High-Performance and Scalable Broadcast Schemes for Deep Learning on GPU Clusters

Ching-Hsiang Chu, Dhabaleswar K. (DK) Panda (**Advisor**)
Department of Computer Science and Engineering, The Ohio State University
chu.368@osu.edu, panda@cse.ohio-state.edu

ABSTRACT

Broadcast operations are a widely used operation in many streaming and deep learning applications to disseminate large amounts of data on emerging heterogeneous High-Performance Computing (HPC) systems. Further, traditional broadcast schemes are not well optimized for upcoming large-scale Graphics Processing Unit (GPU)-based systems. However, utilizing cutting-edge features of modern HPC technologies such like InfiniBand (IB) and NVIDIA GPUs to enable scalable heterogeneous broadcast operations remains an open challenge.

Toward delivering the best performance for streaming and deep learning workloads, we propose high-performance and scalable broadcast schemes that exploit IB-MCAST and NVIDIA GPUDirect technology. We present experimental results and find that they indicate improved scalability and up to 68% reduction of latency compared to the state-of-the-art solutions in the benchmark-level evaluation. Furthermore, the proposed design yields up to 24% performance improvement for the popular deep learning framework, Microsoft cognitive toolkit (CNTK), with no application changes.

1 INTRODUCTION

Emerging high-performance computing (HPC) systems widely employ graphics processing units (GPUs) and InfiniBand (IB) interconnect to boost their performance and scalability. As current applications require processing increasingly large datasets, it is becoming common to utilize large-scale GPU clusters for streaming and deep learning (DL) applications, where *broadcast* operations are often involved for exchanging large amounts of data across GPU-enabled nodes. However, traditional Ring and K-nomial-based broadcast schemes are not well-optimized. The key contributions of this work include:

- High-performance and scalable **zero-copy heterogeneous broadcast** operations, which are required for **streaming applications**, by taking advantage of IB hardware-based multicast (**IB-MCAST**) and NVIDIA GPUDirect Remote Direct Memory Access (RDMA) (**GDR**) technology (Section 2.1)
- **Inter-Process Communication (IPC) features based intra-node broadcast** for Dense-GPU systems (Section 2.2)
- A **Streaming broadcast** design for large-size GPU-to-GPU message transfers, situations that appear commonly in DL applications (Section 2.3)
- A novel Remote Memory Access (RMA)-based **reliability support** for unreliable IB-MCAST (Section 2.4)
- Performance evaluation of the proposed designs for deep learning models on a real-world GPU-enabled InfiniBand cluster (Section 3)
- **Performance prediction** of broadcast schemes for upcoming large-scale GPU clusters [5, 6]

Figure 1 summarizes how the proposed designs advance the state-of-the-art broadcast schemes.

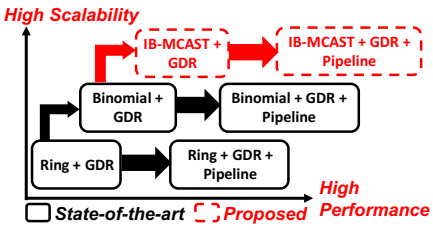


Figure 1: Overview of the existing and proposed broadcast schemes

2 PROPOSED BROADCAST SCHEMES

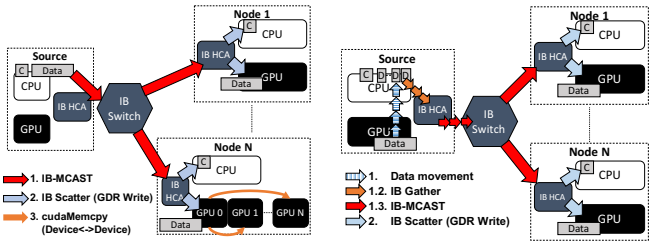
In this section, we briefly summarize the key concepts of the proposed broadcast schemes.

2.1 Designing Heterogeneous Broadcast Schemes

In heterogeneous broadcast operations, both the header and data need to reside in host and GPU memory, respectively, on the receiver side. In this work, we present a **zero-copy broadcast design** [3, 5], that combines GDR and IB-MCAST features. As shown in Figure 2(a), when data arrives, the IB Host Channel Adapter (HCA) performs a *scatter* operation to process header and data *in one IB message.*

2.2 Designing Intra-node Broadcast for Dense-GPU Systems

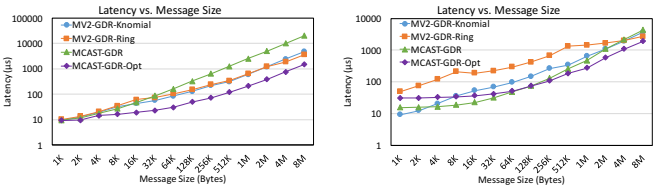
As illustrated in Figure 2(a), we leverage Inter-Process Communication (IPC) feature of NVIDIA GPU for intra-node broadcast [3, 5] on dense-GPU systems. The major benefits include 1) bypassing



(a) Proposed two-level heterogeneous broadcast scheme

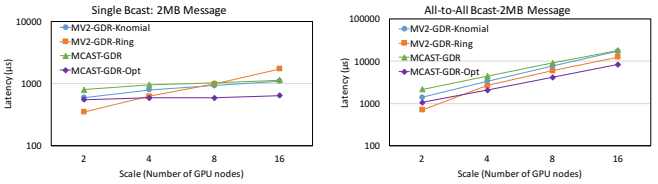
(b) Proposed streaming GPU-to-GPU broadcast scheme

Figure 2: Design overview of the proposed broadcast schemes



(a) Host-to-GPU Broadcast (b) GPU-to-GPU Broadcast

Figure 3: Latency comparison across 16 GPU nodes



(a) Single-source broadcast (b) All-to-all Broadcast

Figure 4: Scalability analysis

CPU 2) maximizing the benefits of the proposed zero-copy design at the inter-node level.

2.3 Optimizing Broadcast for Large Messages

The low PCIe bandwidth of GDR read operations [7] brings significant overhead for GPU-to-GPU broadcast, which is commonly used in DL applications. Therefore, we propose a two-step strategy [5, 6] as illustrated in Figure 2(b): (1) asynchronously stream the GPU-resident data to host memory, and (2) leverage the proposed zero-copy broadcast scheme, these two steps pipelined in a highly overlapped fashion.

2.4 Designing Efficient Reliability Support

Since IB-MCAST only works with the unreliable datagram protocol, we propose a high-performance Remote Memory Access (RMA) based reliability support [4] at the MPI-level although IB-MCAST has shown reasonably reliable data transmission in practice [4]. In the proposed scheme, the receivers retrieve lost IB-MCAST packets via RMA operations without interrupting the sender. Our experimental results show that the proposed scheme introduces no overhead compared to existing solutions [4].

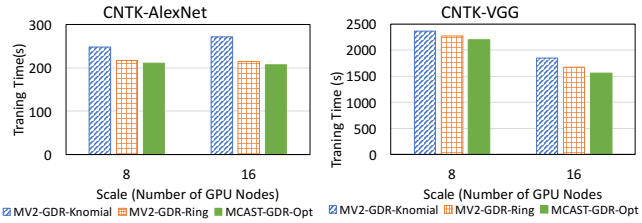
3 PERFORMANCE EVALUATION AND PREDICTION

Experiments were carried out on the **OSU-RI2** cluster, each node equipped with one NVIDIA Tesla K80 GPU. The proposed broadcast designs, labeled as *MCAST-GDR-Opt*, were implemented on top of MVAPICH2-GDR 2.2 [1], where the existing designs are labeled as *MV2-GDR-Ring* and *MV2-GDR-Knomial*.

3.1 Benchmark-level Evaluation

3.1.1 Heterogeneous Broadcast Operation. As exhibited in Figure 3(a), the proposed *MCAST-GDR-Opt* yields up to 70% lower latency than existing schemes.

3.1.2 GPU-to-GPU Broadcast Operation. As shown in Figure 3(b), the proposed design shows significant performance improvement



(a) AlexNet network (b) VGG network

Figure 5: Average training time for CNTK across 16 GPUs

for message sizes ranging from 4 KB to 2 MB, which encompasses typical message sizes in DL applications. The proposed design also exhibits stable latency independent of the number of GPU nodes as shown in Figure 4(a).

3.1.3 All-to-All GPU-based Broadcast Operation. We present the most commonly used all-to-all broadcast in Figure 4(b), which depicts good scalability and up to 68% reduced latency of the proposed design over existing approaches.

3.2 Evaluating Deep Learning Workloads

An optimized Microsoft Cognitive toolkit (CNTK) [2] is used for our evaluation. Figures 5(a) and 5(b) compare average one-epoch training performance of the popular AlexNet and VGG models, respectively. The proposed design reduces by up to 24% the training time compared to existing designs on 16 GPUs, notably with **no application-level code changes**.

4 IMPACT AND FUTURE WORK

The proposed designs will be included as the part of MVAPICH2, which is being used by more than 2,825 organizations in 85 countries worldwide [1].

Future work includes optimizing other broadcast-based collective operations and evaluating the resulting designs across various applications in upcoming large-scale dense-GPU clusters.

REFERENCES

[1] 2017. MVAPICH: MPI over InfiniBand, Omni-Path, Ethernet/iWARP, and RoCE. (2017). http://mvapich.cse.ohio-state.edu/ Accessed: October 3, 2017.

[2] D. S. Banerjee, K. Hamidouche, and D. K. Panda. 2016. Re-Designing CNTK Deep Learning Framework on Modern GPU Enabled Clusters. In *2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. 144–151.

[3] Ching-Hsiang Chu, Khaled Hamidouche, Hari Subramoni, Akshay Venkatesh, Bracy Elton, and D. K. Panda. 2016. Designing High Performance Heterogeneous Broadcast for Streaming Applications on GPU Clusters. In *28th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*. 59–66.

[4] Ching-Hsiang Chu, Khaled Hamidouche, Hari Subramoni, Akshay Venkatesh, Bracy Elton, and D. K. Panda. 2016. Efficient Reliability Support for Hardware Multicast-Based Broadcast in GPU-enabled Streaming Applications. In *First International Workshop on Communication Optimizations in HPC (COMHPC)*. 29–38.

[5] Ching-Hsiang Chu, Xiaoyi Lu, Ammar A. Awan, Hari Subramoni, Bracy Elton, and D. K. Panda. 2017. Exploiting Hardware Multicast and GPUDirect RDMA for Efficient Broadcast. *IEEE Transactions on Parallel and Distributed Systems* (2017). (Under review).

[6] C. H. Chu, X. Lu, A. A. Awan, H. Subramoni, J. Hashmi, B. Elton, and D. K. Panda. 2017. Efficient and Scalable Multi-Source Streaming Broadcast on GPU Clusters for Deep Learning. In *2017 46th International Conference on Parallel Processing (ICPP)*. 161–170.

[7] S. Potluri, K. Hamidouche, A. Venkatesh, D. Bureddy, and D.K. Panda. 2013. Efficient Inter-node MPI Communication Using GPUDirect RDMA for InfiniBand Clusters with NVIDIA GPUs. In *Parallel Processing (ICPP), 2013 42nd International Conference on*. 80–89.