Received XX Month, XXXX; revised XX Month, XXXX; accepted XX Month, XXXX; Date of publication XX Month, XXXX; date of current version XX Month, XXXX.

Digital Object Identifier 10.1109/OJCOMS.2022.1234567

Efficient Algorithm for All-gather Operation in Optical Interconnect Systems

FEI DAI [1,2], YAWEN CHEN [1], ZHIYI HUANG [1], AND HAIBO ZHANG [1]

[1]School of Computing, University of Otago, Dunedin, New Zealand
[2]School of Computing, Eastern Institute of Technology | Te Pūkenga, Napier, Hawke's Bay, New Zealand

CORRESPONDING AUTHOR: FEI DAI (e-mail: tdai@eit.ac.nz).

This article was the extension version of paper presented at the 2023 IEEE International Conference on Communications (ICC 2023) [1].

ABSTRACT In the realm of parallel and distributed computation, All-gather operation, a process where each node in a distributed system gathers data from all others, is pivotal. This operation underpins various high-performance computing (HPC) applications, notably in distributed deep learning (DL), by enabling model and hybrid parallelisms. Although optical interconnection networks promise unmatched bandwidth and reliability for data transfers between distributed nodes, most current All-gather algorithms remain optimized for electrical interconnects, leading to suboptimal performance in optical contexts. This paper proposes ”OpTree”, an advanced scheme distinctly designed for All-gather operation in optical interconnect systems. OpTree constructs an optimal m-ary tree that minimizes communication time by determining the optimal number of communication stages. A comprehensive comparison between OpTree's communication steps and existing All-gather algorithms is provided. Theoretical insights reveal that OpTree substantially curtails communication steps within optical interconnects. Constraints imposed by OpTree on optical communication are also elaborated. Empirical evaluations, through rigorous simulations, establish that: 1) OpTree is effective in generating an optimal m-ary tree for minimizing communication time. 2) For a 1024-node optical ring system, OpTree cuts communication time by 72.97%, 93.15%, and 86.32% against WRHT, Ring, and Neighbor Exchange (NE) schemes, respectively, tested over different message sizes. 3) With varying node counts, the reductions stand at 42.27%, 92.74%, and 85.49% against the same counterparts. 4) As the number of wavelengths increases, communication time further diminishes.

INDEX TERMS Optical Interconnects, All-gather, Communication, Wavelength Division Multiplexing (WDM)

I. INTRODUCTION

As the number of GPUs or other accelerators integrated into systems increases, efficient communication among these devices becomes crucial for running HPC applications. All-to-all communication methods, such as the message passing interface (MPI) All-gather operation, are widely used in many legacy scientific applications to perform Fast Fourier Transforms (FFTs) when data is distributed among multiple processes [2]. The All-gather operation is also gaining attention for performing model or hybrid parallelisms in the training of distributed Deep Neural Networks (DNNs) on GPU clusters [3]. Given the current software and hardware development for machine learning, optimizing All-gather operation in dense-GPU systems is of great importance, both in current systems and in next-generation systems.

Although many algorithms have been proposed to improve the performance of the All-gather operation in electrical interconnect systems, communication bottlenecks still exist in these systems [4]. For instance, in the training of distributed model-parallel DNNs, frequent communications with large data transfers between distributed nodes are required, which can lead to communication bottlenecks in electrical interconnect systems. However, with the advancement of silicon photonics, emerging optical networks can offer increased concurrent communication capacities through Wavelength Division Multiplexing (WDM), promising data transmission rates that are several orders of magnitude higher than current electrical networks. By using optical networks, communication bottlenecks can be avoided for time-sensitive applications in data centers, such as parallel and distributed training of deep learning.

Most All-gather algorithms are designed for electrical interconnect systems using the packet-switching mechanism and have to share bandwidth in the electrical wires. In contrast, optical transmission is usually based on circuit switching, where each optical communication path monopolizes the bandwidth by occupying one wavelength for transmission. As a result, traditional All-gather algorithms do not work well for optical interconnect systems as they do not take into account the unique features of optical communication. While some research has been done on all-to-all broadcast in optical networks, the focus has primarily been on fault tolerance [5] or reducing the number of wavelengths used [6], rather than reducing communication time.

Inspired by uniform routing, where each node evenly distributes communication across the network, and the multi-stage model [6] in optical networks, we propose an efficient scheme, OpTree, for All-gather operation on optical ring interconnect systems, aiming at minimizing the communication time. We use the ring topology because it is more feasible to be deployed and existing prototypes such as TeraRack [4] have already implemented it. To the best of our knowledge, OpTree is the first scheme to optimize the communication time for All-gather operation in optical interconnect systems. Our main contributions are summarized as follows:

- We propose an efficient scheme OpTree based on m-ary tree structure for All-gather operation in optical interconnect networks. OpTree derives the optimal m-ary tree corresponding to the optimal number of communication stages and achieves the minimum communication time among all options of the m-ary tree.
- We analyze and compare the communication steps of OpTree with existing All-gather algorithms. Theoretical results show that OpTree requires a much smaller number of communication steps than existing All-gather algorithms on optical interconnect systems.
- We evaluate OpTree with extensive simulations. Compared with three existing All-gather algorithms WRHT [7], Ring [8], and NE [9], OpTree significantly curtails communication time. Specifically, in a 1024-node optical interconnect system and across various message sizes, OpTree achieves average communication time reductions of 72.97%, 93.15%, and 86.32%, respectively. When tested with varying node counts, OpTree reduce communication time by 42.27%, 92.74%, and 85.49% on average in optical interconnect systems. Experimental results further reveal that OpTree's communication time can be progressively diminished with an increasing number of wavelengths.

The rest of this paper is organized as follows. Section II presents the related work. Section III describes the optical interconnect architecture, preliminary and motivation examples. Section IV illustrates the routing algorithm of OpTree, the analysis of communication steps, communication time, and constraints of optical communications. Section V evaluates OpTree under different scenarios. Section VI discusses this paper's limitations and extensions. Finally, Section VII concludes the article.

II. RELATED WORK

The All-gather operation within electrical interconnect systems, standardized by the Message Passing Interface (MPI), has been the focus of extensive research and engineering efforts. Numerous algorithms have been developed for optimizing All-gather operation, each tailored to specific operational contexts. The Ring algorithm, recognized for its simplicity and included in the MPI library [10], has seen many improved versions adapted for various topologies and networks. To minimize TCP/IP traffic, a NE algorithm was proposed, requiring only half the communication steps of the Ring algorithm [9]. Hierarchical algorithms, as explored in [11]–[13], the multi-leader approach [14], and multi-lane communication methods [15], address bandwidth limitations inherent in electrical links. Topology-aware collective algorithms, such as HierKNEM [16] and Rank Reordering [17], aim to reduce link traversals in both intra- and inter-node communications. Approaches focusing on symmetric multiprocessing (SMP) and multi-core clusters [18]–[21], along with holistic optimization for various topologies [22]–[24], have also been investigated. However, these methods may not be fully compatible with optical interconnect systems.

Collective communication in optical networks was first introduced in [25], analyzing the limits on the number of wavelengths and communication steps for broadcast and gossiping in one-stage and multi-stage models. Further research focused on specific topologies for one-stage collective communication, aiming to minimize the number of required wavelengths. Studied topologies include rings, tori, meshes, hypercubes, and trees [19], [26]–[29]. In [30], a one-stage broadcasting model for WDM networks considered the tap-and-continue characteristics of optical nodes. Subsequent studies on multi-stage models addressed various topologies, such as in [31], which explored uniform all-to-all routing in a symmetric directed ring to reduce wavelength usage. [32] focused on collective communication challenges in multi-stage WDM networks with diverse topologies, including lines, rings, and 2-D/3-D square tori. [6] presented a comprehensive multi-stage routing and wavelength assignment strategy to further decrease the number of wavelengths in optical networks. Recently, [7] proposed the WRHT algorithm for distributed DNN training in optical interconnect systems, organizing it into a hierarchical tree structure with shared wavelengths. Our study distinguishes itself by targeting the optimization of All-gather operation time in optical interconnect systems. To the best of our knowledge, this paper is the first to focus on optimizing the All-gather operation's time in optical interconnect systems, thereby addressing a notable gap in the existing literature.

III. Background and Motivation

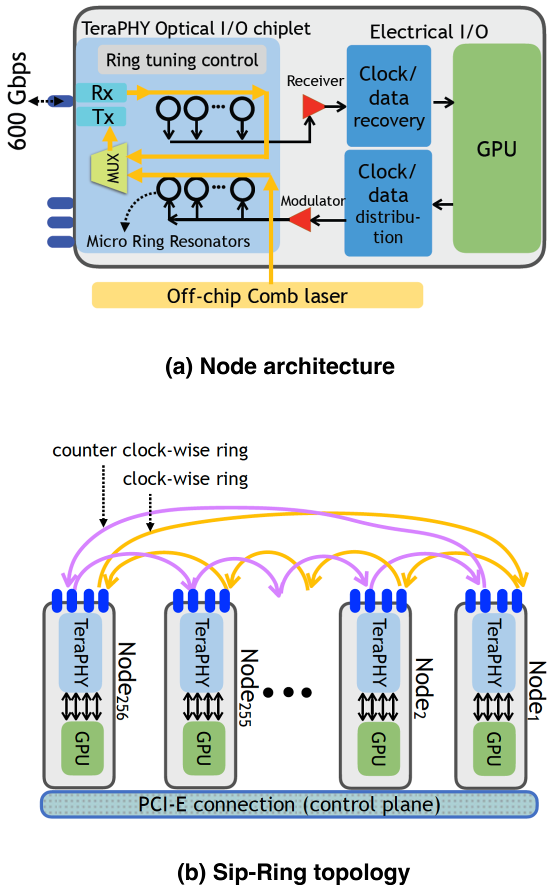
A. Optical Interconnect Architecture

We assume that the OpTree scheme employs the micro-ring resonator (MRR)-based optical switch known as TeraRack [33], though it is also compatible with other similar optical interconnect systems. As depicted in Fig. 1 (a), a TeraRack node is equipped with TeraPHY silicon photonics technology components. While various accelerators can be utilized in TeraRack nodes, we assume the use of homogeneous GPUs as the computing devices. Each optical interface on a TeraRack node features 64 micro-ring resonators, capable of selecting and forwarding any combination of 64 wavelengths. The light source in this system is a comb laser, specifically the SuperNova Light Supply [33]. At the transmitter side (Tx), this external comb laser generates light, which is then channeled into the node using a fiber coupler. The light undergoes rapid modulation at 40 Gbps per wavelength by an array of Micro-Ring Resonators (MRRs), encoding the data from the accelerator for transmission. Conversely, at the receiver side (Rx), a secondary array of MRRs filters out the wavelengths designated for the accelerator, while smoothly allowing the rest to continue through the system. Shifting focus to the overall topology, as depicted in Fig. 1 (b), the nodes are interconnected in a symmetrical ring layout. This configuration incorporates dual rings to facilitate both clockwise and counter-clockwise data transmission. The data plane primarily handles traffic across four single-mode fiber rings, while the control plane manages the Routing and Wavelength Assignment (RWA) configuration. A key feature of this setup is the dynamic allocation of wavelengths around these fiber rings, greatly simplifying the typically complex control-plane logic—a notable challenge in optical datacenter designs. The system parameters are detailed in Section V.

B. Preliminary

The role of the All-gather operation is to make every worker concurrently broadcast the data item until every worker receives the data item from each other. In an optical interconnect system, the All-gather operation corresponds to all-to-all broadcast routing using concurrent optical communications by multiple wavelengths. When the network size is large, the basic approach of one-stage model can lead to an unrealistically large number of wavelengths when realizing all-to-all routing, as can be seen in the following lemma [6].

Lemma 1. *In an N-node optical interconnect system, the minimum number of wavelengths needed for all-to-all routing under one-stage model is $\lfloor \frac{N^2}{4} \rfloor$ in line and $\lceil \frac{N^2}{8} \rceil$ in ring.*



(a) Node architecture

(b) Sip-Ring topology

FIGURE 1. **Optical interconnection architecture: (a) TeraRack node, (b) Double ring topology [33].**

Since the number of wavelengths in the optical interconnect system is limited, multi-stage (k-stage, $k \geq 2$) model can be used to reduce the number of wavelengths, which takes k communication stages to complete all-to-all optical communication [25]. For the k-stage model, the optical signals must be converted to electronic form $k - 1$ times before the All-gather operation is completed. However, most designs for multi-stage models only focus on reducing the number of wavelengths but fail to address the performance of All-gather operation. Therefore, we propose an efficient scheme, OpTree, for All-gather operation on optical interconnect systems based on m-ary tree [34] structure. OpTree can optimize the communication time of All-gather operation. In graph theory, an m-ary tree is a tree structure in which each internal node has no more than m children. A binary tree is a special case where $m = 2$.

Lemma 2. *For an m-ary tree with height k, the upper bound for the maximum number of leaves is m^k.*

C. Motivation

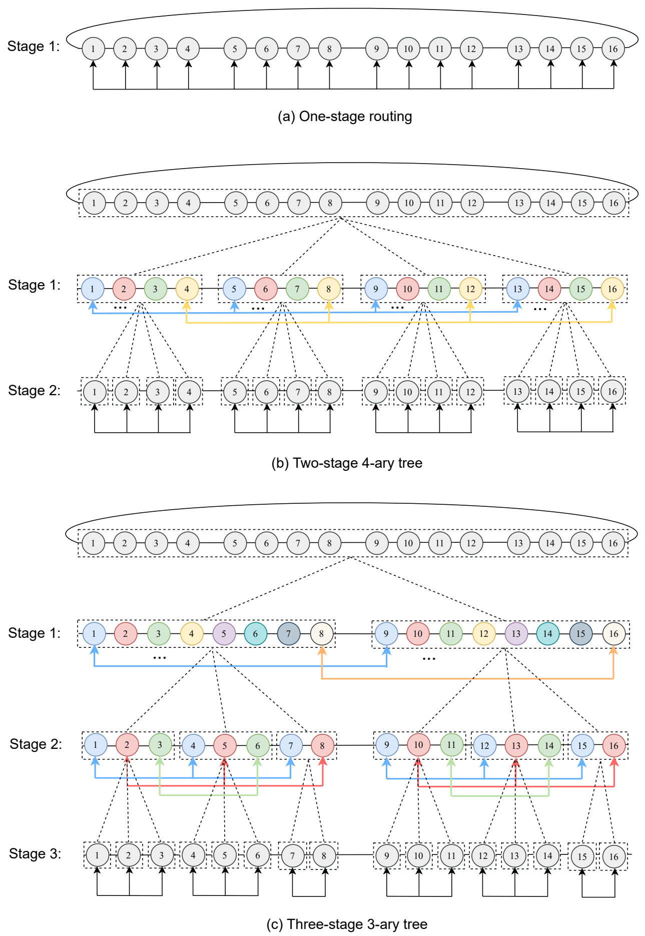
We use a motivation example to illustrate the trade-off between the communication time and the number of stages for our design. We assume that the system has 16 nodes based on the architecture of Fig. 1, the number of available wavelengths is 2, and every node has an amount of data d to collect from all the other nodes. We compare the communication time for the following three schemes: (a) One-stage routing, (b) Two-stage 4-ary tree, and (c) Three-stage 3-ary tree.

One-stage routing: As can be seen in Fig. 2 (a), One-stage routing conducts all-to-all broadcast from each node to all the other nodes without any intermediate relay nodes, which requires $\lceil \frac{16^2}{8} \rceil = 32$ wavelengths according to Lemma 1. Since the available number of wavelengths is two, it takes $\lceil \frac{32}{2} \rceil = 16$ communication steps (time slots) for finishing the All-gather operation, with each step sending amount of data d.

Two-stage 4-ary tree: As shown in Fig. 2 (b), the whole group of 16 network nodes is taken as a super root node of a full 4-ary tree, which is partitioned into 4 children (subgroup) in stage 1. Each child tree node contains 4 network nodes. During stage 1, the network nodes on the ith position in each subgroup perform all-to-all routing using the one-stage model along the ring, with each node sending a data amount of d. Thus, nodes 1, 5, 9, and 13 (marked in blue) route as a subset, and similarly for nodes 2, 6, 10, and 14 (marked in red), and so on. As these four subsets share the optical links in the ring, the wavelength requirement for this stage is $4 \cdot \lceil \frac{4^2}{8} \rceil = 8$, requiring 4 communication steps (time slots) to complete. In stage 2, each subgroup is further partitioned into individual nodes (leaf of the 4-ary tree). Here, subgroup nodes conduct all-to-all routing using the one-stage model on the segment of the ring. The data amount sent by each node is now $4d$, reflecting the d from themselves plus the $3d$ data accumulated from others in their subgroup from stage 1. So, network nodes 1, 2, 3, and 4 perform all-to-all routing as a subset, and 5, 6, 7 and 8 perform all-to-all routing as a subset, and so on. To achieve load balancing in each stage, each wavelength is loaded with a data item of size d. Since these subsets do not share the optical links in this stage, the wavelength requirement is $4 \cdot \lfloor \frac{4^2}{4} \rfloor = 16$, leading to 8 communication steps (time slots) to finish. Therefore, the total number of communication steps required by the two-stage 4-ary tree is $\lceil \frac{8}{2} \rceil + \lceil \frac{16}{2} \rceil = 12$.

Three-stage 3-ary tree: Similarly, as shown in Fig. 2 (c), three-stage 3-ary tree scheme allows each node to contain at most three children with the depth of $\lfloor log_3 16 \rfloor = 3$ (i.e., three stages) for 16 nodes. Accordingly, we can calculate the wavelength requirements for the stages 1, 2 and 3 as $8 \cdot \lceil \frac{2^2}{8} \rceil = 8$, $2 \cdot 3 \cdot \lfloor \frac{3^2}{4} \rfloor = 12$, and $6 \cdot \lfloor \frac{3^2}{4} \rfloor = 12$ respectively. Therefore, the total number of communication steps to finish the All-gather operation is $\lceil \frac{8}{2} \rceil + \lceil \frac{12}{2} \rceil + \lceil \frac{12}{2} \rceil = 16$.

It can be seen that the One-stage model only uses one stage but requires more time slots, and using the multiple-stage m-ary tree method uses more stages and needs less wavelength. From the results of the two-stage 4-ary tree and three-stage 3-ary tree scheme, 4-ary tree scheme has the least number of communication steps among all these three schemes. Therefore, there is a tradeoff between communication steps and the number of stages in m-ary tree to conduct All-gather operation on optical interconnect systems. The challenging problem is: *For a given number of nodes and available wavelengths in the optical interconnect system, how to find the optimal m-ary tree with the optimal number of stages that minimizes communication steps and communication time among all the m-ary tree options.*



(a) One-stage routing

(b) Two-stage 4-ary tree

(c) Three-stage 3-ary tree

FIGURE 2. **Motivation examples: (a) One-stage, (b) Two-Stage 4-ary tree, and (c) Three-stage 3-ary tree.**

IV. The OpTree Scheme

A. Design of OpTree

In response to the aforementioned challenges, we design an efficient scheme OpTree, represented by m-ary tree structure with the number of All-gather stages represented by the tree depth of k. We first present the routing algorithm and then analyze the communication steps and communication time by OpTree.

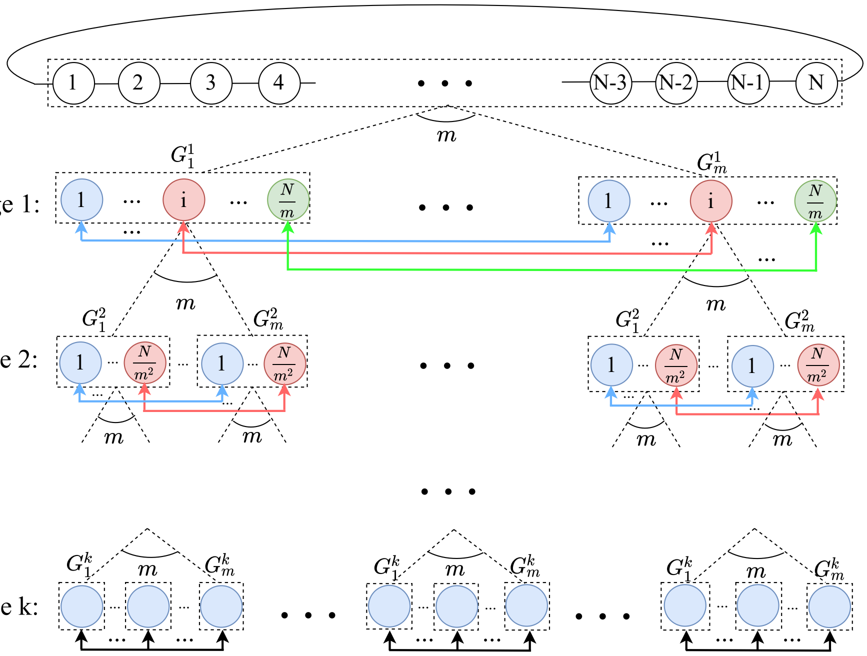


FIGURE 3. **OpTree Algorithm Process.**

1) Routing of OpTree

Assume that the number of nodes in the optical interconnect system is N, and the number of available wavelengths is w. Therefore, the ith level of the m-ary tree corresponds to the ith stage of All-gather operation. As illustrated in Fig. 3, the k-level m-ary tree is constructed by recursively partitioning the group of network nodes into m sub-groups as child tree nodes. According to Lemma 2, each parent tree node has at most $m = N^{\frac{1}{k}}$ child tree nodes, with each child tree node containing $N^{\frac{k-1}{k}}$ network nodes at the ith level of m-ary tree. The depth of the m-ary tree is $k = \lfloor log_m N \rfloor$, which is also the number of stages for All-gather operation. The detailed working principle of OpTree is described below.

Stage 1**:** Initially, the whole group of N network nodes is regarded as the root tree node, which is divided into m child tree nodes denoted as $G_1^1, ..., G_m^1$ with each child node containing $\lceil \frac{N}{m} \rceil$ network nodes. During stage 1, the network node on the ith ($i \in [1, \lceil \frac{N}{m} \rceil]$) position in each of these m sibling nodes can form a subset to perform all-to-all routing by one-stage model. For example, the blue node (node 1) in each of $G_1^1, ..., G_m^1$ broadcasts its data item with size d to all the other blue nodes by one-stage model along the ring.

Stage j**:** Recursively, each node in stage $j-1$ containing $\lceil \frac{N}{m^{j-1}} \rceil$ network nodes is further partitioned into m child nodes denoted as $G_1^j, ..., G_m^j$ with each child node having $\lceil \frac{N}{m^j} \rceil$ network nodes, until each child node only contains one network node at the last stage k. During stage j, the network node in the ith ($i \in [1, \lceil \frac{N}{m^j} \rceil]$) position in each of its m sibling nodes can form a subset to perform all-to-all routing by one-stage model. For example, during stage 2, the blue node (node 1) in each of $G_1^2, ..., G_m^2$ broadcasts its data item to all the other blue nodes among sibling nodes. During stage k, each leaf node contains only one network node, so $G_1^k, ..., G_m^k$ exchange data with each other among sibling nodes by one-stage model. In stage j, the number of network nodes in each tree node is $\lceil \frac{N}{m^j} \rceil$, and each network node has $m^{j-1}d$ data to send by m^{j-1} wavelengths for load balance.

2) Analysis of Communication Steps

In order to achieve load balancing, OpTree algorithm specifies that each wavelength carries the same amount of data d during each communication step. Therefore, the communication time of All-gather operation by OpTree is mainly determined by the total number of communication steps, which we calculate as follows.

Theorem 1. *In an N-node optical ring interconnect system with w available wavelengths, the total number of communication steps needed to perform All-gather operation during k stages ($k \geq 2$) by OpTree is $\lceil \frac{(2k-1)N^{1+\frac{1}{k}}}{8w} \rceil$.*

Proof:
In stage 1, since there are $\lceil \frac{N}{m} \rceil$ subsets sharing the optical links, the number of communication steps is $\lceil \frac{m^2}{8w} \rceil \times \lceil \frac{N}{m} \rceil$. For each of the subsequent $k-1$ stages, the required number of communication steps in stage j ($j \in [2, k]$) is $m^{j-1} \times \lceil \frac{N}{m^j} \rceil \times \lfloor \frac{m^2}{4w} \rfloor = \lfloor \frac{m^2}{4w} \rfloor \times \lceil \frac{N}{m} \rceil$. Let S represent the total number of communication steps for all k stages of OpTree, which can be calculated as follows.

$$S = \lceil \frac{(2k-1)N^{1+\frac{1}{k}}}{8w} \rceil. \qquad (1)$$

Therefore, the theorem holds. ■

By Theorem 1, the optimal m-ary tree corresponding to the optimal number of communication stages can be derived as follows, which can achieve the minimum communication time among all the OpTree options.

Lemma 3. *In an N-node optical ring interconnect system with w available wavelengths, the optimal number of communication steps by OpTree for finishing the All-gather operation is obtained at* $k^* = \left[\frac{lnN+\sqrt{lnN(lnN-2)}}{2}\right]$.

Proof:
Assuming that S is a continuous function and k is a variable in (1), the number of communication steps is minimized when $\frac{\partial S}{\partial k} = 0$. It can be derived that

$$\frac{\partial S}{\partial k} = \frac{N^{1+\frac{1}{k}}}{4w} - \frac{N^{1+\frac{1}{k}} lnN(2k-1)}{8wk^2}.$$

Let $\frac{\partial S}{\partial k} = 0$. S is minimised when

$$k^* = \left[\frac{lnN + \sqrt{lnN(lnN-2)}}{2}\right], \quad (2)$$

where [*] represents integer rounding operation. Hence, this lemma holds. ■

By Lemma 3, the corresponding minimum communication time among all the OpTree options can be derived by replacing k with k^* in (1) as follows.

Theorem 2. *In an N-node optical ring interconnect system with w available wavelengths, the optimal number of communication steps by OpTree is* $\lceil\frac{(2k^*-1)N^{1+\frac{1}{k^*}}}{8w}\rceil$.

Table 1 summarizes the comparisons on the numbers of communication steps among different algorithms for All-gather operation. In traditional Ring All-gather, as described by [9], the All-gather operation is conducted in $N-1$ steps, with each step transmitting an amount of data d. Contrastingly, the neighbor exchange (NE) All-gather method, also detailed in [9], requires $\frac{N}{2}$ steps for completion. The transmission data for each step is $2d$, except for the first. The WRHT algorithm, initially designed for All-reduce operation [7], has been extended to All-gather. This extension retains the original routing scheme of WRHT, but omits the reduction operation for each step, enabling all nodes' data to be gathered without reduction. It takes $1+\left\lceil\frac{\bar{m}(\bar{m}^{\theta-1}-1)}{\bar{m}-1}\right\rceil$ steps to collect all the data into one or a few nodes, and either $\left\lceil(\theta-1)\bar{m}^{\theta-1}\right\rceil$ or $\left\lceil\theta\bar{m}^{\theta-1}\right\rceil$ steps to broadcast the data to any other nodes, where $\theta = \lceil\log_{\bar{m}} N\rceil$ and $\bar{m} = 2w+1$. The total number of communication steps for the WRHT algorithm is either $1+\left\lceil\frac{\bar{m}(\bar{m}^{\theta-1}-1)}{\bar{m}-1}\right\rceil + (\theta-1)\bar{m}^{\theta-1}$ or $1+\left\lceil\frac{\bar{m}(\bar{m}^{\theta-1}-1)}{\bar{m}-1}\right\rceil + \theta\bar{m}^{\theta-1}$. For OSM method, the All-gather operation necessitates $\lceil\frac{N^2}{8w}\rceil$ communication steps, each sending amount of data d.

TABLE 1. **Communication step comparison of different All-gather algorithms in optical interconnect systems.**

Algorithm	Communication steps	Number of steps
		N = 1024, w = 64
Ring	$N-1$	1023
NE	$\frac{N}{2}$	512
WRHT	$1+\left\lceil\frac{(\bar{m}^{\theta}-\bar{m})}{\bar{m}-1}\right\rceil+(\theta-1)\bar{m}^{\theta-1}$	259
OSM	$\lceil\frac{N^2}{8w}\rceil$	128
OpTree	$\lceil\frac{(2k^*-1)N^{1+\frac{1}{k^*}}}{8w}\rceil$	70 (k^* = 7)

3) Communication Time of OpTree

Since we obtain the optimal number of communication steps, we can further derive the total communication time to finish the All-gather operation by OpTree, denoted as T_{comm}:

$$T_{comm} = (\frac{d}{B} + a)S, \quad (3)$$

where a is the O/E/O conversion delay and the reconfiguration delay of the MRRs, B is the bandwidth per wavelength, d is the amount of transferred data to be received initially for each node, and S represents the total number of communication steps. As d, B, and a are all constant values, the optimal communication time using OpTree can be achieved when the number of communication steps is minimized, as indicated by Theorem 3, as shown below.

Theorem 3. *In an N-node optical ring interconnect system with w available wavelengths, the optimal communication time by OpTree for All-gather operation is* $(\frac{d}{B} + a)\lceil\frac{(2k^*-1)N^{1+\frac{1}{k^*}}}{8w}\rceil$.

Proof:
It can be seen from (3) that the parameters of d, B, and a are all constant values. The optimal communication time using OpTree can be achieved when the number of communication steps for All-gather operation S is minimized. According to Theorem 2, the optimal communication time is $\frac{d\lceil\frac{(2k^*-1)N^{\frac{k^*+1}{k^*}}}{8w}\rceil}{B} + a\lceil\frac{(2k^*-1)N^{\frac{k^*+1}{k^*}}}{8w}\rceil$ for All-gather operation in an N-node optical ring interconnect system with w available wavelengths. ■

Algorithm 1 displays pseudocode that uses number of nodes N, available number of wavelengths w, initial amount of transferred data d, and bandwidth per wavelength b as input parameters. It calculates the communication time T_{comm} by employing OpTree algorithm. Lines (8) and (11) conduct all-to-all routing by grouping nodes in each stage. A significant aspect of the algorithm's efficiency is its overall time complexity, which is $O(logN)$. This is attributed to the dominating influence of the loop within Algorithm 1, iterating in a logarithmic manner relative to N. Furthermore, scalar variables such as k, S, w, and j are utilized, each occupying a constant space, thus contributing to the algorithm's space efficiency.

Algorithm 1: OpTree Algorithm for Communication Time

```
Input: $N$, $w$, $d$, $b$
Output: $T_{\text{comm}}$
Calculate $k^* = \left\lceil \frac{lnN+\sqrt{lnN(lnN-2)}}{2} \right\rceil$, where
  $m = N^{\frac{1}{k^*}}$
$S = 0$ // Total number of
    communication steps
// Stage 1
Divide $N$ network nodes into $m$ child tree nodes,
  each containing $\lceil \frac{N}{m} \rceil$ network nodes
$n_{\lambda_r} = \lceil \frac{m^2}{8} \rceil \times \lceil \frac{N}{m} \rceil$ // Number of
    required wavelengths
Network nodes on the $i_{th}$ position perform
  all-to-all routing
$S = S + \lceil \frac{n_{\lambda_r}}{w} \rceil$
// Subsequent Stages
for $j = 2$ to $k$ do
    Divide each node into $m$ child nodes until each
      child node contains one network node
    $n_{\lambda_r} = \lceil \frac{m^2}{8} \rceil \times \lceil \frac{N}{m} \rceil$
    $S = S + \lfloor \frac{n_{\lambda_r}}{4 \cdot w} \rfloor \times \lceil \frac{N}{m} \rceil$
    Network nodes on the $i_{th}$ position perform
      all-to-all routing
end
Compute $T_{\text{comm}} = \left(\frac{d}{B} + a\right) S$
return $T_{comm}$
```

B. Constraints of Optical Communications

Insertion loss and crosstalk occur when optical signals pass through optical interfaces, so discussing the constraints they impose on the OpTree scheme is essential. We first discuss the constraint of signal loss and then the constraint of crosstalk for reliable communications.

1) Signal Loss

Note that optical signal loss is related to signal path length, and the maximum signal path length in the OpTree scheme occurs during the first stage. As shown in Fig. 3, the maximum signal path length is the distance between node 1 in G_1^1 and node 1 in G_m^1. The maximum signal path length L_{max} for the OpTree scheme can be calculated as follows:

$$L_{max} = 1 + ((N^{\frac{1}{k^*}} - 1)N^{1-\frac{1}{k^*}}). \tag{4}$$

Given the maximum signal path length L_{max} of OpTree, we can derive the total optical loss L_l using the following equation:

$$L_l = P_m + L_{max}P_{pass}, \tag{5}$$

where P_m represents the modulator loss in the Tx, and P_{pass} denotes the signal loss when passing an optical interface on the node. According to [35], the OpTree scheme must satisfy the following optical power constraint:

$$P_{laser} \geq L_l + P_p, \tag{6}$$

where P_{laser} is the laser source power, L_l is total optical insertion loss, and P_p refers to the power penalty caused by the extinction ratio. According to (4), (5), and (6), we can estimate the maximum signal path length L_{max} for the OpTree scheme that satisfies the constraint of insertion loss for the optical interconnect system.

2) Crosstalk

The impact of crosstalk noise in the optical interconnect system can be quantified by the signal-to-noise ratio (SNR) [36]. SNR is mathematically defined as the ratio of signal power to the sum of noise power, including crosstalk noise and other noise sources. This relationship is expressed as:

$$\text{SNR} = 10 \log \frac{P_S}{P_N + P_O}, \tag{7}$$

where P_S denotes the optical signal power, P_N represents the crosstalk noise power received at the photo-detector in the receiver, and P_O refers to the power of other noise sources present in the system.

The worst-case crosstalk noise power, P_{N_w}, is primarily dominated by the worst-case crosstalk noise power on the Tx side (P_{Tx}) and the worst-case crosstalk noise power on the Rx side (P_{Rx}) during optical communication. It can be estimated as follows:

$$P_{N_w} = L_{max}P_{Rx} + P_{Tx}. \tag{8}$$

The bit-error-rate (BER) represents the percentage of bits with errors among the total number of bits received during a transmission. It is utilized as a criterion to evaluate the quality of optical communication [37]. The relationship between BER and SNR_w in the optical interconnect system is defined as:

$$\text{BER} = \frac{1}{2} e^{-\frac{\text{SNR}_\text{w}}{4}}. \tag{9}$$

To achieve reliable transmissions in an optical interconnect system, the BER must be lower than 10^{-9} [37]. By using (7), (8), and (9), we can estimate the optical laser power P_S required to achieve reliable optical communications. This estimation represents the crosstalk and laser power constraint that the system needs to satisfy.

V. Evaluation

A. Simulation setup

In our evaluation, the same optical interconnect simulator used in [7] is employed to assess the performance of OpTree. Constructed in Python, this simulator relies on mathematical modeling to replicate the optical communications within optical interconnect systems. While the simulator has not been directly validated against real-world equipment, its design and functionality have been rigorously grounded in mathematical modeling. Cross-validation was performed using analytical and optical communication models to ensure accuracy and consistency. Although originally designed for All-reduce algorithms, enhancements were made to the simulator to incorporate the proposed OpTree and three All-gather algorithms. The simulation parameters for the optical interconnect system are described below.

In the simulation, the bidirectional ring topology of the optical interconnect system is the same as Tera-Rack [33]. The default number of wavelengths is 64, and the bandwidth per wavelength is 40 Gbps. For data transmission, the size of the optical transmission packet is set to 128 bytes. Here, the flit size—the smallest unit of data transfer in the optical network—is defined as 32 bytes. The data type for All-gather operation is assumed to be float32, and each node in the optical interconnect system has one GPU. During the transmission, the reconfiguration delay of MRRs is 25 μs [4], and the O/E/O conversion latency is 1 clock cycle per flit, where a clock cycle refers to one cycle of the CPU clock [38]. We assume that the optical power can support the maximum communication length of OpTree in the simulation.

To evaluate the performance of OpTree, we conduct four sets of experiments. In the first set of experiments, we verify that OpTree can find an optimal m-ary tree for All-gather operation with minimum communication time in different numbers of nodes. In the second set of experiments, we compare the performance of OpTree with WRHT, Ring, and NE, on the simulated optical ring interconnect system using different message sizes. In the third set of experiments, we compare the performance of OpTree with three All-gather algorithms on the simulated optical ring interconnect system by different numbers of nodes. In the fourth set of experiments, we test the impact of different numbers of wavelengths on these All-gather algorithms in the simulated optical interconnect system.

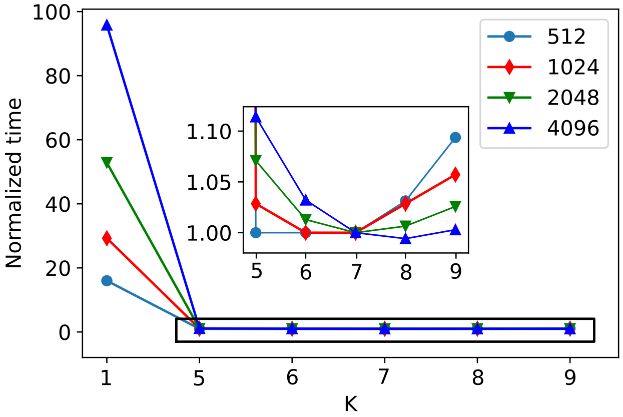


FIGURE 4. Performance evaluation of OpTree across various depths (k) with node counts of 512, 1024, 2048, and 4096.

B. Optimal k-stage m-ary Tree

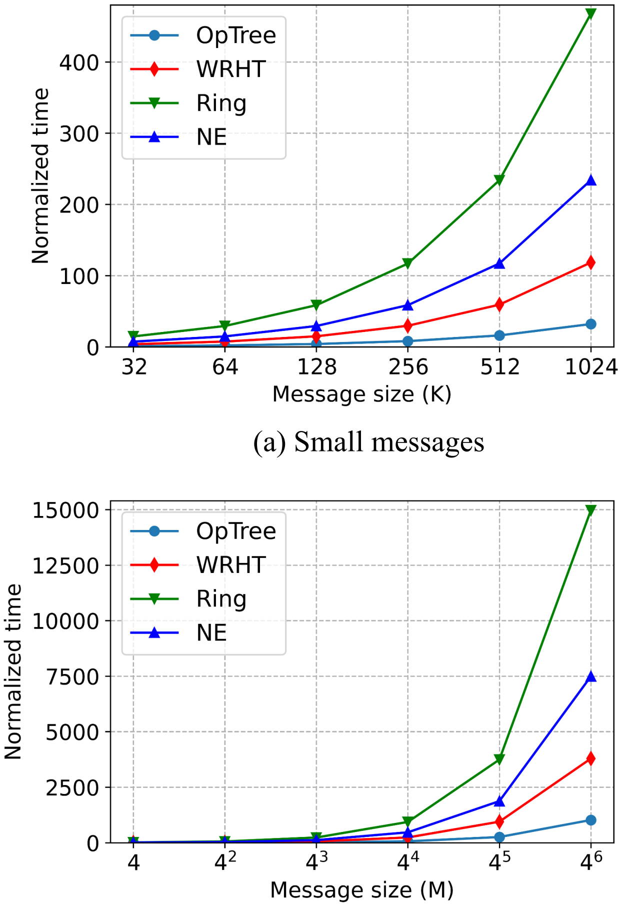
To verify that our proposed OpTree can find a m-ary tree with minimum communication time, we first compare the performance of different m-ary trees with different nodes for All-gather operation in the optical ring interconnect system using the same message size and the same number of wavelengths. In OpTree, the depth of the m-ary tree with N nodes is k, where $k = log_m N$. Therefore, we can use k to denote the different m-ary trees for different numbers of nodes. In the simulation, we vary the number of nodes from 512 to 4096. The number of wavelengths is set as 64, and the message size is set to 4 megabyte (MB). All results presented in Fig. 4 are normalized for comparison purposes. This is done by dividing each result by the initial result obtained from the OpTree method.

In Fig. 4, four distinct line charts offer a comprehensive performance comparison of m-ary trees across varying depths (denoted by k) for the All-gather operation in scenarios involving 512, 1024, 2048, and 4096 nodes. The designation $k = 1$ implies that we are utilizing the OSM for a ring for the All-gather operation. A closer look is facilitated by an inset within Fig. 4, which magnifies specific segments of the line charts, elucidating differences with respect to node numbers. From the presented results, we can discern a nuanced trade-off between the depth of the m-ary trees and the node count in the optical interconnect system. As we delve into the specifics, for depths ranging from 5 to 9 in the m-ary tree, performance under 512 nodes remains static initially, only to spike substantially later on. When considering 1024 nodes, there's an initial dip in performance, which then plateaus, and finally experiences a swift uptick. Performance patterns for 2048 nodes reveal a continual decline until reaching its nadir at a depth of 7, after which there is a resurgence. Interestingly, the behavior observed for 4096 nodes mirrors that of 2048 nodes, with a pronounced initial drop, bottoming out at depth 8, and a subsequent modest rise. The inferences drawn from the simulation data suggest optimal performance levels tied to specific depths for each node configuration: depths 6, 6, 7, and 8 are optimal for 512, 1024, 2048, and 4096 nodes, respectively, in the optical interconnect system. Notably, these empirical findings align seamlessly with our theoretical derivations presented in Lemma 3 and Theorem 3. This concordance underscores the robustness and validity of our proposed OpTree, reinforcing its applicability and precision in the context of real-world optical interconnect systems.

C. Performance on Different Message Sizes

In this set of simulations, we compare the performance of OpTree with existing schemes including WRHT [7], Ring [9], and NE [9] in the 1024-node optical ring interconnect system using small and large messages, respectively. We set the range of small messages from 32 kilobytes (KB) to 1024KB and the range of large messages from 4MB to 4^6MB. The number of wavelengths is set as 64. All results in Fig. 5 are normalized by dividing each by the initial result obtained from the OpTree method.

In Fig. 5, we present a detailed performance comparison that highlights the efficacy of the OpTree, WRHT, Ring, and NE algorithms for the All-gather operation across both small and large message sizes. Diving into the specifics presented in Fig. 5 (a) and (b), the overarching trend is clear: OpTree consistently achieves the quickest total time, regardless of whether we are dealing with petite or expansive message sizes. This consistent efficiency is attributed to OpTree's capacity to determine the optimal number of communication steps and the most efficient communication time, as supported by Theorem 2 and 3 at the given scale. When ranking by efficiency, WRHT emerges as the second most competent across both message size categories. However, its performance does not match up to OpTree. The underlying reason for this shortfall lies in WRHT's stage-wise load imbalance. As the stages multiply, nodes designated to accumulate data within each stage bear an increasingly hefty data load. This surge in data load naturally amplifies the communication steps required, culminating in a longer total time. Conversely, both the Ring and NE algorithms seem to grapple with inefficiencies within the optical interconnect system, especially when juxtaposed with the more adept OpTree and WRHT. Among these, the Ring algorithm's performance is particularly lackluster. This subpar outcome is largely a byproduct of its inability to efficiently reuse wavelengths during message transmissions. When distilled into numbers, the OpTree algorithm's prowess becomes evident. In a 1024-node optical ring interconnect system, OpTree outperforms its counterparts significantly: achieving communication time reductions of 72.97%, 93.15%, and 86.32% when compared to WRHT, Ring, and NE, respectively.



(a) Small messages

(b) Large messages

FIGURE 5. Performance comparison of different All-gather algorithms in 1024-node optical interconnect system.

D. Performance on Different Numbers of Nodes

In this set of simulations, we compare OpTree with WRHT, Ring, and NE algorithms for All-gather operation in the optical interconnect system by setting 512, 1024, 2048, and 4096 nodes, respectively. The message sizes are set from 4MB to 128MB. All results in Fig. 6 are normalized by dividing each by the initial result obtained from the OpTree method.

Fig. 6 presents a comparative analysis of the performance metrics for the OpTree algorithm against the WRHT, Ring, and NE algorithms, especially focusing on the All-gather operation across varying node counts. The sub-figures Fig. 6 (a) through (d) are particularly illuminating. For a majority of scenarios, OpTree consistently outperforms the competition, boasting the least amount of total time. However, an exception arises in the 4096 nodes scenario, but only when the message size exceeds 16MB. Ring's performance seems to be less optimal across the board, with its total time always surpassing its counterparts. NE's efficiency lies somewhere in between, though it distinctly bests Ring in all cases. WRHT's performance trajectory is especially noteworthy. Under a 512-node environment, WRHT and NE have almost parallel total time metrics. As node counts climb from 1024 to 2048, WRHT's efficiency rises, pulling its performance closer to that of OpTree. Intriguingly, once we reach 4096 nodes, WRHT exhibits a marginal edge over OpTree, especially as message sizes expand, albeit with only a slight margin of difference. Delving deeper into the underlying dynamics, several factors come to light. A principal determinant of an algorithm's total time to execute the All-gather operation is its number of communication steps. The rationale behind this is simple: Each wavelength shoulders the responsibility of transmitting a pre-defined data quantum, denoted as d. Hence, the more the communication steps required to culminate the All-gather operation, the greater the aggregate data transfer and the resultant optical device configuration delay. Naturally, algorithms necessitating more communication steps will demand more time. This observation resonates strongly with the data presented in Table 1. The Ring algorithm, for instance, requires $N-1$ communication steps, whereas NE needs only half of that, i.e., $N/2$. This clearly justifies Ring's elongated total time relative to NE. Delving into the nuances of OpTree and WRHT, we discern that OpTree's communication steps follow a power function trajectory with N, while WRHT adopts an exponential function pattern centered around $log_m N$. Given this distinction, as N propels forward, OpTree's communication steps accelerate at a steeper gradient than WRHT. Consequently, in a high-node environment, OpTree's total time surpasses that of WRHT. Compared to WRHT, Ring, and NE, OpTree can reduce communication time by 42.27%, 92.74%, and 85.49% on average in the optical interconnect system under different numbers of nodes. To offer a more statistical perspective on these dynamics, we've tabulated the Average Performance Difference (APD) along with the Standard Deviation (SD) contrasting OpTree's performance against WRHT, Ring, and NE, all stratified by varying node counts, in Table 2.

E. Performance on Different Numbers of Wavelengths

In this section, we test the impact of using different numbers of wavelengths for these four algorithms in the 1024-node optical ring interconnect system. We use the exact message sizes in Section D, and we set the numbers of wavelengths as 4, 16, 64, and 256, respectively. All results in Fig. 7 are normalized by dividing each by the initial result obtained from the OpTree method.

Fig. 7 unfolds a comprehensive comparative analysis, showcasing the performance dynamics of the OpTree, WRHT, Ring, and NE algorithms when employed for the All-gather operation across varying numbers of wavelengths. Observations from Fig. 7 (a) indicate an interesting trend. With just 4 wavelengths in play, NE emerges as the frontrunner, clocking the minimum total time. Ring follows closely, slotting into the second position. Conversely, WRHT lags behind, registering the highest total time, with OpTree trailing just behind it. Shifting our focus to Fig. 7 (b), when the wavelength count jumps to 16, the tables turn. WRHT takes the lead, recording the least total time. OpTree secures the second position, while NE slides to the third. Ring, unfortunately, finds itself at the bottom, taking the most time. For Fig. 7 (c) and (d), set at 64 and 128 wavelengths respectively, OpTree consistently showcases superior efficiency, boasting the minimum total time across increasing message sizes. In stark contrast, Ring lags, clocking the maximum total time. A deeper examination reveals intriguing performance shifts as the wavelength count escalates from 64 to 256. WRHT demonstrates an evident downward trajectory in its total time. To be more specific, when we have 64 wavelengths, WRHT's total time slightly surpasses OpTree's but remains below NE's. On raising the wavelength count to 256, WRHT fares better than Ring, but its performance is just marginally inferior to NE. Delving into the core mechanics, the fluctuating performance trends of these algorithms can be attributed to their innate operational nuances. OpTree's communication steps, for instance, have an inverse relationship with the number of wavelengths. As wavelengths surge, OpTree's total time dwindles. WRHT adopts a unique data collection approach, clustering data in tandem with wavelength counts. Keeping node counts constant, WRHT's communication steps follow a composite power function. Intricacies within this function suggest an optimal balancing point for grouping nodes, identified as m, which hovers around a sweet spot near $w = 16$ ($m = 33$), as inferred from Fig. 7. Ring and NE, however, don't benefit as much from increasing wavelengths. Primarily, this is because they aren't inherently designed for optical interconnect systems, leaving them unable to capitalize on additional wavelengths for efficiency gains. This limitation manifests starkly in Fig. 7, panels (b) through (d), where Ring consistently underperforms, and NE follows suit, albeit to a slightly lesser degree. To provide a quantitative grasp on these intricate dynamics, we've computed and encapsulated the Average Performance Difference (APD) and the Standard Deviation (SD) for OpTree against WRHT, Ring, and NE, stratified by the diverse wavelength counts, all detailed in Table 3.

VI. Discussions

A. Limitations

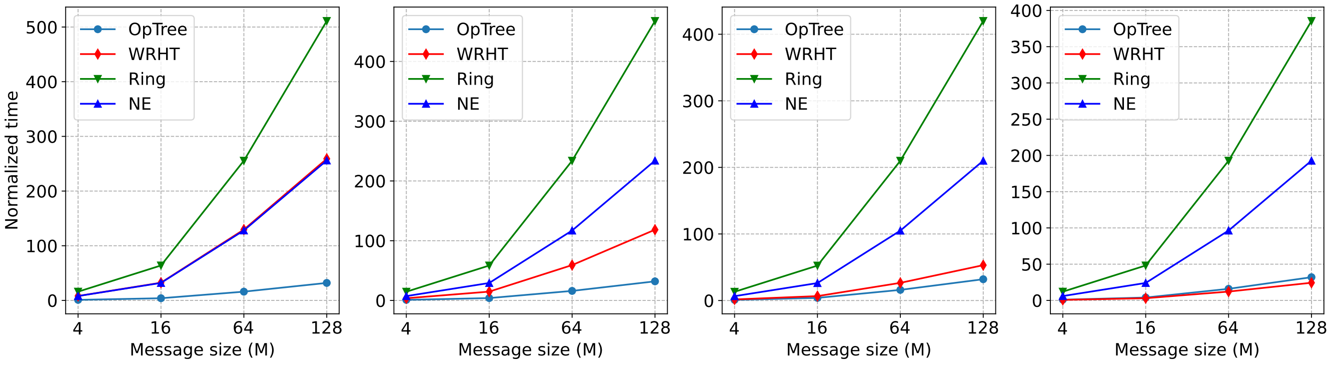
This study aims to develop an efficient routing algorithm for All-gather operation in optical interconnect systems. While our research provides groundbreaking insights, acknowledging its limitations is crucial for interpreting the results accurately and identifying areas for future exploration.

The primary limitation lies in our reliance on a mathematical modeling-based optical interconnect simulator. This approach offers a reliable foundation for preliminary analysis but may not encompass all complexities of real-world implementations. Specifically, the simulator may overlook certain dynamic variables and real-time interactions inherent in physical optical interconnect systems. A more comprehensive analysis would benefit from using hardware-specific execution models, which could provide a nuanced understanding of the algorithm's performance in practical settings. Although such an in-depth exploration is outside our current scope, it presents a valuable opportunity for future research.

Additionally, our focus has predominantly been on algorithm development rather than proposing enhancements to the optical interconnect architecture itself. Our algorithm is highly compatible with TeraRack-like architectures, leveraging the FlowRing algorithm for efficient routing and wavelength assignments. This compatibility highlights the practical applicability of our work. However, adapting our algorithm to other architectures, such as the conventional optical fat-tree architecture prevalent in data centers, would require modifications, particularly in switch configurations, to optimize its integration.

B. Extensions

The adaptability of the OpTree algorithm, originally conceptualized for ring topologies in optical interconnect systems,

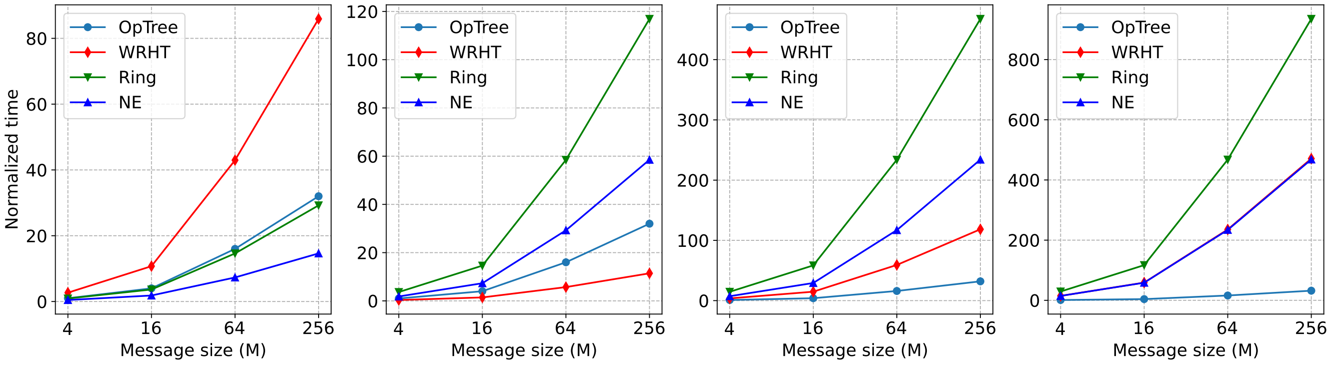


(a) 512 nodes (b) 1024 nodes (c) 2048 nodes (d) 4096 nodes

FIGURE 6. Performance comparison of different All-gather algorithms using different node counts in optical interconnect system.

TABLE 2. Performance difference of the OpTree over other All-gather algorithms using different node counts.

	512	1024	2048	4096	Average	SD
WRHT	87.64%	72.97%	39.76%	-31.27%	42.27%	45.87%
Ring	93.73%	93.15%	92.37%	91.69%	92.74%	0.77%
NE	87.5%	86.32%	84.76%	83.39%	85.49%	1.55%



(a) 4 wavelengths (a) 16 wavelengths (a) 64 wavelengths (a) 128 wavelengths

FIGURE 7. Performance comparison of different All-gather algorithms using different wavelength counts in 1024-node optical interconnect system.

extends to other network configurations, including mesh and torus topologies. This section explores the potential adaptation of the OpTree algorithm to these alternative structures, illustrating its versatility and wide applicability. Consider, for example, the adaptation of OpTree to an $n \times n$ torus topology. The adaptation process commences with the application of the OpTree algorithm to each row of nodes. This initial phase ensures that each node within a row acquires all the necessary data from its peer nodes in the same row. Subsequently, the algorithm is applied across each column, facilitating a comprehensive data exchange amongst nodes in different rows. This methodical, sequential application guarantees that, by the end of the process, every node in the network has obtained all the data from every other node, thereby achieving complete data dissemination across the torus topology. The underlying principles of the OpTree algorithm allow for similar adaptations to other topologies, such as mesh or 3D-torus networks, albeit with some modifications to accommodate the unique characteristics of each topology. However, it is crucial to acknowledge that optimizing the performance of the OpTree algorithm for these diverse topologies necessitates additional research. This includes a thorough investigation into the specific requirements and potential challenges posed by each topology, thereby ensuring the algorithm's efficiency and effectiveness in varied network environments.

In conclusion, while our study has pioneered new approaches in optical interconnect systems, it also opens a broad avenue for future research. The potential extensions of our work, as outlined here, provide a roadmap for future explorations. By building on our findings, subsequent studies can further refine the OpTree algorithm, enhancing its

TABLE 3. Performance difference of the OpTree over other All-gather algorithms under different wavelength counts.

	4	16	64	256	Average	SD
WRHT	62.75%	-180%	72.97%	93.2%	12.23%	111.52%
Ring	-9.48%	72.62%	93.15%	96.57%	63.22%	42.96%
NE	-118.75%	45.31%	86.32%	93.16%	26.51%	85.84%

applicability across a broader range of network topologies and contributing to the advancement of optical interconnect technologies.

VII. CONCLUSION

In this paper, we have introduced OpTree, an efficient scheme for the All-gather operation in optical interconnect systems. Based on an m-ary tree structure, we derived the optimal tree that minimizes communication time among all possible OpTree configurations. When comparing the communication steps of OpTree against existing All-gather algorithms, our theoretical analysis shows that OpTree requires significantly fewer steps in a 1024-node optical interconnect system. Additionally, we have analyzed the constraints imposed by the optical communication in OpTree. In a 1024-node optical interconnect system, simulation results demonstrate that OpTree reduces communication time by 72.97%, 93.15%, and 86.32% on average compared with three other All-gather algorithms. Across different node configurations, OpTree also shows average improvements of 42.27%, 92.74%, and 85.49%. Furthermore, our simulations indicate that increasing the number of wavelengths can notably enhance OpTree's performance. It's important to note, however, that the communication steps in OpTree follow a power function trajectory with respect to the number of network nodes (N). This aspect suggests a potential scalability challenge for very large-scale networks, where the increase in communication steps could become a limiting factor. Future work will not only focus on extending the scheme to other interconnect topologies and heterogeneous computing device scenarios but also on addressing this scalability issue to broaden OpTree's applicability.

ACKNOWLEDGMENT

The authors thank the reviewers for taking the time and effort necessary to review the manuscript. Besides, we acknowledge the use of New Zealand eScience Infrastructure (NeSI) high performance computing facilities as part of this research (Project code: uoo03633).

REFERENCES

[1] F. Dai, Y. Chen, Z. Huang, and H. Zhang, "Optree: An efficient algorithm for all-gather operation in optical interconnect systems," in *ICC 2023 - IEEE International Conference on Communications*, 2023, pp. 428–434.

[2] K. S. Khorassani, C.-H. Chu, Q. G. Anthony, H. Subramoni, and D. K. Panda, "Adaptive and hierarchical large message all-to-all communication algorithms for large-scale dense gpu systems," in *2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid)*. IEEE, 2021, pp. 113–122.

[3] T. Ben-Nun and T. Hoefler, "Demystifying parallel and distributed deep learning: An in-depth concurrency analysis," *ACM Computing Surveys (CSUR)*, vol. 52, no. 4, pp. 1–43, 2019.

[4] M. Khani, M. Ghobadi, M. Alizadeh, Z. Zhu, M. Glick, K. Bergman, A. Vahdat, B. Klenk, and E. Ebrahimi, "Sip-ml: high-bandwidth optical network interconnects for machine learning training," in *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*, 2021, pp. 657–675.

[5] Y. Zhu and J. P. Jue, "Reliable collective communications with weighted srlgs in optical networks," *IEEE/ACM transactions on Networking*, vol. 20, no. 3, pp. 851–863, 2011.

[6] W. Liang and X. Shen, "A general approach for all-to-all routing in multihop wdm optical networks," *IEEE/ACM transactions on networking*, vol. 14, no. 4, pp. 914–923, 2006.

[7] F. Dai, Y. Chen, Z. Huang, and H. Zhang, "Wrht: Efficient all-reduce for distributed dnn training in optical interconnect systems," in *Proceedings of the 52nd International Conference on Parallel Processing*, 2023, pp. 556–565.

[8] J. Chen, L. Zhang, Y. Zhang, and W. Yuan, "Performance evaluation of allgather algorithms on terascale linux cluster with fast ethernet," in *Eighth International Conference on High-Performance Computing in Asia-Pacific Region (HPCASIA'05)*. IEEE, 2005, pp. 6–pp.

[9] J. Chen, Y. Zhang, L. Zhang, and W. Yuan, "Performance of a new allgather algorithm on terascale deepcomp 6800," in *Proceedings of the 2005 Joint DCABES and ICPACE Meeting*, 2005, pp. 73–76.

[10] Y. Saad and M. H. Schultz, "Data communication in parallel architectures," *Parallel Computing*, vol. 11, no. 2, pp. 131–150, 1989.

[11] R. Graham, M. G. Venkata, J. Ladd, P. Shamis, I. Rabinovitz, V. Filipov, and G. Shainer, "Cheetah: A framework for scalable hierarchical collective operations," in *2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*. IEEE, 2011, pp. 73–83.

[12] J. L. Träff, "Efficient allgather for regular smp-clusters," in *European Parallel Virtual Machine/Message Passing Interface Users' Group Meeting*. Springer, 2006, pp. 58–65.

[13] N. T. Karonis, B. R. De Supinski, I. Foster, W. Gropp, E. Lusk, and J. Bresnahan, "Exploiting hierarchy in parallel computer networks to optimize collective operation performance," in *Proceedings 14th international parallel and distributed processing symposium. IPDPS 2000*. IEEE, 2000, pp. 377–384.

[14] K. Kandalla, H. Subramoni, G. Santhanaraman, M. Koop, and D. K. Panda, "Designing multi-leader-based allgather algorithms for multi-core clusters," in *2009 IEEE International Symposium on Parallel & Distributed Processing*. IEEE, 2009, pp. 1–8.

[15] J. L. Träff and S. Hunold, "Decomposing mpi collectives for exploiting multi-lane communication," in *2020 IEEE International Conference on Cluster Computing (CLUSTER)*. IEEE, 2020, pp. 270–280.

[16] T. Ma, G. Bosilca, A. Bouteiller, and J. Dongarra, "Hierknem: An adaptive framework for kernel-assisted and topology-aware collective communications on many-core clusters," in *2012 IEEE 26th International Parallel and Distributed Processing Symposium*. IEEE, 2012, pp. 970–982.

[17] S. H. Mirsadeghi and A. Afsahi, "Topology-aware rank reordering for mpi collectives," in *2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*. IEEE, 2016, pp. 1759–1768.

[18] D. Kranzlmüller, P. Kacsuk, and J. Dongarra, "Recent advances in parallel virtual machine and message passing interface," *The International Journal of High Performance Computing Applications*, vol. 19, no. 2, pp. 99–101, 2005.

[19] S. Kumar and L. V. Kale, "Scaling all-to-all multicast on fat-tree networks," in *Proceedings. Tenth International Conference on Parallel and Distributed Systems, 2004. ICPADS 2004*. IEEE, 2004, pp. 205–214.

[20] P. Sack and W. Gropp, "Faster topology-aware collective algorithms through non-minimal communication," *ACM SIGPLAN Notices*, vol. 47, no. 8, pp. 45–54, 2012.

[21] A. Venkatesh, S. Potluri, R. Rajachandrasekar, M. Luo, K. Hamidouche, and D. K. Panda, "High performance alltoall and allgather designs for infiniband mic clusters," in *2014 IEEE 28th International Parallel and Distributed Processing Symposium*. IEEE, 2014, pp. 637–646.

[22] A. Faraj, X. Yuan, and D. Lowenthal, "Star-mpi: self tuned adaptive routines for mpi collective operations," in *Proceedings of the 20th annual international conference on Supercomputing*, 2006, pp. 199–208.

[23] A. Faraj and X. Yuan, "Automatic generation and tuning of mpi collective communication routines," in *Proceedings of the 19th annual international conference on Supercomputing*, 2005, pp. 393–402.

[24] R. Thakur, R. Rabenseifner, and W. Gropp, "Optimization of collective communication operations in mpich," *The International Journal of High Performance Computing Applications*, vol. 19, no. 1, pp. 49–66, 2005.

[25] J.-C. Bermond, L. Gargano, S. Perennes, A. A. Rescigno, and U. Vaccaro, "Efficient collective communication in optical networks," in *International Colloquium on Automata, Languages, and Programming*. Springer, 1996, pp. 574–585.

[26] M. Sabrigiriraj and M. Meenakshi, "All-to-all broadcast in optical wdm networks under light-tree model," *Computer communications*, vol. 31, no. 10, pp. 2562–2565, 2008.

[27] B. Beauquier, "All-to-all communication for some wavelength-routed all-optical networks," *Networks: An International Journal*, vol. 33, no. 3, pp. 179–187, 1999.

[28] L. Narayanan, J. Opatrny, and D. Sotteau, "All-to-all optical routing in chordal rings of degree four," in *Proceedings of the Symposium on Discrete Algorithms*. Citeseer, 1999, pp. 695–703.

[29] X. Zhang and C. Qiao, "On scheduling all-to-all personalized connection and cost-effective designs in wdm rings," *IEEE/ACM Transactions On Networking*, vol. 7, no. 3, pp. 435–445, 1999.

[30] S. A. Pascu and A. A. El-Amawy, "On conflict-free all-to-all broadcast in one-hop optical networks of arbitrary topologies," *IEEE/ACM Transactions on Networking*, vol. 17, no. 5, pp. 1619–1630, 2009.

[31] J. Opatrny, "Uniform multi-hop all-to-all optical routings in rings," *Theoretical computer science*, vol. 297, no. 1-3, pp. 385–397, 2003.

[32] Q.-P. Gu and S. Peng, "Multihop all-to-all broadcast on wdm optical networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 14, no. 5, pp. 477–486, 2003.

[33] M. Khani, M. Ghobadi, M. Alizadeh, Z. Zhu, M. Glick, K. Bergman, A. Vahdat, B. Klenk, and E. Ebrahimi, "Terarack: A tbps rack for machine learning training," 2020.

[34] J. Tolentino, R. M. Marcelo, and M. A. C. Tolentino, "On twin edge colorings in m-ary trees," *Electronic Journal of Graph Theory and Applications (EJGTA)*, vol. 10, no. 1, pp. 131–149, 2022.

[35] H. Liao, J. Tu, J. Xia, H. Liu, X. Zhou, H. Yuan, and Y. Hu, "Ascend: a scalable and unified architecture for ubiquitous deep neural network computing: Industry track paper," in *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 2021, pp. 789–801.

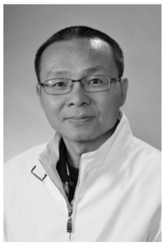
[36] L. H. Duong, M. Nikdast, J. Xu, Z. Wang, Y. Thonnart, S. Le Beux, P. Yang, X. Wu, and Z. Wang, "Coherent crosstalk noise analyses in ring-based optical interconnects," in *2015 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2015, pp. 501–506.

[37] S. Sarkar and N. R. Das, "Study of component crosstalk and obtaining optimum detection threshold for minimum bit-error-rate in a wdm receiver," *Journal of lightwave technology*, vol. 27, no. 19, pp. 4366–4373, 2009.

[38] F. Dai, Y. Chen, Z. Huang, H. Zhang, H. Zhang, and C. Xia, "Comparing the performance of multi-layer perceptron training on electrical and optical network-on-chips," *The Journal of Supercomputing*, pp. 1–22, 2022.

Fei Dai obtained his PhD degree in computer science from the University of Otago, New Zealand in 2023 and Master degree of Software Engineering in Guilin University of Technology, China in 2019. He is currently a Lecturer in Eastern Institute of Technology | Te Pūkenga, New Zealand. His research interests include optical network-on-chips, optical interconnects, performance modeling, optical communication optimization, multi-core architecture, parallel computation, deep learning accelerators, IoT & Embedded system, etc.

Yawen Chen obtained her PhD degree in computer science from The University of Adelaide in Australia in 2008 and Master degree in 2004 from Shandong Normal University in China. She worked as a researcher in Japan Advanced Institute of Science and Technology (JAIST) in 2005. She is currently a Senior Lecturer with the Department of Computer Science, University of Otago, New Zealand. Her research interests include resource optimization and performance evaluation in computer networking and computer architecture.



Zhiyi Huang is an Full Professor at the Department of Computer Science, University of Otago. He received the BSc degree in 1986 and the PhD degree in 1992 from the National University of Defense Technology (NUDT) in China. He was a visiting professor at EPFL and Tsinghua University in 2005, and a visiting scientist at MIT CSAIL in 2009. His research fields include parallel computing, multicore architectures, operating systems, high-performance computing and computer networks. He has more than 100 publications.

Haibo Zhang received MSc degree from Shandong Normal University, China, in 2005 and PhD degree from the University of Adelaide, Australia, in 2009. He is currently an Associate Professor in the Department of Computer Science, University of Otago, New Zealand. He has served on many technical program committee boards and co-chaired several international conferences. His current research interests include wireless networking, optical network-on-chips, etc.