Efficient Allreduce Algorithm for Large-Scale Deep Learning on Distributed Loop Networks

Truong Thao Nguyen, Peng Chen, Yusuke Tanimura
{nguyen.truong,chin.hou,yusuke.tanimura}@aist.go.jp
National Institute of Advanced Industrial Science and Technology,, Tokyo, Japan

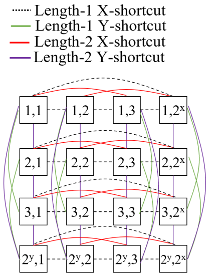

Figure 1: Proposed 2D-DLN network topology.

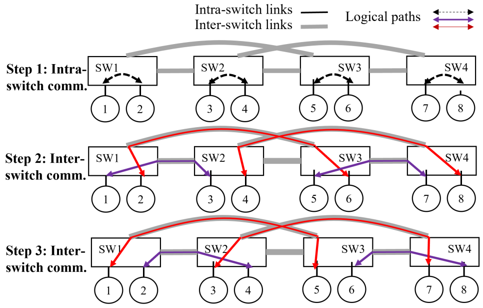
Intra-switch links
Inter-switch links
Logical paths
Step 1: Intra-switch comm.
Step 2: Inter-switch comm.
Step 3: Inter-switch comm.
SW1
SW2
SW3
SW4
1
2
3
4
5
6
7
8

Figure 2: Illustration of our Shifted Halving-Doubling Allreduce Algorithm.

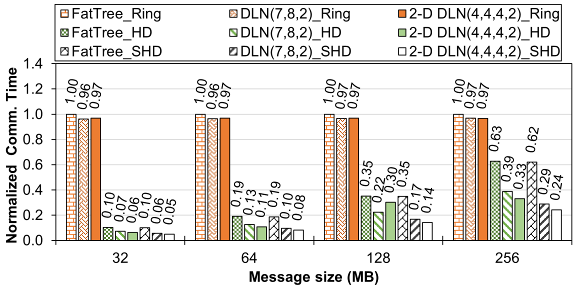

Figure 3: Normalized communication time of different [network]_[allreduce algorithm] to those of FatTree_Ring with 512 processing elements, e.g., GPUs.

1 INTRODUCTION

Training a Deep Learning model on High-Performance Computing systems is becoming a de-facto standard in deep learning. One of the key factors limiting the growth of large-scale training is the collective communication overhead between processing elements (PEs), e.g., an Allreduce operation. With the continual increase in model sizes, e.g., 100s GB, and the number of PEs, e.g., 1,000s of GPUs, communication becomes a major bottleneck. In this context, we aim at finding a network topology and its corresponding collective algorithm that features *bandwidth optimality in $O(\log(P))$ steps with a minimal (or without) network contention.* Our prior work [1] proposed the use of a family of network topologies that exploit small-world network models, e.g, Distributed Loop Network (DLN) and collective algorithms named *Shifted Halving-Doubling (SHD)* which improve the utilization of all the inter-switches links of the DLN topology. In this study, we generalize the SHD algorithm in [1] and propose 2-D DLN which considers both the communication performance and the cost when implemented in a server room.

2 ALGORITHM AND NETWORK CO-DESIGN

Distributed Loop Networks (DLN) topology, where 2^q nodes arranged into a ring with shortcuts connect a node i to node $i + 2^k$, has a distinct benefit to a Fat-tree network in terms of communication latency reduction [1]. However, it is difficult to deploy DLN networks into server rooms where many shortcut links are implemented via diagonal cables which increases the total cable length and deployment cost. We propose 2-D DLN(x, y, z) that arranges switches into a grid of $2^x \times 2^y$ switches with z down link/switch connected to the PEs. All the switches form a directed DLN in each dimension of the grid (Fig. 1). Clearly, 2-D DLN could avoid diagonal cabling problems, thus reducing the cost.

In the conventional Halving-Doubling (HD) collective algorithm, all the PEs of one switch would be paired with the PEs from the same destination switches. As a result, all the PEs of one switch use only one particular shortcut link at one step while other shortcuts are not used which caused network contention. In this work, we present Shifted Halving-Doubling (SHD), which fully utilizes all inter-switch links at each step. As shown in Fig. 2, the inter-switch communication is then performed in q steps, where 2^q is the number of involved switches. At the first step, a PE i-th of a switch u will be paired with the PE i-th of switch $u + 2^i$ where $i = 0 \ldots q - 1$, e.g., use the shortcut of distance 2^i. At the step j-th, we shift the shortcut used by PE i to be the one with distance $2^{(i+j) \bmod q}$.

3 SIMULATION RESULTS

We simulate the performance of an Allreduce micro-benchmark using the Simgrid framework v3.21 [2] with the same setting as mentioned in [1]. We compare our proposal with the baselines (FatTree_Ring and FatTree_HD) in Fig. 3. The halving-doubling family algorithms are relatively better than ring-based algorithms in the case of a big number of computing elements (GPUs). Furthermore, the communication time of SHD is only 75-80% of HD in most of the cases (in both DLN or 2-D DLN networks). This result clearly illustrates the impact of our design with shifted halving-doubling technique aiming at reducing network contention. Finally, we found that the communication of SHD on a 2-D DLN network is slightly faster than those on DLN. Therefore, we state that the new 2-D DLN network topologies have the same performance as 1-D DLN but have a clear advantage in its implementation cost.

4 ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI under Grant Number JP21K17751.This presentation is based on results obtained from a project, JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

REFERENCES

[1] Truong Thao Nguyen et al. 2021. An Allreduce Algorithm and Network Co-design for Large-Scale Training of Distributed Deep Learning. In *2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing*. 396–405.

[2] SimGrid: Versatile Simulation of Distributed Systems. [n. d.]. http://simgrid.gforge.inria.fr/. [01 April 2019].