Latest Advances in MVAPICH2 MPI Library for NVIDIA GPU Clusters with InfiniBand

Presentation at GTC 2014

by

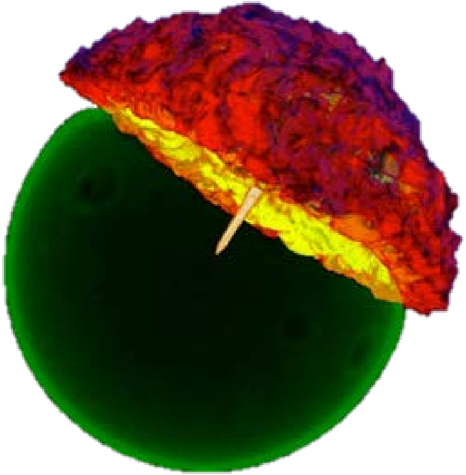
Dhabaleswar K. (DK) Panda

The Ohio State University

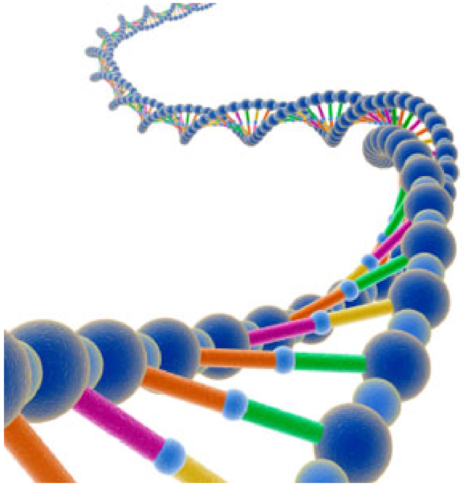
E-mail: panda@cse.ohio-state.edu

http://www.cse.ohio-state.edu/~panda

Current and Next Generation HPC Systems and Applications



- Growth of High Performance Computing (HPC)
  - Growth in processor performance
    - Chip density doubles every 18 months
  - Growth in commodity networking
    - Increase in speed/features + reducing cost
  - Growth in accelerators (NVIDIA GPUs)

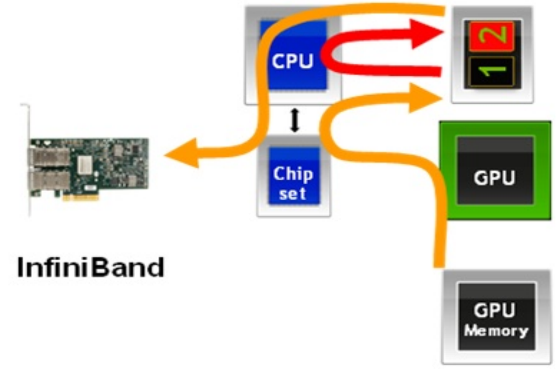


Outline

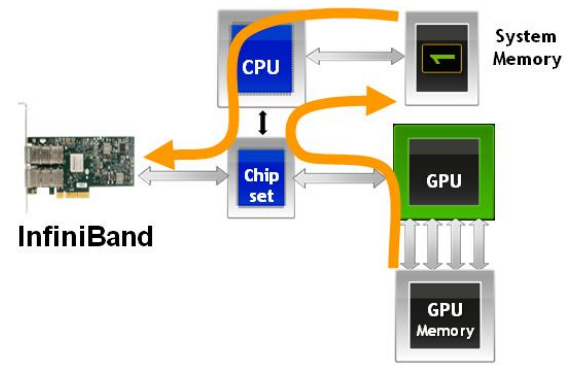
- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
  - Two-sided Communication
  - One-sided Communication
  - MPI Datatype Processing
  - More Optimizations
- MPI and OpenACC
- On going work

MVAPICH2-GPU: CUDA-Aware MPI

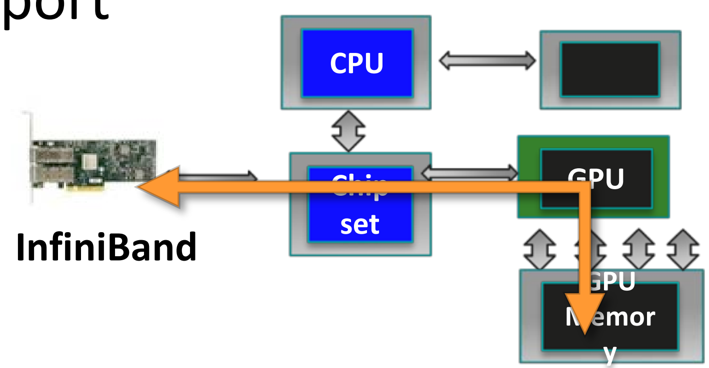
- Before CUDA 4: Additional copies
  - Low performance and low productivity



- After CUDA 4: Host-based pipeline
  - Unified Virtual Address
  - Pipeline CUDA copies with IB transfers
  - High performance and high productivity

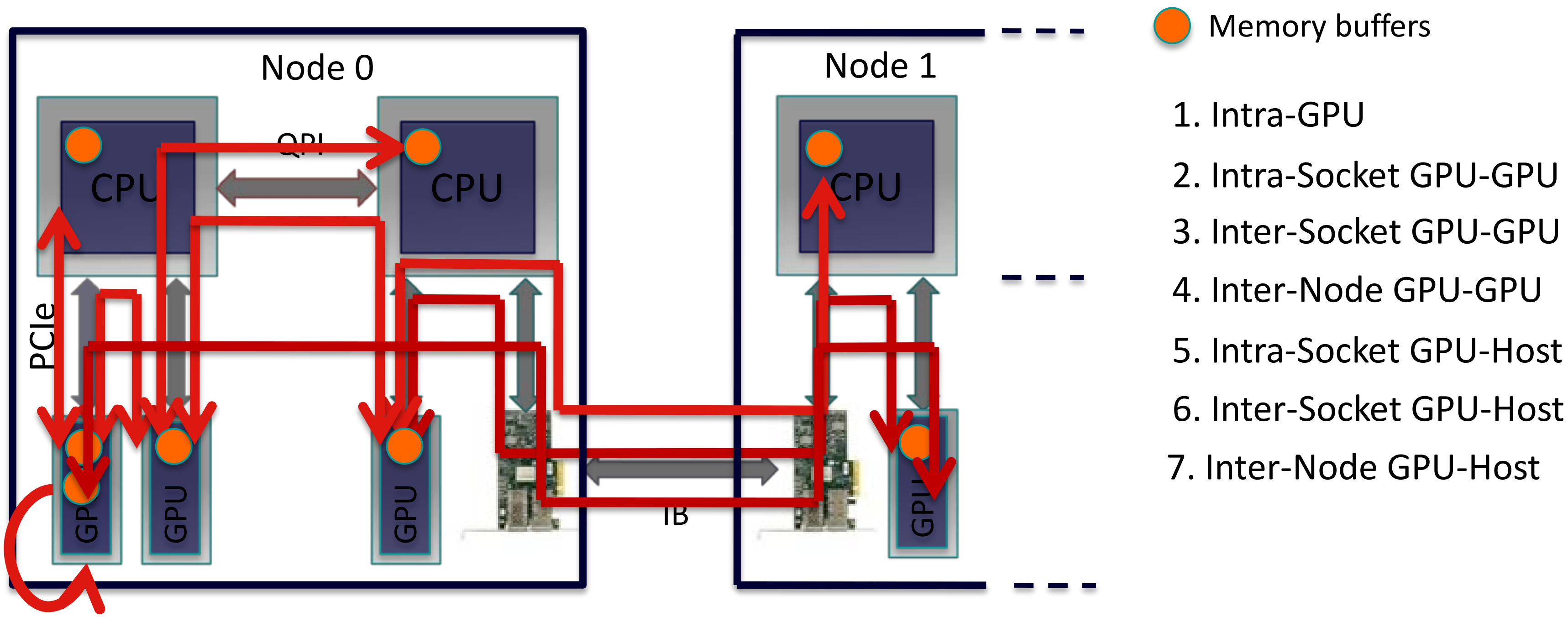


- After CUDA 5.5: GPUDirect-RDMA support
  - GPU to GPU direct transfer
  - Bypass the host memory
  - Hybrid design to avoid PCI bottlenecks



Data Movement on GPU Clusters

- Connected as PCIe devices – Flexibility but Complexity



8. Inter-Node GPU-GPU with IB adapter on remote socket

and more . . .

- For each path different schemes: Shared_mem, IPC, GDR, pipeline ….
- Critical for runtimes to optimize data movement while hiding the complexity

MVAPICH2/MVAPICH2-X Software

- High Performance open-source MPI Library for InfiniBand, 10Gig/iWARP and RDMA over Converged Enhanced Ethernet (RoCE)
  - MVAPICH (MPI-1) ,MVAPICH2 (MPI-2.2 and MPI-3.0), Available since 2002
  - MVAPICH2-X (MPI + PGAS), Available since 2012
  - **Support for NVIDIA GPUs, Available since 2011**
  - Used by more than 2,150 organizations (HPC Centers, Industry and Universities) in 72 countries
  - More than 205,000 downloads from OSU site directly
  - Empowering many TOP500 clusters
    - 7th ranked 519,640-core cluster (Stampede) at TACC
    - 11th ranked 74,358-core cluster (Tsubame 2.0) at Tokyo Institute of Technology
    - 16th ranked 96,192-core cluster (Pleiades) at NASA
    - 105th ranked 16,896-core cluster (Keenland) at GaTech and many others . . .
  - Available with software stacks of many IB, HSE and server vendors including Linux Distros (RedHat and SuSE)
  - http://mvapich.cse.ohio-state.edu

Outline

- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
    - Two-sided Communication
    - One-sided Communication
    - MPI Datatype Processing
    - More Optimizations
- MPI and OpenACC
- On going work

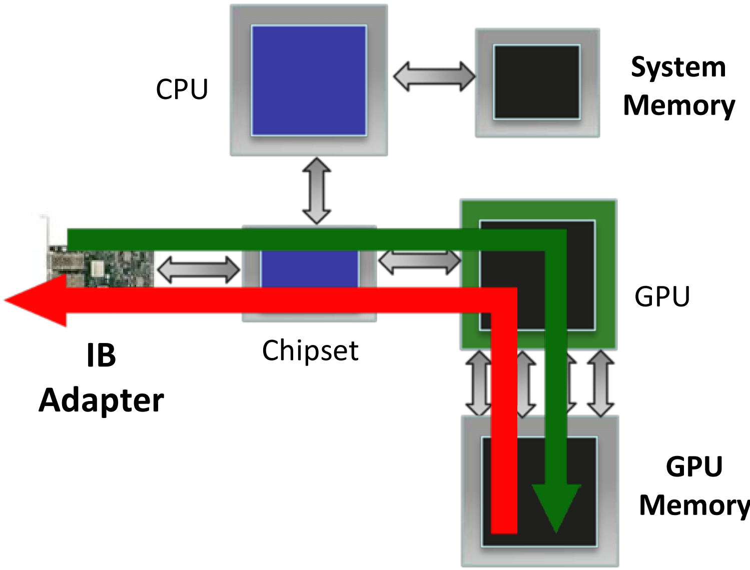
GPUDirect RDMA (GDR) with CUDA

- Hybrid design using GPUDirect RDMA
  - GPUDirect RDMA and Host-based pipelining
  - Alleviates P2P bandwidth bottlenecks on SandyBridge and IvyBridge
- Support for communication using multi-rail
- Support for Mellanox Connect-IB and ConnectX VPI adapters
- Support for RoCE with Mellanox ConnectX VPI adapters

S. Potluri, K. Hamidouche, A. Venkatesh, D. Bureddy and D. K. Panda, Efficient Inter-node MPI Communication using GPUDirect RDMA for InfiniBand Clusters with NVIDIA GPUs, Int'l Conference on Parallel Processing (ICPP '13)

SNB E5-2670 /

IVB E5-2680V2



SNB E5-2670

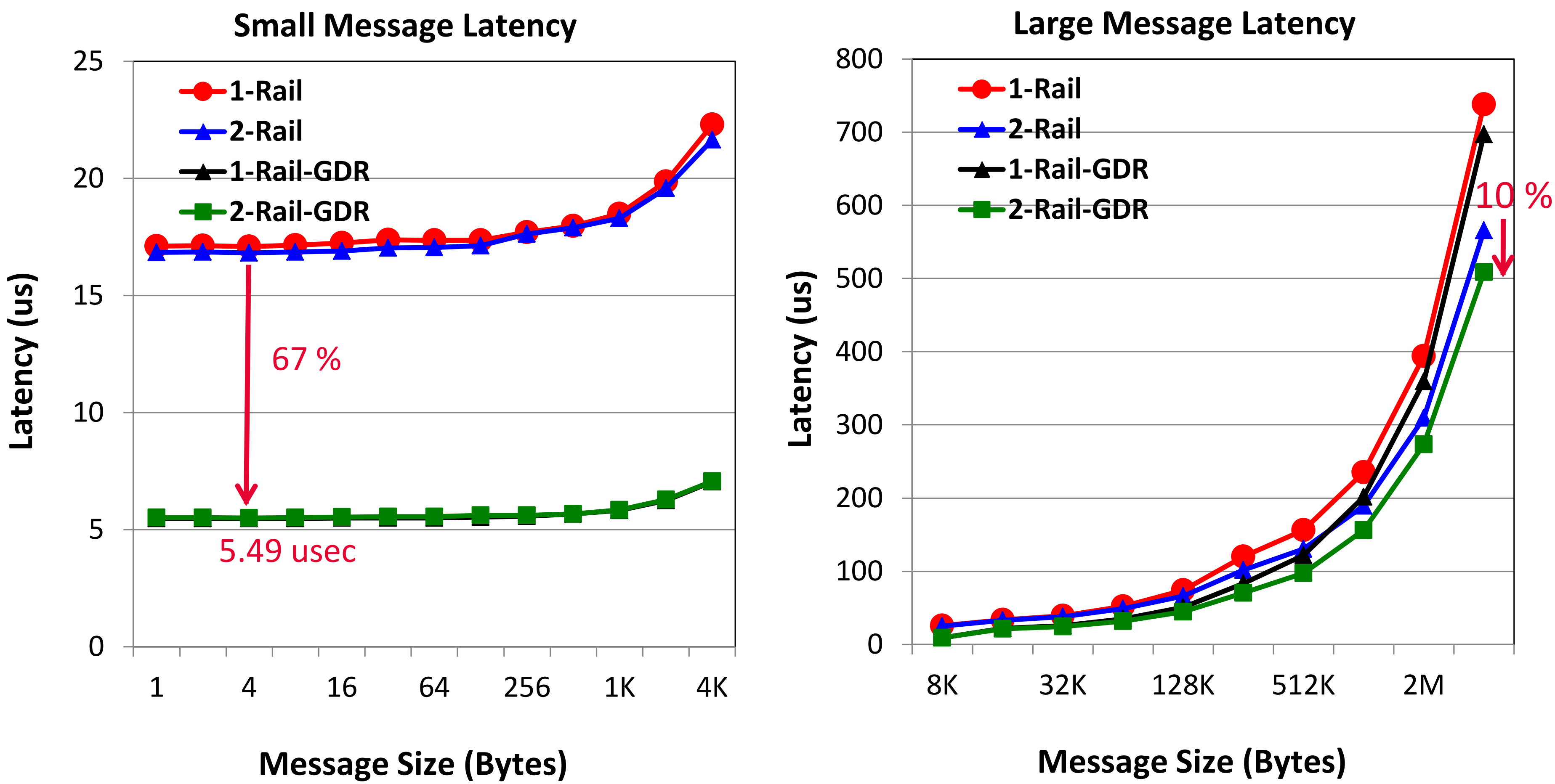
P2P write: 5.2 GB/s
P2P read: < 1.0 GB/s

IVB E5-2680V2

P2P write: 6.4 GB/s
P2P read: 3.5 GB/s

Performance of MVAPICH2 with GPUDirect-RDMA: Latency

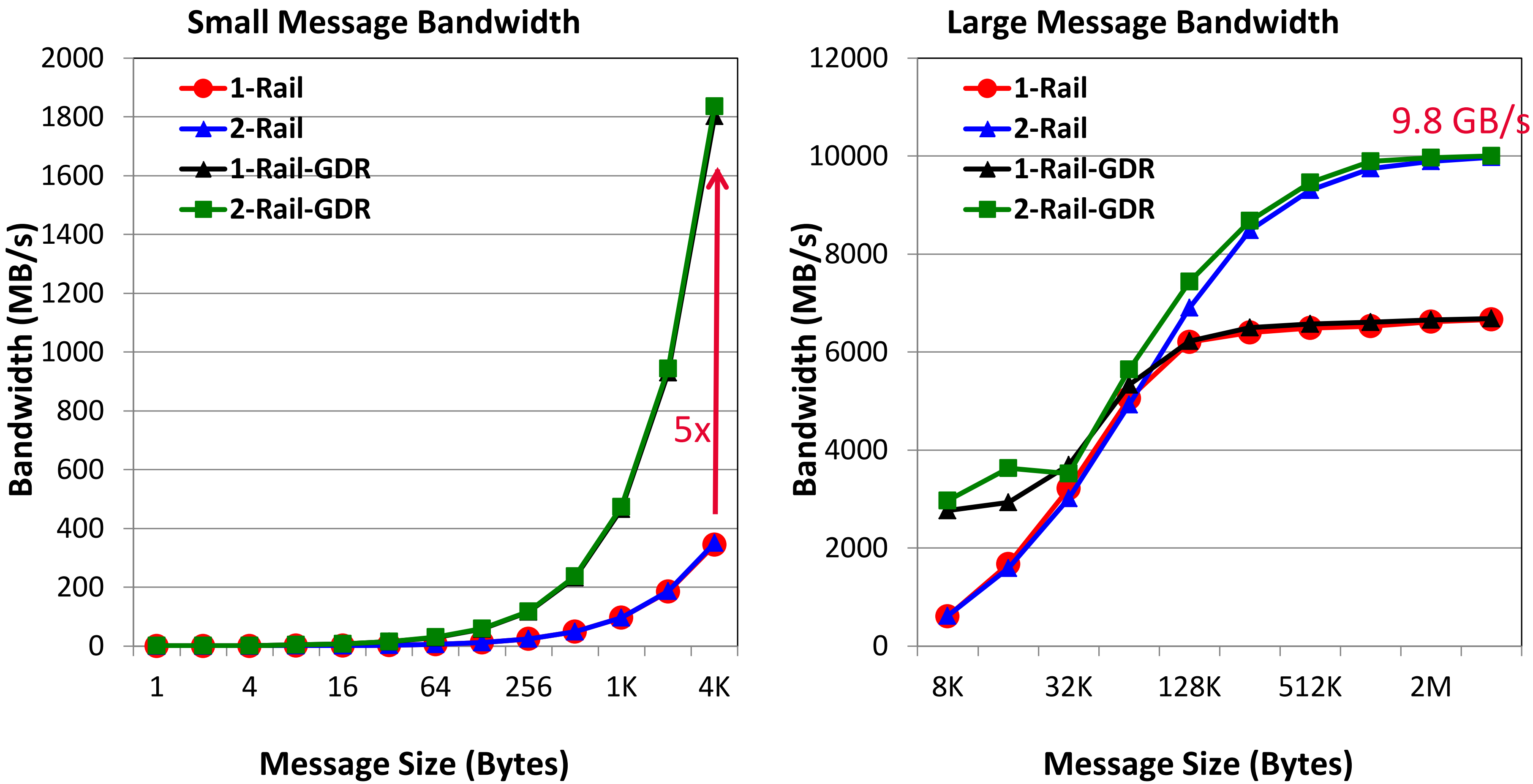
GPU-GPU Internode MPI Latency



Based on MVAPICH2-2.0b
Intel Ivy Bridge (E5-2680 v2) node with 20 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch

Performance of MVAPICH2 with GPUDirect-RDMA: Bandwidth

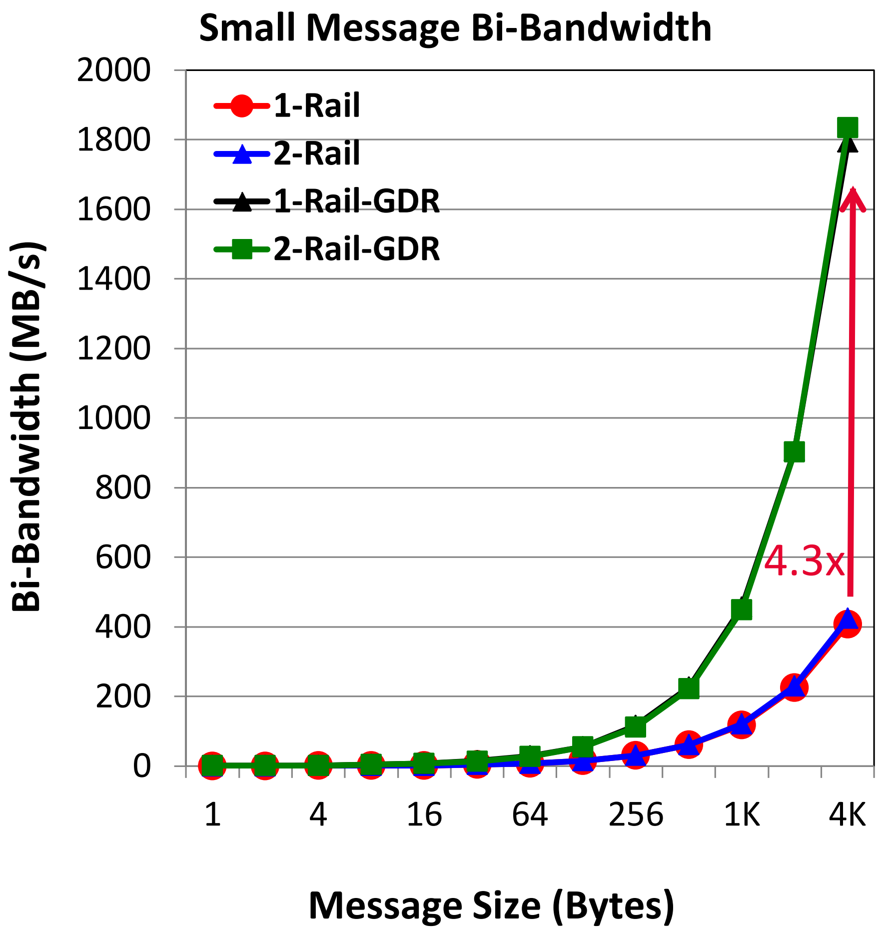
GPU-GPU Internode MPI Uni-Directional Bandwidth

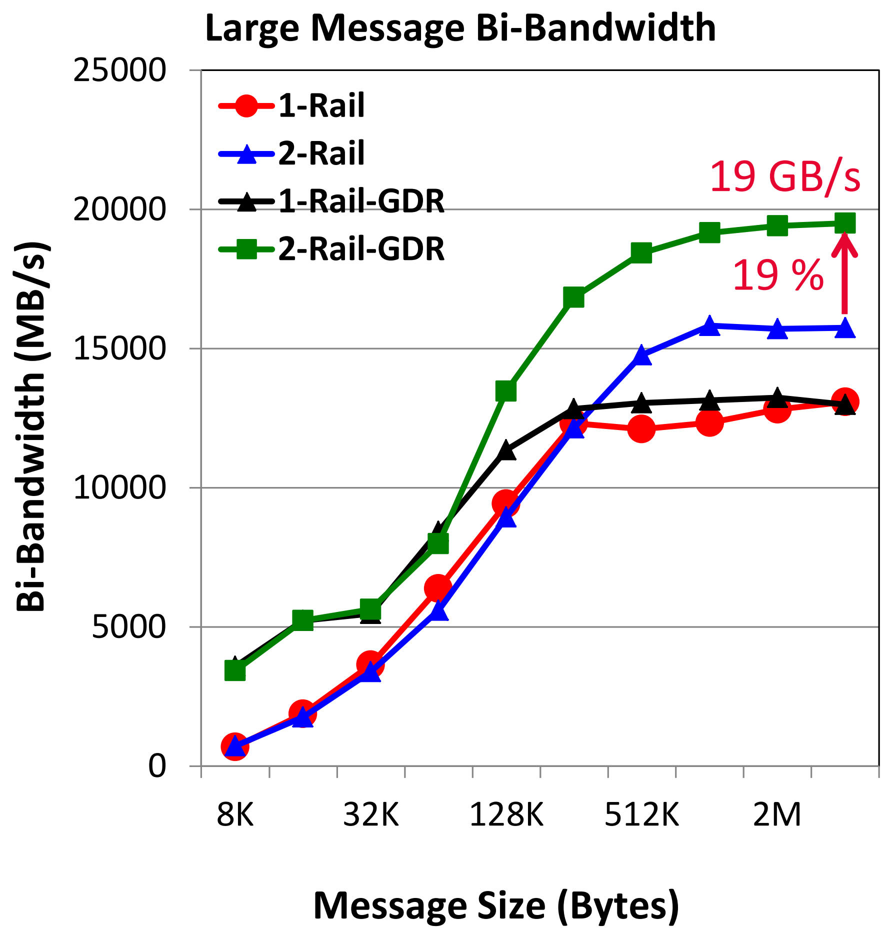


Based on MVAPICH2-2.0b
Intel Ivy Bridge (E5-2680 v2) node with 20 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch

Performance of MVAPICH2 with GPUDirect-RDMA: Bi-Bandwidth

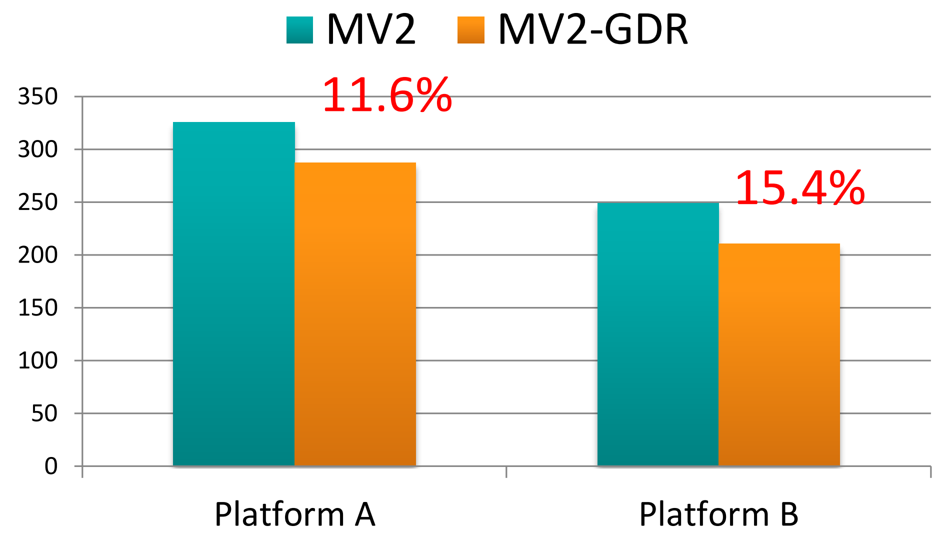
GPU-GPU Internode MPI Bi-directional Bandwidth

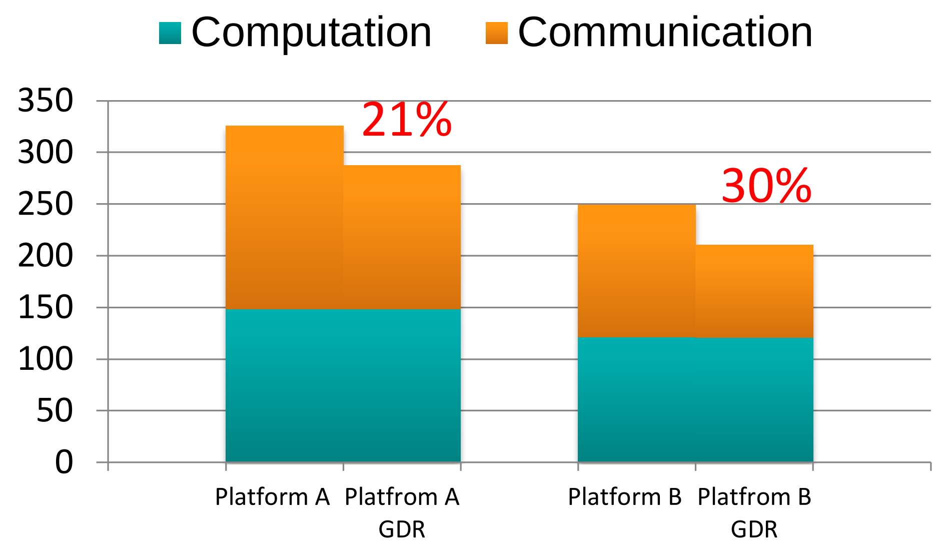




Based on MVAPICH2-2.0b
Intel Ivy Bridge (E5-2680 v2) node with 20 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch

Applications-level Benefits: AWP-ODC with MVAPICH2-GPU





Platform A: Intel Sandy Bridge + NVIDIA Tesla K20 + Mellanox ConnectX-3

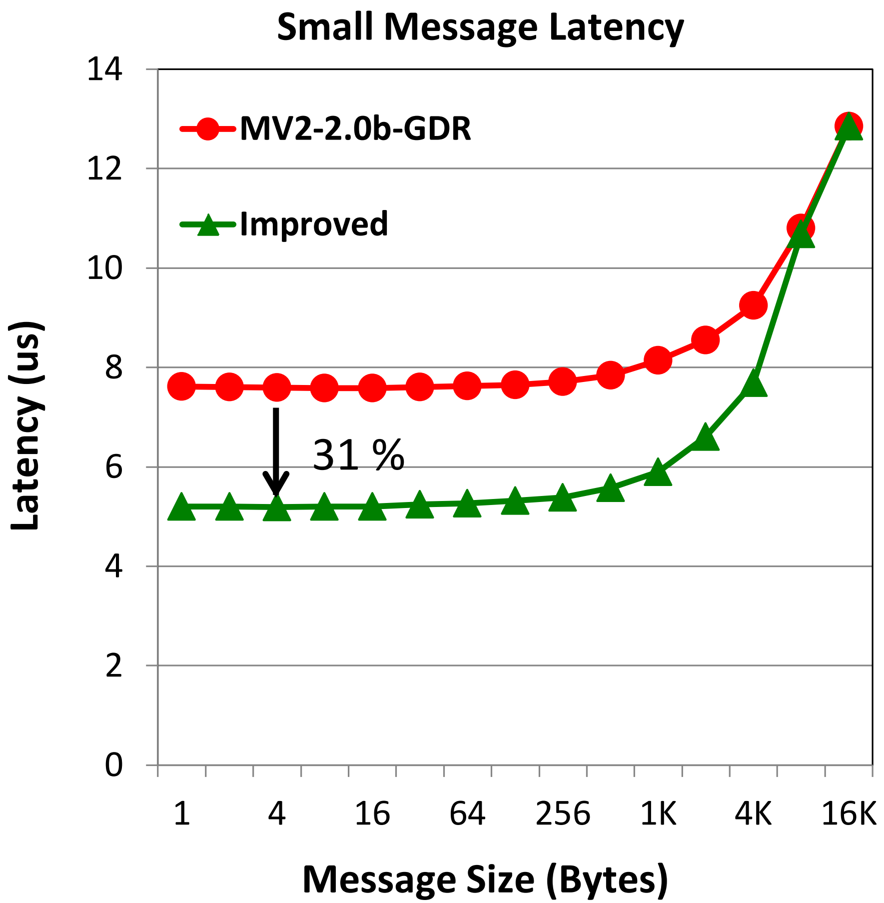
Platform B: Intel Ivy Bridge + NVIDIA Tesla K40 + Mellanox Connect-IB

- A widely-used seismic modeling application, Gordon Bell Finalist at SC 2010
- An initial version using MPI + CUDA for GPU clusters
- Takes advantage of CUDA-aware MPI, two nodes, 1 GPU/Node and 64x32x32 problem
- GPUDirect-RDMA delivers better performance with newer architecture

Based on MVAPICH2-2.0b, CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch
Two nodes, one GPU/node, one Process/GPU

Continuous Enhancements for Improved Point-to-point Performance

GPU-GPU Internode MPI Latency

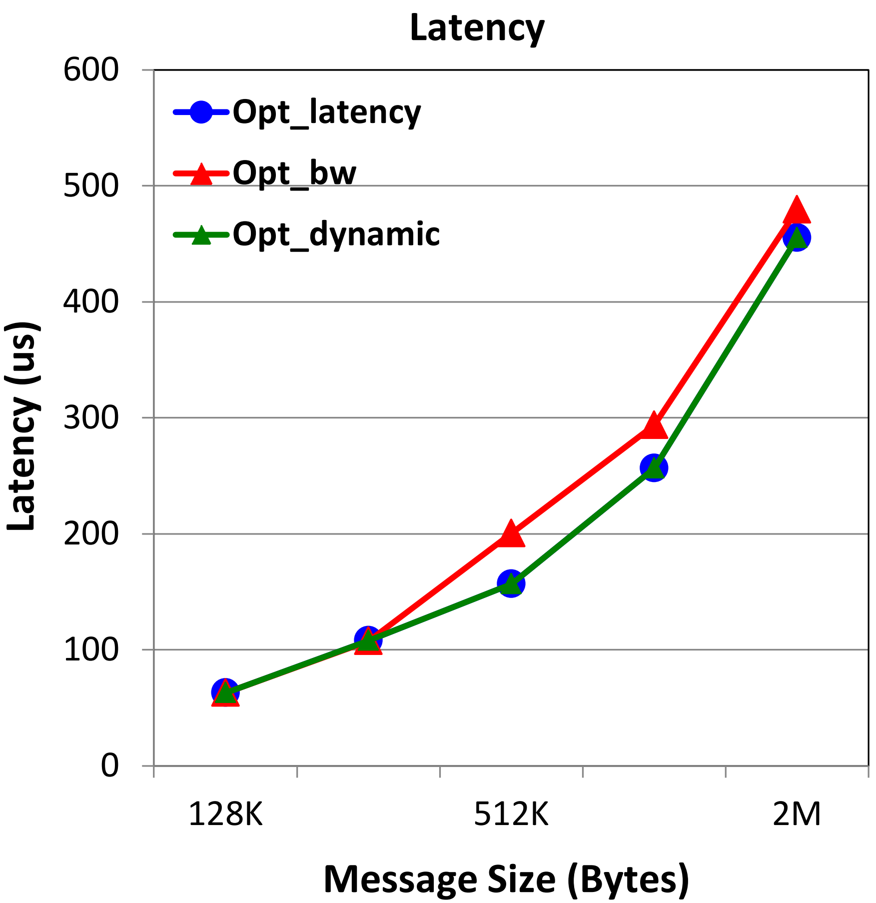


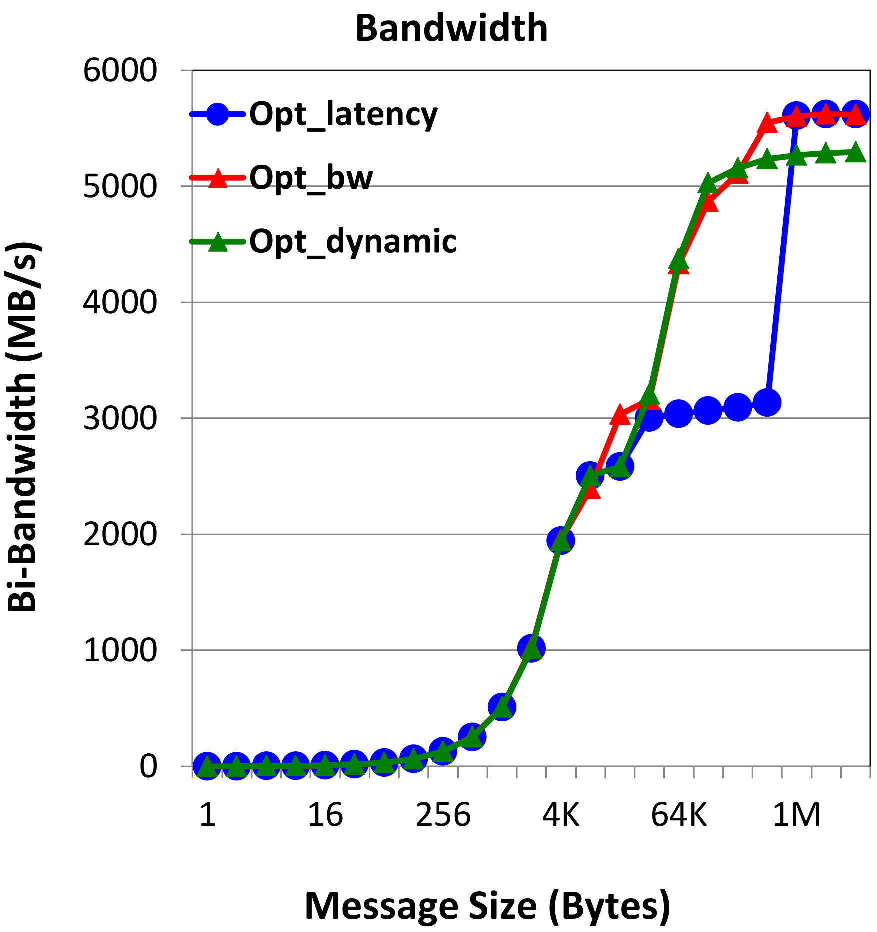
- Reduced synchronization and while avoiding expensive copies

Based on MVAPICH2-2.0b + enhancements
Intel Ivy Bridge (E5-2630 v2) node with 12 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch

Dynamic Tuning for Point-to-point Performance

GPU-GPU Internode MPI Performance





Based on MVAPICH2-2.0b + enhancements
Intel Ivy Bridge (E5-2630 v2) node with 12 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.0 with GPUDirect-RDMA Patch

Outline

- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
  - Two-sided Communication
  - One-sided Communication
  - MPI Datatype Processing
  - More Optimizations
- MPI and OpenACC
- Conclusion

One-sided communication

- Send/Recv semantics incur overheads
  - Distributed buffer information
  - Message matching
  - Additional copies or rendezvous exchange

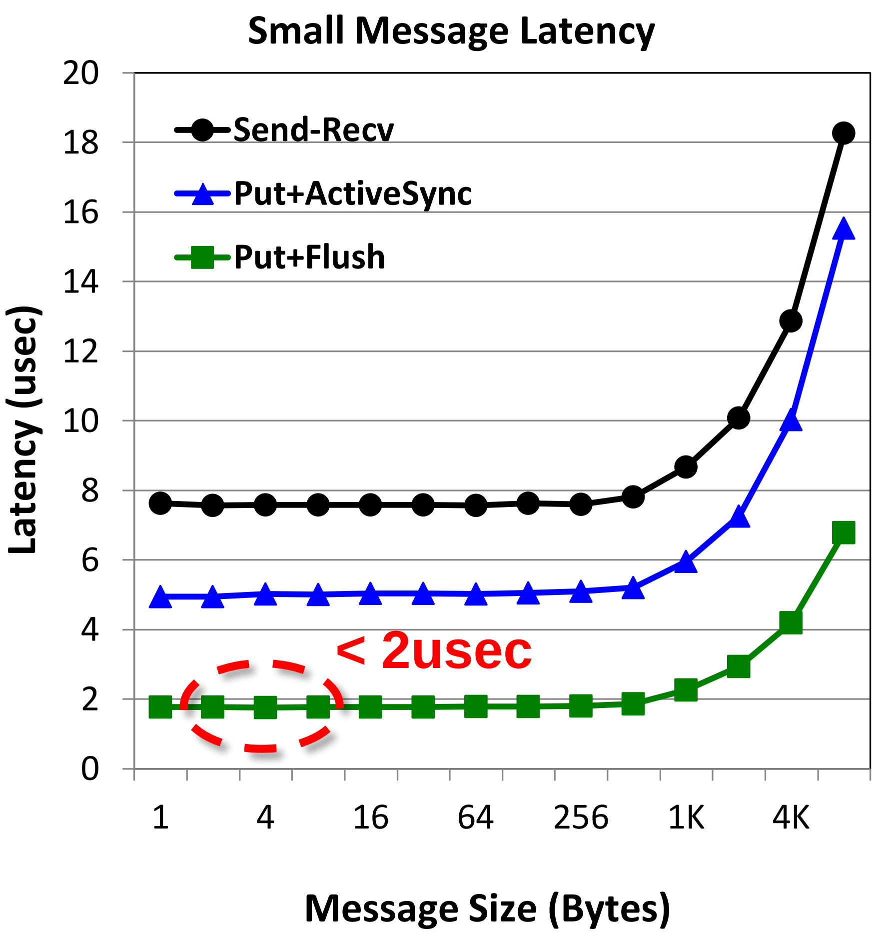
4 bytes	Host-Host	GPU-GPU
IB send/recv	0.98	1.84
MPI send/recv	1.25	6.95

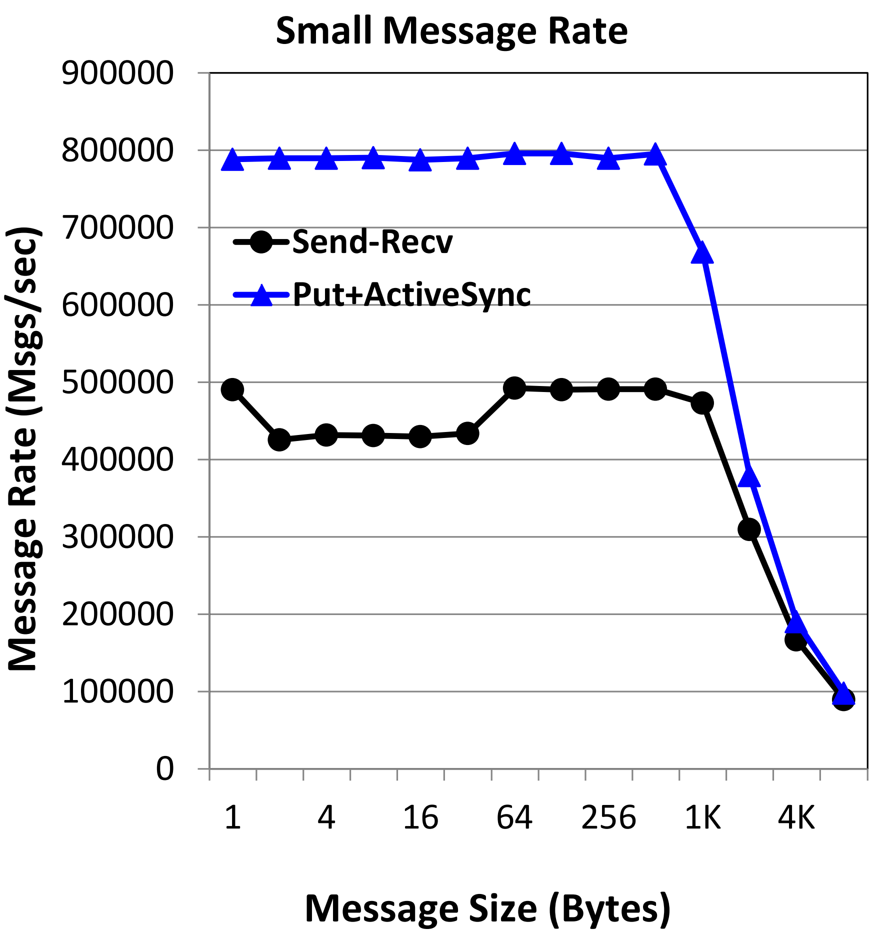
Table: Latency (half round trip) on SandyBridge nodes with FDR connect-IB

- One-sided communication
  - Separates synchronization from communication
  - Direct mapping over RDMA semantics
  - Lower overheads and better overlap

MPI-3 RMA Support with GPUDirect RDMA

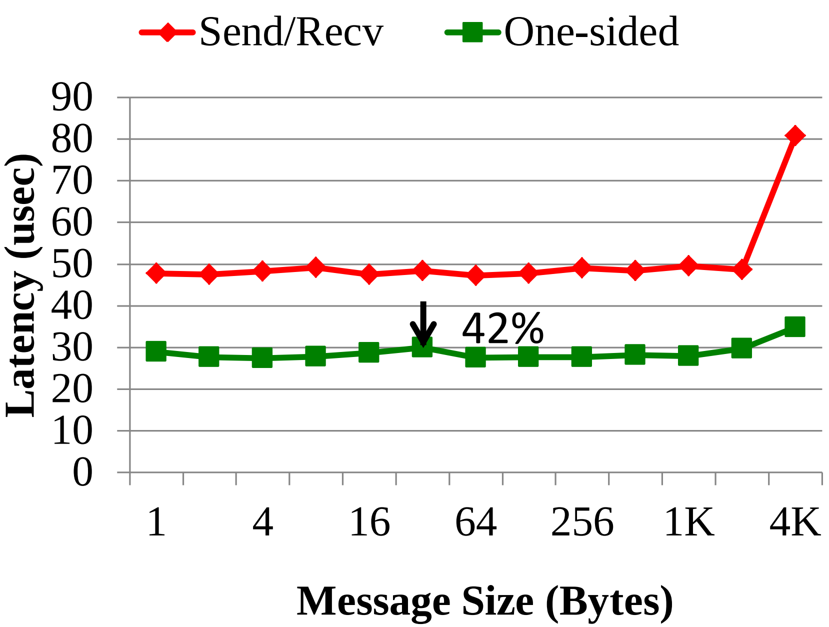
MPI-3 RMA provides flexible synchronization and completion primitives



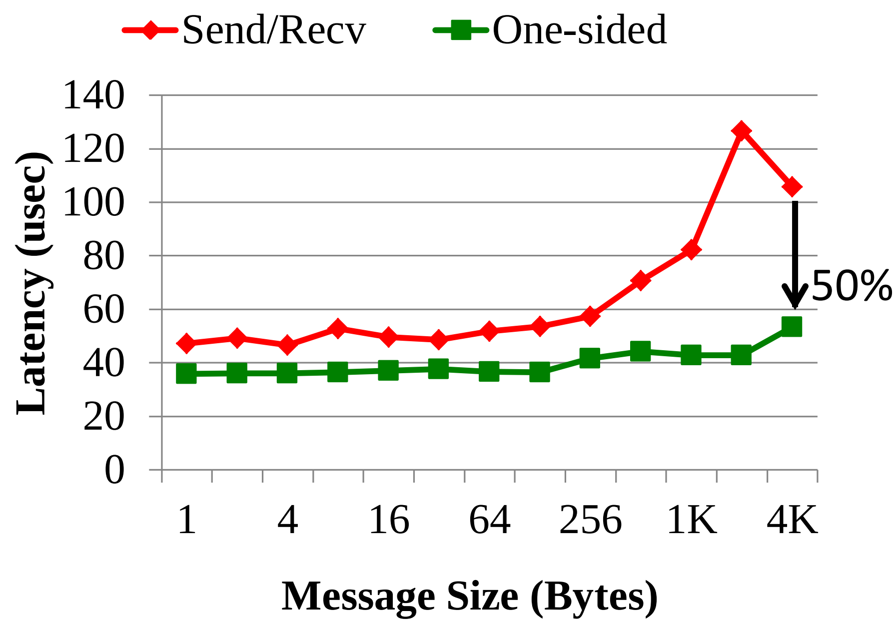


Based on MVAPICH2-2.0b + Extensions
Intel Sandy Bridge (E5-2670) node with 16 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.1 with GPUDirect-RDMA Plugin

Communication Kernel Evaluation: 3Dstencil and Alltoall



3D Stencil with 16 GPU nodes



AlltoAll with 16 GPU nodes

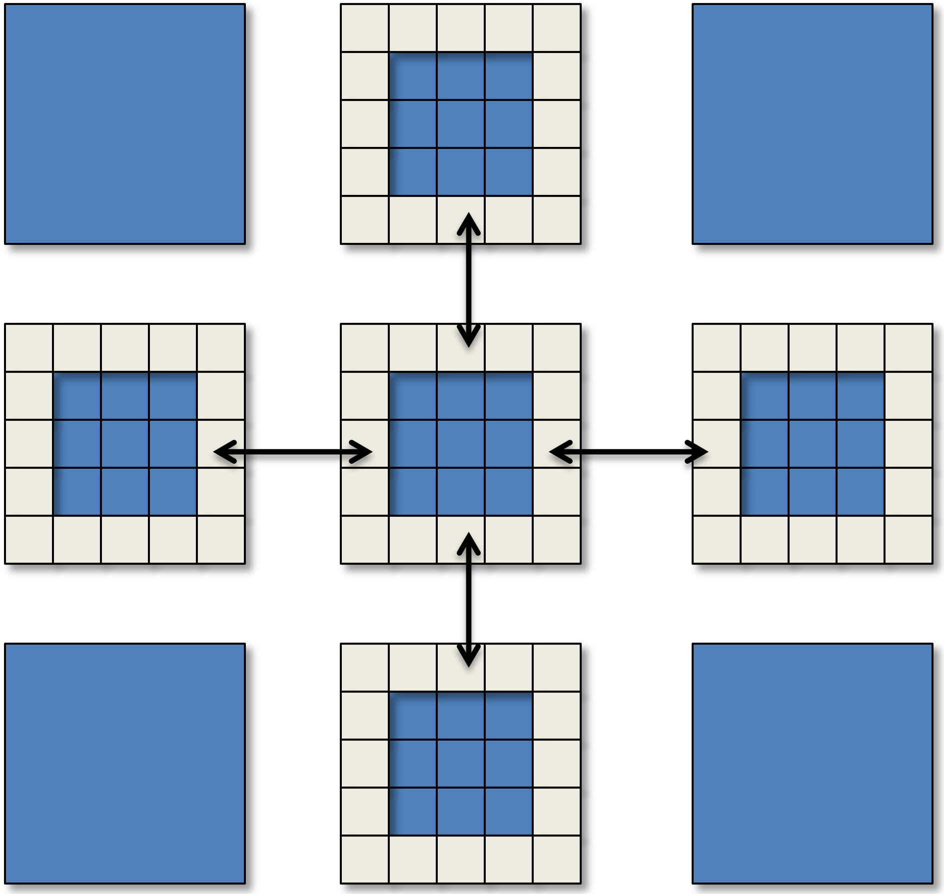
Based on MVAPICH2-2.0b + Extensions
Intel Sandy Bridge (E5-2670) node with 16 cores
NVIDIA Tesla K40c GPU, Mellanox Connect-IB Dual-FDR HCA
CUDA 5.5, Mellanox OFED 2.1 with GPUDirect-RDMA Plugin

Outline

- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
  - Two-sided Communication
  - One-sided Communication
  - MPI Datatype Processing
  - More Optimizations
- MPI and OpenACC
- Conclusion

Non-contiguous Data Exchange

Halo data exchange

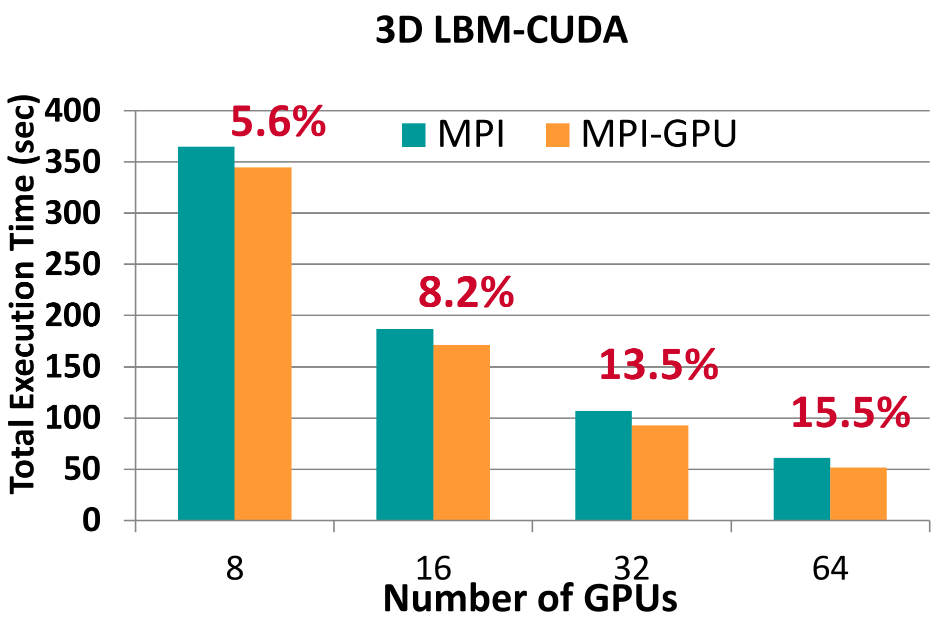


- Multi-dimensional data
  - Row based organization
  - Contiguous on one dimension
  - Non-contiguous on other dimensions
- Halo data exchange
  - Duplicate the boundary
  - Exchange the boundary in each iteration

MPI Datatye Processing

- Comprehensive support
  - Targeted kernels for regular datatypes - vector, subarray, indexed_block
  - Generic kernels for all other irregular datatypes
- Separate non-blocking stream for kernels launched by MPI library
  - Avoids stream conflicts with application kernels
- Flexible set of parameters for users to tune kernels
  - Vector
    - MV2_CUDA_KERNEL_VECTOR_TIDBLK_SIZE
    - MV2_CUDA_KERNEL_VECTOR_YSIZE
  - Subarray
    - MV2_CUDA_KERNEL_SUBARR_TIDBLK_SIZE
    - MV2_CUDA_KERNEL_SUBARR_XDIM
    - MV2_CUDA_KERNEL_SUBARR_YDIM
    - MV2_CUDA_KERNEL_SUBARR_ZDIM
  - Indexed_block
    - MV2_CUDA_KERNEL_IDXBLK_XDIM

Application-Level Evaluation (LBMGPU-3D)



- LBM-CUDA (Courtesy: Carlos Rosale, TACC)
  - Lattice Boltzmann Method for multiphase flows with large density ratios
  - **3D LBM-CUDA: one process/GPU per node, 512x512x512 data grid, up to 64 nodes**

- Oakley cluster at OSC: two hex-core Intel Westmere processors, two NVIDIA Tesla M2070, one Mellanox IB QDR MT26428 adapter and 48 GB of main memory

Outline

- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
    - Two-sided Communication
    - One-sided Communication
    - MPI Datatype Processing
    - More Optimizations
- MPI and OpenACC
- Conclusion

More Optimizations!!!

- Topology-detection:
  - Avoid the inter-sockets QPI bottlenecks
  - Dynamic threshold selection between GDR and host-based transfers
- All these and other features will be available with the next release of MVAPICH2-GDR => coming very soon

Outline

- Communication on InfiniBand Clusters with GPUs
- MVAPICH2-GPU with GPUDirect-RDMA (GDR)
    - Two-sided Communication
    - One-sided Communication
    - MPI Datatype Processing
    - More Optimizations
- MPI and OpenACC
- Conclusion

OpenACC

- OpenACC is gaining popularity
- Several sessions during GTC
- A set of compiler directives (#pragma)
- Offload specific loops or parallelizable sections in code onto accelerators

```
#pragma acc region
{
    for(i = 0; i < size; i++) {
        A[i] = B[i] + C[i];
    }
}
```

- Routines to allocate/free memory on accelerators

```
buffer = acc_malloc(MYBUFSIZE);
acc_free(buffer);
```

- Supported for C, C++ and Fortran
- Huge list of modifiers – **copy, copyout, private, independent, etc..**

Using MVPICH2 with the new OpenACC 2.0

- acc_deviceptr to get device pointer (in OpenACC 2.0)
  - Enables MPI communication from memory allocated by compiler when it is available in OpenACC 2.0 implementations
  - MVAPICH2 will detect the device pointer and optimize communication
  - Delivers the same performance as with CUDA

```
A = malloc(sizeof(int) * N);

……

#pragma acc data copyin(A)
{

#pragma acc parallel for
//compute for  loop

MPI_Send(acc_deviceptr(A), N, MPI_INT, 0, 1, MPI_COMM_WORLD);

}
……
free(A);
```

How can I get Started with GDR Experimentation?

- MVAPICH2-2.0b with GDR support can be downloaded from
  https://mvapich.cse.ohio-state.edu/download/mvapich2gdr/
- System software requirements
  - Mellanox OFED 2.1
  - NVIDIA Driver 331.20 or later
  - NVIDIA CUDA Toolkit 5.5
  - Plugin for GPUDirect RDMA
    (http://www.mellanox.com/page/products_dyn?product_family=116)
- Has optimized designs for point-to-point communication using GDR
- Work under progress for optimizing collective and one-sided communication
- Contact MVAPICH help list with any questions related to the package mvapich-help@cse.ohio-state.edu
- MVAPICH2-GDR-RC1 with additional optimizations coming soon!!

Conclusions

- MVAPICH2 optimizes MPI communication on InfiniBand clusters with GPUs
- Provides optimized designs for point-to-point two-sided and one-sided communication, and datatype processing
- Takes advantage of CUDA features like IPC and GPUDirect RDMA
- Delivers
  - High performance
  - High productivity

  With support for latest NVIDIA GPUs and InfiniBand Adapters

Acknowledgments

Dr. Davide Rossetti and others @NVIDIA

Talk on Hybrid HPL for Heterogeneous Clusters

Want to improve the top500 ranking of your heterogeneous GPU Cluster?

Yes !!

Do not miss our next talk –

S4535 - Accelerating HPL on Heterogeneous Clusters with NVIDIA GPUs

Tuesday, 03/25 (today)

Room LL21A

17:00 – 17:25

Thank You!

panda@cse.ohio-state.edu

Network-Based Computing Laboratory

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

MVAPICH Web Page

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