ULTIMA: Robust and Tail-Optimal All-Reduce for Distributed Deep Learning

Ertza Warraich*•, Leonard Liu*•, Omer Shabtai†, Yonatan Piasetzky†, Shay Vargaftik‡, Matty Kadosh†, Lalith Suresh‡, and Muhammad Shahbaz*

*Purdue University †NVIDIA ‡VMware Research — •Student

1 ABSTRACT

Synchronous distributed data-parallel training [17] is now the de-facto standard for training large-scale deep-learning models (comprising billions of parameters) that form the backbone of many mainstream enterprise applications, including computer vision [4, 8, 15, 32], natural-language processing [19, 21, 33], recommendation and prediction systems [6, 7, 9, 10, 23], networking [11, 12], healthcare [14, 20, 22, 34], and smart cities [2, 29]. Under this scheme, the training occurs in rounds. Workers locally train a copy of the model on a fragment of data and then share the model updates (a.k.a gradients) among themselves over the network to compute an aggregated result. The aggregate is then used to update the model locally for the next round of training. Distributed deep-learning (DDL) is, therefore, inherently a computation- and communication-intensive workload and is becoming even more so with growing model sizes and complexity [27] as well as ever-increasing amount of training data [1, 3].

To train such large models, extensive efforts are underway in reducing both the computation and communication time of DDL jobs, albeit in isolation. On the one hand, we have GPUs [26] and emerging hardware accelerators, like Tensor Processing Units (TPUs) [13], that are drastically bringing down the computation time—reducing it by 62× in the last seven years. While, on the other hand, we have recent proposals based on programmable switches [30] that aim at reducing the communication time by 2–5× (via in-network aggregation) [25]. Yet, when seen together, both these efforts mainly help in improving the average completion time of a deep-learning job (either by accelerating computation or communication). The vast array of system-level variabilities (e.g., device failures, OS and hypervisor scheduling, and resource contention) and network-level delays (e.g., congestion, packet drops and retransmissions, and out-of-order delivery) still lead to long tails; hence, resulting in poor overall performance for these training jobs.

In this paper, we make the case for ULTIMA, a collective-communication system for All-Reduce that ensures bounded, predictable completion times for deep-learning jobs in the presence of myriad computation and communication variabilities. ULTIMA exploits the inherent resiliency and the stochastic nature of deep-learning systems to work with approximated gradients and provides an efficient balance between (tail) performance and the resulting accuracy of the trained models. Others are already utilizing this characteristic of deep learning to optimize hardware design (e.g., chip area [24, 35]), minimize traffic overhead [5, 18, 28], or offload certain DDL tasks to the network switches [16, 25, 30, 31]. For example, to improve communication time, ATP [16] and SwitchML [25] utilize fixed-point arithmetic to execute gradient aggregation in programmable switches, with acceptable approximation loss. Various gradient-compression schemes [5, 18, 28] employ lossy compression to reduce network traffic overhead, while limiting deviation from the achievable model accuracy. Similarly, hardware designers are incorporating approximate operations (e.g., approx. multipliers [24, 35]) in their architectures to minimize resource and energy usage—to scale to ever-increasing DDL models.

In ULTIMA, we replace the (tail-prone) deterministic, run-to-completion computation and communication stages of a DDL system with best-effort, *time-bounded* implementations. (1) ULTIMA introduces the notion of *Adaptive Timeouts* to restrict the time a deep-learning job spends doing computation (forward/backward pass and aggregation) and communication (gradient sharing). (2) ULTIMA implements a new *Bounded Transport* to maximize the gradients received during each window. Unlike TCP or RDMA that are prone to tail effects due to out-of-order delivery and retransmissions, ULTIMA's transport only implements flow- and congestion-control while delivering gradients as fast as possible within the given time window. (3) It also incorporates *Native Multicast* and an accompanying *Transpose-Allreduce Collective* to further reduce the gradient delivery time from `O(N)` to `O(1)`. (4) Finally, to minimize the impact of missed or lost gradients, ULTIMA implements *Hadamard Transform* to ensure, for any drop pattern (e.g., tail drops), a receiver still obtains an unbiased estimate of the gradients resulting in minimal loss in terms of model accuracy. Our preliminary results are promising and show that ULTIMA can reach full model accuracy with 16% faster tail completion times under steady-state while incurring negligible approximation loss (0.2%) during bursty (system and network) conditions, compared to the state-of-the-art systems.

REFERENCES

[1] 2013. Criteo terabyte click logs dataset. https://labs.criteo.com/2013/12/download-terabyte-click-logs.

[2] Giuseppe Ciaburro and Gino Iannace. 2020. Improving smart cities safety using sound events detection based on deep neural network algorithms. In *Informatics*, Vol. 7. Multidisciplinary Digital Publishing Institute, 23.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[4] Xianzhi Du, Mostafa El-Khamy, Jungwon Lee, and Larry Davis. 2017. Fused DNN: A deep neural network fusion approach to fast and robust pedestrian detection. In *2017 IEEE winter conference on applications of computer vision (WACV)*. IEEE, 953–961.

[5] Jiawei Fei, Chen-Yu Ho, Atal N Sahu, Marco Canini, and Amedeo Sapio. 2021. Efficient sparse collective communication and its application to accelerate distributed deep learning. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*. 676–691.

[6] Shuo Feng, Huiyu Zhou, and Hongbiao Dong. 2019. Using deep neural network with small dataset to predict material defects. *Materials & Design* 162 (2019), 300–310.

[7] Mingsheng Fu, Hong Qu, Zhang Yi, Li Lu, and Yongsheng Liu. 2018. A novel deep learning-based collaborative filtering model for recommendation system. *IEEE transactions on cybernetics* 49, 3 (2018), 1084–1096.

[8] Abhinav Goel, Caleb Tung, Yung-Hsiang Lu, and George K Thiruvathukal. 2020. A survey of methods for low-power deep learning and computer vision. In *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*. IEEE, 1–6.

[9] Brian Heredia, Joseph D Prusa, and Taghi M Khoshgoftaar. 2018. Social media for polling and predicting United States election outcome. *Social Network Analysis and Mining* 8, 1 (2018), 1–16.

[10] Zhenhua Huang, Guangxu Shan, Jiujun Cheng, and Jian Sun. 2019. TRec: an efficient recommendation system for hunting passengers with deep neural networks. *Neural Computing and Applications* 31, 1 (2019), 209–222.

[11] Nathan Jay, Noga Rotman, Brighten Godfrey, Michael Schapira, and Aviv Tamar. 2019. A deep reinforcement learning perspective on internet congestion control. In *International Conference on Machine Learning*. PMLR, 3050–3059.

[12] Yang Jia, Meng Wang, and Yagang Wang. 2019. Network intrusion detection algorithm based on deep neural network. *IET Information Security* 13, 1 (2019), 48–53.

[13] Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. 2017. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th annual international symposium on computer architecture*. 1–12.

[14] Asif Iqbal Khan, Junaid Latief Shah, and Mohammad Mudasir Bhat. 2020. CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest x-ray images. *Computer methods and programs in biomedicine* 196 (2020), 105581.

[15] Michał Koziarski and Bogusław Cyganek. 2017. Image recognition with deep neural networks in presence of noise–dealing with and taking advantage of distortions. *Integrated Computer-Aided Engineering* 24, 4 (2017), 337–349.

[16] ChonLam Lao, Yanfang Le, Kshiteej Mahajan, Yixi Chen, Wenfei Wu, Aditya Akella, and Michael M Swift. 2021. ATP: In-network Aggregation for Multi-tenant Learning.. In *NSDI*. 741–761.

[17] Shen Li, Yanli Zhao, Rohan Varma, Omkar Salpekar, Pieter Noordhuis, Teng Li, Adam Paszke, Jeff Smith, Brian Vaughan, Pritam Damania, et al. 2020. Pytorch distributed: Experiences on accelerating data parallel training. *arXiv preprint arXiv:2006.15704* (2020).

[18] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and William J Dally. 2017. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887* (2017).

[19] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504* (2019).

[20] Pradeep Kumar Mallick, Seuc Ho Ryu, Sandeep Kumar Satapathy, Shruti Mishra, Gia Nhu Nguyen, and Prayag Tiwari. 2019. Brain MRI image classification for cancer detection using deep wavelet autoencoder-based deep neural network. *IEEE Access* 7 (2019), 46278–46287.

[21] Daniel W Otter, Julian R Medina, and Jugal K Kalita. 2020. A survey of the usages of deep learning for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems* 32, 2 (2020), 604–624.

[22] Yanxing Qi, Yi Guo, and Yuanyuan Wang. 2020. Image quality enhancement using a deep neural network for plane wave medical ultrasound imaging. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control* 68, 4 (2020), 926–934.

[23] S Ramesh, C Yaashuwanth, K Prathibanandhi, Adam Raja Basha, and T Jayasankar. 2021. An optimized deep neural network based DoS attack detection in wireless video sensor network. *Journal of Ambient Intelligence and Humanized Computing* (2021), 1–14.

[24] Alexander Rucker, Muhammad Shahbaz, and Kunle Olukotun. 2021. Chopping off the Tail: Bounded Non-Determinism for Real-Time Accelerators. *IEEE Computer Architecture Letters* 20, 2 (2021), 110–113.

[25] Amedeo Sapio, Marco Canini, Chen-Yu Ho, Jacob Nelson, Panos Kalnis, Changhoon Kim, Arvind Krishnamurthy, Masoud Moshref, Dan RK Ports, and Peter Richtárik. 2019. Scaling distributed machine learning with in-network aggregation. *arXiv preprint arXiv:1903.06701* (2019).

[26] Dave Steinkraus, Ian Buck, and PY Simard. 2005. Using GPUs for machine learning algorithms. In *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*. IEEE, 1115–1120.

[27] Dhananjay Theckedath and RR Sedamkar. 2020. Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks. *SN Computer Science* 1, 2 (2020), 1–7.

[28] Jianqiao Wangni, Jialei Wang, Ji Liu, and Tong Zhang. 2017. Gradient sparsification for communication-efficient distributed optimization. *arXiv preprint arXiv:1710.09854* (2017).

[29] Jacek Lukasz Wilk-Jakubowski, Pawel Stawczyk, Stefan Ivanov, and Stanko Stankov. 2022. Control of acoustic extinguisher with Deep Neural Networks for fire detection. *Elektronika ir Elektrotechnika* 28, 1 (2022), 52–59.

[30] Bruno Missi Xavier, Rafael Silva Guimarães, Giovanni Comarela, and Magnos Martinello. 2021. Programmable switches for in-networking classification. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 1–10.

[31] Zhaoqi Xiong and Noa Zilberman. 2019. Do switches dream of machine learning? toward in-network classification. In *Proceedings of the 18th ACM workshop on hot topics in networks*. 25–33.

[32] Cheng Xu, Duo Chai, Jie He, Xiaotong Zhang, and Shihong Duan. 2019. InnoHAR: A deep neural network for complex human activity recognition. *Ieee Access* 7 (2019), 9893–9902.

[33] Wenpeng Yin, Katharina Kann, Mo Yu, and Hinrich Schütze. 2017. Comparative study of CNN and RNN for natural language processing. *arXiv preprint arXiv:1702.01923* (2017).

[34] Xin Yuan, Weite Li, Kui Lin, and Jinglu Hu. 2019. A Deep Neural Network Based Hierarchical Multi-Label Classifier for Protein Function Prediction. In *2019 International Conference on Computer, Information and Telecommunication Systems (CITS)*. IEEE, 1–5.

[35] Georgios Zervakis, Hassaan Saadat, Hussam Amrouch, Andreas Gerstlauer, Sri Parameswaran, and Jörg Henkel. 2021. Approximate Computing for ML: State-of-the-Art, Challenges and Visions. In *ACM ASPDAC*.