

# 分布式机器学习网络通信优化技术

## Communication Optimization Technology of Distributed Machine Learning Network

张汉钢,邓鑫源,宋 晔,薛旭伟,郭秉礼,黄善国(北京邮电大学,北京 100876)

Zhang Hangang, Deng Xinyuan, Song Ye, Xue Xuwei, Guo Bingli, Huang Shanguo (Beijing University of Posts and Telecommunications, Beijing 100876, China)

### 摘要:

Ring all-reduce 算法被广泛应用在分布式机器学习之中,其同步过程会受到慢节点的影响进而降低整个系统的效率。对 Ring all-reduce 中的 Reduce\_Scatter 和 Allgather 2 个阶段进行分析,针对 Reduce\_Scatter 数据汇总过程提出优化策略,其主要思想是将慢节点多出的计算时间与通信时间进行重叠。使用 OMNet++ 对 Ring all-reduce 和优化策略进行对比仿真,仿真结果与理论分析相一致,该策略相比 Ring all-reduce 算法最高能缩短 25.3% 的训练时间。

### 关键词:

Ring all-reduce 算法;分布式机器学习;Ring all-reduce 优化策略

doi: 10.12045/j.issn.1007-3043.2024.02.005

文章编号: 1007-3043(2024)02-0027-04

中图分类号: TP391

文献标识码: A

开放科学(资源服务)标识码(OSID):



### Abstract:

Ring all-reduce algorithm is widely used in distributed machine learning, its synchronization process will be affected by slow nodes and reduce the efficiency of the whole system. The two stages of Reduce\_Scatter and Allgather in Ring all-reduce are analyzed, and an optimization strategy is proposed for the data summary process of Reduce\_Scatter. The main idea is to overlap the extra calculation time and communication time of slow nodes. OMNet++ is used to compare and simulate Ring all-reduce and optimization strategy. The simulation results are consistent with the theoretical analysis, and the strategy can shorten the training time by up to 25.3% compared with Ring all-reduce algorithm.

### Keywords:

Ring all-reduce algorithm; Distributed machine learning; Ring all-reduce optimization strategy

引用格式: 张汉钢,邓鑫源,宋晔,等. 分布式机器学习网络通信优化技术[J]. 邮电设计技术, 2024(2): 27-30.

## 1 概述

近些年随着互联网的蓬勃发展及大数据技术的不断演进,在图像识别、语言翻译、自动驾驶等相关领域,机器学习取得了众多突破并被广泛成功应用。在进行机器学习训练时,通常需要大量的数据用于计算,以保证模型的准确性。在海量数据的背景下,训练所使用的数据集的规模越来越大,训练的模型越来越复杂,单个机器无法满足存储与算力的需求,分布式机器学习将是未来发展的必然趋势。分布式机器

学习需要快速、高效的连接网络作为支撑,该网络结构有 2 种主要的并行方式:数据并行与模型并行。数据并行是最为常见的并行模式,广泛应用于大规模分布式系统中。在采用数据并行的方式进行训练时,各个机器之间需要对大量的数据进行同步,例如 GPT-3 就拥有超过 1 750 亿个机器学习参数。通信时间在训练时间中占比较大,如何减少通信开销成为了热门课题。数据并行使用 all-reduce 操作进行梯度同步,其中 Ring all-reduce 算法在高性能计算领域得到广泛应用。

Ring all-reduce 基于同步通信,在计算节点的性能存在差异时,整体的计算性能会被较慢的计算节点拖慢。为解决同步通信的这一弊端,异步并行(ASP)与

收稿日期: 2024-01-16

延迟同步并行(SSP)被相继提出,但 Ring all-reduce 架构难以支持 ASP 和 SSP 等模型的一致性协议。本文进一步分析 Ring all-reduce 算法特性,提出一种对其同步参数的过程进行加速的优化方案,降低慢节点带来的影响。

## 2 优化策略及理论分析

### 2.1 Ring all-reduce 基本分析

百度在 2017 年提出将 Ring all-reduce 架构引入到分布式机器学习中来,使这个架构受到广泛关注。Ring all-reduce 算法分为 2 个部分: Reduce\_Scatter 和 Allgather。Reduce\_Scatter 阶段按照  $n$  个训练节点数将数据划分为  $n$  个 chunk,将从上一相邻节点接收到的 chunk 同本地 chunk 进行 reduce 操作并发,该阶段包含  $n-1$  个步骤,如图 1 所示。Allgather 阶段是将汇总后的梯度段依次传递,用接收到的内容替换本地内容,该阶段包含  $n-1$  个步骤(见图 2)。

在 Github 上百度公开给出名为 baidu-allreduce 的

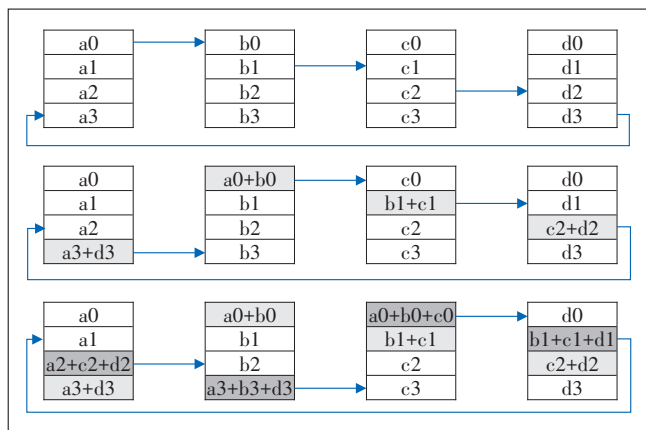


图 1 Reduce\_Scatter 阶段示意

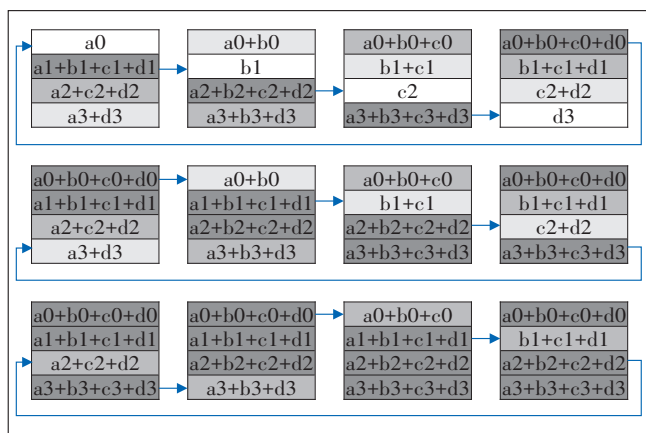


图 2 Allgather 阶段示意

小型 c++ 库,具体演示了环形的 all-reduce 与 Allgather 过程。Ring all-reduce 的同步过程体现在进行 reduce 时,调用了同步 CUDA 流(cudaStreamSynchronize)来等待所有 GPU 都完成操作。reduce 操作执行完毕也就意味着完成了一轮同步过程,随后向后一个节点进行发送操作。

当某个节点的 reduce 阶段所用时间显著变长,即该节点为慢节点时,若按照 Ring all-reduce 过程进行同步,则需要等慢节点多产生的时间,而这多出的时间将直接加在 Ring all-reduce 原有的时间开销上,影响到 Ring all-reduce 算法的执行效率。为了减轻慢节点带来的负面影响,如何利用慢节点较正常节点多出来的时间便成为了关键,下文给出一种优化策略思路。

### 2.2 优化策略

以 4 节点为例对 Ring all-reduce 的 Reduce\_Scatter 阶段进行分析。设置  $T_{rs}$  为节点接收数据、进行 reduce 操作和发送数据的时间,  $T_{trans}$  为传输数据的时间。A 节点将  $a_0$  发送至 B 节点, B 节点将收到的  $a_0$  与本地的  $b_0$  进行汇总,并在下一次通信时将汇总的内容发送至 C 节点(见图 3)。在这一时期, B 节点因 reduce 过程缓慢而成为慢节点,  $T_{rs}$  变长, C 节点需要等待的  $T_{rs}+T_{trans}$  时间也随之变长,将导致训练效率下降。

本文提出一种优化策略,设置  $T$  表示在没有慢节点情况下的  $T_{rs}$  的平均值,则 Reduce\_Scatter 阶段 1 个步骤的实际开销  $T_{rs}=T+\Delta T$ ,其中  $\Delta T$  为因外在因素导致的时间波动;  $\alpha T$  为节点触发策略的时间阈值,  $\alpha > 1$ 。当出现慢节点时,  $\Delta T$  变大;当  $T_{rs}=T+\Delta T > \alpha T$  时,则触发策略,具体过程如图 4 所示。

假设 B 节点是慢节点,  $T_i$  为 C 节点等待时间。当  $T_{rs}=T+\Delta T > T_i = \alpha T + T_{trans}$  时,改变 C 节点的操作步骤。此时 C 节点将不再等待接收 B 节点发送的  $a_0 + b_0$ ,而是直接将本地的  $c_0$  发送至  $d_0$ ; B 节点达到  $T_{rs}$  时将跳过 C 节点直接与 D 节点进行通信。为了更直观地分析时间

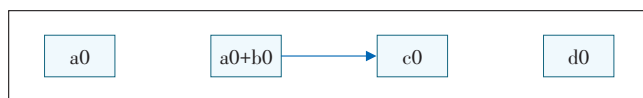


图 3 Ring all-reduce 的通信步骤



图 4 改变策略后的通信步骤

上的重叠,假设如下。

- a) 各个结点之间传输数据的时间  $T_{trans}$  都一致。
- b) 若节点为慢节点,其  $T_{RS}=T+\Delta T$  且  $\Delta T>0$ 。
- c) 若节点为非慢节点,其  $T_{RS}=T$ 。
- d) 设定B节点为慢节点。

如图5所示,当B节点是慢节点且C节点等待至  $\alpha T+T_{trans}$  时,C节点直接向D节点传递数据,D节点先收到来自C节点的数据并等待B节点发送来的数据。在收到2份数据后,D节点一并进行汇总。截至D节点汇总所有数据时,策略带来的时间减少值如式(1)所示。

$$T_{dec} = T + \Delta T + 2T_{trans} + 2T - (T + \Delta T + T_{trans} + T) = T_{trans} + T \quad (1)$$

由式(1)可知,当  $\Delta T$  较大时,在  $T+\Delta T>\alpha T+T_{trans}$  情况下策略带来的时间减少值为固定时间。

如图6所示,当  $T_{RS}=T+\Delta T<T_t=\alpha T+T_{trans}$  时,慢节点B发送的消息先到,等到C节点发送的数据到达后,D节点一并进行汇总。此时策略带来的时间减少值为:

$$T_{dec} = T + \Delta T + 2T_{trans} + 2T - (\alpha T + 2T_{trans} + T) = \Delta T + (2 - \alpha)T \quad (2)$$

当  $\Delta T$  与  $\alpha$  取合适的值时,可以使  $T_{dec}$  取得一个较大值,即在有慢节点的情况下能减轻慢节点带来的负面影响,达到优化效果。

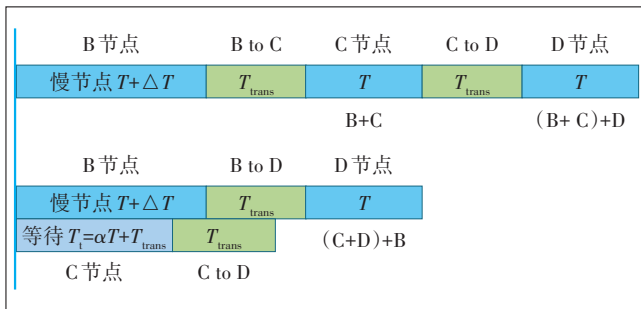


图5 B节点发送的消息先到达D节点

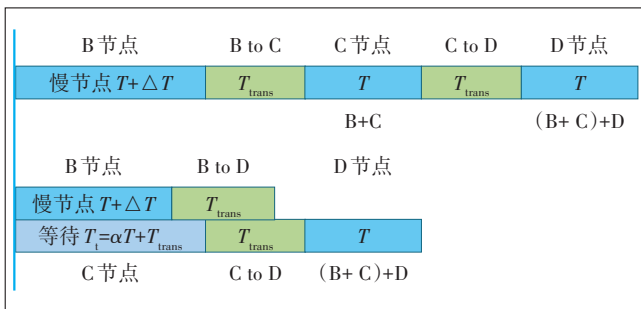


图6 C节点发送的消息先到达D节点

### 3 仿真实验设置及结果分析

采用OMNet++框架对 Ring all-reduce 算法和改进策略进行模拟。因为 Allgather 阶段这2种方式的开销相同,故仅比较2种方式完成 Reduce\_Scatter 所用的时间(分别是  $T_{Ring}$  和  $T_{Strategy}$ )。基于第2.2节做出的基本假设,对相关的仿真参数进行设定:  $T=3 \mu s$ ;  $T_{trans}$  为固定链路时延  $5 \mu s$  与数据包传递时间之和;数据包大小取  $8 \text{ KB}$ ;链路带宽为  $100 \text{ Gbit/s}$ 。设置不同的  $\Delta T$  与  $\alpha$  的值,优化效果即  $T_{dec}$  与  $T_{Ring}$  的比值。

如图7所示, Ring all-reduce 算法完成 Reduce\_Scatter 阶段所用时间随着  $\Delta T$  的增加而线性增加,而采用优化策略完成 Reduce\_Scatter 阶段所用时间均得到减少,说明采取优化策略能减少时间开销。进一步分析,如图8所示,当  $\alpha$  不变时,优化效果大体上随着  $\Delta T$  的增大而先增后减。当  $\Delta T$  较小时,  $T_{RS}=T+$

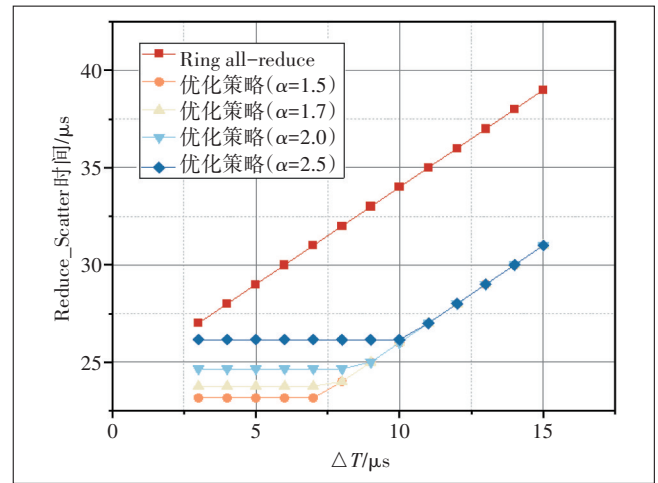


图7 Ring all-reduce 与优化策略时间开销比较

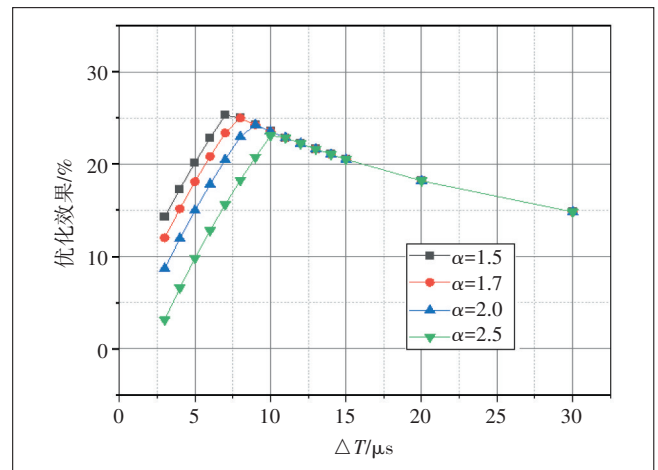


图8  $\Delta T$  与  $\alpha$  对优化效果的影响

$\Delta T < T_i = \alpha T + T_{trans}$ , 优化效果会随着  $\alpha$  的增大而减小。结果符合式(2)预期。

当  $T_{RS} = T + \Delta T > T_i = \alpha T + T_{trans}$ , 优化效果与  $\alpha$  的变化无关(见图9), 结果符合式(1)预期。

$T + \Delta T < \alpha T + T_{trans}$  时, 即慢节点跳过后一节点发送的数据先到达,  $\alpha$  增大, 说明后一节点最长等待时间将延长, 触发机制的机会将相对减少, 机制带来的增益受到限制, 优化效果随之减弱。

当  $\Delta T$  是一个较大偏置时, 有  $T + \Delta T > \alpha T + T_{trans}$ , 即后一节点发送的数据先到达而慢节点发送的数据后到达, 此时获得的时间减少值  $T_{dec}$  是一个定值。  $\Delta T$  的增加会导致  $T_{Ring}$  的时间线性增加,  $T_{dec}$  与  $T_{Ring}$  的比值也随之减少, 即优化效果减弱(见图10)。

## 4 总结

在分布式计算中, Ring all-reduce 是一种常用的通

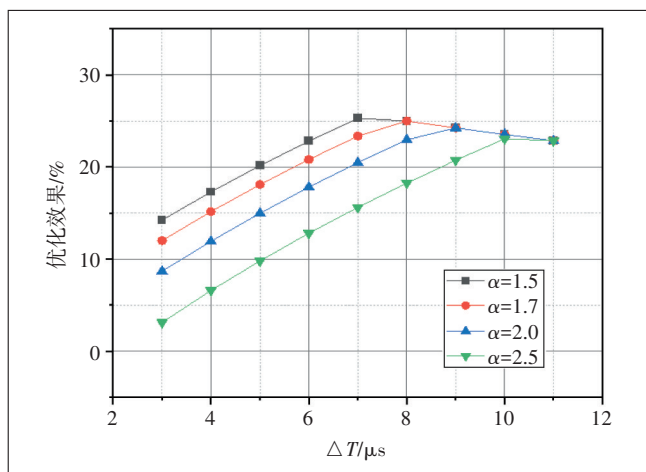


图9  $\alpha$  变化对优化效果影响

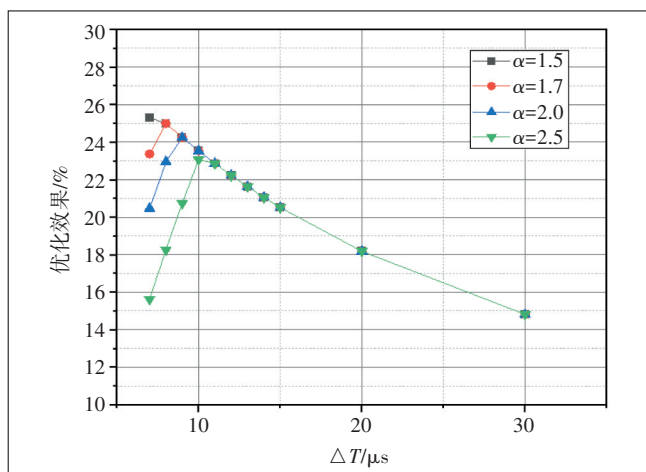


图10  $\Delta T$  变化对优化效果影响

信模式, 然而由于网络中存在慢节点, 通信延迟分布不均匀, 会影响整体性能。本文聚焦 Ring all-reduce 算法的 Reduce\_Scatter 阶段, 提出一种创新性的优化策略实现对慢节点的感知与通信调整, 减小慢节点对整体通信性能的负面影响。仿真结果显示, 相较于传统的 Ring all-reduce 算法, 本文提出的优化策略成功降低了通信的时间开销, 训练时长最高节省了 25.3%。

## 参考文献:

- [1] SEVILLA J, HEIM L, HO A, et al. Compute trends across three eras of machine learning [C]//2022 International Joint Conference on Neural Networks (IJCNN). Padua: IEEE, 2022: 1-8.
- [2] QIAO S Y, ZHANG Z S, SHEN W, et al. Gradually updated neural networks for large-scale image recognition [C]//Proceedings of the 35th International Conference on Machine Learning. New York: PMLR, 2018: 4188-4197.
- [3] XIONG W, WU L, ALLEVA F, et al. The microsoft 2017 conversational speech recognition system [C]//2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Calgary: IEEE, 2018: 5934-5938.
- [4] HE D, LU H Q, XIA Y C, et al. Decoding with value networks for Neural Machine Translation [C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc, 2017: 177-186.
- [5] BOJARSKI M, DEL TESTA D, DWORAKOWSKI D, et al. End to end learning for self-driving cars [EB/OL]. [2023-12-02]. <https://arxiv.org/abs/1604.07316>.
- [6] KESKAR N S, MUDIGERE D, NOCEDAL J, et al. On large-batch training for deep learning: generalization gap and sharp minima [EB/OL]. [2023-12-02]. <https://arxiv.org/abs/1609.04836>.
- [7] GENG J K, LI D, WANG S. ElasticPipe: an efficient and dynamic model-parallel solution to DNN training [C]//Proceedings of the 10th Workshop on Scientific Cloud Computing. Phoenix: Association for Computing Machinery, 2019: 5-9.
- [8] PENG Y H, ZHU Y B, CHEN Y R, et al. A generic communication scheduler for distributed DNN training acceleration [C]//Proceedings of the 27th ACM Symposium on Operating Systems Principles. Huntsville: Association for Computing Machinery, 2019: 16-29.

## 作者简介:

张汉钢, 北京邮电大学硕士在读, 主要从事分布式训练通信加速机制的研究工作; 邓鑫源, 北京邮电大学硕士在读, 主要从事 OMNet++ 仿真网络的相关工作; 宋晔, 北京邮电大学硕士在读, 主要从事仿真算力节点网络的研究工作; 薛旭伟, 拔尖人才教授, 博士生导师, 主要研究方向为面向数据中心和高性能计算的超快全光交换技术、可调谐硅光器件设计、高精度时频同步技术等; 郭秉礼, 副教授, 硕士生导师, 主要从事数据中心与高性能计算中的光互联网络技术的相关研究工作; 黄善国, 教授, 博士生导师, 主要从事多维光交换与光网络的研究工作。