortran 90, UPC, CAF, ...
  - Automated configuration
- Website/submission improvements
- End-user tools for data analysis
- Reporting guidelines
  - Especially for vendor comparisons
  - Cores
  - Processors
  - Threading
    - OpenMP
    - HyperThreading, Simultaneous Mulithreading, ...
    - ViVA (Virtual Vector Architecture)

ICL UT
INNOVATIVE COMPUTING LABORATORY

VISIT THE ICL TEAM AT THE ORNL BOOTH (331) AT SC2004