A Reconfigurable Optical Network for Distributed Deep Learning

Wenzhe Li
Institute of Computing Technology, CAS
Beijing, China
liwenzhe@ncic.ac.cn

Guojun Yuan
Institute of Computing Technology, CAS
Beijing, China
yuanguojun@ncic.ac.cn

Changliang Wu
Institute of Computing Technology, CAS
Beijing, China
wuchangliang@ncic.ac.cn

Pingping Huang
Institute of Computing Technology, CAS
Beijing, China
huangpingping20g@ncic.ac.cn

Zhan Wang
Institute of Computing Technology, CAS
Beijing, China
wangzhan@ncic.ac.cn

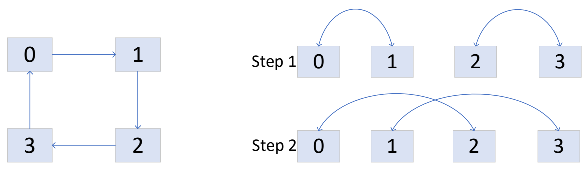
***Abstract*—We propose a reconfigurable optical network to accelerate distributed deep learning. The optical network reconfigures the topology to adapt different all-reduce algorithms and models. We demonstrate the performance improvement with a 32-node prototype.**

Keywords—Distributed deep learning, Reconfigurable optical network, Data-parallelism, All-reduce algorithm

I. INTRODUCTION

Due to the success of the deep learning on language processing, objective detection, image classification, the amount of computing power used in the largest AI training has doubled on average every 3.4 months [1], while Moore's law has a 2-year doubling time. The hardware plays an important role in the development of deep learning.

As the size of input datasets increases, data-parallelism has become an attractive solution for the training of deep learning. However, a large number of parameters have to be synchronized among all devices during data-parallelism training, putting a heavy burden on network. For example, GPT-3, which achieves excellent performance on producing human-like text, has over 175 billion machine learning parameters [1]. Communication time accounts for a large proportion of the training time, and it has become the main bottleneck of large-scale distributed deep learning. It is important to improve the network performance for distributed deep learning (DDL) systems.



(a) Ring all-reduce (b) RD all-reduce

Fig. 1 The comparison of two all-reduce algorithms.

Data parallelism uses all-reduce operations for gradient synchronization. Many all-reduce algorithms have been proposed to improve the performance of data parallelism, including Ring, Recursive Doubling (RD), and Double Binary Tree, and others. Different algorithms are suitable for different training scenarios. Ring algorithm (shown in Fig. 1(a)) has high bandwidth utilization, thereby achieving good performance in the middle-size system. However, Ring algorithm suffers from low efficiency when it is used for a larger-size system. RD algorithm (shown in Fig. 1(b)) outperforms Ring algorithm in large-size systems due to fewer steps. But RD algorithm requires that there are 2N devices in training systems. With various all-reduce algorithms, the all-reduce operations in different training systems can be optimized. A high performance computing (HPC) system usually employs topology such as torus [3] or dragonfly [4], which is a fixed topology. However, HPC systems with fixed topology cannot fulfill the requirement of different all-reduce algorithms and models. Recently, reconfigurable optical network demonstrates its advantage over traditional multi-tier electrical network in Google's data centers [3]. At the same time, deep learning applications have predictable traffic patterns, improving the reliability and efficiency of the reconfiguration scheme.

Therefore, we utilize a reconfigurable optical network architecture to improve the performance of data parallelism by adapting different all-reduce algorithms. To analyze the performance of network topologies under different all-reduce algorithms, we develop a distributed training simulator based on real distributed applications. Then, we simulate the 7 all-reduce algorithms and 5 candidate topologies with different system sizes. Each all-reduce algorithm has an optimal topology, and this result can be used for the reconfiguration strategy. In addition, we build a 32-node prototype of the reconfigurable training system and measure the training speed of model resnet50. Experimental results show that the reconfigurable optical network can improve overall performance by reconfiguring the optical layer.

II. RECONFIGURABLE OPTICAL NETWORK ARCHITECTURE

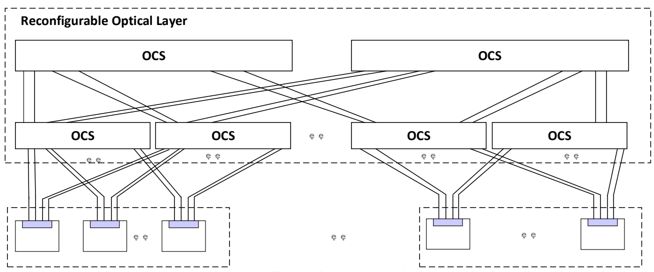


Fig. 2 The structure of reconfigurable optical network for distributed training system

The structure of the reconfigurable optical network for distributed training is shown in Fig. 2. The reconfigurable optical layer will reconfigure the interconnection between nodes according to the allreduce algorithm. The optical layer consists of optical circuit switches (OCS), which can establish paths between pairs of nodes. Each server connects to the optical layer through multiple optical ports.

The network topology can be configured into Torus, Ring, and Binary Tree.

III. Real Training Data Based on Simulation

To simulate realistic training traffic, we first run several models on a single 2080Ti GPU. We use the Horovod Timeline function to get the trace file of training, which includes the beginning and end times of each iteration, and the communication volume after each iteration. We then import these files into our OMNeT++based simulator.

TABLE I. Simulation Parameters

Link delay	5 µs
Bandwidth	100 Gbps
Fusion threshold	1 MB
Fusion time	10µs
Inner node delay	87µs
Topology	One_swith Torus Fattree Ring Bianry tree
Memory bandwidth	127.8 Gbps

As shown in Table I, we obtain the parameters from the server that runs the real models. We simulate four models: Lenet, Resnet50, Vgg13, and Bert as training models. We implement 7 allreduce algorithms, including BTree, Ring, PSLike, Torus, DBTree, and RD. We use the One_switch topology, which means all nodes are connected with one non-blocking electrical switch, as an ideal network. We also simulate L3_fattree topology to compare the network performance.

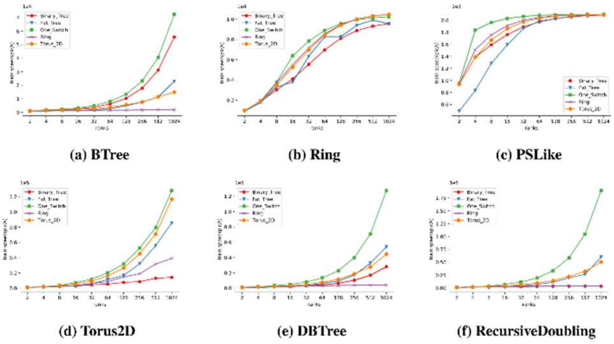


Fig. 3 The comparison of training speed under Lenet

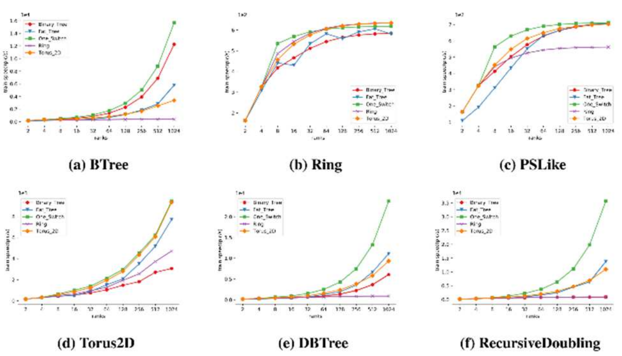


Fig. 4 The comparison of training speed under Resnet50

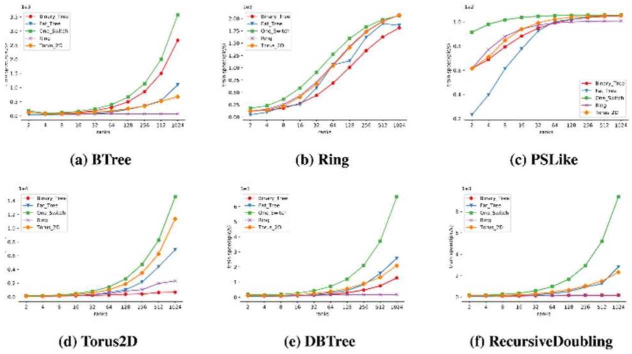


Fig. 5 The comparison of training speed under Vgg13

The simulation results under Lenet are shown in Fig. 3. Ring allreduce achieves the highest training speed with the Binary Tree topology (except the ideal network). The Torus topology has an obvious advantage over the other three network topologies under Torus allreduce. For Ring and PSlike allreduce, the Ring topology has a better performance compared to others with small-size system. For DBTree and RD allreduce, the Torus topology outperforms the Ring and Binary Tree topologies.

Fig. 4 illustrates the comparison results under Resnet50. For PSlike allreduce, the Torus topology has a performance advantage, which is different from the results of Lenet. Other allreduce algorithms show similar trends to those of Lenet. Fig. 5 illustrates the comparison results under Vgg13. For Ring, PSLike, RD, DBTree and Torus allreduce, the Torus topology outperforms the Ring and Binary Tree topologies. For BTree allreduce, the Binary Tree is the optimal topology. We can conclude that the optimal topology is related to allreduce algorithm and training model.

IV. Prototype of Reconfigurable Optical Network

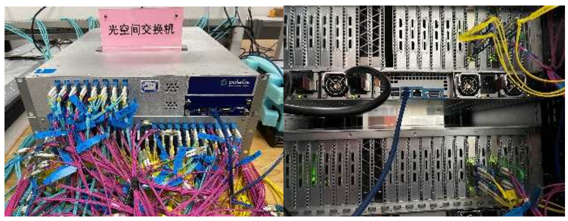


Fig. 6 The picture of the prototype

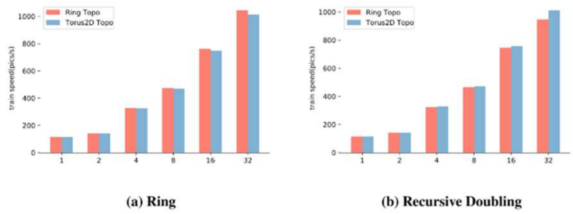


Fig. 7 Experiment results of the prototype

As shown in Fig.6, we build a prototype with 32 servers, each containing 4 100Gbps ports and 1 M40 GPU. We connect all ports with a 32×32 and a 96×96 Polatis MEMS switch. The topology can be configured into Ring or Torus. We use Resnet50 as the training model on the prototype system. We employ Ring and RD allreduce algorithms. As we scale the system to 32 nodes, Resnet50 with Ring allreduce algorithm achieves higher training speed with the Ring topology. When RD allreduce algorithm is deployed, we reconfigure the system into the Torus topology (shown in Fig.7) to improve the overall performance. These results demonstrate that the reconfigurable optical network is feasible and efficient for DDL.

V. Conclusion

In this paper, we utilize the reconfigurable optical network to accelerate the data parallelism in DDL. We simulate various models, network topologies, and allreduce algorithms using real training applications. We also build a 32-node distributed training system with the reconfigurable optical network. The results demonstrate that the training speed of Resnet50 can be improved with the reconfigurable optical network.

Acknowledgment

This work is supported in part by National Science and Technology Innovation Program 2030 (No. 2020AAA0104402), NSFC (No. 61972380 and 62102396), and Youth Innovation Promotion Association of Chinese Academy of Sciences (2021099).

References

[1] Sevilla, J., Heim, L., Ho, A., Besiroglu, T., Hobbhahn, M., & Villalobos, P , "Compute trends across three eras of machine learning", 2022 International Joint Conference on Neural Networks (IJCNN) (pp. 1-8). 2022.

[2] Dale, R. (2021). "GPT-3: What's it good for? Natural Language Engineering", 27(1), 113-118. doi:10.1017/S1351324920000601J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol M. Dhanak, P. D. Godbole and R. A. Patil, "Torus network labeling in High Performance computing," 2016 International Conference on Computing Communication Control and automation (ICCUBEA), Pune, India, 2016, pp. 1-4, doi: 10.1109/ICCUBEA.2016.7859992.. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] M. Dhanak, P. D. Godbole and R. A. Patil, "Torus network labeling in High Performance computing," 2016 International Conference on Computing Communication Control and automation (ICCUBEA), Pune, India, 2016, pp. 1-4, doi: 10.1109/ICCUBEA.2016.7859992.

[4] G. Faanes et al., "Cray Cascade: A scalable HPC system based on a Dragonfly network," SC '12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, Salt Lake City, UT, USA, 2012, pp. 1-9, doi: 10.1109/SC.2012.39.

[5] Leon Poutievski, Omid Mashayekhi, Joon Ong, Arjun Singh, Mukarram Tariq, Rui Wang, Jianan Zhang, Virginia Beauregard, Patrick Conner, Steve Gribble, Rishi Kapoor, Stephen Kratzer, Nanfang Li, Hong Liu, Karthik Nagaraj, Jason Ornstein, Samir Sawhney, Ryohei Urata, Lorenzo Vicisano, Kevin Yasumura, Shidong Zhang, Junlan Zhou, and Amin Vahdat. 2022. "Jupiter evolving: transforming google's datacenter network via optical circuit switches and software-defined networking", In Proceedings of the ACM SIGCOMM 2022 Conference (SIGCOMM '22), New York, NY, USA, 66–85.