NETWORK-BASED COMPUTING LABORATORY

Performance Analysis and Evaluation of Mellanox ConnectX InfiniBand Architecture with Multi-Core Platforms

Sayantan Sur, Matt Koop, Lei Chai

Dhabaleswar K. Panda

Network Based Computing Lab,

The Ohio State University

OHIO STATE

Presentation Outline

- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

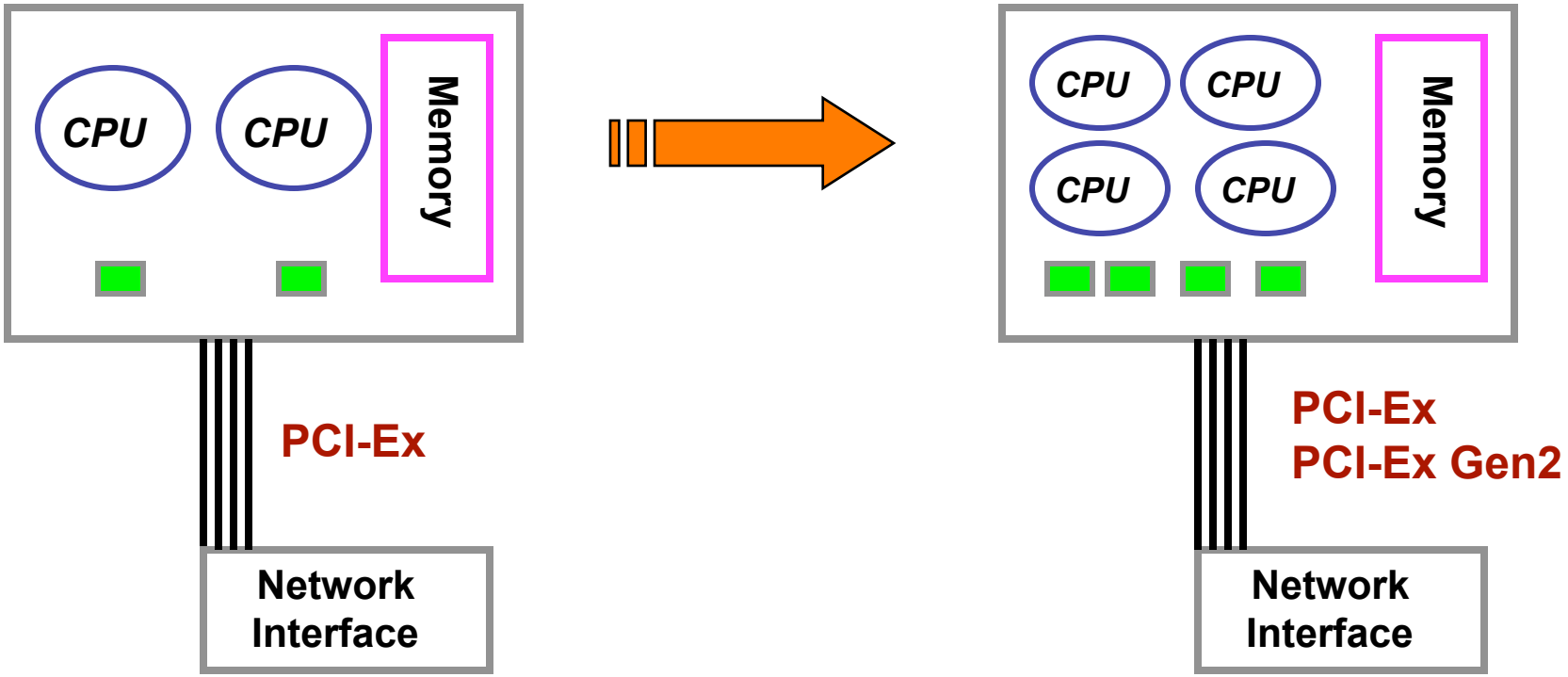
Introduction

- InfiniBand is a popular interconnect used for High-Performance Computing
- The advent of *Multi-core computing* is changing overall system architecture and performance parameters
- Trend: increasing cores / network-interface
  - Older systems had 2-4 processors and 1 network card
  - Most current systems have 8-cores and 1 network card
  - Number of cores will increase as per Moore's Law!
- *Next generation of network-interfaces may need to be redesigned to improve performance on Multi-core platforms!*

InfiniBand Overview

- InfiniBand is an emerging HPC interconnect
  - Industry Standard; Open Source software
- Very good performance with many features
  - Minimum Latency: ~1.2us, Peak Bandwidth: ~1500MB/s
  - RDMA, Atomic Operations, Shared Receive Queue
  - Hardware multicast, Quality of Service …
- Several generations of InfiniBand hardware
  - Third Generation: ***InfiniHost III***
  - Fourth Generation: ***ConnectX***
- Several transport mechanisms are supported
  - Reliable Connection (RC)
  - Unreliable Datagram (UD)
  - Scalable Reliable Connection (SRC) (New)
    - Initial API discussion has started

Increasing Communication Volume in Multi-Core Clusters



• The amount of communication handled per network interface is increasing exponentially, as Moore's law allows for more CPU cores to be fabricated

Presentation Outline

- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

Problem Statement

- Can network-interfaces be designed to offer greater levels of performance for Multi-core system architecture?
- Can we experimentally ascertain performance improvements offered by next generation ConnectX architecture?

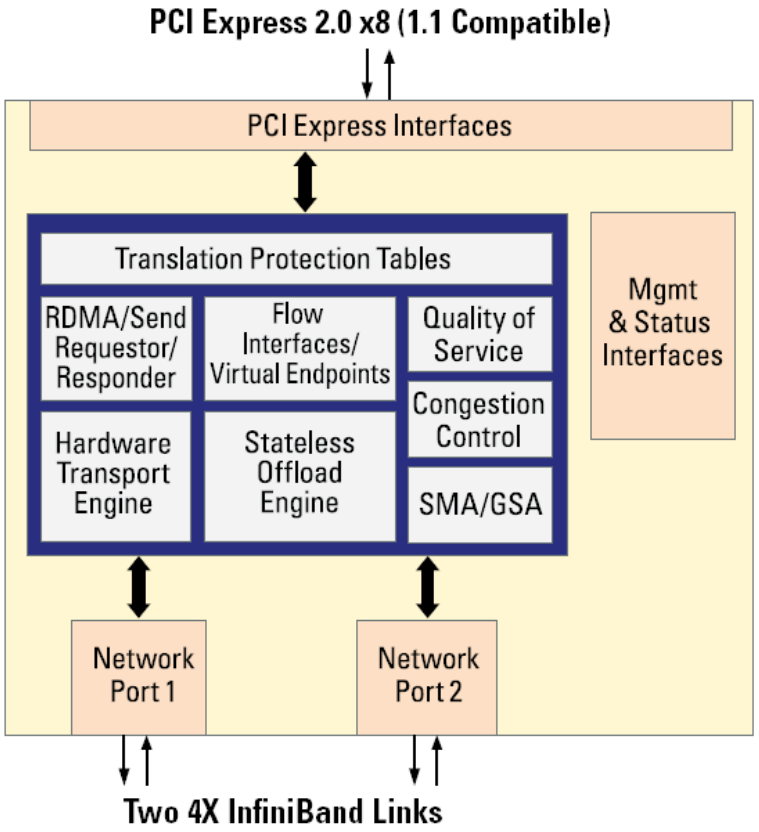
Approach Used

- Micro-benchmarks
  - Design suitable experiments to study the performance at the lowest network-level
    - RDMA Write/Read Latency
    - RDMA Write bandwidth
    - Multi-pair performance
- Application level Benchmarks
  - Communication specific benchmark HALO
  - Molecular dynamics simulation benchmark LAMMPS

Presentation Outline

- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

ConnectX Overview



ConnectX Architecture, courtesy Mellanox

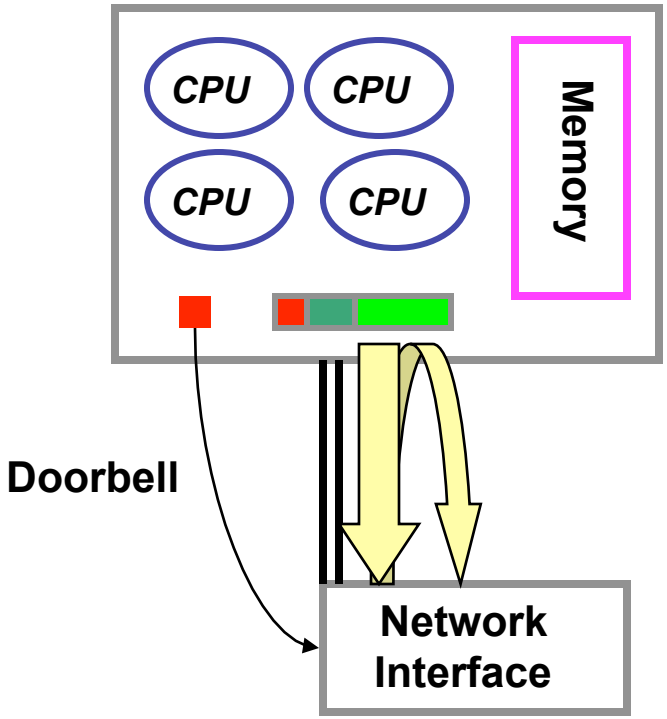
- Fourth Generation Silicon
  - DDR (Double Data Rate)
    - PCI-Express Gen1
  - QDR (Quad Data Rate)
    - PCI-Express Gen2
- Flexibility to configure each individual port to either InfiniBand or 10G
- Hardware support for Virtualization
- Quality of Service
- Stateless Offloads
- In this presentation, we focus on the InfiniBand device

Performance improvements in ConnectX

- Designed to improve the processing rate of incoming packets
- Scheduling done in hardware with no firmware involvement in critical path
- Improved WQE processing capabilities
  - Difference b/w RDMA and Send/Receive: 0.2us
  - For earlier generation InfiniHost III it was: 1.15us
- Use of PIO for very small message sizes
  - “Inline” allows data to be encapsulated in the WR
  - PIO mode allows entire WR to be sent to the HCA without any DMA operations

PIO vs DMA

- In DMA mode, the processor writes a smaller command over I/O bus and HCA arranges for DMA transfer
  - Lower CPU usage
  - More I/O bus transactions
- In PIO mode, the processor writes data over I/O bus to a HCA command buffer
  - Increased CPU usage
  - Less I/O bus transactions
- For small messages, CPU usage is negligible, while number of transactions can be significantly reduced



I/O Bus Transcations for DMA and PIO modes

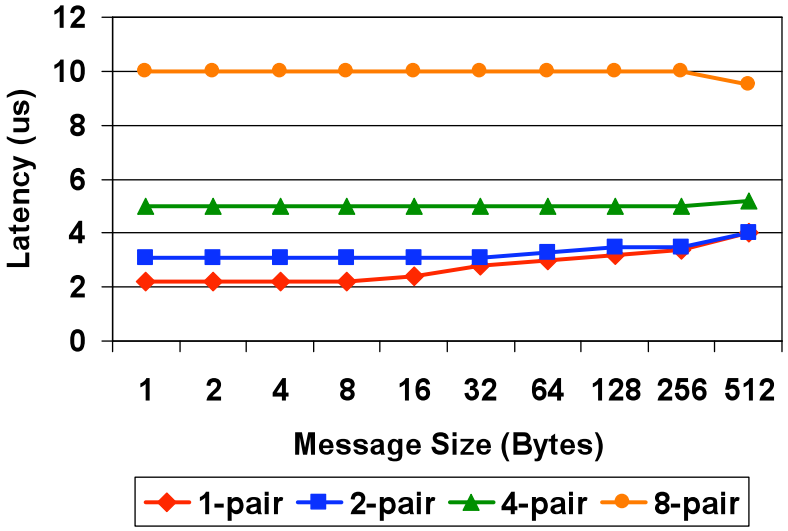
Presentation Outline

- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

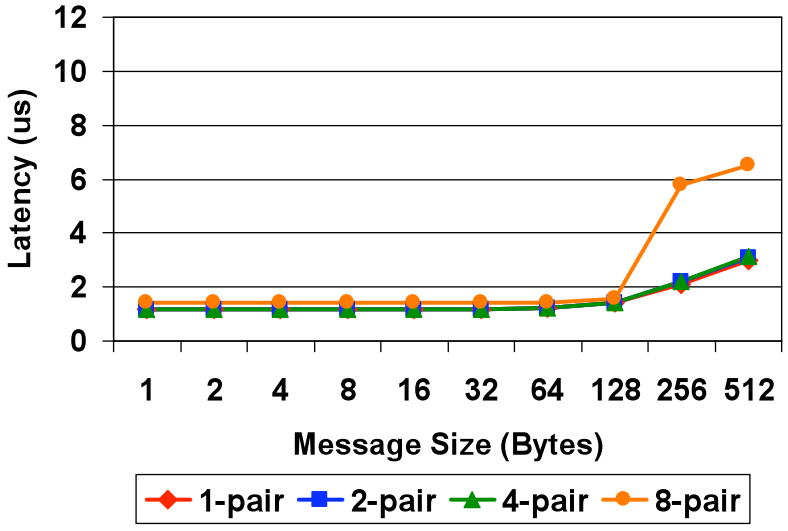
Experimental Platform

- Four node Intel Bensley platform
- Dual Intel Clovertown 2.33 GHz Quad Core
- 4 GB main memory (FBDIMM)
- 3 x8 PCI-Express slots
  - ConnectX DDR HCAs (MT25408)
    - Firmware 2.0.139
    - Expected similar performance with current FW
  - InfiniHost III DDR HCAs (MT25218)
    - Firmware version 5.2.0
- OpenFabrics Gen2 stack (OFED-1.2)
- Linux kernel 2.6.20-rc5
- Two identical MTS2400 switches

Multi Pair RDMA Write Latency



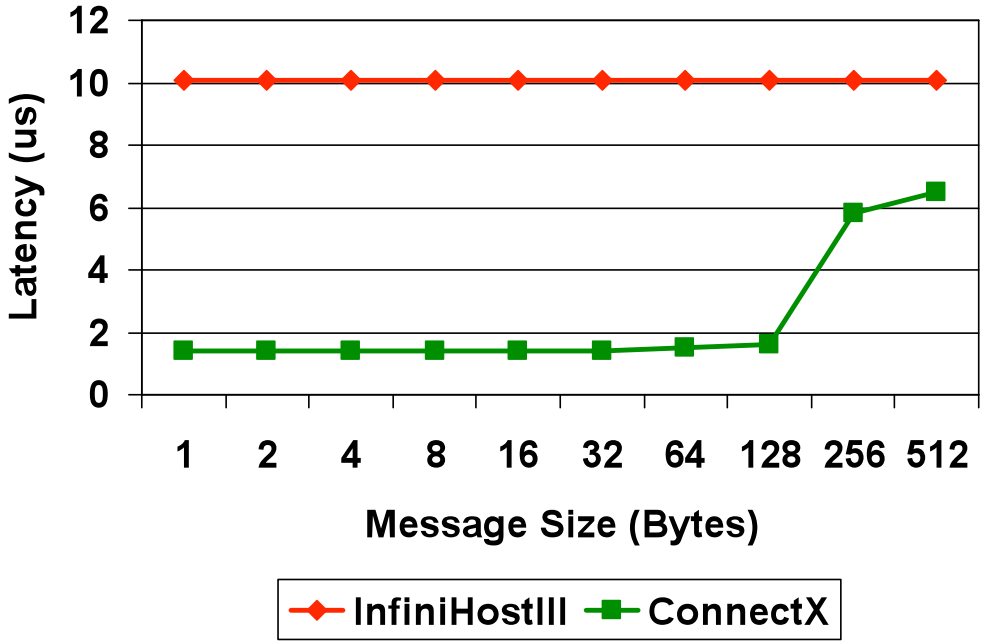
Multi-Pair Latency on InfiniHost III



Multi-Pair Latency on ConnectX

- OpenFabrics/Gen2 level RDMA Write test with multiple pairs
- InfiniHost III latencies increase linearly with number of pairs
- ConnectX latencies are almost same regardless of pairs
- For 256 bytes, size of WQE exceeds PIO limit and DMA is used
  - Our platform doesn't execute so many PCI-E reads concurrently as issued by ConnectX firmware
- For 8 pairs, ConnectX *factor of 6 improvement* for <= 512 bytes

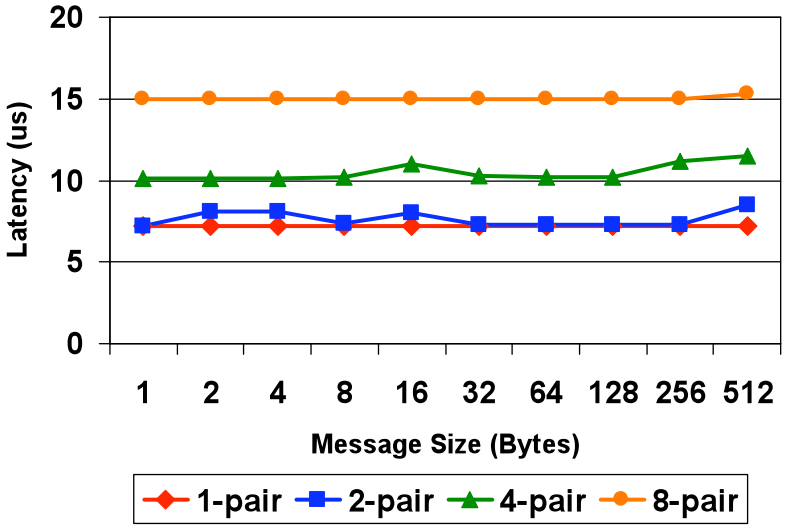
Multi Pair RDMA Write Latency Comparison



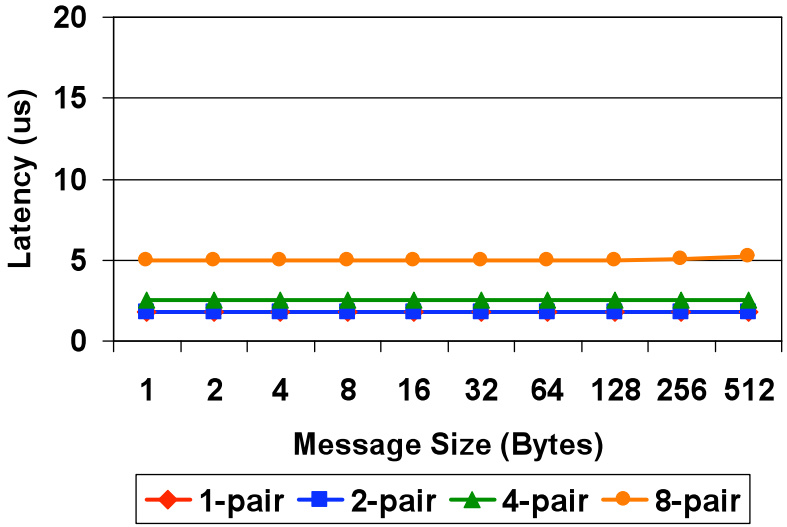
Multi-Pair Latency Comparison for 8-pairs

- For 256 bytes, size of WQE exceeds PIO limit and DMA is used
  - Our platform doesn't execute so many PCI-E reads concurrently as issued by ConnectX firmware
- For 8 pairs, ConnectX *factor of 6 improvement* for <= 128 bytes

Multi Pair RDMA Read Latency



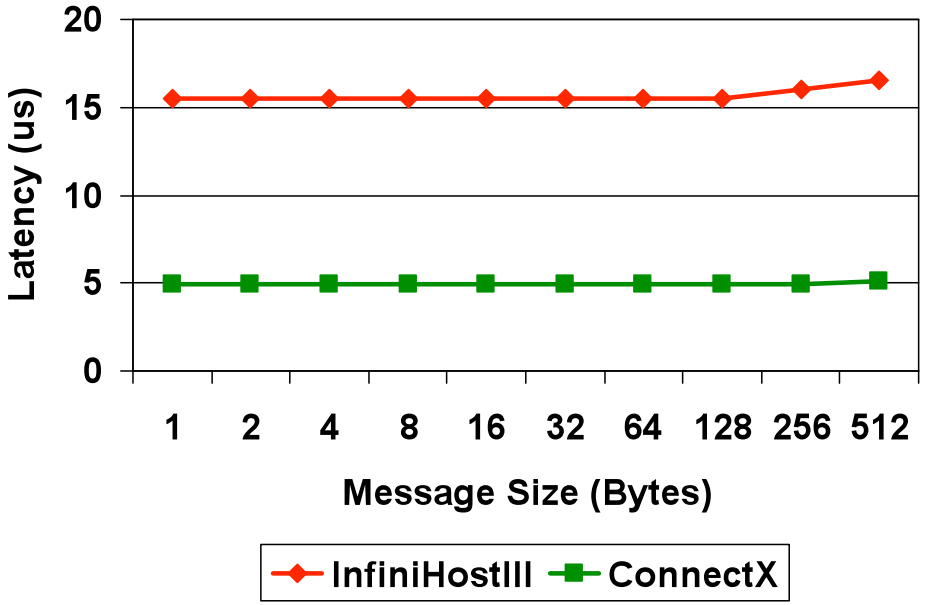
Multi-Pair Read Latency on InfiniHost III



Multi-Pair Read Latency on ConnectX

- OpenFabrics/Gen2 level RDMA Read test with multiple pairs
- InfiniHost III latencies increase linearly with number of pairs
- ConnectX latencies increase by lesser increments
  - For Read, no PIO can be used for data, DMA needs to be used
  - Our chipset does not seem to issue as many concurrent reads
  - With our settings, ConnectX can issue up to *16 concurrent reads* on the bus

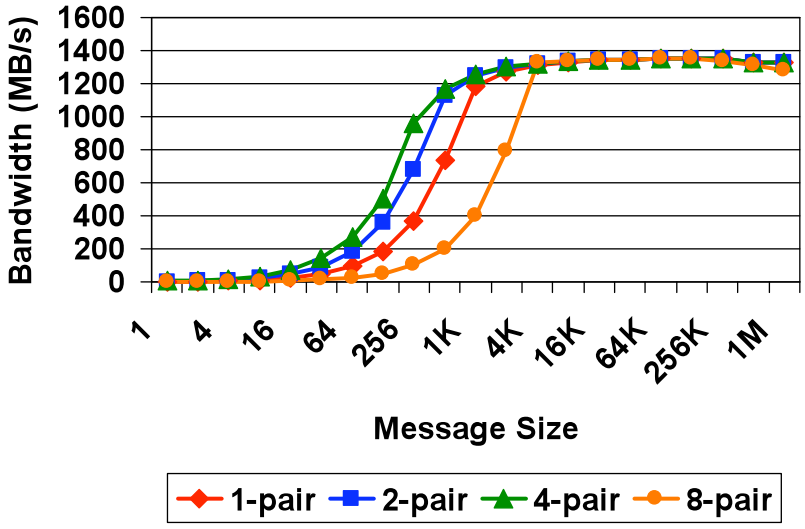
Multi Pair RDMA Read Latency Comparison

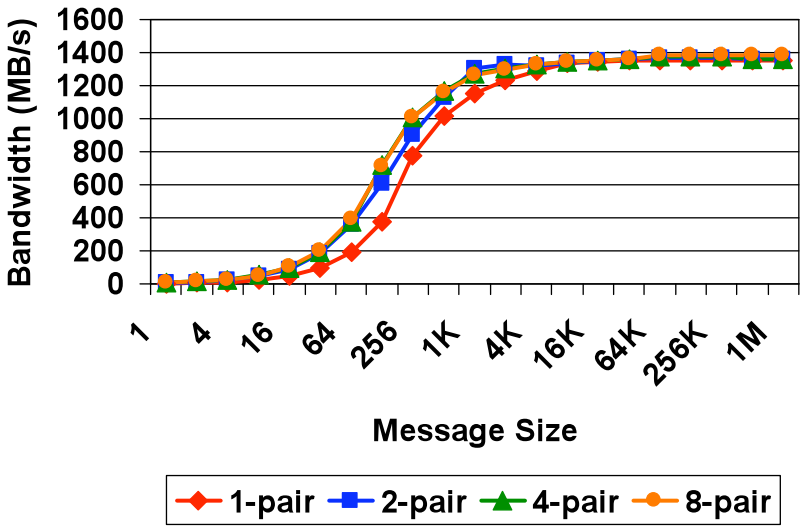


Multi-Pair Latency Comparison for 8-pairs

- For 8 pairs, ConnectX *factor of 3 improvement* for <= 512 bytes

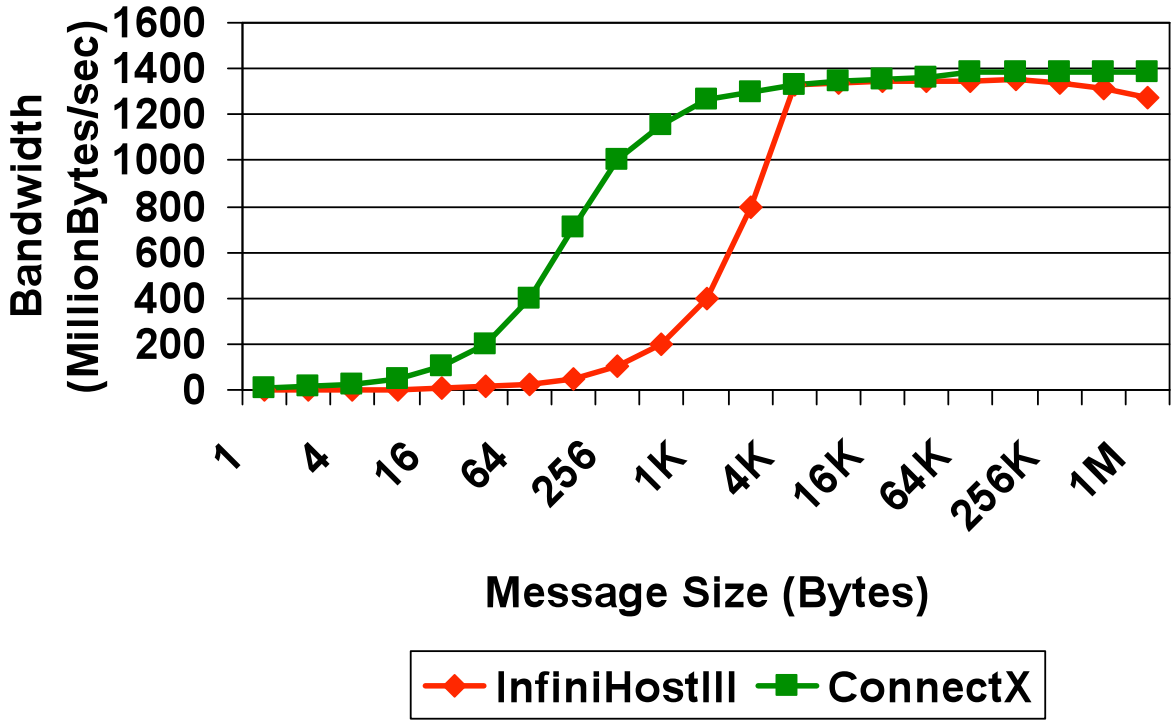
Multi Pair RDMA Write Bandwidth





- OpenFabrics/Gen2 level RDMA Write bw test with multiple pairs
- InfiniHost III bandwidth decreases with number of pairs
- ConnectX bandwidth improves with number of pairs
- For ConnectX, even 1-pair can achieve closer to maximum bandwidth!
- For 8 pairs, ConnectX *factor of 10 improvement* for 256 bytes

Multi Pair RDMA Write Bandwidth Comparison



Multi-Pair Bandwidth Comparison for 8-pairs

- For 8 pairs, ConnectX *factor of 10 improvement* for 256 bytes

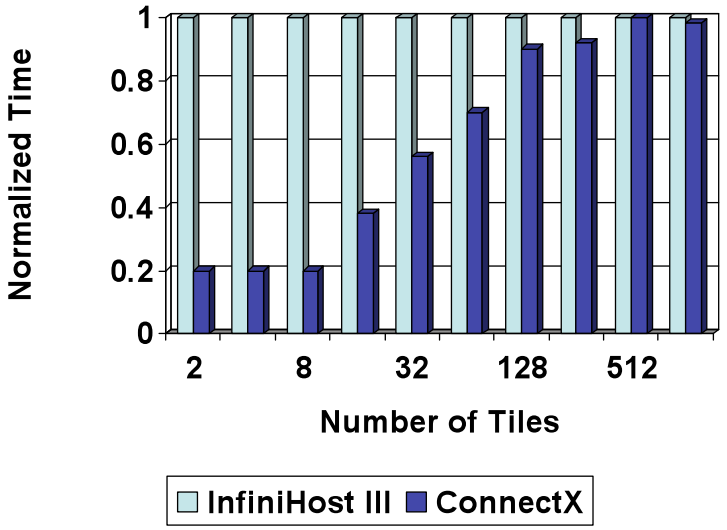
Presentation Outline

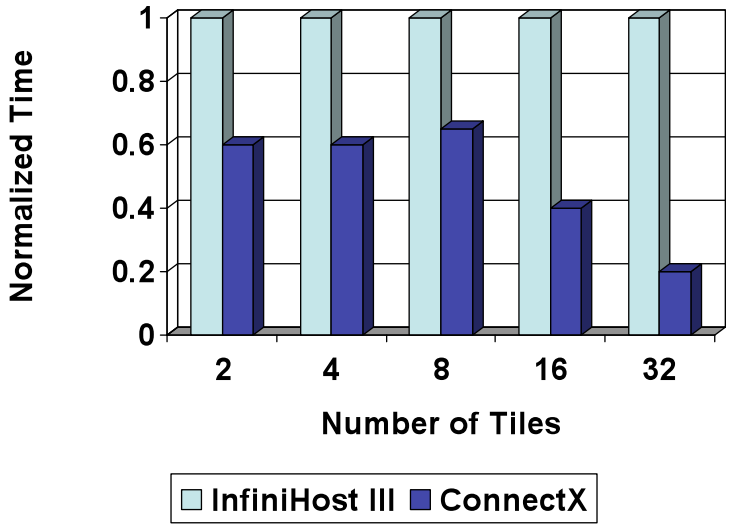
- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

MVAPICH and MVAPICH2 Software Distributions

- **High Performance MPI Library for InfiniBand and iWARP Clusters**
  - **MVAPICH (MPI-1) and MVAPICH (MPI-2)**
  - **Used by more than 540 organizations in 35 countries**
  - **Empowering many TOP500 clusters**
  - **Available with software stacks of many InfiniBand, iWARP and server vendors including Open Fabrics Enterprise Distribution (OFED)**
  - ***http://mvapich.cse.ohio-state.edu***

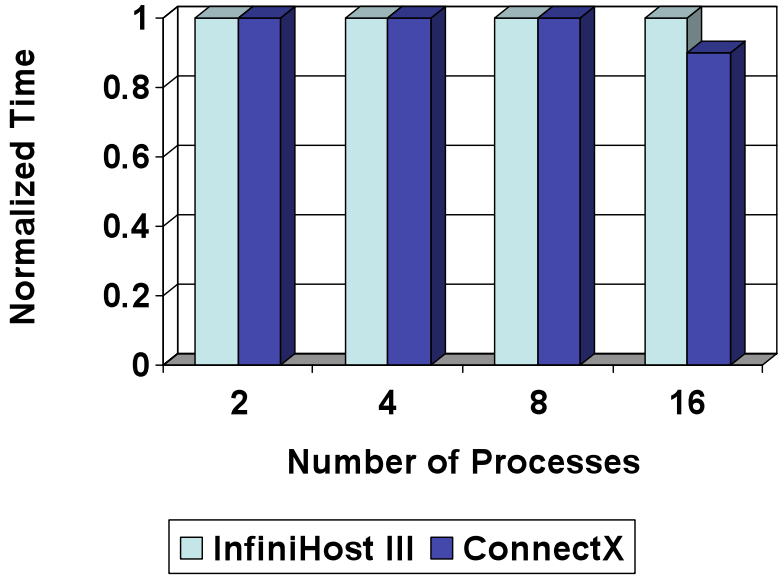
Halo Communication Benchmark





- Simulates Layered Ocean model communication characteristics
- MVAPICH-0.9.9 is used to execute this benchmark
- Processes scattered in cyclic manner
- For small number of tiles {2, 64}, ConnectX performance is better by a *factor of 2 to 5*

LAMMPS Benchmark



- Molecular dynamics simulator from Sandia National Labs
- MVAPICH-0.9.9 is used to execute this benchmark
- Processes scattered in cyclic manner
- "Loop Time" is reported as per benchmark specs
- `in.rhodo` benchmark used
- For 16 processes, ConnectX outperforms InfiniHost III by 10%
  - This is explainable by higher bandwidths for ConnectX when all 8-pairs of processes are communicating

Presentation Outline

- Introduction and Motivation
- Problem Statement and Approach Used
- Overview of ConnectX Architecture
- Micro-benchmark Level Evaluation
- Application Level Evaluation
- Conclusions and Future Work

Conclusions and Future Work

- In this work we presented
  - Network and Application level performance characteristics of ConnectX architecture
  - Outlined some of the major architectural improvements in ConnectX
  - Performance improvements for multi-core architectures are impressive
- In the future, we want to
  - Study performance on larger scale clusters
  - Leverage novel features of ConnectX such as: Scalable Reliable Connection (SRC), QoS, Reliable Multicast etc. into future MVAPICH designs

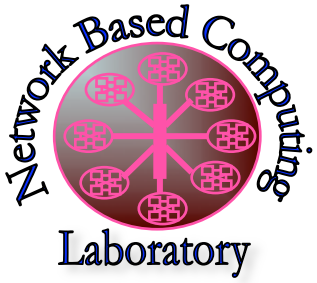
Acknowledgements

Our research is supported by the following organizations

- Current Funding support by

- Current Equipment support by


Web Pointers



http://nowlab.cse.ohio-state.edu/

MVAPICH Web Page
http://mvapich.cse.ohio-state.edu