Research Challenges

- Can a set of tools be designed to optimize performance of an MPI library during installation?
- Can a set of dynamic tools with low overhead be designed to optimize performance on a per-user and per-application basis during production runs?
- How to configure MPI libraries on a given system to deliver optimal performance with applications?
- What kind of benefits (in terms of performance, scalability and memory efficiency) can be achieved using the proposed framework?

Proposed Framework

- MPI Library Optimization Techniques
  - Latency
  - Bandwidth
  - Overlap
  - Memory
  - MPI-T

Static Tools
- Point-to-Point
- Collectives
- Static Tools
- DMA
- Heterogeneous Systems

Dynamic Tools
- SyS. State
- Profiling
- Dynamic Tools
- MPI

Underlying Network, Processor and System Technologies
- Networking Technology
- Network Topology and Routing
- Accelerators and Co-processors

Project Milestones

- Design of lightweight profiling interface
- Architecture-based Pi-PT and collectives tuning
- Initial support since MVAPICH2-2.0
- Architecture-based tuning of RMA operations
- Dynamic tuning per application per system

Collective Tuning Framework

- MPI_Bcast : Stampede (N= system size, m= message size)
- MPI_Reduce : Stampede (N= System Size, m= Message size)
  - For each system, for each configuration (message-size, system-size) the best algorithm for both intra-node and inter-node is dynamically selected
  - Legend : if the algorithm is a two-level, then left side represents inter-node and right side represents intra-node schemes (ex: Pipeline-Bcast-shmem-intra)

Collective Tuning for MPI+OpenMP Programming

- Partial subscription nature of hybrid MPI+OpenMP programming requires a new level of collective tuning
- For PPN=2 (Processes Per Node), the tuned version of MPI_Reduce shows 51% improvement on 2,048 cores
- We see 4% and 24% improvement with LULESH and HPCGCC applications, respectively on 512 cores (8 OpenMP threads per MPI processes)

Application Case Studies: HoomdBlue, SMG2000, Neuron, Amber, MiniAMR and MILC

- Performance of MPI-3 RMA operations as well as collectives heavily rely on network and processor characteristics
- MVAPICH2 library uses a tuning framework to select optimal point-point, collective algorithms and the best RMA protocol based on system characteristics yielding best performance
- Tuning RMA operations for a given system reduces the latency of Put and Get operations by 50% and 47% for 16Kb messages

- HoomdBlue is a Molecular Dynamics simulation using a custom force field.
- GUPDirect specific features selection and tuning significantly benefit the HoomdBlue application. We observe a factor of 2x improvement on 32 GPU nodes, with both 64K and 256K particles

- UD-based transport protocol selection benefits the SMG2000 and the Neuron application
  - With SMG2000 the improvement is 22% and 6% on 1,024 and 4,096 cores, respectively
  - For Neuron application, 15% and 27% improvement is seen for 768 and 1,024 processes respectively

- Tuning the Eager threshold has a significant impact on application performance by avoiding the synchronization of rendezvous protocol and thus yielding better communication computation overlap
  - With MiniAMR application kernel, increasing the eager threshold shows up to 8% reduction in the total communication time
  - For the Amber application, tuning results in 19% improvement at 256 processes

User-mode Memory Registration (UMR) based tuning
- Non-contiguous data processing is very common on HPC applications. MVAPICH2 offers efficient designs for MPI Datatype support using novel hardware features such as UMR
- UMR-based protocol selection benefits the MILC application. We see 4% and 6% improvement in execution time at 512 and 640 processors, respectively

Education, Outreach and Training

- Impacting all the XSEDE users running on Stampede, Comet and Gordon
- Many workshops/tutorials presented at recent events (XSEDE, SC, ISC, Cluster, MUG, etc.)
- A large number of publications and several Ph.D theses have come out of the work

Contact Information

- DK Panda (OSU), W. Barth (TACC), A. Majumdar (SDSC) and K. Tomko (OSC)
- http://mvapich.cse.ohio-state.edu

SI2-SSI: A Comprehensive Performance Tuning Framework for the MPI Stack

- Award #1148371 - Lead Institution
- Award #1148428 - TACC
- Award #1147926 - SDSC