DGT: A Communication-Efficient Differential Gradient Transmission Protocol For Distributed Deep Learning

Huaman Zhou
University of Electronic Science and Technology of China
Chengdu, Sichuan
hmzhou@std.uestc.edu.cn

Zonghang Li
University of Electronic Science and Technology of China
Chengdu, Sichuan
zhli@std.uestc.edu.cn

Hongfang Yu
University of Electronic Science and Technology of China
Chengdu, Sichuan
yuhf@uestc.edu.cn

ABSTRACT

In recent years, researchers have paid more attention to distributed deep learning to train large-scale models with massive data in parallel, which is much more efficient than training on a single machine. However, machines still need to communicate model gradients to realize collaboration, results in unneglectable communication overhead and limits the scalability when deployed on bandwidth-limited and intermittent connection network. This paper proposes *Differential Gradient Transmission* (DGT), a communication-efficient gradient transmission protocol designed for Distributed Deep Learning, which transfers gradients in multi-channels with different reliability and priority according to their contribution to model convergence to effectively accelerate the model training. Experiments on a cluster with 6 GTX 1080TI GPUs and 1Gbps network show that DGT decreases the training time to achieve the specified test accuracy 0.8 by 15.1% on AlexNet model and 18.3% on VGG-11 model, indicating that DGT makes a good trade-off between transmission delay and model convergence, and obtains desired training acceleration.

CCS CONCEPTS

- **Computing methodologies → Distributed artificial intelligence**.

KEYWORDS

Deep Learning, Distributed Artificial Intelligence, Differential Gradient Transmission, Approximate Transmission Protocol

ACM Reference Format:
Huaman Zhou, Zonghang Li, and Hongfang Yu. 2019. DGT: A Communication-Efficient Differential Gradient Transmission Protocol For Distributed Deep Learning. In *Proceedings of 3rd Asia-Pacific Workshop on Networking (APNet'19).* ACM, Beijing,China, 2 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

The success of emerging AI technologies is largely due to massive data and large-scale models, which requires extremely high storage capacity and computing power to perform model training, and takes weeks or even months to achieve convergence on a singe machine. To this end, researchers proposed Distributed Deep Learning, a paradigm to train deep learning models over multiple machines, to accelerate the model training and achieved great success[4]. In the classical data parallel mode, several machines train a shared global model collaboratively with local data, which avoids the communication overhead caused by the transmission of massive raw data and accelerates the training speed through parallel computing.

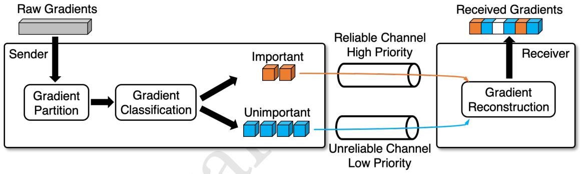

Figure 1: Overview of DGT.

However, machines still need to communicate model gradients to realize collaboration. In general distributed deep learning systems, for the purpose of preventing impaired model convergence due to lost gradient or transmission error, machines upload all the model gradients accurately to perform aggregation using reliable transmission protocols (e.g. TCP). However, uploading all the model gradients accurately results in extremely high transmission delay, and situations would get worse with large models and harsh network environment. Fortunately, the gradient-based optimization algorithms (e.g. SGD[1]) have natural tolerance for gradients with noise, which can be considered as Approximate Computing[5] paradigms (computation can tolerate inaccurate or incomplete data). Recently, Liu et al.[3] proposed Approximate Transmission Protocol (ATP) for general approximate computing applications, which allows packets to be lost within a certain limit, to reduce the transmission delay caused by transmitting large dataset. We introduce ATP in Distributed Deep Learning to transmit model gradients. Preliminary results show that ATP failed to obtain desired acceleration of total training time, due to ATP drops gradients indiscriminately, leading to risk of convergence oscillation and accuracy reduction.

In order to accelerate the model training, we combine the above reliable transmission protocols and approximate transmission protocols to make trade-offs between transmission delay and model convergence. This paper proposes *Differential Gradient Transmission* (DGT), a communication-efficient gradient transmission protocol designed for Distributed Deep Learning. As shown in Fig 1, DGT classifies the model gradients into important and unimportant , then transmits the important gradients via reliable channel with high priority to provide model convergence guarantee, instead, the unimportant gradients are transmitted in "best-effort delivery" mode via unreliable channels with low priority to minimum transmission delay. The challenges include: 1) How to classify the model gradients, and 2) How to transmit model gradients discriminately to obtain a good trade-off between transmission delay and model convergence. We will discuss these challenges and provide solutions in the next section.

APNet'19, August 2019, Beijing, China
2019. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Table 1: Alexnet and VGG-11 trained on Fashion-MNIST

Model	Protocol	BCT (s)	Total Iters	Total Time (s)
AlexNet	Baseline	36.72	**1300**	47494
	DGT	**23.71**	1700	**40314**
VGG-11	Baseline	105.34	**420**	44246
	DGT	**73.75**	490	**36141**

To the best of the authors' knowledge, we are the first to combine reliable transmission protocols and approximate transmission protocols for transmitting gradients in Distributed Deep Learning. We evaluate DGT on AlexNet and VGG-11 models with the real-world Fashion-MNIST dataset on a cluster with 6 GTX 1080TI GPUs and 1Gbps network. Experimental results show that DGT decreases the training time to achieve the specified test accuracy 0.8 by 15.1% on AlexNet model and 18.3% on VGG-11 model, indicating that DGT makes a good trade-off between transmission delay and model convergence, and therefore obtains desired training acceleration.

2 METHODS

In this section, we will give a brief description of the key techniques used in DGT, including: 1) Gradient Classification, and 2) Differential Transmission.

Gradient Classification. General deep learning software frameworks (e.g. MXNET[2]) update and transmit model parameters at tensor granularity, i.e. the parameters of an neural network layer. To refine the difference in tensors, DGT divides a tensor into several sub-blocks with fixed size n, and estimates the contribution of each sub-block based on its gradients. In our implementation, we designed an algorithm named *Sliding Average Norm Estimation*, which updates the contribution $C_k^j(\tau)$ of the j-th sub-block in the k-th tensor at τ-th iteration as:

$$C_k^j(\tau) = \alpha C_k^j(\tau - 1) + (1 - \alpha)\frac{1}{n}\sum_{i=1}^{n}|g_i|, \quad g_i \in G_k^j \tag{1}$$

where $C_k^j(0) = 0$, G_k^j is the gradient matrix of the j-th sub-block in the k-th tensor, and α $(0 \leq \alpha \leq 1)$ is a constant value called contribution momentum factor.

DGT ranks all the sub-blocks in the k-th tensor according to their contribution, and classifies the gradients of top-p% sub-blocks as important and the rest as unimportant. We define p as a threshold to classify gradients, which controls the amount of important gradients, and a well-designed threshold p can balance the transmission delay and the model convergence efficiently. In our preliminary design, DGT updates the threshold p in a heuristic way as $p = p_0 \cdot loss_\tau / loss_0$ (where p_0 is the initial ratio of important gradients), based on the fact that the gradients tend to be sparse and most gradients become unimportant as the training goes on.

Differential Transmission. DGT schedules the important gradients to the channel with reliable transmission protocol (e.g. TCP) to provide model convergence guarantee, instead, the unimportant gradients are scheduled to channels with unreliable transmission protocol (e.g. UDP) to minimum transmission delay. Specially, the unimportant gradients that are late to reach the receiver will be actively discarded when all the important gradients are received, to avoid wasting too much time waiting for the unimportant gradients and therefore reduce transmission delay.

Moreover, DGT allows the important gradients to be transmitted preferentially in network through tabbing high priority at DSCP field in IP packets, to further reduce transmission delay. However, the unimportant gradients are tabbed with low priority, making them more likely to be dropped or delayed when congestion occurs.

3 RESULTS

The experimental platform is built on MXNET, a flexible and efficient library for deep learning, based on which we implement DGT on its component KVSTORE and an open source communication library PS-LITE. We evaluate DGT on a cluster with 6 GTX 1080TI GPUs, 4 switches and 1Gbps network, and select the classic AlexNet and VGG-11 models and Fashion-MNIST dataset to demonstrate the performance of DGT. We consider the batch completion time (BCT), the total number of iterations (Total Iters) and the total time (Total Time) to achieve test accuracy 0.8 on AlexNet and VGG-11 models in our experiments. We set $n = 1024, \alpha = 0.3, p_0 = 1$, and let baseline represents the reliable transmission protocol which is widely used in general deep learning software framework (e.g. TCP in MXNET and TensorFlow).

Table 1 shows that DGT decreases BCT by 35.4% on AlexNet model and 29.9% on VGG-11 model, since important gradients are prioritized for transmission in network and a few unimportant gradients that are late to reach the receiver are actively discarded when all the important gradients are received. Although the loss of unimportant gradients will bring negative effects to model convergence, DGT still decrease total training time by 15.1% on Alexnet model and 18.3% on VGG-11 model, indicating that DGT makes a good trade-off between transmission delay and model convergence, and obtains desired training acceleration.

4 CONCLUSION

In this paper, we proposes a communication-efficient gradient transmission protocol for Distributed Deep Learning named *Differential Gradient Transmission* (DGT), which transfers gradients in multi-channels with different reliability and priority according to their contribution to model convergence. We test DGT on the classic AlexNet and VGG-11 models and the real-world Fashion-MNIST dataset. Empirical results show that DGT accelerates distributed training effectively on bandwidth-limited network. In future work, we will further 1) study a more reasonable threshold update strategy for p to find a better trade-off between transmission delay and model convergence; 2) use more channels to provide finer-grained control on reliability and priority; 3) test DGT on more advanced models and datasets; and 4) expand DGT on larger-scale clusters with crowded network for more empirical results.

REFERENCES

[1] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*. Springer, 177–186.

[2] Tianqi Chen, Mu Li, Yutian Li, et al. 2015. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274* (2015).

[3] Ke Liu, Shin-Yeh Tsai, and Yiying Zhang. 2019. ATP: a Datacenter Approximate Transmission Protocol. *arXiv preprint arXiv:1901.01632* (2019).

[4] Hiroaki Mikami, Hisahiro Suganuma, Yoshiki Tanaka, et al. 2018. Imagenet/resnet-50 training in 224 seconds. *arXiv preprint arXiv:1811.05233* (2018).

[5] Swaminathan Natarajan. 1995. *Imprecise and approximate computation*. Vol. 318. Springer Science & Business Media.