SAND2020-1246PE

EXASCALE COMPUTING PROJECT

Using LDMS for Performance and Proxy Representativeness Characterization

Jeanine Cook (SNL) and Jefferey Kuehn (LANL)

Omar Aaziz (SNL)

Courtenay Vaughan (SNL)

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Light-Weight Distributed Metric System Architecture

LDMS Application -Level Samplers

Network (Aries)

- NIC, Router Tiles
  - Packets, FLITS, stalls
- Latency/message size
- Extract congestion metrics

Meminfo

- Memory watermark
- Large page count

PAPI (Processor)

- Frontend
- Backend
- Execution
- Cache/memory hierarchy

Kokkos

- Time stamps Kokkos/code regions
  - Correlation with other sampler data

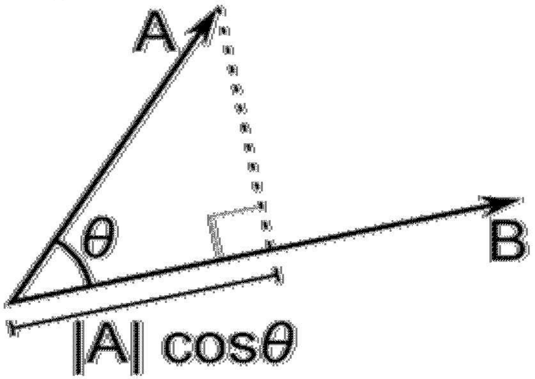
How Can LDMS be used?

- Performance characterization
  - Network, node memory
- Identify hardware bottlenecks
  - Vendor analyses such as Intel’s Top-Down Microarchitecture Analysis (TMA)
    - E.g., Memory latency and/or bandwidth bound, compute bound, etc.
- Collect application signatures
  - Cosine similarity
    - Can use it for all sorts of analysis!

LDMS is cross-platform tool: Intel (KNL, BDX, HSX, SKX), IBM P9, ARM, GPUs (coming soon)

Cosine Similarity

- Is a property of dot (inner) product in vector spaces in two or more dimensions
  - Think: “Projection of **A** in the direction of **B**”
- Uses cosθ as an angular distance metric
- Quantifies distance of A and B independent of th
  magnitude



- Application signatures ➔ Vectors A and B
  - Events that characterize system design constraints
    - E.g., Pipeline frontend

      UOPS_EXECUTED:THREAD

      UOPS_EXECUTED.CYCLES_GE_1_UOP_EXEC

      UOPS_EXECUTED.CYCLES_GE_2_UOPS_EXEC

      UOPS_EXECUTED.CYCLES_GE_3_UOPS_EXEC

      UOPS_EXECUTED.CYCLES_GE_4_UOPS_EXEC

      UOPS_EXECUTED.CORE_CYCLES_NONE

      UOPS_EXECUTED.STALL_CYCLES

      ARITH.FPU_DIV_ACTIVE

How can Cosine Similarity be Used?

- Distance between performance signatures ➔ Compare applications/workloads
  - Compare proxy/parent applications
    - Representativeness, redundancy, gaps, artifacts
  - Compare application behavior across different problems/inputs
  - Compare application behavior across different versions
    - Determine what changed (e.g., cache, memory, frontend, backend, network)
  - Performance portability
  - Set error bounds on proxy-based performance projections for parent apps
    - Future work

ECP
EXASCALE COMPUTING PROJECT

Thank you!