OVER-THE-AIR PERSONALIZED FEDERATED LEARNING

Hasin Us Sami

ECE Department
University of California, Riverside
hsami003@ucr.edu

Başak Güler

ECE Department
University of California, Riverside
bguler@ece.ucr.edu

ABSTRACT

Federated learning is a distributed framework for training a machine learning model over the data stored by wireless devices. A major challenge in doing so is the communication overhead from the devices to the server. Over-the-air federated learning is a recent framework to address this challenge, which utilizes the superposition property of the wireless multiple access channel to enable computations to be performed in the wireless medium. Current over-the-air aggregation frameworks, on the other hand, train a single model for all users, which can degrade performance in heterogeneous environments where the data distributions of the users can differ from one another. This work presents a personalized over-the-air federated learning framework towards addressing this challenge. Our experiments demonstrate significant performance improvement in terms of the test accuracy over conventional federated learning.

***Index Terms*—** Over-the-air machine learning, distributed training, personalized federated learning

1. INTRODUCTION

Federated learning (FL) is a recent distributed learning framework to train machine learning models over the massively distributed data in wireless networks [1, 2]. Training in conventional FL is an iterative process coordinated by a server who maintains a global model. At each iteration, the server sends the current state of the global model to the wireless devices (users), who then update the global model by training on their local dataset. The local models are then aggregated by the server to update the global model.

A major challenge in FL is the communication overhead of sending the local models from the users to the server, where the number of users and the model parameters can reach millions [3]. Over-the-air FL has emerged as a recent framework towards addressing this challenge, by utilizing the waveform superposition property of a multi-access channel to perform over-the-air aggregation of local model parameters, which can reduce the communication overhead by a factor of the total number of users [4, 5, 6]. In contrast to conventional FL where the server has to first reconstruct the local models to perform the aggregation, over-the-air FL allows the local models to be aggregated directly in the channel. Different user scheduling policies based on channel conditions and model updates can further increase the communication and computational efficiency of the system [7]. To ensure proper synchronization among the users, a truncated channel inversion policy is proposed in [8, 9]. An optimal power allocation strategy is proposed in [10] to further enhance the learning performance. Reference [11] utilizes noise perturbation in over-the-air aggregation to improve privacy.

In most real-life FL scenarios, the dataset distributions of the users vary from one another, in which case training a single model can lead to severe performance degradation for the individual users [12, 13, 14]. Specifically, when the local datasets of the users are non-i.i.d. (where i.i.d. stands for independent identically distributed), the model tends to favor some of the users while heavily degrading the performance of others [13]. Personalized FL is a recent framework to address this data heterogeneity challenge in FL systems [15, 16, 17, 18, 19, 12, 20]. To this end, [21] introduces a proximal term in the local objective functions to minimize the divergence among local updates, leading to higher accuracy than conventional FL. Reference [22] incorporates a hierarchical clustering algorithm to cluster the users based on the similarity of their local updates. Users in each cluster perform FL independently, which has been shown to improve accuracy in non-iid settings. References [18, 19] leverage cosine similarity between the gradient updates of each pair of users for clustering, to determine the proximity of local optimizations. Reference [23] proposes an adaptive clustering technique where the number of clusters vary with time based on the observed statistics. Reference [12] and [20] develop efficient clustering techniques by training multiple models for different groups of users, and assigning the users into the groups based on which group model outputs the minimum loss over the users' local data.

In this paper, we propose a personalized over-the-air federated learning scheme. Our approach is motivated by clustered FL for personalization [12, 20], where N users are grouped into K clusters, and a designated model is trained for each cluster. We then propose a MIMO encoder-decoder design where the encoding operation *aligns* the transmitted waveforms for the local models belonging to the users in the same cluster. The encoding operation also ensures that the *aggregate* of the local models for each cluster can be decoded by the server. While the proposed approach relies solely on the space dimensions and does not involve explicit frequency partitioning between the users, our approach can also be combined with a frequency partitioning scheme to increase the number of users sharing each subband. To evaluate the performance of the proposed approach, we perform experiments for image classification on the MNIST and CIFAR-10 datasets [24, 25] under heterogeneous (non-iid) data distributions for the individual users. Our experiments demonstrate that the proposed approach can significantly improve the performance of over-the-air FL in terms of the test accuracy of individual users.

Our specific contributions are as follows:

1. We propose the first over-the-air personalized FL framework, which can significantly reduce the communication latency of personalized FL by allowing all users to share the same spectrum band for transmission.

Research was sponsored by the OUSD (R&E)/RT&L and was accomplished under Cooperative Agreement Number W911NF-20-2-0267. The views and conclusions are those of the authors.

2. We develop a MIMO encoding-decoding design for personalized FL, which aligns the local models received from users belonging to different clusters in different subspaces over-the-air, and enables the server to recover the aggregate of the local models for each cluster.
3. We evaluate the performance of over-the-air aggregation in terms of test accuracy of the individual users under the non-iid data distribution setting and demonstrate that personalized over-the-air aggregation can significantly outperform the performance compared to conventional (non-personalized) over-the-air aggregation.

2. SYSTEM MODEL

We consider a network with N wireless users and a single server. User $i \in [N]$ has a local dataset $\mathcal{D}_i$ consisting of $|\mathcal{D}_i| := D_i$ data points. The users are heterogeneous in terms of their dataset distribution, where the local dataset of each user is realized from a class of K distributions $p_1, \ldots, p_K$. The number of users whose local dataset is distributed according to p_k is given by N_k, where $\sum_{k\in[K]} N_k = N$.

In this work, we consider a personalized FL approach known as clustered federated learning [12, 20], where users are partitioned into K clusters and a global model $\mathbf{w}^k$ is trained for each cluster $k \in [K]$. The goal is to find the optimal model parameters $\{\mathbf{w}^k\}_{k\in[K]}$ to minimize a loss function,

$$\min_{\mathbf{w}^1,\ldots,\mathbf{w}^K} \sum_{i\in[N]} \alpha_i \min_{k\in[K]} F_i(\mathbf{w}^k), \tag{1}$$

where $F_i(\mathbf{w}^k)$ is the loss function of user $i \in [N]$ evaluated over the local dataset $\mathcal{D}_i$, and α_i is a weight parameter assigned to user i, often set as $\alpha_i = \frac{D_i}{D}$ where $D = \sum_{i\in[N]} D_i$.

Training is carried out through an iterative process, where the estimate of model $\mathbf{w}^k$ at training round t is denoted by $\mathbf{w}^k(t)$. At each training round, the server broadcasts the current state of the K global models $\{\mathbf{w}^k(t)\}_{k\in[K]}$ to the users. Then, user i locally computes the loss of each $\{\mathbf{w}^k(t)\}_{k\in[K]}$ on its local dataset $\mathcal{D}_i$, and selects the cluster that minimizes the local loss, which is denoted as,

$$c_{it} := \arg\min_{k\in[K]} F_i(\mathbf{w}^k(t)) \tag{2}$$

and then locally updates the model $\mathbf{w}^{c_{it}}(t)$ through multiple stochastic gradient descent (SGD) steps, creating a local model $\mathbf{w}_i(t)$ and sends its local model to the server.

Finally, the server aggregates the local models from the users in each cluster, to update the global models for the next training round,

$$\mathbf{w}^k(t+1) = \sum_{i\in\mathcal{S}_k^t} \alpha_i(t)\mathbf{w}_i(t) \tag{3}$$

where

$$\mathcal{S}_k^t := \{i : c_{it} = k \text{ and } i \in [N]\} \tag{4}$$

represents the set of users assigned to cluster k at training round t, according to (2) and $\alpha_i(t) \triangleq \frac{D_i}{\sum_{j\in\mathcal{S}_{c_{it}}^t} D_j}$ is the weight parameter of user i at round t.

Note that the aggregation operation in (3) requires the server to sum the local models corresponding to each cluster of users separately. As such, a naive application of over-the-air aggregation, in which all users would send their model parameters into the wireless channel, would lead to the aggregation of local models belonging to users from different clusters in the wireless medium. Accordingly, the server would receive the sum of all user models, and would not be able to distinguish the aggregate of the local models belonging to different clusters.

In the following, we introduce an over-the-air personalized FL framework to address this challenge, which enables all users to share the spectrum while ensuring that the server can recover the aggregate of the local models belonging to different clusters. For ease of exposition, we assume that all the users have equal-sized datasets in the sequel, i.e., $D_i = \frac{D}{N}$ for all $i \in [N]$, for which (3) becomes,

$$\mathbf{w}^k(t+1) = \frac{1}{|\mathcal{S}_k^{(t)}|} \sum_{i\in\mathcal{S}_k^t} \mathbf{w}_i(t) \tag{5}$$

noting that our approach can be generalized without loss of generality when dataset sizes are different across the users, by letting each user scale their local model according to $\alpha_i(t)$ before sending it to the server.

3. OVER-THE-AIR PERSONALIZED FEDERATED LEARNING

We consider a MIMO transmission model where each user is equipped with N_T transmitter antennas. In addition, we consider an access point, integrated with the server, equipped with N_R receiver antennas. We represent the channel parameters from user i to the access point with an $N_R \times N_T$ matrix $\mathbf{H}_i(t)$ at round t. We consider a Rayleigh channel where each element of $\mathbf{H}_i(t)$ for $i \in [N]$ is i.i.d. from a complex Gaussian distribution $\mathcal{CN}(0, \sigma^2)$. The channel varies from one training round to another.

For each user i, we define an $N_T \times d$ dimensional encoding matrix $\mathbf{V}_{i,k}(t)$ for $k \in [K]$ at training round t. Given $c_{it} \in [K]$ from (2), which denotes the cluster user i is assigned to at round t, user i encodes its local model $\mathbf{w}_i(t)$ by using the encoder $\mathbf{V}_{i,c_{it}}(t)$ and transmits the encoded model $\mathbf{V}_{i,c_{it}}(t)\mathbf{w}_i(t)$ to the channel. Due to the superposition property of the wireless channel, the received signal at the access point is the summation of the signals transmitted from all users. We denote the received signal at the access point with an $N_R \times 1$ vector:

$$\mathbf{y}(t) = \sum_{i\in[N]} \mathbf{H}_i(t)\mathbf{V}_{i,c_{it}}(t)\mathbf{w}_i(t) + \mathbf{n}(t) \tag{6}$$

at round t, which can also be written as,

$$\mathbf{y}(t) = \sum_{k\in[K]} \sum_{i\in\mathcal{S}_k^t} \mathbf{H}_i(t)\mathbf{V}_{i,k}(t)\mathbf{w}_i(t) + \mathbf{n}(t) \tag{7}$$

where $\mathbf{n}(t)$ represents the noise vector consisting of independent zero mean Gaussian random variables with $\mathbb{E}[\mathbf{n}(t)\mathbf{n}(t)^T] = \mathbf{I}$.

Upon receiving (7), the server decodes the aggregate of the local models for each cluster to update the global models as shown in (5). To do so, we define a $d \times N_R$ decoding matrix $\mathbf{U}_k$ for each cluster $k \in [K]$. Then, the decoding operation for cluster k is given by,

$$\widehat{\mathbf{z}}_k(t) = \mathbf{U}_k\mathbf{y}(t) \tag{8}$$

where, $\widehat{\mathbf{z}}_k(t)$ is the estimate of the aggregate of the local models assigned to cluster k, denoted by,

$$\mathbf{z}_k(t) \triangleq \sum_{i\in\mathcal{S}_k^t} \mathbf{w}_i(t). \tag{9}$$

Algorithm 1 Over-the-Air Personalized Federated Learning

for each cluster $k \in [K]$ **do**
  Initialize $\mathbf{w}^k(0)$ ▷ $\mathbf{w}^k(t)$ is the global model of cluster k at round t
for each round $t = 1, 2, ..., T$ **do**
  for each client $i \in [N]$ in parallel **do**
    for $k \in [K]$ **do**
      compute $F_i(\mathbf{w}^k(t))$ ▷ loss function of user i on global model of cluster k
    cluster estimate $c_{it} = \text{argmin}_{k\in[K]} F_i\left(\mathbf{w}^k(t)\right)$
    $\mathbf{w}_i(t+1) \leftarrow \text{CLIENTUPDATE}(i, \mathbf{w}^{c_{it}}(t))$
    encoding $\rightarrow \mathbf{V}_{i,c_{it}}(t)\mathbf{w}_i(t)$
  Over-the-Air Aggregation:
  $\mathbf{y}(t) = \sum_{k\in[K]} \mathbf{A}_k \mathbf{z}_k(t)$ ▷ Equation (13)
  Server executes:
  for $k \in K$ **do**
    Decode cluster aggregate $\rightarrow \widehat{\mathbf{z}}_k(t) = \mathbf{U}_k \mathbf{y}(t)$
    Update cluster global model $\rightarrow \mathbf{w}^k(t+1) = \frac{1}{|\mathcal{S}_k(t)|}\widehat{\mathbf{z}}_k(t)$
function CLIENTUPDATE($u, \mathbf{w}$)
  for $i = 1, ..., E$ **do** ▷ E is the number of local updates
    for uniformly random selected data sample $\zeta_i \in \mathcal{D}_i$ **do**
      $\mathbf{w} \leftarrow \mathbf{w} - \eta\nabla F(\mathbf{w}; \zeta_i)$ ▷ η is the learning rate
  return $\mathbf{w}$

The key intuition behind our encoding-decoding scheme is to *align* the models received from the same cluster in a way that enables the server to recover the aggregate of the local models for each cluster, using the received signal which corresponds to the sum of the signals received from all users. To do so, we define,

$$\mathbf{A}_k = \mathbf{H}_i(t)\mathbf{V}_{i,k}(t) \quad \forall i \in \mathcal{S}_k^t \tag{10}$$

where $\mathbf{A}_k$ for $k \in [K]$ is an $N_R \times d$ matrix, where each element is generated i.i.d. according to a Gaussian distribution $\mathcal{N}(0, \beta^2)$. Parameter β controls the average transmit power. The system of linear equations in (10) has a solution if $N_T \geq N_R$. Then, we define the encoder of user $i \in \mathcal{S}_k^t$ as,

$$\mathbf{V}_{i,k} = \mathbf{H}_i^{\dagger}(t)\mathbf{A}_k \tag{11}$$

where $\mathbf{H}_i^{\dagger}(t) = \mathbf{H}_i^H(\mathbf{H}_i(t)\mathbf{H}_i^H(t))^{-1}$ is the pseudo-inverse of $\mathbf{H}_i(t)$, and $\mathbf{H}_i^H(t)$ is the Hermitian transpose of $\mathbf{H}_i(t)$, respectively. When $N_T < N_R$, a solution to (10) does not exist, in which case one can use a least-squares approach to jointly optimize $\mathbf{A}_k$ and $\mathbf{V}_{i,k}$, which we leave as future work. Using (10), equation (7) can be written as,

$$\mathbf{y}(t) = \sum_{k\in[K]} \mathbf{A}_k \left(\sum_{i\in\mathcal{S}_k^t} \mathbf{w}_i(t) \right) + \mathbf{n}(t) \tag{12}$$

$$= \sum_{k\in[K]} \mathbf{A}_k \mathbf{z}_k(t) + \mathbf{n}(t), \tag{13}$$

and similarly (8) is given by,

$$\widehat{\mathbf{z}}_k(t) = \mathbf{U}_k \sum_{k\in[K]} \mathbf{A}_k \mathbf{z}_k(t) + \mathbf{U}_k \mathbf{n}(t). \tag{14}$$

For the design of the decoder, we rewrite the first term in (13) as,

$$\sum_{k\in[K]} \mathbf{A}_k \mathbf{z}_k = \mathbf{A}_{\bar{k}} \mathbf{z}_{\bar{k}}(t) + \mathbf{A}_k \mathbf{z}_k(t) \tag{15}$$

where $\mathbf{A}_{\bar{k}} \triangleq \begin{bmatrix} \mathbf{A}_1 & \cdots & \mathbf{A}_{k-1} & \mathbf{A}_{k+1} & \cdots & \mathbf{A}_K \end{bmatrix}$ is an $N_R \times (K-1)d$ dimensional cascaded matrix of $\mathbf{A}_j$ for $j \in [K]$ and $j \neq k$ and $\mathbf{z}_{\bar{k}}(t) \triangleq \begin{bmatrix} \mathbf{z}_1(t) & \cdots & \mathbf{z}_{k-1}(t) & \mathbf{z}_{k+1}(t) & \cdots & \mathbf{z}_K(t) \end{bmatrix}^T$ is a $(K-1)d \times 1$ dimensional cascaded model vector where each element is the aggregated model vector of cluster $j \in [K]$ and $j \neq k$.

In order to decode the aggregate of the local models for cluster $k \in [K]$, we design the decoder to satisfy the following two conditions,

$$\mathbf{U}_k \mathbf{A}_k = \mathbf{I} \tag{16}$$

and

$$\mathbf{U}_k \mathbf{A}_{\bar{k}} = \mathbf{0}. \tag{17}$$

The constraint in (17) implies that $\mathbf{U}_k$ should be in the null space of $\mathbf{A}_{\bar{k}}$ [26], from which we can define the decoder as:

$$\mathbf{U}_k = \left((\mathbf{U}_k^o)^H \mathbf{A}_k \right)^{-1} (\mathbf{U}_k^o)^H \tag{18}$$

where $[\mathbf{U}_k^0 \ \mathbf{U}_k^1]\Sigma_k \mathbf{B}_k$ is the SVD of $\mathbf{A}_{\bar{k}}$ and $\mathbf{U}_k^0$ is a $d \times N_R$ matrix whose columns corresponds to a null-space basis of $\mathbf{A}_{\bar{k}}$. As such, the decoder matrix from (18) satisfies both (16) and (17). Finally, note that condition (17) can be stated as,

$$\mathbf{U}_k \mathbf{A}_1 = \cdots = \mathbf{U}_k \mathbf{A}_{k-1} = \mathbf{U}_k \mathbf{A}_{k+1} = \cdots = \mathbf{U}_k \mathbf{A}_K = \mathbf{0} \tag{19}$$

from which, combined with (16) and (14), we have that,

$$\widehat{\mathbf{z}}_k(t) = \mathbf{z}_k(t) + \mathbf{U}_k \mathbf{n}(t) \quad \forall k \in [K]. \tag{20}$$

The individual steps of our algorithm is provided in Algorithm 1.

We note that the number of required antennas can be further reduced by leveraging gradient compression and sparsification techniques. For instance, instead of sending the entire local model (of size d), users can adopt the *allReduce* rand-k sparsification scheme [27] where users send only the model parameters corresponding to $k \ll d$ locations, where all users are assigned to the same set of k random locations (assigned by the server), which also allows over-the-air aggregation.

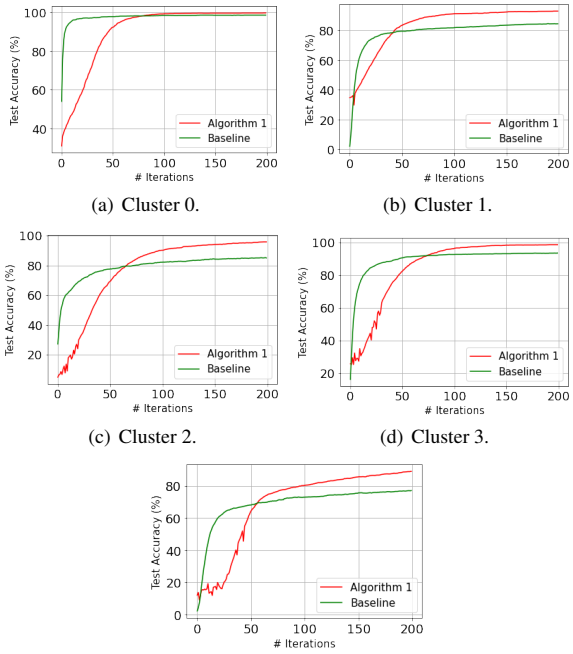
4. EXPERIMENTS

In our experiments, we consider an image classification task with $N = 25$ users using the MNIST [24] and CIFAR-10 [25] datasets. We consider a non-i.i.d. data distribution among the users, where the local dataset of each user consists of the data samples corresponding to two distinct labels. In particular, for both MNIST or CIFAR-10 datasets (both contain 10 labels), the training samples containing labels $\{2j, 2j+1\}$ are distributed among users $\{5j, \ldots, 5j+4\}$ for $j = 0, \ldots, 4$. We set the total number of clusters to $K = 5$. We apply the *allReduce* rand-k sparsification scheme in our experimental setup to avoid using a large number of antennas at the transmitter and receiver. The specific parameters used for our experimental setup are demonstrated in Table 1.

In Fig. 1, we demonstrate the average test accuracy for the users in each cluster, compared to the baseline conventional FL algorithm [1] where a single global model is trained for all users. As shown in Fig. 1, in the conventional setup, the average accuracy of the users in all but one cluster are heavily degraded, hence a single global model did not provide comparable performance for all users. In particular, the average accuracy of all clusters is 95.2% for our framework, while the average accuracy for the baseline scheme is 86% percent, hence our scheme leads to around 11% improvement in the test accuracy.

Table 1: Parameters used for the experimental setup

Parameters	Value
Total number of users, N	25
Total number of clusters, K	5
Number of model parameters, d (MNIST)	159010
Number of model parameters, d (CIFAR-10)	164954
Number of sparsified gradient parameters	3000
Number of transmitter antennas, N_T	15000
Number of receiver antennas, N_R	15000
Learning rate, η	0.0001

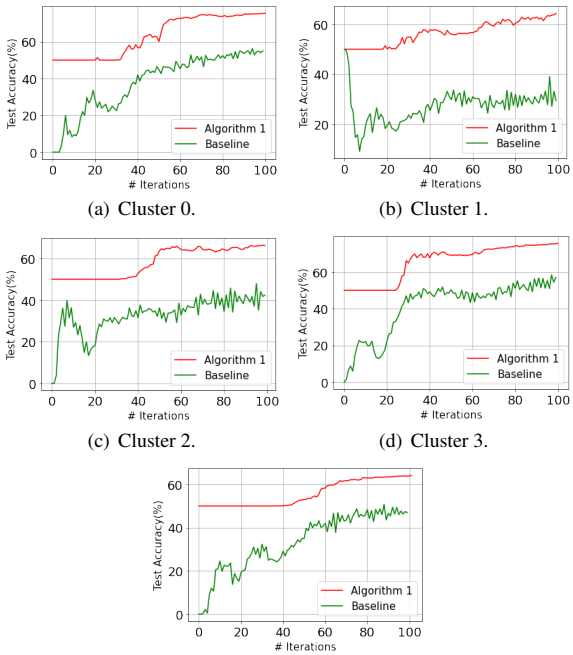


(a) Cluster 0. (b) Cluster 1.

(c) Cluster 2. (d) Cluster 3.

(e) Cluster 4.

Fig. 1: Test accuracy vs number of iterations (MNIST dataset)

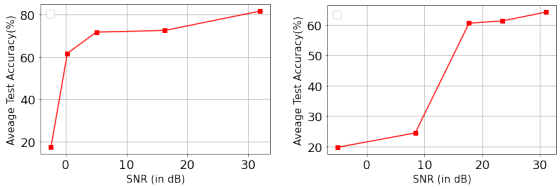


(a) Cluster 0. (b) Cluster 1.

(c) Cluster 2. (d) Cluster 3.

(e) Cluster 4.

Fig. 2: Test accuracy vs number of iterations (CIFAR-10 dataset)



(a) MNIST dataset (b) CIFAR-10 dataset

Fig. 3: Average test accuracy of all users vs SNR

As shown in Fig. 2, for the CIFAR-10 dataset, the discrepancy observed in the performance is even higher. In the baseline scheme, the average accuracy is around 45% whereas the average accuracy for our proposed scheme is 74.2% after 140 iterations which indicates 64.8% improvement over the test accuracy of baseline scheme. In Fig. 3, we demonstrate the average test accuracy of the users over all iterations versus signal to noise ratio (SNR), by varying the parameter β that controls the transmit power. As observed in Fig. 3, a higher SNR leads to better robustness against noise and thus higher accuracy.

5. CONCLUSION

We have proposed an over-the-air personalized federated learning protocol for communication-efficient distributed learning under heterogeneous settings, in particular, when the users have non-i.i.d. data distributions. Our approach builds on a clustered federated learning approach and an encoder design that aligns the transmitted signals from the users belonging to the same cluster. Then, a zero-forcing decoder is designed for each cluster to null the interference caused by the remaining clusters. We provide experiments on the MNIST and CIFAR-10 datasets and demonstrate the performance improvement over the federated learning benchmark. Future directions include enabling synchronization among the users to prevent information leakage from individual local models.

6. REFERENCES

[1] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Int. Conf. on Artificial Int. and Stat. (AISTATS)*, pp. 1273–1282, 2017.

[2] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1205–1221, 2019.

[3] Y. Sun, H. Ochiai, and H. Esaki, "Decentralized deep learning

for mobile edge computing: A survey on communication efficiency and trustworthiness," *CoRR*, vol. abs/2108.03980, 2021.

[4] M. M. Amiri and D. Gündüz, "Machine learning at the wireless edge: Distributed stochastic gradient descent over-the-air," in *2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 1432–1436, 2019.

[5] K. Yang, T. Jiang, Y. Shi, and Z. Ding, "Federated learning via over-the-air computation," *IEEE Transactions on Wireless Communications*, vol. 19, no. 3, pp. 2022–2035, 2020.

[6] M. Krouka, A. Elgabli, C. B. Issaid, and M. Bennis, "Communication-efficient split learning based on analog communication and over the air aggregation," *CoRR*, vol. abs/2106.00999, 2021.

[7] X. Ma, H. Sun, Q. Wang, and R. Q. Hu, "User scheduling for federated learning through over-the-air computation," *CoRR*, vol. abs/2108.02891, 2021.

[8] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 491–506, 2020.

[9] G. Zhu, Y. Du, D. Gündüz, and K. Huang, "One-bit over-the-air aggregation for communication-efficient federated edge learning: Design and convergence analysis," *IEEE Transactions on Wireless Communications*, vol. 20, no. 3, pp. 2120–2135, 2021.

[10] X. Cao, G. Zhu, J. Xu, Z. Wang, and S. Cui, "Optimized power control design for over-the-air federated edge learning," *ArXiv*, vol. abs/2106.09316, 2021.

[11] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, "Differentially private aircomp federated learning with power adaptation harnessing receiver noise," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, pp. 1–6, 2020.

[12] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh, "Three approaches for personalization with applications to federated learning," *CoRR*, vol. abs/2002.10619, 2020.

[13] T. Li, M. Sanjabi, A. Beirami, and V. Smith, "Fair resource allocation in federated learning," in *International Conference on Learning Representations*, 2020.

[14] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *arXiv preprint arXiv:1806.00582*, 2018.

[15] K. Wang, R. Mathews, C. Kiddon, H. Eichner, F. Beaufays, and D. Ramage, "Federated evaluation of on-device personalization," *CoRR*, vol. abs/1910.10252, 2019.

[16] D. Li and J. Wang, "Fedmd: Heterogenous federated learning via model distillation," *CoRR*, vol. abs/1910.03581, 2019.

[17] Y. Luo, X. Liu, and J. Xiu, "Energy-efficient clustering to address data heterogeneity in federated learning," in *ICC 2021 - IEEE International Conference on Communications*, pp. 1–6, 2021.

[18] M. Duan, D. Liu, X. Ji, R. Liu, L. Liang, X. Chen, and Y. Tan, "Fedgroup: Efficient clustered federated learning via decomposed data-driven measure," 2021.

[19] F. Sattler, K.-R. Müller, and W. Samek, "Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 8, pp. 3710–3722, 2021.

[20] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran, "An efficient framework for clustered federated learning," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing System (NeurIPS)*, 2020.

[21] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, 2020.

[22] C. Briggs, Z. Fan, and P. Andras, "Federated learning with hierarchical clustering of local updates to improve training on non-iid data," in *2020 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–9, 2020.

[23] Y. Kim, E. A. Hakim, J. Haraldson, H. Eriksson, J. M. B. da Silva, and C. Fischione, "Dynamic clustering in federated learning," in *ICC 2021 - IEEE International Conference on Communications*, pp. 1–6, 2021.

[24] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database," *http://yann. lecun. com/exdb/mnist*, 2010.

[25] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," tech. rep., Citeseer, 2009.

[26] Q. H. Spencer, A. L. Swindlehurst, and M. Haardt, "Zero-forcing methods for downlink spatial multiplexing in multiuser mimo channels," *IEEE transactions on signal processing*, vol. 52, no. 2, pp. 461–471, 2004.

[27] N. F. Eghlidi and M. Jaggi, "Sparse communication for training deep networks," *CoRR*, vol. abs/2009.09271, 2020.