2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW) | 979-8-3503-6460-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/IPDPSW63119.2024.00196

Energy-Aware Decentralized Learning with Intermittent Model Training

Martijn de Vos*, Akash Dhasade*, Paolo Dini†, Elia Guerra§‡,
Anne-Marie Kermarrec*, Marco Miozzo†, Rafael Pires* and Rishi Sharma*
*EPFL, Switzerland
†Centre Tecnològic de Telecomunicacions de Catalunya, Spain
‡Independent researcher

***Abstract*—SKIPTRAIN is a novel Decentralized Learning (DL) algorithm, which minimizes energy consumption in decentralized learning by strategically *skipping* some training rounds and substituting them with *synchronization* rounds. These training-silent periods, besides saving energy, also allow models to better mix and produce models with superior accuracy than typical DL algorithms. Our empirical evaluations with 256 nodes demonstrate that SKIPTRAIN reduces energy consumption by 50% and increases model accuracy by up to 12% compared to D-PSGD, the conventional DL algorithm.**

***Index Terms*—Decentralized Learning, Energy Efficiency, Peer-to-Peer Learning Systems.**

I. INTRODUCTION

Decentralized Learning (DL) represents an attractive alternative to centralized machine learning (ML), as it addresses privacy concerns by not moving training data while eliminating the dependency on a central server [1]–[4]. An important, yet overlooked problem of DL is the training energy consumed by DL algorithms. Typically, nodes in DL algorithms perform the following operations each round: *(i)* locally training the model; *(ii)* exchanging the model with neighbors; and *(iii)* aggregating models received from neighbors. Most of the energy consumption happens at training time *(i)*, while that of communication, *i.e.*, *(ii)* and *(iii)*, remains low. Specifically, using the model adopted by Guerra *et al.* [5], model training is more than $200\times$ costlier in terms of energy than model sharing and aggregation.

From an energy perspective, increasing the amount of sharing and aggregation operations has a negligible impact on the energy consumption. Furthermore, executing only sharing and aggregation rounds leads the local models towards the global consensus model, like the one produced by the central server in federated learning (FL) or through a decentralized all-reduce operation [6]. Based on this insight, we introduce SKIPTRAIN, our novel DL algorithm where nodes skip some *training* rounds (*i.e.*, training, sharing, and aggregation) in favor of *synchronization* rounds (*i.e.*, sharing and aggregation).

SKIPTRAIN-CONSTRAINED extends SKIPTRAIN to operate in scenarios in which nodes have energy constraints *i.e.*, where they are typically limited by individual energy budgets, such as in Internet-of-Things (IoT) networks [7]. Each node, depending on its energy capacity, makes an individual probabilistic decision in every round to either engage in training or skip training in favor of synchronization.

We evaluate the efficiency and performance of SKIPTRAIN on non independent and identically distributed (non-IID) data distributions using the CIFAR-10 and FEMNIST datasets. A unique aspect of our experimental setup is the integration of energy traces that we compiled by extending existing data [8]. SKIPTRAIN achieves a 50% reduction in energy consumption and increases model accuracy by up to 7% compared to decentralized parallel stochastic gradient descent (D-PSGD), a standard and popular DL algorithm. In energy-constrained settings, SKIPTRAIN-CONSTRAINED increases model accuracy by up to 12% compared to D-PSGD. We conclude that SKIPTRAIN is an effective approach that gives DL practitioners a flexible tool to decrease the energy impact of their DL tasks.

II. OUR APPROACH

SKIPTRAIN In SKIPTRAIN, a training round is similar to a complete round in D-PSGD. A node i carries out the training on the local data, sharing of the model with neighbors, and aggregation of the received models. During a synchronization round, however, only the sharing and aggregation steps are executed. SKIPTRAIN follows a pattern of alternating between a batch of Γ_{train} training rounds and Γ_{sync} synchronization rounds. The target is to alternate training and synchronization rounds such that the overall number of rounds does not increase when compared to standard D-PSGD.

SKIPTRAIN-CONSTRAINED In settings where devices run on batteries, nodes cannot perform arbitrarily many training rounds because of energy constraints. Specifically, each node $i \in [N]$ has a computational budget τ_i that defines the maximum number of training rounds that can be executed before its battery is depleted. One way to incorporate these constraints in D-PSGD would be to carry out consecutive training rounds until the allocated energy budget is exhausted. For the remaining rounds, the node will execute only synchronization rounds. We refer to this approach as GREEDY and use this as a baseline. While SKIPTRAIN with injected synchronization rounds between training rounds is energy-efficient, energy budgets may still limit the number of training rounds that a node can perform. In SKIPTRAIN-CONSTRAINED, nodes

§Work done while affiliated with CTTC during a research visit at EPFL. This publication has been partially funded by European Union Horizon 2020 research and innovation programme under Grant Agreement No. 953775 (GREENEDGE).

979-8-3503-6460-6/24/$31.00 ©2024 IEEE
DOI 10.1109/IPDPSW63119.2024.00196

perform synchronization rounds just like SKIPTRAIN. However, in a training round, each node independently performs or skips training based on training probabilities (p_i) derived out of its own energy budgets. Specifically, if T is the total number of rounds executed by SKIPTRAIN, the maximum number of training rounds that a node executes is given by: $T_{\text{train}} = \frac{\Gamma_{\text{train}}}{\Gamma_{\text{train}}+\Gamma_{\text{sync}}} T$, where Γ_{train} and Γ_{sync} are the number of consecutive training and synchronization rounds, respectively. We define the training probability of a node i as: $p_i = \min\left(\frac{\tau_i}{T_{\text{train}}}, 1\right)$, where τ_i is the computational budget of i.

III. EVALUATION

A. Experimental setup

Datasets In our experiments, we emulate 256 nodes connected on d-regular topologies, with $d \in \{6, 8, 10\}$. SKIPTRAIN is evaluated on two well-known image classification datasets: CIFAR-10 [9] and FEMNIST [10]. For the first dataset, we consider a 2-shard non-IID data distribution. In FEMNIST, we pick the top-256 clients with the highest number of samples.

Training and Metrics We use Convolutional Neural Network (CNN) architectures adapted from previous work [10]–[12]. These models are trained with stochastic gradient descent (SGD) and the Cross-Entropy loss function. We tuned the learning rate (η) of each model with D-PSGD on a validation set obtained by extracting 50% of the samples from the test set. Hence, the validation and test sets are disjoint. We evaluate the Top-1 accuracy on the validation and test sets, computed every $\Gamma_{\text{train}} + \Gamma_{\text{sync}}$ rounds. We use the validation set to optimize our hyperparameters, including Γ_{train} and Γ_{sync} which are introduced by SKIPTRAIN and the test set to determine model accuracies during all other experiments.

Energy Model Following the same approach of [5], the energy consumption of the training process for a node $i \in [N]$, during a generic iteration t, is the product of the power consumption of the hardware, $P^t_{hw,i}$, and the duration of the task Δ^t_i: $\mathcal{E}^t_i = P^t_{hw,i}\Delta^t_i$. Therefore, the total energy consumption of all the nodes during T rounds is given by: $\mathcal{E} = \sum_{t=1}^{T}\sum_{i=1}^{N}\mathcal{E}^t_i$. In our evaluation, we consider networks consisting of four different smartphones and we derive the energy consumption by the training process for each device type. We first derive the power consumption of each device type from the Burnout benchmark [13]. Then, we obtain the inference time of one data sample for MOBILENET-V2 from the AI benchmark [14]. We scale this inference time with the number of parameters in the model, local steps, and the batch size to get the total inference time. Finally, we compute the training time following the methodology of the FedScale [8], *i.e.*, scale the inference time with the batch size and a multiplier of $3\times$.

Constrained Energy Budgets We obtain the maximum number of training rounds that can be executed by each device (τ_i) as the number of rounds to exhaust a certain percentage of the battery capacity: 10% and 50% for CIFAR-10 and FEMNIST, respectively.

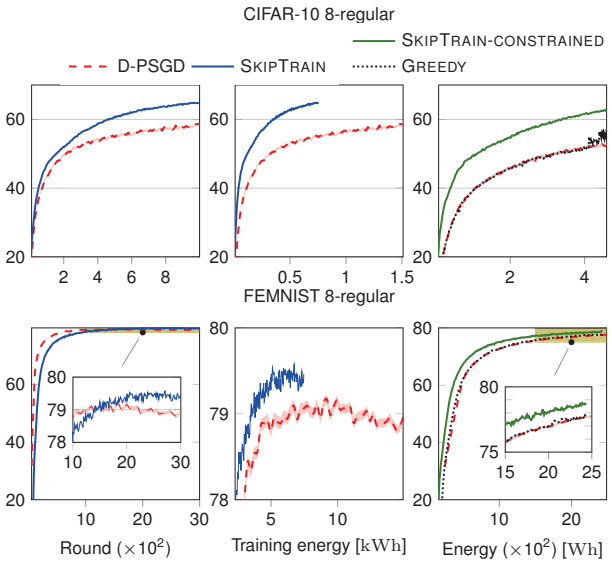

Fig. 1. Comparing SKIPTRAIN to D-PSGD in terms of test accuracy and energy with 8-regular topology (left and middle). We also compare SKIPTRAIN-CONSTRAINED with a greedy baseline (right).

B. Performance

SKIPTRAIN We compare the accuracy and the energy consumption of SKIPTRAIN on a fixed number of total rounds T on the CIFAR-10 and FEMNIST datasets. Figure 1 (left and middle) shows the average test accuracy vs. rounds, and test accuracy vs. training energy consumed for the optimized combination of Γ_{sync} and Γ_{train}. SKIPTRAIN consistently outperforms D-PSGD on CIFAR-10 by reaching on average 6% higher accuracy across all topologies. On the FEMNIST dataset, SKIPTRAIN reaches similar test accuracy values as D-PSGD while significantly reducing the energy consumption. We observe that SKIPTRAIN consumes up to $2\times$ less energy on both datasets as SKIPTRAIN performs half the training rounds.

SKIPTRAIN-CONSTRAINED In Figure 1 (right), we present the test accuracy of each algorithm against the training energy consumed. On the CIFAR-10 dataset, SKIPTRAIN-CONSTRAINED outperforms both D-PSGD and GREEDY by reaching up to 10% and 7% higher accuracies, respectively. On the FEMNIST dataset, the performance gap is smaller, but the trend remains the same.

IV. CONCLUSION

We introduced SKIPTRAIN, a novel DL algorithm devised to reduce energy consumption in decentralized learning environments by selectively substituting training rounds with synchronization rounds. Our next step is to address a main limitations of our approach: the bias towards high-energy-capacity devices. This is a problem because a focus on energy efficiency can inadvertently bias the system towards high-energy-capacity devices due to their more frequent participation in training rounds.

REFERENCES

[1] X. Lian, C. Zhang, H. Zhang, C.-J. Hsieh, W. Zhang, and J. Liu, "Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent," *Advances in neural information processing systems*, vol. 30, 2017.

[2] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized sgd with changing topology and local updates," in *International Conference on Machine Learning*, PMLR, 2020, pp. 5381–5393.

[3] A. Dhasade, N. Dresevic, A.-M. Kermarrec, and R. Pires, "TEE-based decentralized recommender systems: The raw data sharing redemption," in *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, IEEE, 2022, pp. 447–458. DOI: 10.1109/IPDPS53621.2022.00050.

[4] M. De Vos, S. Farhadkhani, R. Guerraoui, A.-M. Kermarrec, R. Pires, and R. Sharma, "Epidemic learning: Boosting decentralized learning with randomized communication," *Advances in Neural Information Processing Systems*, vol. 36, 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/file/7172e147d916eef4cb1eb30016ce725f-Paper-Conference.pdf.

[5] E. Guerra, F. Wilhelmi, M. Miozzo, and P. Dini, "The cost of training machine learning models over distributed data sources," *IEEE Open Journal of the Communications Society*, vol. 4, pp. 1111–1126, 2023. DOI: 10.1109/OJCOMS.2023.3274394.

[6] M. Yu, Y. Tian, B. Ji, C. Wu, H. Rajan, and J. Liu, "Gadget: Online resource optimization for scheduling ring-all-reduce learning jobs," in *IEEE INFOCOM 2022-IEEE Conference on Computer Communications*, IEEE, 2022, pp. 1569–1578.

[7] A. Imteaj, U. Thakker, S. Wang, J. Li, and M. H. Amini, "A survey on federated learning for resource-constrained iot devices," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 1–24, 2021.

[8] F. Lai, Y. Dai, S. Singapuram, J. Liu, X. Zhu, H. Madhyastha, and M. Chowdhury, "Fedscale: Benchmarking model and system performance of federated learning at scale," in *International Conference on Machine Learning*, PMLR, 2022, pp. 11 814–11 827.

[9] A. Krizhevsky, G. Hinton, *et al.*, "Learning multiple layers of features from tiny images," 2009.

[10] S. Caldas, S. M. K. Duddu, P. Wu, T. Li, J. Konečný, H. B. McMahan, V. Smith, and A. Talwalkar, *Leaf: A benchmark for federated settings*, 2019. arXiv: 1812.01097.

[11] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Artificial intelligence and statistics*, PMLR, 2017, pp. 1273–1282.

[12] A. Dhasade, A.-M. Kermarrec, R. Pires, R. Sharma, and M. Vujasinovic, "Decentralized learning made easy with DecentralizePy," in *Proceedings of the 3rd Workshop on Machine Learning and Systems*, 2023, pp. 34–41. DOI: 10.1145/3578356.3592587.

[13] A. Ignatov *et al.*, *Burnout benchmark*, 2022. [Online]. Available: https://burnout-benchmark.com/index.html.

[14] A. Ignatov, R. Timofte, W. Chou, K. Wang, M. Wu, T. Hartley, and L. Van Gool, "Ai benchmark: Running deep neural networks on android smartphones," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, 2018, pp. 0–0.