FedLAM: Low-latency Wireless Federated Learning via Layer-wise Adaptive Modulation

Linping Qu, *Graduate Student Member, IEEE,* Shenghui Song, *Senior Member, IEEE,* and Chi-Ying Tsui, *Senior Member, IEEE*

***Abstract*—In wireless federated learning (FL), the clients need to transmit the high-dimensional deep neural network (DNN) parameters through bandwidth-limited channels, which causes the communication latency issue. In this paper, we propose a layer-wise adaptive modulation scheme to save the communication latency. Unlike existing works which assign the same modulation level for all DNN layers, we consider the layers' importance which provides more freedom to save the latency. The proposed scheme can automatically decide the optimal modulation levels for different DNN layers. Experimental results show that the proposed scheme can save up to 73.9% of communication latency compared with the existing schemes.**

***Index Terms*—Federated learning (FL), latency, modulation**

I. Introduction

Federated learning (FL) is a promising learning framework [1] where only the deep neural network (DNN) parameters rather than the source data will be uploaded to the server, which enhances the data privacy of users. However, since large-size DNN parameters are transmitted through bandwidth-limited wireless channels in every communication round and a number of communication rounds are needed, long latency becomes a critical bottleneck for wireless FL.

Existing works tackle the latency issue mainly from two aspects. From the aspect of learning techniques, fast convergence, model quantization and sparsification are proposed. The authors of [2]–[4] analyzed the convergence rate of FL and proposed methods to reduce the number of communication rounds in FL. The authors of [5]–[7] proposed model sparsification to reduce the number of communicated DNN parameters. The authors of [8]–[10] proposed quantization to reduce the bit-width of every DNN parameters. The reduction of communication rounds and parameter size can help reduce the communication latency. From the aspect of communication techniques, traditional adaptive modulation schemes [11], [12] adjusted the modulation level adaptively, where high modulation level is adopted when the channel has a high signal-to-noise ratio (SNR) and a low modulation level is adopted for a poor-SNR channel. It only considered the channel quality without considering the property of learning. The authors of [13], [14] proposed new adaptive modulation method which considered both learning property and channel quality, but without considering the importance of different DNN layers and just assigned the same modulation level for all DNN layers, which limits the diversity of assigning adaptive modulation levels.

In this paper, we propose a layer-wise adaptive modulation scheme to reduce the latency of wireless FL. For different training stages and channel quality, not only the DNN adopts a optimized modulation level, but also every layer in that DNN can decide its optimal modulation level, which gets more freedom for modulation levels assignment and enhance the saving ratio of latency. The contributions of this paper are concluded as follows:

1) We investigate the importance of different DNN layers and quantify their importance by the eigenvalue of the hessian matrix for the given layer.
2) A modified convergence analysis of FL is introduced by taking the layer importance into the analysis. Based on that, an optimization problem is built whose target is to achieve maximum loss drop under given latency.
3) Each DNN layer can decide its own modulation level, which boosts the freedom of modulation assignment and helps to achieve more latency reduction.

The remainder of this paper is organized as follows. In Section II, we describe the wireless FL system and give the learning performance model and latency model. In Section III, we formulate an optimization problem and solve it. In Section IV, we give our experimental results and discussions. Finally, we conclude our paper in Section V.

II. System Model

In this section, we introduce the wireless FL system, and describe the learning performance model and latency model.

A. Wireless Federated Learning

As depicted in Fig. 1, the FL system contains one remote server and multiple clients. Basically, FL solves the following optimization problem [15]:

$$\min_{\mathbf{w}\in\mathbb{R}^D} f(\mathbf{w}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{w}), \tag{1}$$

where $\mathbf{w}$ represents the DNN model with D parameters in total, n denotes the number of clients, and $f_i(\mathbf{w})$ denotes the local loss function of the i-th client.

In wireless FL, both the uplink and downlink communication channels should be noisy, which introduces communication error to model parameters. In the r-th communication round, the global model at the server, denoted as $\mathbf{w}_r$, will be broadcast to the clients. After receiving the global model,

Linping Qu, Shenghui Song, and Chi-Ying Tsui are with the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong (e-mail: lqu@connect.ust.hk; eeshsong@ust.hk; eetsui@ust.hk).

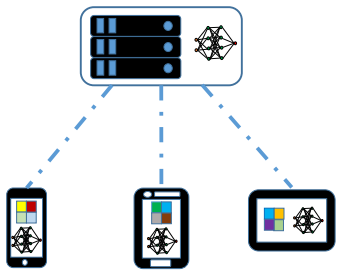


Fig. 1: FL over noisy wireless channels.

clients will perform E steps of local training based on their local dataset. The local updating process can be modeled as

$$\mathbf{w}_{r,e+1}^i = \mathbf{w}_{r,e}^i - \eta \hat{\nabla} f_i(\mathbf{w}_{r,e}^i), e = 0, \cdots, E-1 \quad (2)$$

where $\hat{\nabla} f_i(\mathbf{w}_{r,e}^i)$ represents the stochastic gradient computed from the local mini-batch datasets. And the first step of local training is based on the received noisy global model, which means

$$\mathbf{w}_{r,0}^i = \tilde{\mathbf{w}}_r, \quad (3)$$

where $\tilde{\mathbf{w}}_r$ denotes the model with communication noise. Upon completing E steps of local training, the i-th client obtains the local model update as

$$\Delta \mathbf{w}_r^i = \mathbf{w}_{r,E}^i - \mathbf{w}_{r,0}^i. \quad (4)$$

Then, the server aggregates the noisy local model updates by

$$\mathbf{w}_{r+1} = \mathbf{w}_r + \frac{1}{n} \sum_{i=1}^{n} (\tilde{\Delta} \mathbf{w}_r^i). \quad (5)$$

Above process iterates until FL converges.

B. Learning Performance Model

Due to the fact that the server can support high transmitting power, in this paper, we assume the downlink has little communication error and only focus on the error in the uplink channels. For FL with uplink communication bit errors, from [16], the global training loss drop of two adjacent communication rounds can be formulated as

$$\begin{aligned} \mathbb{E}f(\mathbf{w}_r) - \mathbb{E}f(\mathbf{w}_{r+1}) \geq & \frac{\eta}{2} \sum_{t=0}^{\tau-1} \mathbb{E}\|\nabla f(\bar{\mathbf{w}}_{r,t})\|^2 \\ & - \frac{L}{2n^2} \sum_{i\in[n]} \mathbb{E}\|\tilde{\Delta} \mathbf{w}_r^i - \Delta \mathbf{w}_r^i\|^2 \\ & - \frac{L^2(n+1)\tau(\tau-1)\eta^3\sigma^2}{2n} - \frac{L\tau\eta^2\sigma^2}{2n}. \end{aligned} \quad (6)$$

Summing (6) over communication round $r = 0, ..., R-1$, we get the training loss drop for the total learning process:

$$\begin{aligned} \mathbb{E}f(\mathbf{w}_0) - \mathbb{E}f(\mathbf{w}_R) \geq & \frac{\eta}{2} \sum_{r=0}^{R-1} \sum_{t=0}^{\tau-1} \mathbb{E}\|\nabla f(\bar{\mathbf{w}}_{r,t})\|^2 \\ & - \frac{L}{2n^2} \sum_{r=0}^{R-1} \sum_{i\in[n]} \mathbb{E}\|\tilde{\Delta} \mathbf{w}_r^i - \Delta \mathbf{w}_r^i\|^2 \\ & - \frac{L^2(n+1)\tau(\tau-1)\eta^3\sigma^2 R}{2n} - \frac{L\tau\eta^2\sigma^2 R}{2n}. \end{aligned} \quad (7)$$

Above learning convergence assumes every DNN layer the same importance. However, in this paper, we consider the layers' different importance. As widely used in [17]–[19], the top eigenvalue of the layer's hessian matrix can be used to quantify the DNN layers' importance. Using H_k to denote the importance of the k-th layer, (7) can be updated into

$$\begin{aligned} \mathbb{E}f(\mathbf{w}_0) - \mathbb{E}f(\mathbf{w}_R) \geq & \frac{\eta}{2} \sum_{r=0}^{R-1} \sum_{t=0}^{\tau-1} \mathbb{E}\|\nabla f(\bar{\mathbf{w}}_{r,t})\|^2 \\ & - \frac{L}{2n^2} \sum_{r=0}^{R-1} \sum_{i\in[n]} \sum_{k\in l} \frac{H_k}{\sum_{k=1}^{l} H_k} \mathbb{E}\|\tilde{\Delta} \mathbf{w}_r^{i,k} - \Delta \mathbf{w}_r^{i,k}\|^2 \\ & - \frac{L^2(n+1)\tau(\tau-1)\eta^3\sigma^2 R}{2n} - \frac{L\tau\eta^2\sigma^2 R}{2n}. \end{aligned} \quad (8)$$

The second term on the RHS of (8) captures the square error of model updates which is a function of BER as shown in [20]. In this work, the BER is impacted by the modulation levels and their relationship can be formulated as [21]

$$b \cong \frac{2}{max(log_2 M, 2)} \cdot \sum_{i=1}^{max(M/4,1)} Q\left(\sqrt{\frac{2E_s}{N_0}} sin\frac{(2i-1)\pi}{M}\right). \quad (9)$$

The authors in [13] only consider the limited cases of (9), i.e., when $M > 4$ and $E_s/N_0 \gg 1$. But in this paper, we consider the general cases without special requirement on modulation levels and $E_s/N0$.

Based on above relationships, we can get the relationship between learning performance and modulation levels.

C. Latency Model

The latency of wireless FL comprises three parts: downlink broadcasting, local computing, and uplink transmission. Assuming BPSK is used in downlink broadcasting, the latency can be calculated as

$$T_d = \frac{DN}{2B_d}, \quad (12)$$

where N denotes the bit-length of each DNN parameter, and B_d denotes downlink bandwidth. Assuming OFDMA technique is used in uplink communication and each client is assigned with a bandwidth of B_u, the latency of one client for uplink transmission can be obtained as

$$T_u = \sum_{k=1}^{l} \frac{D_k N}{2B_u log_2(M^k)}, \quad (13)$$

where l denotes the DNN layers and D_k denotes the parameter size of the k-th layer. And the local computing latency can be calculated as

$$T_c = \frac{VC}{f}, \quad (14)$$

where V represents the volume of the local training data, C denotes the required operation cycles for processing one training data, and f denotes the frequency of the computing unit.

For the entire training which consumes R communication

$$\max_{M_r^{i,k}} \frac{\mathbb{E}f(\mathbf{w}_0) - \mathbb{E}f(\mathbf{w}_R)}{T} = \frac{\frac{\eta}{2}\sum_{r=0}^{R-1}\sum_{t=0}^{\tau-1}\mathbb{E}\|\nabla f(\bar{\mathbf{w}}_{r,t})\|^2 - \frac{L}{2n^2}\sum_{r=0}^{R-1}\sum_{i\in[n]}\sum_{k\in l}\frac{H_k}{\sum_{k=1}^{l}H_k}\mathbb{E}\|\tilde{\Delta}\mathbf{w}_r^{i,k} - \Delta\mathbf{w}_r^{i,k}\|^2 - \frac{L^2(n+1)\tau(\tau-1)\eta^3\sigma^2 R}{2n} - \frac{L\tau\eta^2\sigma^2 R}{2n}}{\frac{DNR}{2B_d} + \frac{VCR}{f} + \sum_{r=0}^{R-1}\sum_{k=1}^{l}\frac{D_k N}{2B_u log_2 M_r^k}}. \tag{10}$$

$$\max_{M_r^{i,k}} \frac{f(\mathbf{w}_r) - \mathbb{E}f(\mathbf{w}_{r+1})}{T_r} = \frac{\frac{\eta}{2}\sum_{t=0}^{\tau-1}\mathbb{E}\|\nabla f(\mathbf{w}_{r,t})\|^2 - \frac{L}{2n}\sum_{k\in l}\frac{H_k}{\sum_{k=1}^{l}H_k}\mathbb{E}\|\tilde{\Delta}\mathbf{w}_r^{i,k} - \Delta\mathbf{w}_r^{i,k}\|^2 - \frac{L^2(n+1)\tau(\tau-1)\eta^3\sigma^2}{2n} - \frac{L\tau\eta^2\sigma^2}{2n}}{\frac{DN}{2B_d} + \frac{VC}{f} + \sum_{k=1}^{l}\frac{D_k N}{2B_u log_2 M_r^k}}. \tag{11}$$

rounds, the total latency is

$$T = \sum_{r=0}^{R-1} T_r = \sum_{r=0}^{R-1}\left(\frac{DN}{2B_d} + \frac{VC}{f} + \sum_{k=1}^{l}\frac{D_k N}{2B_u log_2 M_r^k}\right) = \frac{DNR}{2B_d} + \frac{VCR}{f} + \sum_{r=0}^{R-1}\sum_{k=1}^{l}\frac{D_k N}{2B_u log_2 M_r^k}. \tag{15}$$

III. Problem Formulation and Solution

In this section, we will formulate the latency issue of FL into an optimization problem and then solve it.

A. Problem Formulation

We define the objective function of low-latency FL as (10), where the numerator captures the loss drop and the denominator denotes the total time cost of R communication rounds. The meaning of (10) is to gain the biggest loss drop at the given time cost. With high modulation level, the time cost in the denominator can be reduced, but the loss drop in the numerator will also be reduced, which shows the trade-off between latency and learning performance.

However, (10) is just an ideal objective function because whose solution relies on the overall information of all clients and of all communication rounds. In practice, each client needs to make real-time decision without knowing mutual and future information. So we modify (10) into (11) to make it practical, which only requires the information for a given client in the current communication round.

B. Problem Solution

First, to quantify the DNN layers' importance, the power iteration method [22] can be used to simplify the calculation of top eigenvalue.

Then, given the objective function of (11), by choosing different modulation levels for different DNN layers, we can get different value of (11). Considering the fact that the modulation levels are discrete integers, like [2, 4, 8, 16]-PSK for example, we can find the optimal solution of (11) by enumeration.

For l- layers DNN model, each layer can be assigned with the provided 4 different PSK modulation schemes and the searching space is 4^l. When l is large, it will take noticeable time to find the optimal solution. To tackle this issue, we further propose a group method for very deep-layers DNN. The layers having similar importance score will be assigned with a same modulation level. Take the 20-layers DNN for example, the original searching space is 4^{20} which is too huge to find the optimal solution. But if the layers are divided into 5 groups based on their importance, the searching space will be reduced into 4^5 which is easy to solve.

IV. Experiments and Discussions

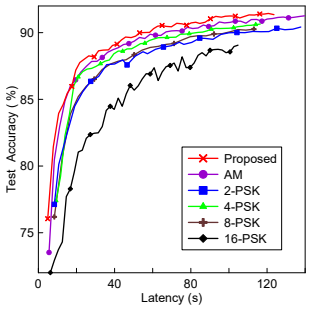
In this section, we conduct experiments to verify the performance of the proposed layer-wise adaptive modulation scheme.

A. Experiment Settings

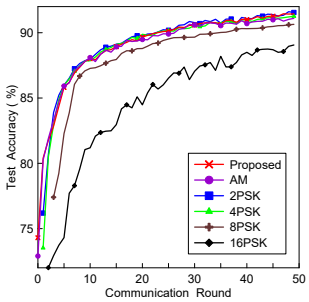
The communication settings are similar to the "AM" [14]. And we considered three sets of FL tasks: MNIST [23] dataset classified by LeNet-300-100 [24], Fashion-MNIST [25] dataset classified by vanilla CNN [26], CIFAR-10 [27] dataset classified by ResNet-20 [28]. The datasets were split in an identical and independently distributed (i.i.d) manner. The local training was configured with a learning rate of 0.01, using the SGD optimizer. The number of local training steps was set to 5.

B. Results and Analysis

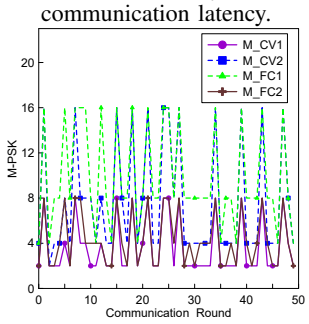
Fig. 2 shows the experimental results of Fashion-MNIST. The results of MNIST is similar, so we omit its figures due to page limit. Fig. 2(a) shows that the proposed scheme consumes least latency for achieving the same test accuracy since it has more freedom to change the modulation for each layer. Fig. 2(b) shows how learning accuracy improves with communication rounds. Fig. 2 (c) shows how the modulation levels of every layer changes during training. It can be observed that the first and last layers (CV1 and FC2) generally adopt higher modulation levels than the middle layers (CV2, FC1). This can be explained that the first and last layer are the more important layers which determine the input feature and output result. So low modulation levels are preferred to avoid errors and guarantee the correctness of input and output. And



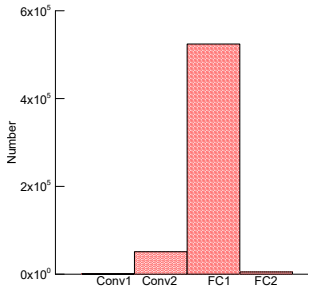
(a) Test accuracy vs. communication latency.



(b) Test accuracy vs. communication round.

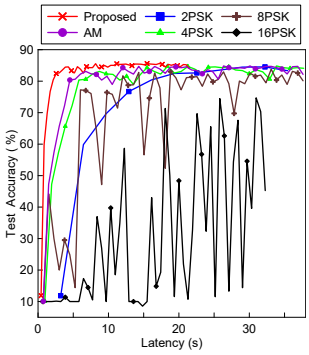


(c) Modulation level of each layer changes in training.

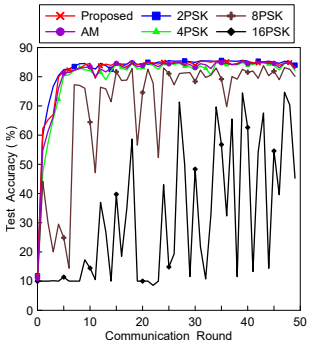


(d) Parameter size of each layer in the DNN model.

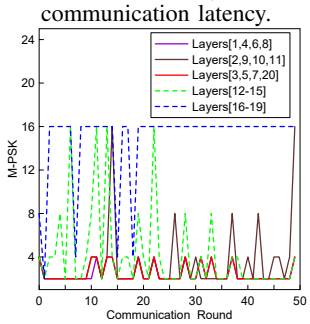
Fig. 2: Experiment of Fashion-MNIST.



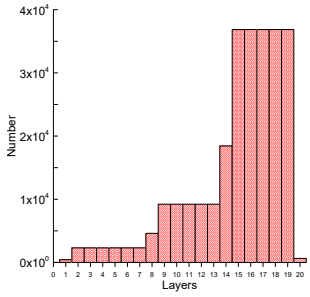
(a) Test accuracy vs. communication latency.



(b) Test accuracy vs. communication round.



(c) Modulation level of each layer changes in training.



(d) Parameter size of each layer in the DNN model.

Fig. 3: Experiment of CIFAR-10.

from Fig. 2(d), the parameter size of the first or the last layer is very small compared with the middle two layers, so the low modulation levels in the first and last layer introduces negligible latency.

In the CIFAR-10 experiment as shown in Fig. 3, the layers for ResNet-20 is 20. So the layers group method is adopted to divide 20 layers into 5 groups. The first eight layers and the last layer are assigned with low modulation levels due to their importance and small size. The middle layers, especially for the [16-19] layers that have large parameter size are assigned with high modulation levels for saving latency purpose.

Concluding all the three experiments, the first and the last layer with higher importance and smaller parameter size are preferred to be assigned with low modulation levels. The middle layers generally with less importance and larger parameter size are suggested to assign higher modulation levels.

TABLE I: Summary of Experimental Results.

	MNIST Acc.=97.8%	Fashion-MNIST Acc.=90.6%	CIFAR-10 Acc.=83.1%
2PSK	46.4s	137.7s	25.8s
4PSK	39.3s	114.1s	21.3s
8PSK	37.3s	117.3s	36.1s
AM [14]	28.8s	94.9s	14.3s
Proposed	21.6s	72.6s	3.5s
Saving	25%	23.5%	73.9%

The detailed data for performance comparison is summarized in Table. I. The comparison is based on the latency for achieving a given test accuracy. Since the test accuracy of "16PSK" is too bad, there is no need to compare with it and thus not listed in the table. The proposed scheme is compared with "2PSK", "4PSK", "8PSK", and "AM". The saving ratio is for the proposed scheme comparing with the "AM" scheme.

C. The Cost of Computing Hessian Eigenvalue

The proposed layer-wise modulation scheme needs to calculate the hessian matrix of every layer, which introduces extra latency. The hessian value calculation is equivalent to one further step of differential on gradients [29]. So the extra overhead of hessian calculation is same as one extra step of back propagation. Due to the advanced AI hardware like GPU and AI chips, the latency bottleneck is more on communication side rather than computation side [30]. So the one step of back propagation will not introduce noticeable latency compared with the overall latency.

V. Conclusion

In this paper, we investigated the latency issue of wireless FL. By theoretically analyzing the impact of modulation on learning performance and latency, we formulated an optimization problem and proposed the DNN layer-wise adaptive modulation scheme. The proposed scheme can adjust the uplink modulation levels for every DNN layer individually, which can achieve a good trade-off between learning performance and latency.

REFERENCES

[1] H. B. McMahan, F. Yu, P. Richtarik, A. Suresh, D. Bacon, et al., "Federated learning: Strategies for improving communication efficiency," in *Proceedings of the 29th Conference on Neural Information Processing Systems (NIPS), Barcelona, Spain*, 2016, pp. 5–10.

[2] H. Wu and P. Wang, "Fast-convergent federated learning with adaptive weighting," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1078–1088, 2021.

[3] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. V. Poor, "Fast-convergent federated learning," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 1, pp. 201–218, 2020.

[4] M. van Dijk, N. V. Nguyen, T. N. Nguyen, L. M. Nguyen, Q. Tran-Dinh, and P. H. Nguyen, "Asynchronous federated learning with reduced number of rounds and with differential privacy from less aggregated gaussian noise," *arXiv preprint arXiv:2007.09208*, 2020.

[5] B. Wang, J. Fang, H. Li, and B. Zeng, "Communication-efficient federated learning: A variance-reduced stochastic approach with adaptive sparsification," *IEEE Transactions on Signal Processing*, 2023.

[6] W. Luping, W. Wei, and L. Bo, "Cmfl: Mitigating communication overhead for federated learning," in *2019 IEEE 39th international conference on distributed computing systems (ICDCS)*. IEEE, 2019, pp. 954–964.

[7] S. Li, Q. Qi, J. Wang, H. Sun, Y. Li, and F. R. Yu, "Ggs: General gradient sparsification for federated learning in edge computing," in *ICC 2020-2020 IEEE International Conference on Communications (ICC)*. IEEE, 2020, pp. 1–7.

[8] Y. Wang, Y. Xu, Q. Shi, and T.-H. Chang, "Quantized federated learning under transmission delay and outage constraints," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 1, pp. 323–341, 2021.

[9] M. Lan, Q. Ling, S. Xiao, and W. Zhang, "Quantization bits allocation for wireless federated learning," *IEEE Transactions on Wireless Communications*, vol. 22, no. 11, pp. 8336–8351, 2023.

[10] L. Qu, S. Song, and C.-Y. Tsui, "Feddq: Communication-efficient federated learning with descending quantization," in *GLOBECOM 2022-2022 IEEE Global Communications Conference*. IEEE, 2022, pp. 281–286.

[11] B. S. K. Reddy and B. Lakshmi, "Adaptive modulation and coding with channel state information in ofdm for wimax," *IJ Image, Graphics and Signal Processing*, vol. 1, pp. 61–69, 2015.

[12] S. Hadi and T. Tiong, "Adaptive modulation and coding for lte wireless communication," in *IOP Conference Series: Materials Science and Engineering*. IOP Publishing, 2015, vol. 78, p. 012016.

[13] X. Xu, G. Yu, and S. Liu, "Adaptive modulation for wireless federated learning," in *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE, 2021, pp. 1–6.

[14] X. Xu, G. Yu, and S. Liu, "Adaptive modulation for wireless federated edge learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, no. 4, pp. 1096–1109, 2023.

[15] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," *Proceedings of Machine Learning and Systems*, vol. 2, pp. 429–450, 2020.

[16] L. Qu, S. Song, C.-Y. Tsui, and Y. Mao, "How robust is federated learning to communication error? a comparison study between uplink and downlink channels," in *2024 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2024, pp. 1–6.

[17] Z. Dong, Z. Yao, A. Gholami, M. W. Mahoney, and K. Keutzer, "Hawq: Hessian aware quantization of neural networks with mixed-precision," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 293–302.

[18] Z. Dong, Z. Yao, D. Arfeen, A. Gholami, M. W. Mahoney, and K. Keutzer, "Hawq-v2: Hessian aware trace-weighted quantization of neural networks," *Advances in neural information processing systems*, vol. 33, pp. 18518–18529, 2020.

[19] S. Yu, Z. Yao, A. Gholami, Z. Dong, S. Kim, M. W. Mahoney, and K. Keutzer, "Hessian-aware pruning and optimal neural implant," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 3880–3891.

[20] L. Qu, Y. Mao, S. Song, and C.-Y. Tsui, "Energy-efficient channel decoding for wireless federated learning: Convergence analysis and adaptive design," *IEEE Transactions on Wireless Communications*, 2024.

[21] J. Lu, K. B. Letaief, J.-I. Chuang, and M. L. Liou, "M-PSK and M-QAM BER computation using signal-space concepts," *IEEE Transactions on communications*, vol. 47, no. 2, pp. 181–184, 1999.

[22] Z. Yao, A. Gholami, K. Keutzer, and M. W. Mahoney, "Pyhessian: Neural networks through the lens of the hessian," in *2020 IEEE international conference on big data (Big data)*. IEEE, 2020, pp. 581–590.

[23] Y. LeCun, "The MNIST database of handwritten digits," *[Online]. Available: http://yann. lecun. com/exdb/mnist/*, 1998.

[24] S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *arXiv preprint arXiv:1510.00149*, 2015.

[25] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017.

[26] 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.

[27] A. Krizhevsky, "Learning multiple layers of features from tiny images," Tech. Rep., 2009.

[28] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[29] Z. Yan, X. S. Hu, and Y. Shi, "Swim: Selective write-verify for computing-in-memory neural accelerators," in *Proceedings of the 59th ACM/IEEE Design Automation Conference*, 2022, pp. 277–282.

[30] X. Yao, C. Huang, and L. Sun, "Two-stream federated learning: Reduce the communication costs," in *2018 IEEE Visual Communications and Image Processing (VCIP)*. IEEE, 2018, pp. 1–4.