SAND2016-4771C

Exceptional service in the national interest

Sandia National Laboratories

RMA-MT: A Benchmark Suite for Assessing MPI Multi-threaded RMA Performance

Matthew G. F. Dosanjh, Taylor Groves, Ryan E. Grant, Ron Brightwell, Patrick G. Bridges

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration | CCR Center for Computing Research

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Trends Towards Exascale

Sandia National Laboratories

- Trends in High Performance Computing
  - More cores, threads, and nodes
    - OpenMP
    - Qthreads
    - Argobots
    - Tasking Models
  - Fine grained communication
  - Current performance warrants exploration of other models
- Parallelism models for Exascale systems
  - MPI Thread Multiple for intra-node parallelism
  - MPI RMA for inter-node parallelism
  - Lack of benchmarks for these models, especially together

What do we need to know?

Sandia National Laboratories

- Does RMA-MT work?
- How does RMA-MT perform?
- How do I choose what RMA functions to use in my RMA-MT code?

The RMA-MT Benchmark Suite

Sandia National Laboratories

- Two levels of measurement
  - Microbenchmarks for Performance Measurement
  - Proxy applications for Application impact
- Based on existing two-sided benchmarks/mini apps
  - R. Thakur and W. Gropp's Multithreaded Latency and Bandwidth
  - Sandia Micro Benchmarks (SMBs)
  - Mantevo Miniapplications
    - HPCCG
    - MiniFE
    - MiniMD

Goals of the Benchmark Suite

Sandia National Laboratories

- Test Functionality
  - Check for Communication Correctness
  - Check for impact on solutions
- Measure Performance
  - Latency
  - Bandwidth
  - Message Rate
  - Proxy Applications
- Explore Synchronization and Transfer Options

Sandia National Laboratories

BACKGROUND

MPI RMA

Sandia National Laboratories

- MPI's implementation of one sided communication
- Four synchronization methods
  - Fence
  - Post/Start/Complete/Wait
  - Lock/Unlock
  - Lock-all/Unlock-all
- Avoids the majority of MPI processing
  - Serialized Data Structures
  - Unexpected Message Costs
- Combined Synchronization Costs
  - Communication and computation overlap
  - Fine grained communication

MPI Thread Multiple

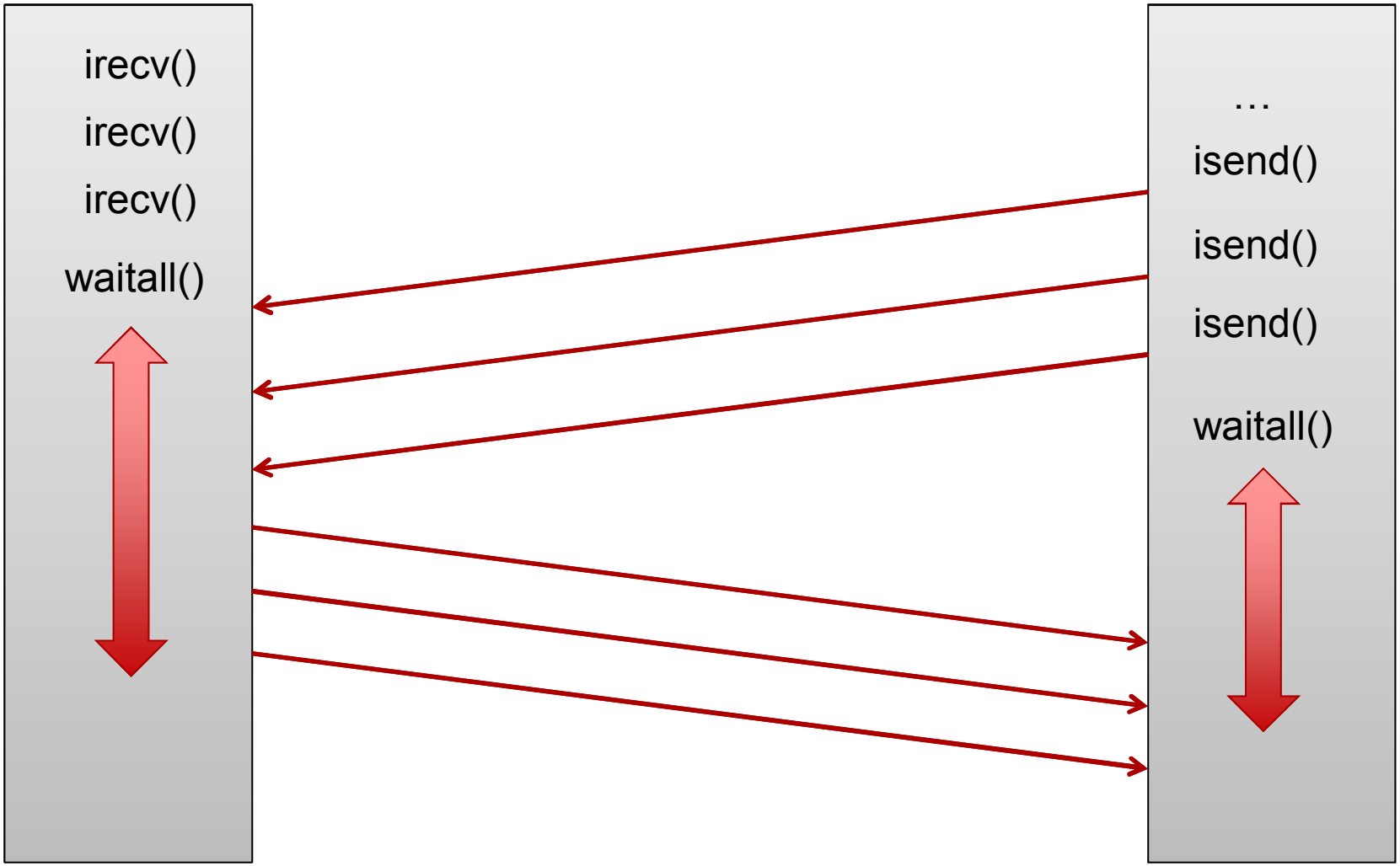
Sandia National Laboratories

- MPI's thread-safe mode
  - Makes new communication patterns easier
    - Fine grained messaging
    - Communication and computation overlap
  - Tested on Two sided
- Current Open Source Implementations are inefficient
  - Synchronous data structures
  - Implemented using global locking

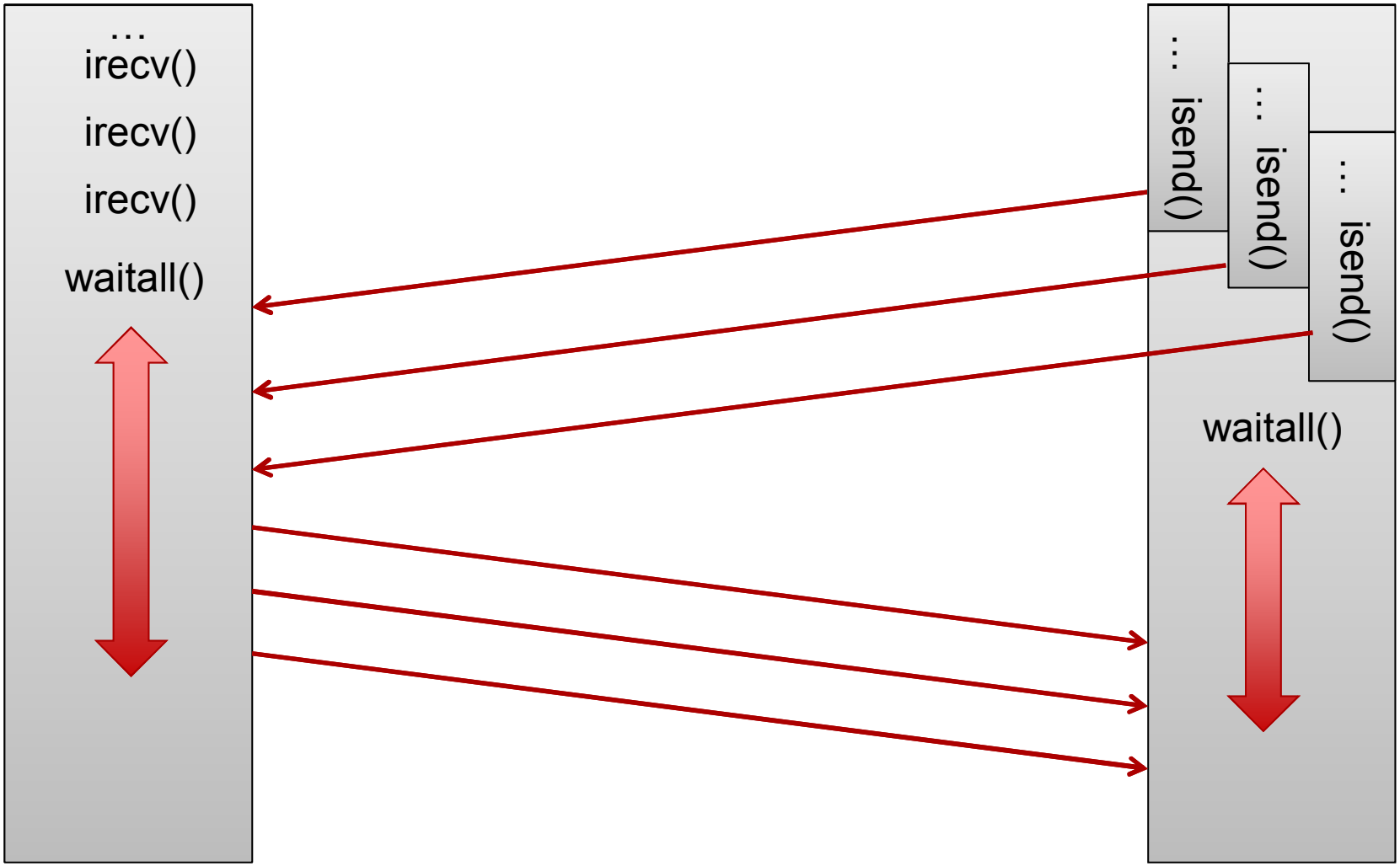
Sandia National Laboratories

CREATING THE BENCHMARKS

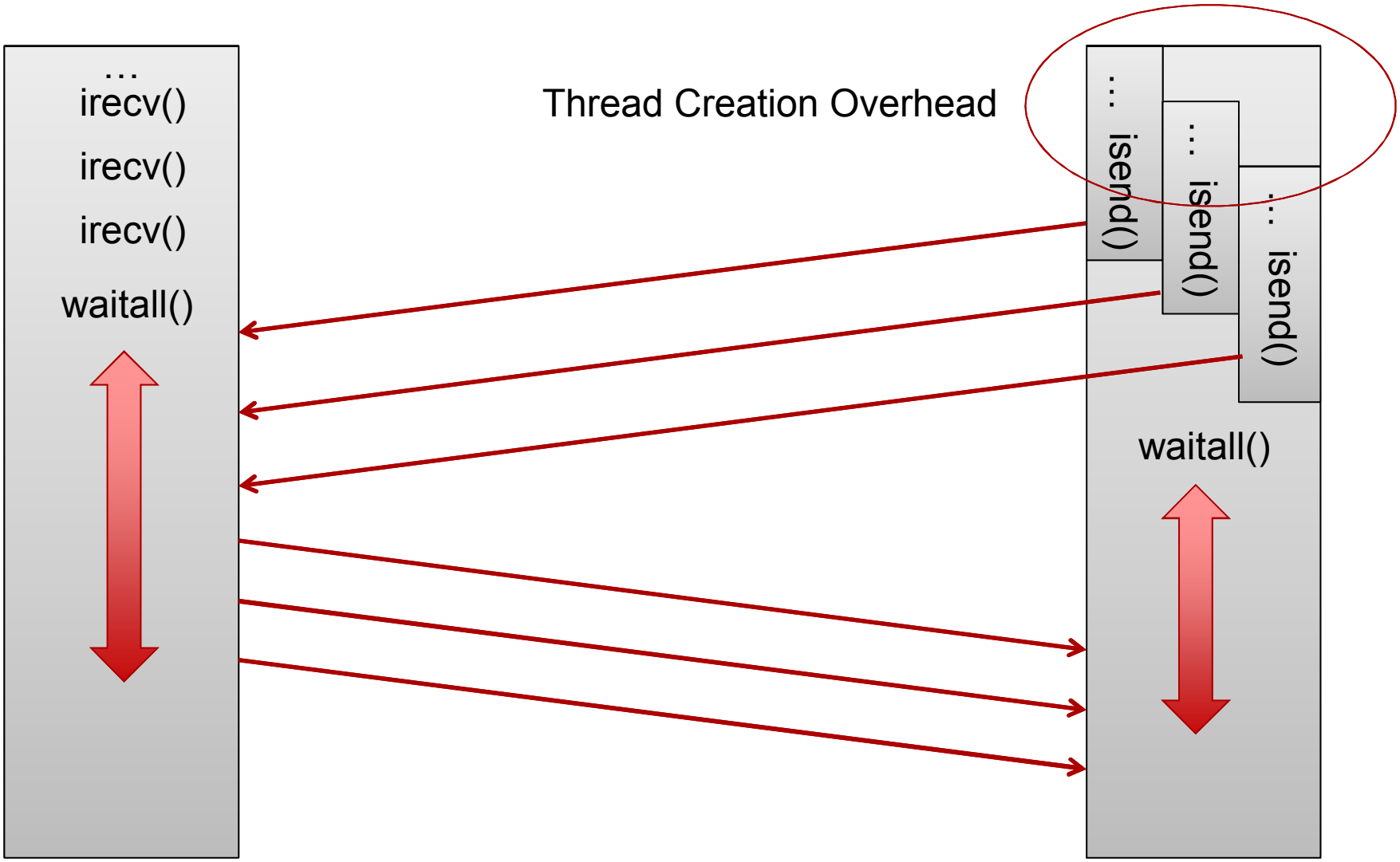
Basic Bandwidth



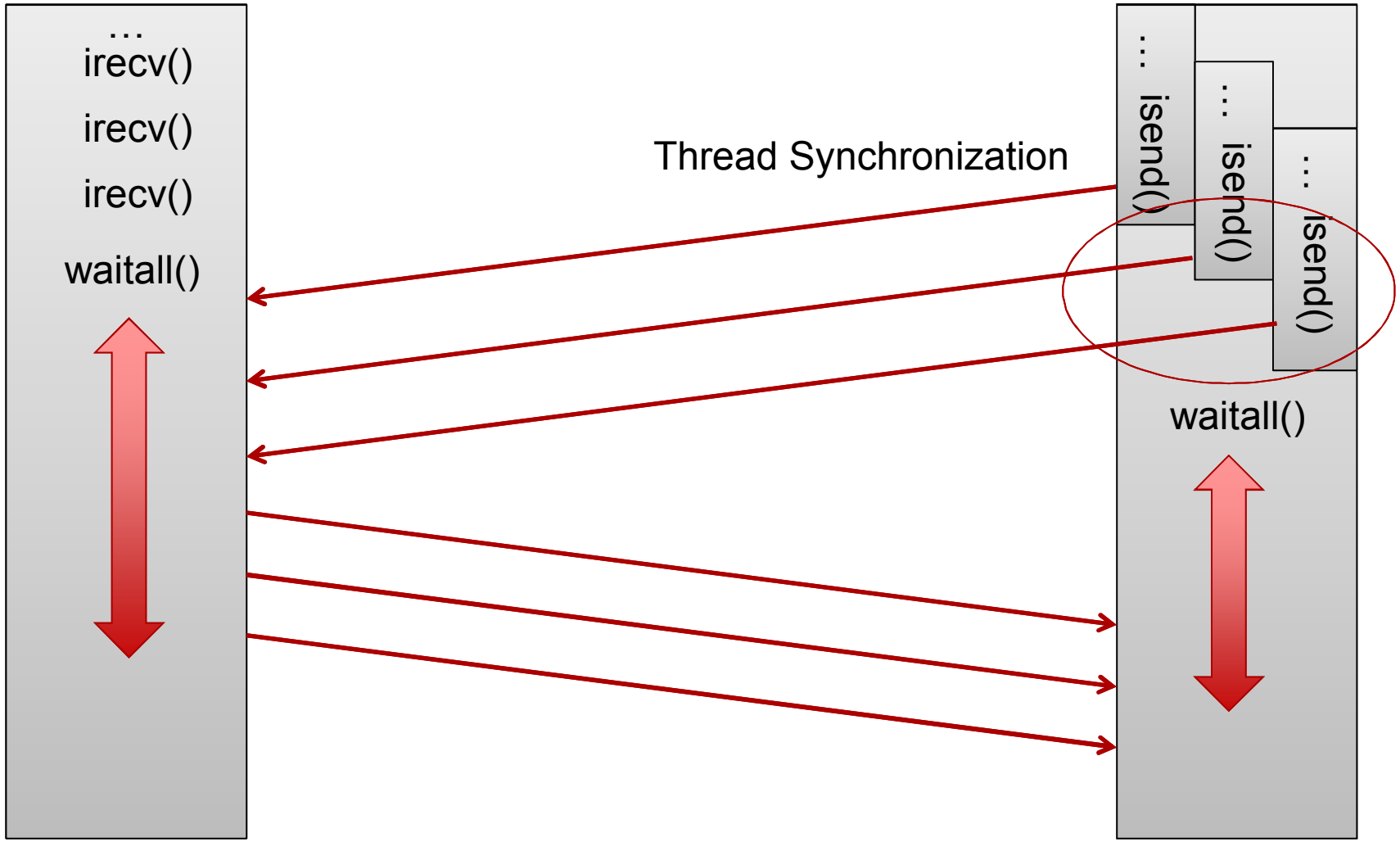
Multithreaded Bandwidth



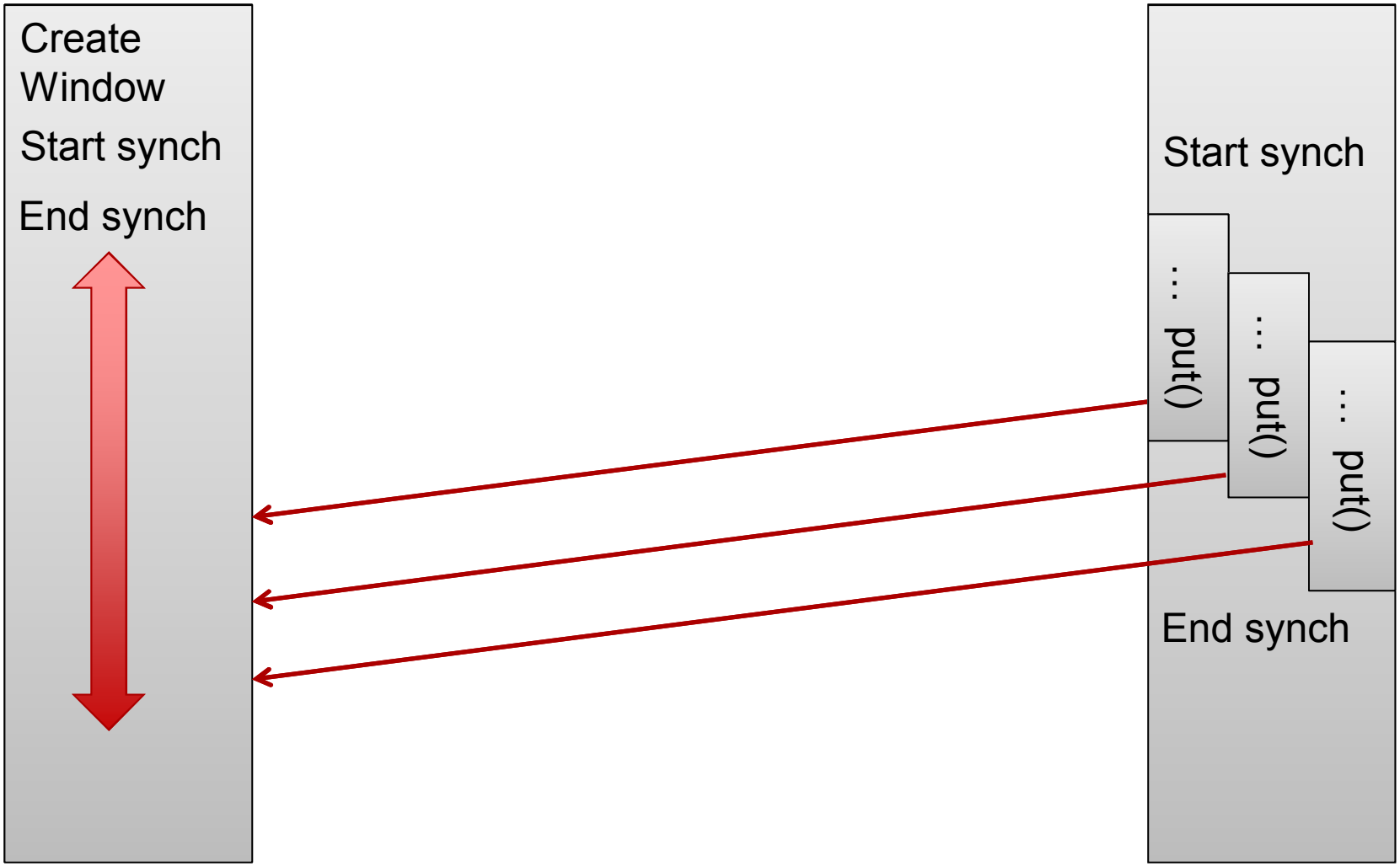
Multithreaded Bandwidth



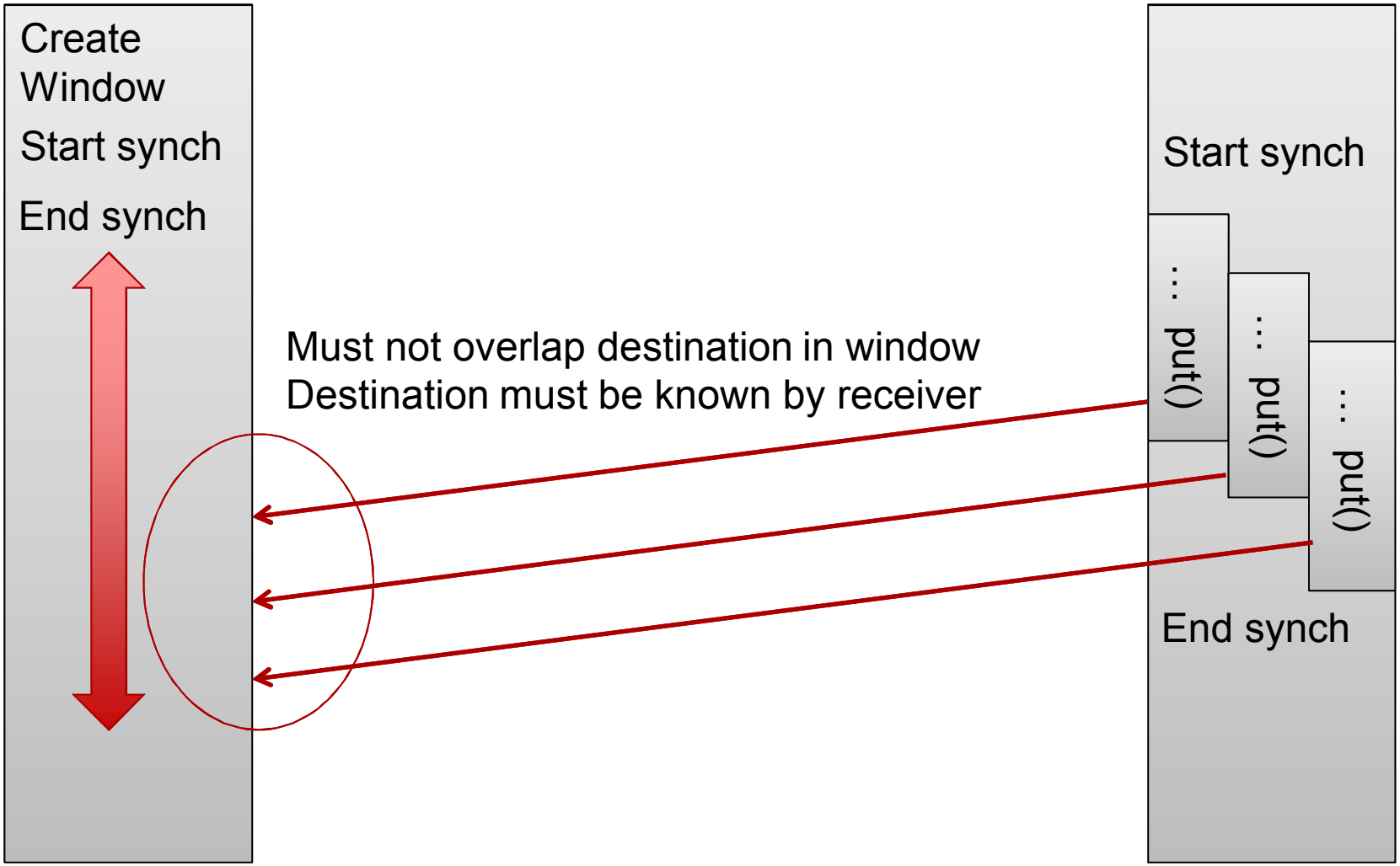
Multithreaded Bandwidth



RMA-MT Bandwidth



RMA-MT Bandwidth



RMA-MT Goals

Sandia National Laboratories

- Does it work?
  - Check data resulting from transfer
- How does is perform?
  - Time Measurement
  - Accounting for thread overhead
- What options do I choose?
  - Comparing transfer options (Put/Get)
  - Comparing synchronization options

Sandia National Laboratories

RESULTS

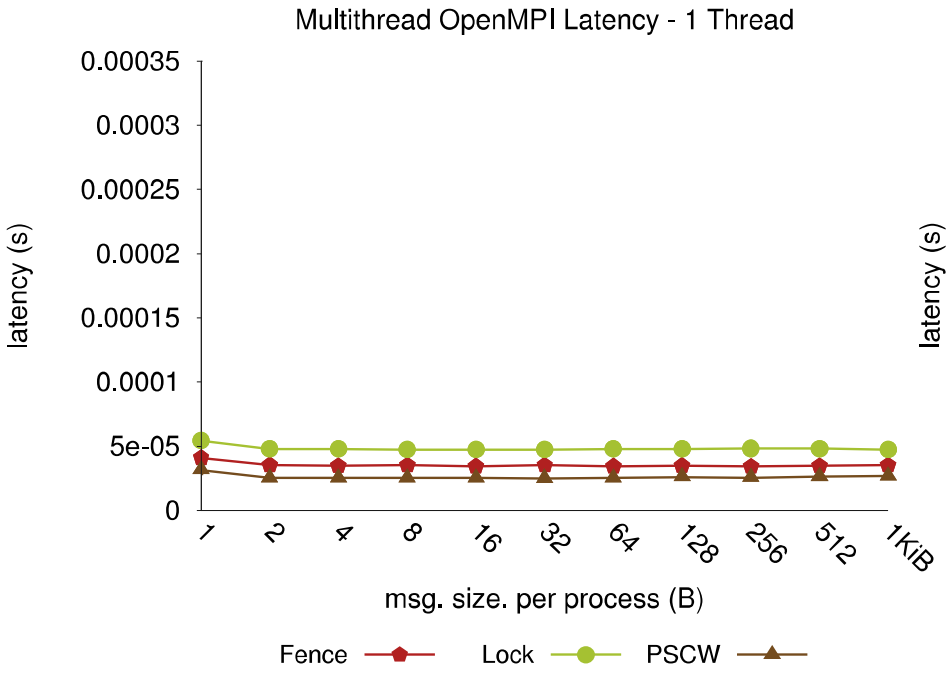
Experimental Setup

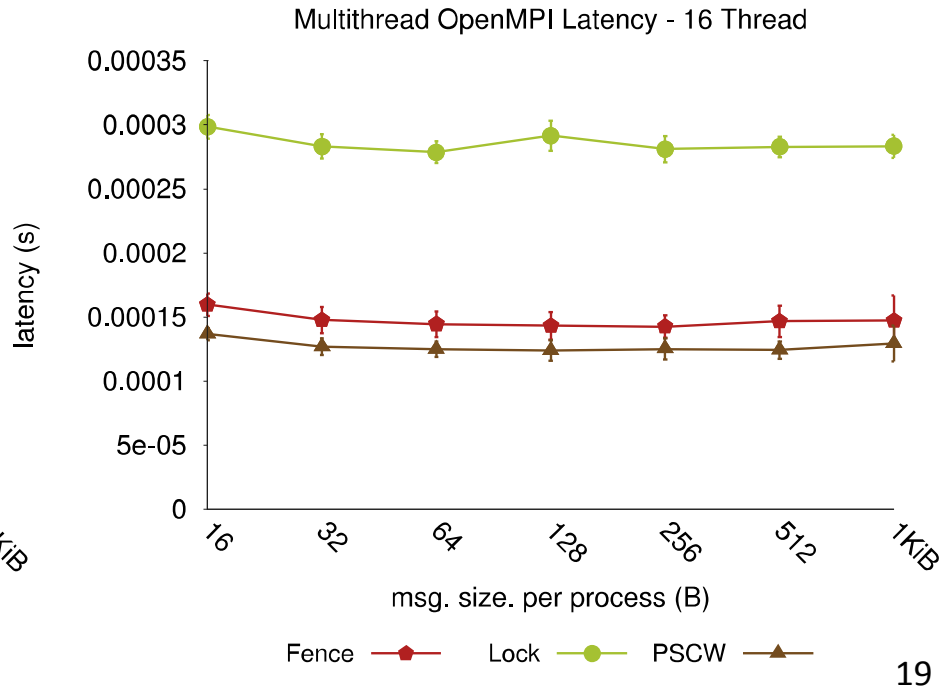
Sandia National Laboratories

- Hardware
  - 2 – Xeon E5 8 core processors running at 2.6 GHz
  - Qlogic Infiniband network architecture
- Software
  - OpenMPI Development Branch using OSC/PT2PT
  - MVAPICH numbers are available in the paper
- Miniapps
  - HPCCG uses Lock-all/Unlock-all
  - MiniFE and MiniMD use Fence

Latency – OpenMPI

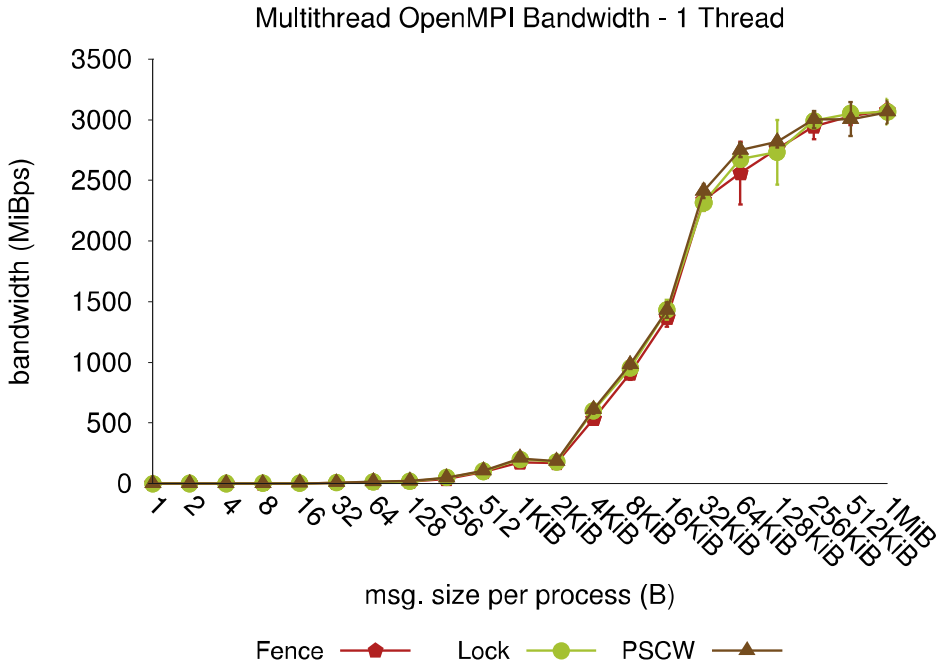
- All synchronization methods perform worse with Threads
- PSCW performs best in both cases
  - PSCW has 4 low latency steps
  - PSCW has an active synchronization on the receiver
- Lock has extra synchronization overhead
  - Acquires a reader-writer lock

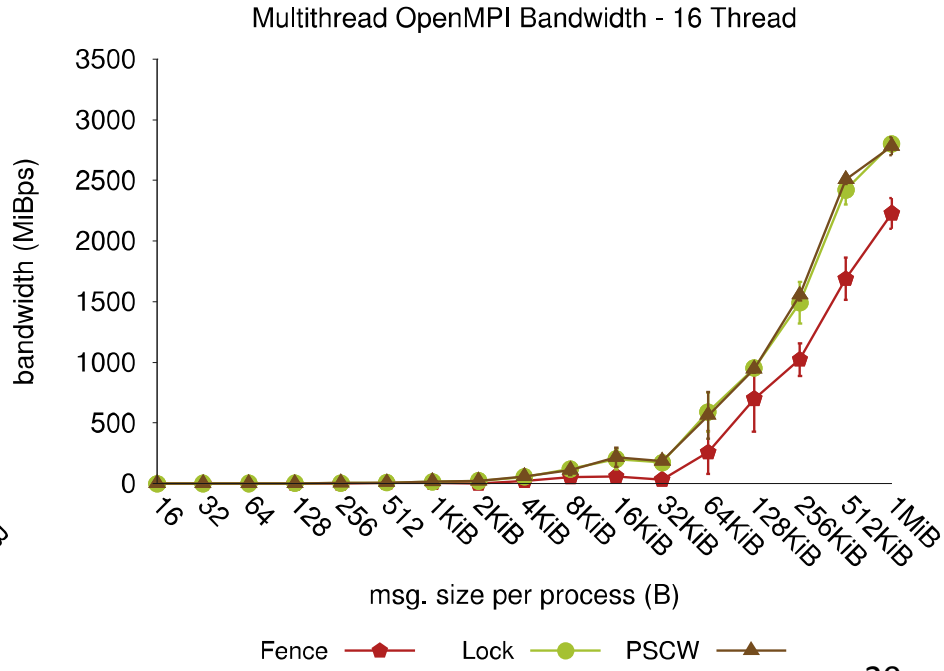




Bandwidth - OpenMPI

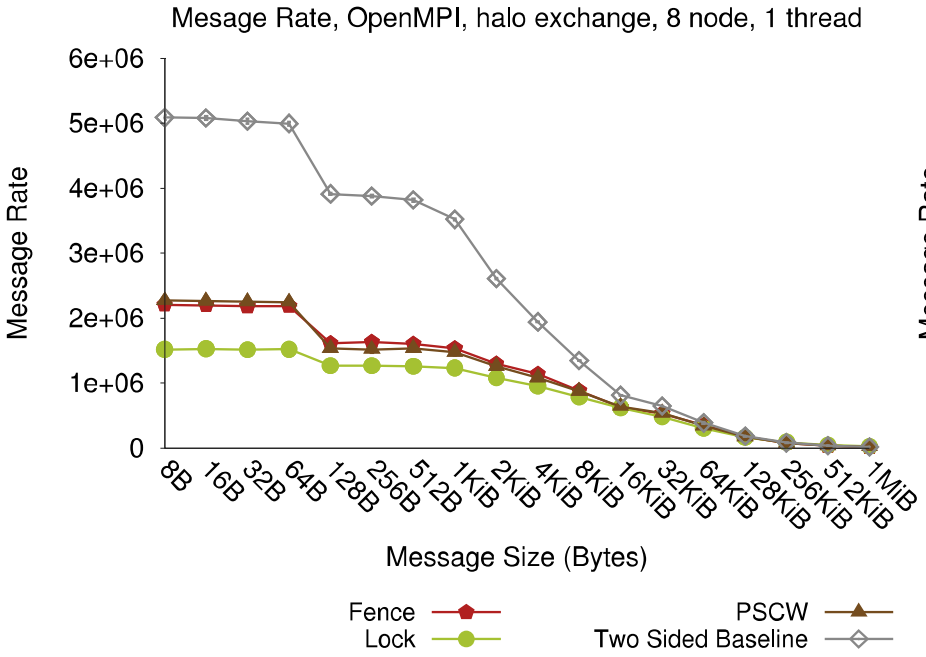
- Threads perform worse for messages smaller than 1MiB
- PSCW and Lock perform best in both cases
  - Each require synchronization with each process communicated with
  - Bandwidth test is one process communicating to a second
  - Lock's overhead is amortized with multiple transfers in an epoch
- Fence performs worst in 16 threaded case
  - Fence is implemented as a collective

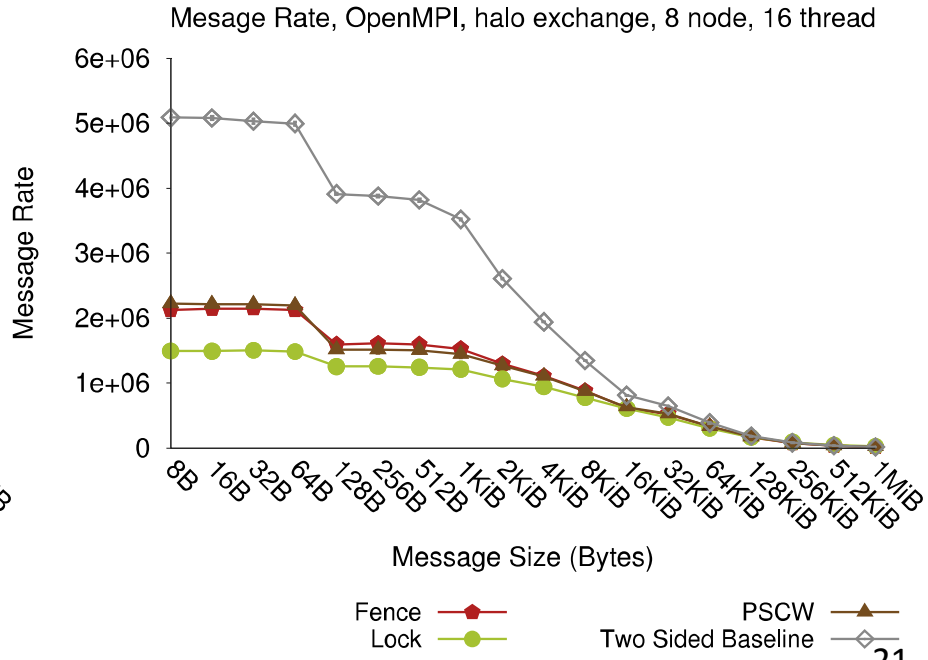




Message Rate - OpenMPI

- Performance is less impacted by threads
- Fence and PSCW perform best in both cases
  - Fence synchronizes all neighbors with one call
  - PSCW has a relatively low overhead
- Lock performs worst
  - Likely due to overhead acquiring the reader-writer lock





Discussion

Sandia National Laboratories

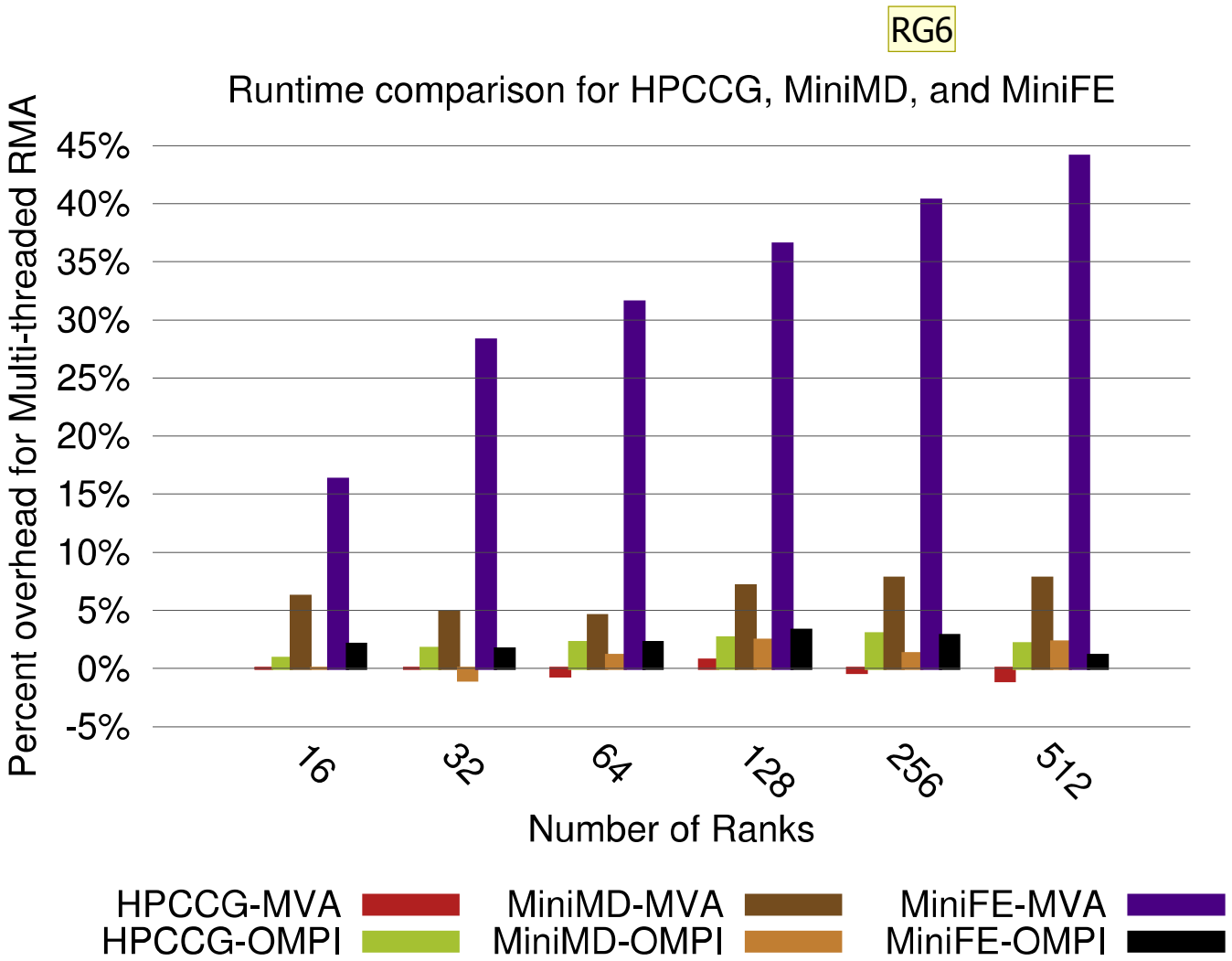
- Fence for the Mini Apps
  - Fence is performant in Message Rate Halo Exchange
  - Halo exchange matches the communication of the MiniApps
  - Fence provides a simpler coding paradigm
- Lockall functionality issues
  - Due to the development trunk
- MVAPICH numbers are available in the paper

Functionality

Sandia National Laboratories

- Running these tests exposed both performance and correctness problems with RMA-MT code paths in MPI implementations
  - 3 Segmentation Faults
  - 22 failed assertions
  - 6 incorrect transfers
  - Out of 280 runs
- We've shared these benchmarks with developers who have used them to test, fix, and optimize their implementations

Proxy Applications



Slide 24

RG6 do you plan on saying here? It's easy to get lost in the crummy MVAPICH performance and miss the real message of thsi slide

Ryan Grant, 5/17/2016

Performance

Sandia National Laboratories

- RMA-MT currently performs worse than single-threaded two-sided
  - Two-sided implementation of one sided calls
  - Global locks in MVAPICH
  - Queued until synchronization in MVAPICH
- These are software engineering challenges

Future Work

Sandia National Laboratories

- How does RMA-MT interact with different network architectures at scale?
- How will RMA-MT impact application performance using Passive Target synchronization?
- How will RMA-MT impact asynchronous communication algorithms?
- Can we adapt these benchmarks to other one sided communication libraries such as OpenSHMEM?

Sandia National Laboratories

QUESTIONS?