Semi-Decentralized Federated Learning with Collaborative Relaying

Michal Yemini, Rajarshi Saha, Emre Ozfatura,
Deniz Gündüz and Andrea J. Goldsmith

Abstract

We present a semi-decentralized federated learning algorithm wherein clients collaborate with each other in relaying their local updates to a central parameter server (PS). At every communication round to the PS, each client computes a local consensus of the updates from its neighboring clients and eventually transmits a weighted average to the PS. To ensure that even clients with poor connectivity are able to relay their local updates to the PS, we appropriately optimize the weights used by each client in computing the local consensus. The optimized weights ensure that the variance of the global update at the PS is reduced, consequently improving the rate of convergence. Numerical simulations substantiate our theoretical claims and demonstrate settings with intermittent connectivity between the clients and the PS where our proposed algorithms shows an improved convergence rate and accuracy.

I. Introduction

Federated learning (FL) algorithms iteratively optimize a common objective function to learn a shared model over data samples that are localized over multiple distributed clients [1]. FL approaches aim to reduce the required communication overhead and improve clients' privacy by letting each client train a local model on its local (private) dataset and forwarding them periodically to a centralized parameter server (PS).

Michal Yemini and Andrea J. Goldsmith are with the Faculty of Electrical and Computer Engineering, Princeton University. (Emails: myemini@princeton.edu, goldsmith@princeton.edu). Rajarshi Saha is with the Department of Electrical Engineering Stanford University. (Email: rajsaha@stanford.edu). Emre Ozfatura and Deniz Gündüz are with the Department of Electrical and Electronic Engineering, Imperial College London. (Emails: m.ozfatura@imperial.ac.uk, d.gunduz@imperial.ac.uk).

M. Yemini, R. Saha, and A.J. Goldsmith are partially supported by the AFOSR award #002484665 and a Huawei Intelligent Spectrum grant.

E. Ozfatura and D. Gündüz received funding from the European Research Council (ERC) through Starting Grant BEACON (no. 677854) and the UK EPSRC (grant no. EP/T023600/1) under the CHIST-ERA program.

In practical FL scenarios, some clients are stragglers and cannot send their estimates regularly, either because: (i) they cannot finish their computation within a prescribed deadline, or (ii) they cannot transmit their estimate to the PS successfully due to communication limitations [2]. Specifically, clients can suffer from intermittent connectivity to the PS, where their wireless communication channel is temporary blocked [3]–[9]. Stragglers deteriorate the convergence of FL as the computed local estimates become stale and useless, and can even result in bias in the final model in the case of persistent stragglers. However, the case of communication stragglers that are limited due to loss of direct communication opportunities to the PS is inherently different from those that result from limited computation resources at the client, since the former can be solved by relaying the updates via neighboring clients.

Communication quality at the wireless edge as a key design principle is considered in the federated edge learning (FEEL) framework [10], which takes into account the wireless channel characteristics from the clients to the PS to optimize the convergence and final model performance at the PS. So far the FEEL paradigm has mainly focused on direct communication from the clients to the PS, and aimed at improving the performance by resource allocation across clients [10]–[19]; this model has ignored possible cooperation between clients in the case of intermittent communication blockages.

Motivated by our prior works [20]–[22], where client cooperation is used to improve the connectivity to the cloud and to reduce the latency and scheduling overhead, this work proposes and analyzes a new FEEL paradigm, where the clients cooperate to mitigate the detrimental effects of communication stragglers. In this proposed method clients send each other their current updates so that each client can send to the PS a weighted average of its current update and those of its neighbors. Using this approach, the PS receives new updates from clients with intermittently failing uplink connections that would otherwise become stale and be discarded. Moreover, to reduce expected distance to the optimal point at the PS, we optimize the weights given to each clients' neighbors in order to ensure that the transmitted updates to the PS (i) achieve weak unbiasedness that preserves the objective function at the PS, and (ii) minimizes the convergence time of the learning algorithm. We provide an analysis and theoretical guarantees for the improvement in convergence rate that our proposed scheme achieves.

Most existing works on FL assume error-free rate-limited orthogonal communication links, assuming that the wireless imperfections are taken care of by an underlying communication protocol. However, such a separation between the communication medium and the learning

protocol can be strictly suboptimal [10]. An alternative approach is to treat the communication of the model updates to the PS as an uplink communication problem and jointly optimize the learning algorithm and the communication scheme, taking wireless channel imperfections into account [10]. Within this framework an original and promising approach is the *over-the-air computation (OAC)* [15]–[17], which exploits the signal superposition property of the wireless medium to convey the sum of the model updates that are transmitted by each client in an uncoded fashion. In addition to bandwith efficiency, the OAC framework also provides a certain level of anonymity to the clients due to its superposition nature; and hence, enhances the privacy of the participating clients [18], [19]. We emphasize here that, in OAC framework, the PS receives the aggregate model, and it is not possible to entangle the individual model updates. Therefore, any strategy that utilizes a PS side aggregation mechanism with individual model updates to address unequal client participation is not compatible with the OAC framework. One of the major advantages of our proposed scheme is that it mitigates the drawbacks of unequal client participation without requiring the knowledge of the identity of the transmitting clients or their individual updates at the PS. Therefore, our solution is compatible with OAC

A. Related works

The conventional FL framework [1] is orchestrated by a centralized entity called PS, which helps participating clients to reach a consensus on the model parameters by aggregating their locally trained models. However, such a consensus mechanism requires the exchange of a large number of model parameters between the PS and the clients resulting in a significant communication overhead at the PS. A decentralized collaborative learning framework has been introduced as an alternative to centralized FL, in which the PS is removed to mitigate a potential communication bottleneck and a single point of failure. In decentralized learning, each client shares its local model with the neighbouring clients through device-to-device (D2D) communications, and model aggregation is executed at each client in parallel. In a sense, in decentralized learning, each client becomes a PS. The aggregation strategy at each client is determined according to the network topology, that is the connection pattern between the clients, and often a fixed topology is considered [23]–[31]. These results can be further extended to scenarios with time varying topologies [32]–[34].

An alternative approach to both centralized and decentralized schemes is the *hierarchical FL (HFL)* framework [22], [35]–[37], in which multiple PSs are employed for the aggregation

to prevent a communication bottleneck. In HFL, clients are divided into clusters and a PS is assigned to each cluster to perform local aggregation, while the aggregated models at the clusters are later aggregated at the main PS in a subsequent step to obtain the global model. This framework has significant advantages over centralized and decentralized schemes, particularly when the communication takes place over wireless channels since it allows spatial reuse of available resources [22].

Although HFL has certain advantages, this framework requires employing multiple PSs that may not be practical in certain scenarios. Instead, the idea of hierarchical collaborative learning can be redesigned to combine hierarchical and decentralized learning concepts which is referred as *semi-decentralized FL*, where the local consensus follows decentralized learning with D2D communication, whereas the global consensus is orchestrated by the PS [38], [39]. One of the major challenges in FL that is not considered in the aforementioned works on semi-decentralized FL is the partial client connectivity [40], [41]. Unequal client participation due intermittent connectivity exacerbates the impact of data heterogeneity [42]–[45], and increases the generalization gap.

The connectivity of the clients is a particularly significant challenge in FEEL, where the clients and the PS communicate over unreliable wireless channels. Due to their different physical environments and distances to the PS, clients may have different connectivity to each other and the PS. This problem has been recently addressed in [11]–[14], [46]–[49] by considering customized client selection mechanisms to seek a balance between the participation of the clients and the latency for the model aggregation in order to speed up the learning process. In this work, we adopt a different approach to the connectivity problem, and instead of designing a client selection mechanism, or optimizing resource allocation to balance client participation, we introduce a *knowledge relaying* mechanism that takes into account the nature of individual clients' connectivity to the PS and ensures that, in case of poor connectivity, their local knowledge is conveyed to the PS with the help of their neighboring clients. Finally, another related body of work considers coded computation as a possible solution to mitigate stragglers [50]–[53]. However, in this approach, data is strategically allocated to clients to create redundant computations that can be exploited at the PS using the additional information of the stragglers' identities. In contrast, our approach does not require redundancies in clients' data and can be applied together with OAC where, the PS is blind to the identities of the transmitting clients.

B. Main Contributions

The main contributions of our paper can be summarized as follows:

- We propose a new semi-decentralized FL framework, which exploits local connections between clients to relay each other's estimates to the PS, to mitigate the negative impacts of intermittent client-PS connections on the learning performance. In the proposed framework. clients cooperate not only to learn a common model by exchanging model updates with the PS, but also to improve their connectivity.
- We optimize our proposed collaborative relaying approach to preserve the unbiasedness of the local updates and to minimize the expected convergence time to an optimal model.
- Our approach can be applied to PSs that are blind to the identities of the transmitting clients. Therefore, it is suitable to be used in conjunction with OAC.
- By conducting extensive simulations, we numerically show the superiority of the proposed framework compared to FedAvg, particularly when the data heterogeneity is taken into account to mimic realistic scenarios.

C. Paper Organization

The rest of the paper is organized as follows: Section II presents the FL system model and the proposed FL collaborative relaying scheme. Section III derives conditions for the unbiasedness of our collaborative relaying scheme and presents an upper bound on its expected distance to optimality. Furthermore, using these analytical guarantees, Section IV optimizes our collaborative relaying scheme to reduce the upper bound on the expected suboptimality gap, while Section V presents numerical results that validate our theoretical analysis and highlight the performance improvement in terms of training accuracy the collaborative relaying provides. Finally, Section VI concludes this paper.

II. System Model for Collaborative Relaying

Consider n clients in a FL environment communicating with a central PS, which trains a model with d parameters represented by the d-dimensional vector $\mathbf{x} \in \mathbb{R}^d$. The clients communicate periodically with the PS, over intermittently connected links, to minimize an empirical loss function we define below.

Let $\mathcal{L}(\mathbf{x}, \boldsymbol{\zeta})$ be the loss evaluated for a model $\mathbf{x}$ at the data point $\boldsymbol{\zeta}$. Denote the local loss at client i by $f_i : \mathbb{R}^d \times \mathcal{Z}_i \to \mathbb{R}$, where $f(\mathbf{x}; \mathcal{Z}_i) = \frac{1}{|\mathcal{Z}_i|} \sum_{\boldsymbol{\zeta} \in \mathcal{Z}_i} \mathcal{L}(\mathbf{x}; \boldsymbol{\zeta})$. Here, $\mathcal{Z}_i$ is the local dataset of client i. The PS aims to solve the following Empirical Risk Minimization (ERM) problem:

$$\mathbf{x}^* = \underset{\mathbf{x} \in \mathbb{R}^d}{\arg\min.}\, f(\mathbf{x}) \triangleq \underset{\mathbf{x} \in \mathbb{R}^d}{\arg\min.}\, \frac{1}{n} \sum_{i=1}^{n} f_i(\mathbf{x}; \mathcal{Z}_i). \tag{1}$$

We optimize the results in this paper assuming uniform sizes for local datasets, i.e., $|Zc_i| = |Zc_j|$ for every $i, j \in [n]$. However, our results hold for any system that fulfills Assumptions 1-3 that are stated in III.

A. FL with Local SGD at Clients

Denote by $\nabla f_i(\mathbf{x})$ the gradient of the loss function at client i, i.e., $\nabla f_i(\mathbf{x}) \triangleq \nabla f(\mathbf{x}; Zc_i)$. Additionally, let $g_i(\mathbf{x})$ denote a stochastic gradient of $f(\mathbf{x}; \mathcal{Z}_i)$.

Let $\mathcal{T}$ be the *period* of local averaging, i.e., the number of local iterations *after* which the PS receives clients' estimates for the model parameters. At the beginning r^{th} *round* of FL, the PS broadcasts the global model $\mathbf{x}^{(r)}$ to the clients. For local iteration $k \in [0 : \mathcal{T}]$ of the r^{th} round, client i applies the following local update rule:

$$\mathbf{x}_i^{(r,k+1)} = \mathbf{x}_i^{(r,k)} - \eta_r g_i \left(\mathbf{x}_i^{(r,k)} \right), \tag{2}$$

where η_r is the local learning rate for round r, and $\mathbf{x}_i^{(r,0)} = \mathbf{x}^{(r)}$.

B. Communication Model

For the sake of the simplicity of exposition, we consider the extreme case where communication link is either unavailable or perfect. Furthermore, we depict in Fig. 1 the considered communication model.

Communication between clients and PS: We assume a model where the communication from clients to the PS is intermittent. We model the intermittent connectivity of client i to the PS at round r by the Bernoulli random variable $\tau_i(r) \sim \text{Bernoulli}(p_i)$, where $\tau_i(r) = 1$ denotes an uplink communication opportunity between client i and the PS at round r whereas $\tau_i(r) = 0$ denotes that the uplink communication channel between client i and the PS at round r is blocked. We assume that the connectivity is permanent in the downlink.

Remark 1. *We assume that the connectivity probabilities p_i, $i \in [n]$ are known, however, in general they can be easily estimated, using, for example, pilot signals. Moreover, clients can*

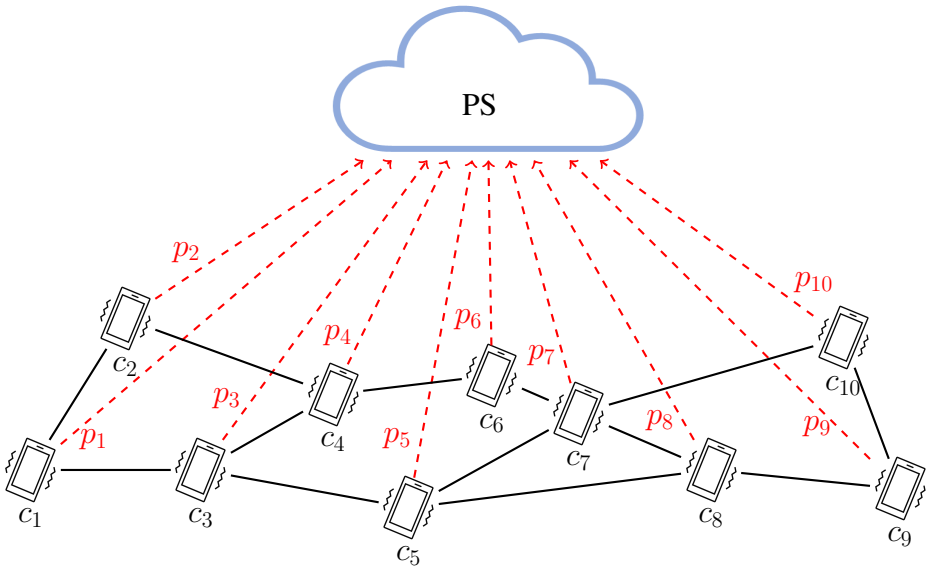

Fig. 1: System model with communication between clients, and from clients to the PS. The red arrows denote intermittently connected links from clients to the PS. Here, c_i denotes client i. Additionally, p_i is the probability that the link from client i to the PS is connected.

share their p_i with each other using local links in a pre-training phase. On the other hand, we do not assume that the instantaneous connectivity information $\tau_i(r)$, $r \in [n]$, is available to any of the clients.

Communication between Clients: We model the connectivity amongst the clients by an undirected graph $G = (V, E)$ where $V = [n]$ and $\{i, j\} \in E$ if and only if client i can communicate with client j. Therefore, communication between clients is bidirectional. We denote by $\mathcal{N}_i$ the set of neighbors of client i, that is

$$\mathcal{N}_i = \{j \in V : \{i, j\} \in E\}.$$

For example, in the system depicted in Fig. 1, $\mathcal{N}_1 = \{2, 3\}$ and $\mathcal{N}_2 = \{1, 4\}$.

Since communicating clients can send updates to one another, each client can send to the PS a weighted average of its own update and those of its neighbors. In this way, the PS can receive estimates from clients with failing uplink connections. Based on this observation, next we present our collaborative relaying procedure.

Algorithm 1: COLREL-CLIENT (Client Collaborative Relaying)

Input: Round number r, step-size η_r, round length $\mathcal{T}$, set of neighbors of client i $\mathcal{N}_i$, α_{ij} for every $j \in \mathcal{N}_i \cup \{i\}$.

Output: $\Delta\widetilde{\mathbf{x}}_i^{r+1}$.

Receive $\mathbf{x}^{(r)}$ from PS

Set $\mathbf{x}_i^{(r,0)} = \mathbf{x}^{(r)}$

for $k \leftarrow 0$ ***to*** $\mathcal{T} - 1$ **do**

  Generate a random gradient $g_i\left(\mathbf{x}_i^{(r,k)}\right)$

  $\mathbf{x}_i^{(r,k+1)} = \mathbf{x}_i^{(r,k)} - \eta_r g_i\left(\mathbf{x}_i^{(r,k)}\right)$

end

Set $\Delta\mathbf{x}_i^{r+1} = \mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}$

Send $\Delta\mathbf{x}_i$ to every $j \in \mathcal{N}_i$

Receive $\Delta\mathbf{x}_j$ from every $j \in \mathcal{N}_i$

Compute $\Delta\widetilde{\mathbf{x}}_i^{r+1} = \sum_{j\in\mathcal{N}_i\cup\{i\}} \alpha_{ij} \cdot \Delta\mathbf{x}_j^{r+1}$

Transmit $\Delta\widetilde{\mathbf{x}}_i^{r+1}$ to the PS

C. Collaborative Relaying of Local Updates

Let $\Delta\mathbf{x}_j^{r+1}$ denote the update of client i at the end of the $\mathcal{T}$th local iteration in round r; i.e.,

$$\Delta\mathbf{x}_j^{r+1} = \mathbf{x}_j^{(r,\mathcal{T})} - \mathbf{x}^{(r)}.$$

We assume that the model update of client i at the end of the $\mathcal{T}$th iteration is readily available to its neighbors in $\mathcal{N}_i$. Then, client i forms a combination of its own update and those of its neighbors

$$\Delta\widetilde{\mathbf{x}}_i^{r+1} = \sum_{j\in\mathcal{N}_i\cup\{i\}} \alpha_{ij} \cdot \Delta\mathbf{x}_j^{r+1} = \sum_{j\in\mathcal{N}_i\cup\{i\}} \alpha_{ij}\left(\mathbf{x}_j^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right), \tag{3}$$

where α_{ij} is a nonnegative weight used by client i to relay the update from client j. Note that the complexity of this averaging is low since it involves a linear operation, i.e., weighted averaging, of $O(\max_{i\in[n]} |\mathcal{N}_i| + 1)$.

D. PS Aggregation

We assume that the PS does not explicitly select which subset of clients it wants to receive information from, but rather receives information from all *communicating* clients every $\mathcal{T}$ units

Algorithm 2: COLREL-PS (PS Aggregation)

Input: Number of rounds R, a set of clients $[n]$, scalar w.

Output: An estimate $\mathbf{x}^{(R)}$ of optimal value $\mathbf{x}^\star$

Set $\mathbf{x}^0 = \mathbf{0}$

for $r \leftarrow 0$ ***to*** $R-1$ **do**

  Send $\mathbf{x}^{(r)}$ to all clients

  Set $\tau_i(r+1) = 1$ if PS successfully receives $\Delta\widetilde{\mathbf{x}}_i^{r+1}$ from client i and set $\tau_i(r+1) = 0$ otherwise.

  Compute $\mathbf{x}^{(r+1)} = \mathbf{x}^{(r)} + \frac{1}{n}\sum_{i\in[n]} \tau_i(r+1)\Delta\widetilde{\mathbf{x}}_i^{r+1}$

end

of time.

In a FL setup, the PS can use the following rescaled sum of the received updates:

$$\mathbf{x}^{(r+1)} = \mathbf{x}^{(r)} + w\sum_{i\in[n]} \tau_i(r+1)\Delta\widetilde{\mathbf{x}}_i^{r+1} = \mathbf{x}^{(r)} + w\sum_{i\in[n]} \tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}} \alpha_{ij}\left(\mathbf{x}_j^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right). \tag{4}$$

This update can be computed over-the-air and does not require the PS to know the identities of the communicating clients at each round. Additionally, we set $w = \frac{1}{n}$ to preserve the unbiasedness of the objective function at the PS, as we discuss next.

We depict the overall collaborative relaying (ColRel) FL procedure in Algorithm 1 and Algorithm 2.

III. UNBIASEDNESS AND EXPECTED SUBOPTIMALITY GAP

This section presents sufficient conditions for the unbiasedness of the proposed collaborative relaying method. Under the unbiasedness conditions, we derive an upper bound on the expected suboptimality gap of the proposed method.

A. Sufficient Condition for the Unbiasedness of Collaborative Relaying

Recall that α_{ji} is the coefficient factor client j gives the updated estimate it receives from client i. Client i and each of its neighbors, $j \in \mathcal{N}_i$, send to the PS $\alpha_{ji}\Delta\mathbf{x}_i^{r+1}$ on behalf of client i. Specifically, in the system depicted in Fig. 1, the updated estimate of client 1, i.e., $\Delta\mathbf{x}_1^{r+1}$ is

scaled and transmitted to the PS by clients $1, 2$ and 3. Specifically, the PS receives from clients $1, 2$ and 3 the following values at the end of round r:

$$\begin{aligned}
\tau_1(r+1)\Delta\widetilde{\mathbf{x}}_1^{r+1} &= \tau_1(r+1)\left(\alpha_{11}\cdot\Delta\mathbf{x}_1^{r+1}+\alpha_{12}\cdot\Delta\mathbf{x}_2^{r+1}+\alpha_{13}\cdot\Delta\mathbf{x}_3^{r+1}\right)\\
\tau_2(r+1)\Delta\widetilde{\mathbf{x}}_2^{r+1} &= \tau_2(r+1)\left(\alpha_{21}\cdot\Delta\mathbf{x}_1^{r+1}+\alpha_{22}\cdot\Delta\mathbf{x}_2^{r+1}+\alpha_{24}\cdot\Delta\mathbf{x}_4^{r+1}\right)\\
\tau_3(r+1)\Delta\widetilde{\mathbf{x}}_3^{r+1} &= \tau_3(r+1)\left(\alpha_{31}\cdot\Delta\mathbf{x}_1^{r+1}+\alpha_{33}\cdot\Delta\mathbf{x}_3^{r+1}+\alpha_{34}\cdot\Delta\mathbf{x}_4^{r+1}\right).
\end{aligned} \tag{5}$$

Therefore, the overall contribution of the update of client 1 received at the PS is

$$\tau_1(r+1)\alpha_{11}\cdot\Delta\mathbf{x}_1^{r+1}+\tau_2(r+1)\alpha_{21}\cdot\Delta\mathbf{x}_1^{r+1}+\tau_3(r+1)\alpha_{31}\cdot\Delta\mathbf{x}_1^{r+1}. \tag{6}$$

In the general case, the total received update of client i at the PS is given by

$$\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\Delta\mathbf{x}_i^{r+1}. \tag{7}$$

Then, the following lemma presents a sufficient condition on α_{ji} and w for unbiasedness.

Lemma 1. *Let $w=\frac{1}{n}$ and α_{ij} be such that*

$$E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\right]=p_i\alpha_{ii}+\sum_{j:j\in\mathcal{N}_i}p_j\alpha_{ji}=1. \tag{8}$$

Then, for every $i\in[n]$

$$w\cdot E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\Delta\mathbf{x}_i^{r+1}\middle|\Delta\mathbf{x}_i^{r+1}\right]=\frac{1}{n}\cdot\Delta\mathbf{x}_i^{r+1}. \tag{9}$$

Proof. Since for every $j\in[n]$ $\tau_j(r+1)$ is statistically independent of $\Delta\mathbf{x}_i^{r+1}$ we have that

$$w\cdot E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\Delta\mathbf{x}_i^{r+1}\middle|\Delta\mathbf{x}_i^{r+1}\right]=w\cdot E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\right]\Delta\mathbf{x}_i^{r+1}. \tag{10}$$

Substituting $w=\frac{1}{n}$ and (8) concludes the proof. □

We note that if condition (8) is fulfilled, then we also have

$$\frac{1}{n}E\left[\sum_{i\in[n]}\tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\right]=\frac{1}{n}\sum_{i\in[n]}E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\right]=\frac{1}{n}\cdot n=1. \tag{11}$$

Note, however, that since $\tau_j(r+1)$ are Bernoulli random variables Lemma 1 does *not* imply that

$$\frac{1}{n}\cdot\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\Delta\mathbf{x}_i^{r+1}=\frac{1}{n}\cdot\Delta\mathbf{x}_i^{r+1}. \tag{12}$$

Finally, the standard model of FL with random client sampling but with no connectivity among clients is captured by substituting $w = \frac{1}{n}$, $\mathcal{N}_i = \emptyset, p_i = p$ and $\alpha_{ii} = 1$, $\alpha_{ij} = 0$ for all $i, j \in [n]$ and $j \neq i$.

B. Expected Suboptimality Gap

Next, we present an upper bound on the expected distance to optimality as a function of the weights α_{ij}, $i, j \in [n]$.

Main assumptions: Let $\langle \mathbf{a}, \mathbf{b} \rangle$ denote the dot product between $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$, that is $\langle \mathbf{a}, \mathbf{b} \rangle = \mathbf{a}^T\mathbf{b}$. Additionally, let $\langle \mathbf{a}, \mathbf{b} \rangle \mathbf{a}^T\mathbf{b}$ and $\|\mathbf{a}\| = \sqrt{\mathbf{a}^T\mathbf{a}}$ denote the Euclidean norm of $\mathbf{a}$.

Assumption 1. *The loss functions f_i are L-smooth with respect to $\mathbf{x}$. That is, for every $i \in [n]$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ we have that*

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|. \tag{13}$$

Assumption 2. *The stochastic gradients $g_i(\mathbf{x})$ are unbiased and have bounded variance, i.e.:*

1) *$E(g_i(\mathbf{x})) = \nabla f_i(\mathbf{x})$ and*
2) *there exists σ^2 such that $E(\|g_i(\mathbf{x}) - \nabla f_i(\mathbf{x})\|^2) \leq \sigma^2$ for every $i \in [n]$, $\mathbf{x} \in \mathbb{R}^d$.*

Assumption 3. *The loss functions f_i are μ strongly convex. That is, for every $i \in [n]$ and $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$ we have that*

$$\langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \mu\|\mathbf{x} - \mathbf{y}\|^2. \tag{14}$$

Statement of the main theorem: Denote $\mathbf{A} = (\alpha_{ij})_{i,j\in[n]}$,

$$\mathcal{N}_{il} = (\mathcal{N}_i \cup \{i\}) \cap (\mathcal{N}_l \cup \{l\}), \tag{15}$$

and

$$S(\boldsymbol{p}, \mathbf{A}) = \sum_{i,l\in[n]} \sum_{j:j\in\mathcal{N}_{il}} p_j(1 - p_j)\alpha_{ji}\alpha_{jl}. \tag{16}$$

Theorem 1 (Expected Distance to Optimality)**.** *Let*

$$B(\boldsymbol{p}, \mathbf{A}) = \frac{2L^2}{n^2} S(\boldsymbol{p}, \mathbf{A}) \quad \text{and} \quad r_0(\boldsymbol{p}, \mathbf{A}) = \max\left\{\frac{L}{\mu}, 4\left(\frac{B(\boldsymbol{p}, \mathbf{A})}{\mu^2} + 1\right), \frac{1}{\mathcal{T}}, \frac{4n}{\mu^2\mathcal{T}}\right\}. \tag{17}$$

Additionally, let $\eta_r = \frac{4\mu^{-1}}{r\mathcal{T}+1}$, and

$$C_1(\boldsymbol{p}, \mathbf{A}) = \frac{4^2}{\mu^2} \cdot \frac{2\sigma^2}{n^2} S(\boldsymbol{p}, \mathbf{A}),$$

$$C_2 = \frac{4^2}{\mu^2} \cdot L^2 \frac{\sigma^2}{n} e,$$
$$C_3(\boldsymbol{p}, \mathbf{A}) = \frac{4^4}{\mu^4} \cdot \left(L^2 \sigma^2 e + \frac{2L^2 \sigma^2 e}{n^2} S(\boldsymbol{p}, \mathbf{A}) \right). \tag{18}$$

Then, under Assumptions 1-3 and condition (8)*, for every* $r \geq r_0(\boldsymbol{p}, \mathbf{A})$ *we have*

$$E \left\| \mathbf{x}^{(r+1)} - x^\star \right\|^2 \leq \frac{(r_0 \mathcal{T} + 1)}{(r\mathcal{T} + 1)^2} \left\| \mathbf{x}^{(0)} - x^\star \right\|^2 + C_1(\boldsymbol{p}, \mathbf{A}) \frac{\mathcal{T}}{k\mathcal{T} + 1} + C_2 \frac{(\mathcal{T} - 1)^2}{k\mathcal{T} + 1} + C_3(\boldsymbol{p}, \mathbf{A}) \frac{\mathcal{T}}{(k\mathcal{T} + 1)^2} \tag{19}$$

given the update dynamic captured by (2)-(4)*.*

It follows from Theorem 1 that

$$E \left\| \mathbf{x}^{(r+1)} - x^\star \right\|^2 = O\left(\frac{\left\| \mathbf{x}^{(0)} - x^\star \right\|^2}{r^2} + \frac{S(\boldsymbol{p}, \mathbf{A})}{r} \right), \tag{20}$$

therefore, the minimization of the distance to optimality is achieved by minimizing the term $S(\boldsymbol{p}, \mathbf{A})$ under the unbiasedness condition (8). Furthermore, minimizing the term $S(\boldsymbol{p}, \mathbf{A})$ can also reduce the value of $r_0(\boldsymbol{p}, \mathbf{A})$.

IV. Optimizing the Weights α_{ij}

From Theorem 1 and (33) we can minimize the upper bound on the expected distance to optimality by solving the following optimization problem

$$\begin{aligned} \min_{\mathbf{A}} S(\boldsymbol{p}, \mathbf{A}) &:= \sum_{i,l \in [n]} \sum_{j : j \in \mathcal{N}_{il}} p_j (1 - p_j) \alpha_{ji} \alpha_{jl}, \\ \text{s.t.:} \sum_{j : j \in \mathcal{N}_i} p_j \alpha_{ji} &= 1, \quad \forall i \in [n], \\ \alpha_{ji} &\geq 0 \quad \forall i, j \in [n]. \end{aligned} \tag{21}$$

Lemma 2. *Function* $S(\boldsymbol{p}, \mathbf{A})$ *is convex with respect* $\mathbf{A}$ *for* $\boldsymbol{p} \in [0, 1]^n$*.*

We present the proof of this lemma in Appendix D.

Let $\mathbf{A}_i$ denote the ith column of $\mathbf{A}$, that is, $\mathbf{A}_i = (\mathbf{A}_{1i}, \ldots, \mathbf{A}_{ni})^T$. We can rewrite $S(\boldsymbol{p}, \mathbf{A})$ as follows:

$$S(\boldsymbol{p}, \mathbf{A}) = \sum_{i \in [n]} \left[\sum_{j \in \mathcal{N}_i \cup \{i\}} p_j (1 - p_j) \alpha_{ji}^2 + \sum_{l \in [n], l \neq i} \sum_{j : j \in \mathcal{N}_{il}} p_j (1 - p_j) \alpha_{ji} \alpha_{jl} \right]. \tag{22}$$

Since the domain of the convex problem (21) is separable with respect to $\mathbf{A}_i$, we can use the Gauss-Seidel method to iteratively solve (21) and converge to an optimal solution [54, Proposition

2.7.1]. Let $\boldsymbol{A}_i^{(\ell)}$ denote the approximated value for $\mathbf{A}_i$ in the ℓth iteration. We choose the initial solution $\mathbf{A}_{ji}^{(0)} = \frac{1}{(|\mathcal{N}_i|+1)\cdot p_j} \cdot \mathbb{1}_{\{j\in\mathcal{N}_i\cup\{i\}:p_j>0\}}$. Then improve our solution iteratively by the Gauss-Seidel method until convergence. That is, at every iteration ℓ we compute $\boldsymbol{A}^{\ell}$ as follows

$$\mathbf{A}_i^{(\ell)} = \begin{cases} \widehat{\mathbf{A}}_i^{(\ell)} & \text{if } \ \ell \mod n + n\cdot \mathbb{1}_{\{\ell \mod n=0\}} = i \\ \mathbf{A}_i^{(\ell-1)} & \text{otherwise} \end{cases} \tag{23}$$

where $\mathbb{1}_{\{\cdot\}}$ denotes the indicator function, and

$$\begin{aligned} \widehat{\mathbf{A}}_i^{(\ell)} = \arg\min & \left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}} p_j(1-p_j)\alpha_{ji}^2 + 2\sum_{l\in[n],l\neq i}\sum_{j:j\in\mathcal{N}_{il}} p_j(1-p_j)\alpha_{ji}\alpha_{jl}^{(\ell-1)} \right], \\ \text{s.t.:} & \sum_{j:j\in\mathcal{N}_i\cup\{i\}} p_j\alpha_{ji} = 1, \\ & \alpha_{ji} \geq 0 \quad \forall j\in[n]. \end{aligned} \tag{24}$$

Define

$$L_{ji} = \{l : l\in[n], l\neq i, j\in\mathcal{N}_{il}\} \quad \text{and} \quad \beta_{ji} = \sum_{l\in L_{ji}} \alpha_{jl}^{(\ell-1)}.$$

Using Lagrange multipliers, we show in Appendix E that the optimal value for $\widehat{\mathbf{A}}_i^{(\ell)}$ is as follows:

$$\widehat{\alpha}_{ji}^{(\ell)} = \begin{cases} \left(-\beta_{ji} + \frac{\lambda_i}{2(1-p_j)}\right)^+ & \text{if } p_j\in(0,1) \text{ and } j\in\mathcal{N}_i\cup\{i\} \text{ and } \max_{k\in\mathcal{N}_i\cup\{i\}} p_k < 1, \\ \frac{1}{\sum_{k\in[n]} \mathbb{1}_{\{p_k=1, k\in\mathcal{N}_i\cup\{i\}\}}} & \text{if } p_j = 1 \text{ and } j\in\mathcal{N}_i\cup\{i\}, \\ 0 & \text{otherwise} \end{cases} . \tag{25}$$

where $(a)^+ \triangleq \max\{a, 0\}$ and λ_i is set such that $\sum_{j:j\in\mathcal{N}_i\cup\{i\}} p_j \left(-\beta_{ji} + \frac{\lambda_i}{2(1-p_j)}\right)^+ = 1$. We can find λ_i using the bisection method over the interval $\left[0, \max_{j\in\mathcal{N}_i\cup\{i\}} \{2(1-p_j)(\beta_{ji}+1)\}\right]$.

We summarize the *centralized* optimization procedure for $\boldsymbol{A}$ in Algorithm 3, where $\mathbf{1}$ denotes a column vector of ones. This centralized algorithm can be used if the PS is not blind to the identity of the transmitting clients and knows the connectivity graph G. Alternatively, Algorithm 3 can be also be implemented at clients that have similar information that is passed to them by other clients. Additionally, we present in Algorithm 4 a communication efficient version of Algorithm 3 that can be implemented by the clients in a distributed fashion. We note that each client does not need to fully know the matrix $\boldsymbol{A}$, but only the weights of all its direct neighbors

and its second degree neighbors (i.e., neighbors of its neighbors). This distributed algorithm can be used to optimized the weights when the PS is blind to the identities of the transmitting clients, as is the case with OAC. For simplicity of the presentation of Algorithm 4, we assume that every client i knows the set $\mathcal{N}_i^2 = \{k \in [n] : k \in \cup_{j\in[n]} L_{ji} \cup \{i\}\}$ and the probabilities p_j for all $j \in \mathcal{N}_i^2$. Recall that the probabilities p_j are assume to be known as we note in Remark 1 .

Computation complexity: The overall computational complexity of Algorithm 3 is $O(L \cdot (n^2 + K))$, where K is the number of iterations used in the bisection method for optimizing λ_i. Similarly, the computational complexity of Algorithm 4 *per client* is $O\left(\frac{L(n^2+K)}{n} + \frac{L(n-1)}{n} \cdot n\right)$, where the first term captures the computational complexity when a client computes (25) and the second term captures the worst-case complexity of substituting a column of the matrix $\boldsymbol{A}$.

Algorithm 3: COPT-α Centralized optimization of the weight matrix $\boldsymbol{A}$

Input: A set of clients $[n]$, a connectivity graph G, the function $S(\boldsymbol{p}, \mathbf{A})$, vector pf transmission probabilities $\boldsymbol{p}$, maximal number of iteration L, termination threshold δ.

Output: A matrix $\boldsymbol{A}^{(L)}$ that approximately minimizes (21)

Set $\mathbf{A}_{ji}^{(0)} = \frac{1}{(|\mathcal{N}_i|+1)\cdot p_j} \cdot \mathbb{1}_{\{j\in\mathcal{N}_i\cup\{i\}:p_j>0\}}$

Set $\ell = 0$

Set $\delta_\ell = \delta + 1$

Set $\boldsymbol{s} = S(\boldsymbol{p}, \mathbf{A}^{(0)}) \cdot \mathbf{1}$

while $\ell \leq L - 1$ *and* $\delta_\ell > \delta$ **do**

  Set $\ell = \ell + 1$

  Set $i = \ell \mod n + n \cdot \mathbb{1}_{\{\ell \mod n=0\}}$

  Compute $\widehat{\mathbf{A}}_i^{(\ell)}$ according to (25)

  Set $\mathbf{A}_k^{(\ell)}$ according to (23) for every $k \in [n]$

  Set $s_{\text{new}} = S(\boldsymbol{p}, \mathbf{A}^{(\ell)})$

  Set $\delta_\ell = s - s_{\text{new}}$

  Set $s = s_{\text{new}}$

end

Algorithm 4: DOPT-α Distributed optimization of the weight matrix $\boldsymbol{A}$

Input: Every client i knows the set $\mathcal{N}_i^2 = \{z \in [n] : z \in \cup_{j\in[n]} L_{ji} \cup \{i\}\}$, the probabilities p_j for all $j \in \mathcal{N}_i^2$, the number of clients n, and the maximal number of iteration L.

Output: Weight matrices $\boldsymbol{A}^{(L)}(i)$, $i \in [n]$ that approximately minimize (21)

Each client i set $\mathbf{A}_{st}^{(0)}(i) = \frac{1}{(|\mathcal{N}_t|+1)\cdot p_s} \cdot \mathbb{1}_{\{s\in\mathcal{N}_t\cup\{t\}:p_s>0\}} \cdot \mathbb{1}_{\{s,t\in\mathcal{N}_i^2\}}$

Set $\ell = 0$

while $\ell \leq L-1$ **do**

  Set $\ell = \ell + 1$

  Each client k sets $\mathbf{A}^{(\ell)}(k) = \mathbf{A}^{(\ell-1)}(k)$

  Set $i = \ell \mod n + n \cdot \mathbb{1}_{\{\ell \mod n=0\}}$

  Client i computes $\widehat{\mathbf{A}}_i^{(\ell)}(i)$ according to (25)

  Client i sets $\mathbf{A}_i^{(\ell)}(i) = \widehat{\mathbf{A}}_i^{(\ell)}(i)$

  Client i broadcasts $\mathbf{A}_i^{(\ell)}(i)$ to each of its neighbors $\mathcal{N}_i$

  Each client $j \in \mathcal{N}_i$ sets $\mathbf{A}_i^{(\ell)}(j) = \mathbf{A}_i^{(\ell)}(i)$

  The clients $\mathcal{N}_i$ broadcast $\mathbf{A}_i^{(\ell)}(i)$ to their neighbors $k \in \mathcal{N}_i^2 \setminus \mathcal{N}_i$

  Each client $k \in \mathcal{N}_i^2 \setminus (\mathcal{N}_i \cup \{i\})$ sets $\mathbf{A}_i^{(\ell)}(k) = \mathbf{A}_i^{(\ell)}(i)$;

end

V. Numerical Simulations

A. Simulation setup

To validate our theoretical results with numerical simulations, we use the CIFAR-10 [55] image classification dataset that contains 50,000 training and 10,000 test images from 10 classes. We consider the training set to be distributed across 10 clients according to both Independent and Identically distributed (IID) and non-IID fashions. Non-IID-ness of the data distribution amongst clients is often prevalent in FL setups and to emulate the same, we consider the *sort and partition* approach in which the training data is initially sorted based on the labels, and then they are divided into blocks and distributed amongst the clients randomly based on a parameter s, that measures the skewness of the data distribution. More precisely, s defines the maximum number of different labels present in the local dataset of each user, and therefore, smaller s

implies more skew in the data distribution. We use $s = 3$, meaning each client has images from at most 3 classes.

All clients locally train a ResNet-20 model for our image classification task. ResNet-20 consists of 0.27M parameters and is a popular architecture [56] that utilizes skip-connections / residual blocks to solve the problem of exploding / vanishing gradients. The plotted results of all simulations have been averaged over 5 independent realizations. In between every communication round to the parameter server (PS), the clients execute 8 local training steps of local-SGD. In all the experiments, we utilize the SGD optimizer at the clients. For non-IID setting, we also follow similar strategy to [57] such that the cumulative model updates of the clients are considered as pseudo-gradients and a momentum parameter is updated at PS based on these pseudo gradients and later used to update global model. We used a learning rate of 0.1 for SGD, a coefficient of $1e-4$ for ℓ_2-regularization to prevent overfitting, and a batch-size of 64. All simulations were done on NVIDIA GeForce GTX 1080 Ti with a CUDA Version 11.4.

B. Simulation Results

We now discuss our numerical experiments. To illustrate the advantages of our proposed scheme we compare it with three benchmark strategies namely,

- **FedAvg with perfect client connectivity to PS**. We consider federated averaging (FedAvg) when all clients are able to successfully transmit their local updates to the PS at every communication round. Denoted as **FedAvg - No Dropout** in the plots, this serves as a natural upper bound to the performance of any algorithm proposed in the presence of dropouts.
- **Blind FedAvg with intermittent client connectivity to PS**. As a natural performance lower bound in the presence of intermittent client connectivity, we consider a naïve federated averaging strategy, denotes as **FedAvg - Dropout (Blind)**, in which the PS is unaware of the identity of clients. In this strategy, for the clients that are unable to send their updates to the PS due to a dropout, the PS simply assumes that their update is zero. Essentially, the PS adds all the local updates it receives at any communication round, and divides it by the total number of clients irrespective of the knowledge of the number of actual successful transmissions. Such blind averaging strategies are often the norm for Over-The-Air FL settings.

- **Non-Blind FedAvg with intermittent client connectivity to PS** As another benchmark, we also consider a non-blind strategy, **FedAvg - Dropout (Non-Blind)** where the PS is aware of the identity of the clients, and knows exactly, how many and which clients have successfully been able to send their local update to the PS. This is common in point-to-point learning settings. In this case, the PS simply ignores the clients that have been unable to send their updates, and averages the successful updates by dividing the global aggregate at the PS by the number of successful transmissions.

We compare the above-mentioned strategies with our proposed collaborative relaying strategy in the presence of intermittent client connectivity to the PS. In order to highlight the difference in performance from the above-mentioned benchmarks, we consider the following setups for simulating our proposed scheme, COLREL*: Collaborative Relaying*:

- **Heterogeneity in client connectivity**. We distinguish between the cases when the all the clients have the same probability of successfully transmitting their local updates to the PS, vs. when some clients have more reliable connectivity to the PS than others. In this setting, we study the effect of optimizing the relay-weights that are allocated by every client to each of its neighbors while computing a local consensus. Relay weights are a function of the decentralized topology among clients and the link quality between clients and the PS.
- **Effect of topology**. We consider connectivity among clients according to *fully-connected (FC)* and *ring* network topologies. Clients can relay their updates with perfect reliability only to their immediate neighbors as specified by the decentralized topology. Consequently, this affects the convergence rate of the algorithm.
- **Heterogeneity in data distribution**. We also consider the effect of heterogeneity in data distribution across clients, and how collaborative relaying helps mitigate the performance deterioration in the presence of intermittent connectivity.

We now describe and discuss these setups in more detail below.

1) IID local data distribution across clients: We first consider the scenario where the data distribution is IID. Figs. 2 and 3 correspond to homogeneous connectivity where all clients have an equal probability of successful transmission to the PS ($p = 0.5$ and $p = 0.2$ respectively). For Figs. 2 and 3, we only consider the fully connected (FC) decentralized topology amongst

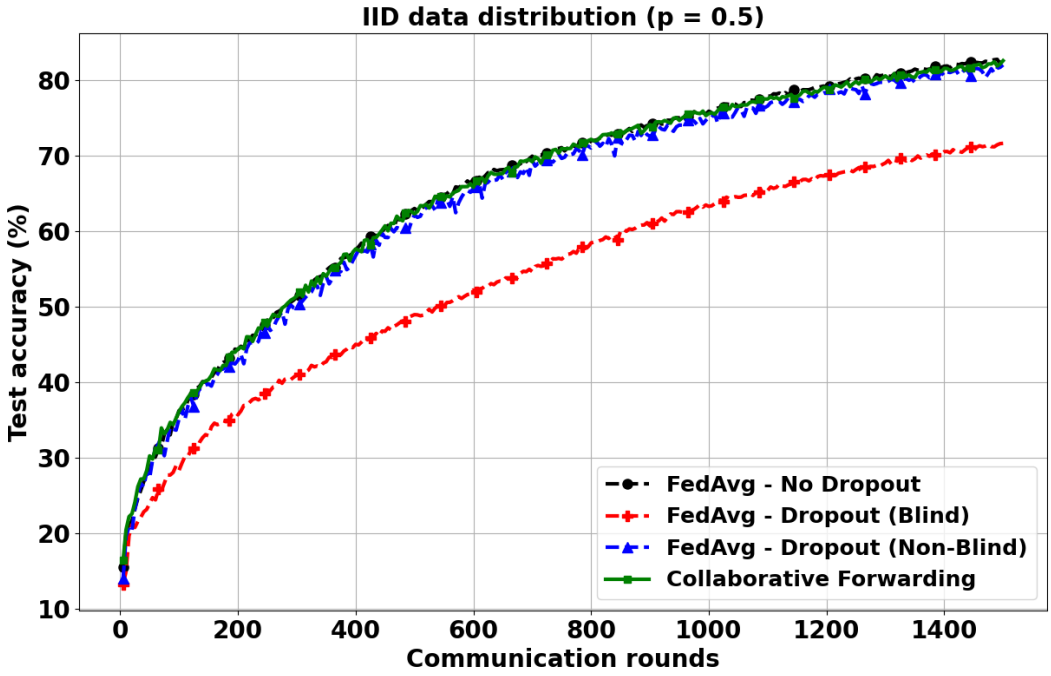

Fig. 2: Homogeneous connectivity with $p_i = 0.5, \forall i \in [n]$ and FC topology.

the clients. We also assume the transmissions amongst the clients (for collaboration) are always successful. From the plots, we see that for our proposed collaborative strategy performs at par with FedAvg without dropout, compared to the naive blind strategy. Moreover, for very small probabilities of successful transmissions, like $p = 0.2$, it performs even better than the non-blind averaging strategy, implying that decentralized collaboration amongst clients is pretty effective. We also note that we have ensured all the simulations to have the same step-size for a fair comparison. In other words, we have *not* optimized the learning-rate for individual simulations.

2) Heterogeneity in client connectivity and Effect of Network Topology: In subsequent simulations, Figs. 4 and 5 depict the comparison of different algorithms when every client has a different probability of successfully transmitting to the PS. We consider these probabilities to be $[0.1, 0.2, 0.3, 0.1, 0.1, 0.5, 0.8, 0.1, 0.2, 0.9]$. We have deliberately chosen to ensure that some clients have a very low probability of transmission, namely $p_1 = p_4 = p_5 = p_8 = 0.1$, some others moderate, and a couple of them high, i.e., $p_7 = 0.8$ and $p_{10} = 0.9$. For this setting, we distinguish the cases with and without optimized weights. The weights are optimized in order to minimize

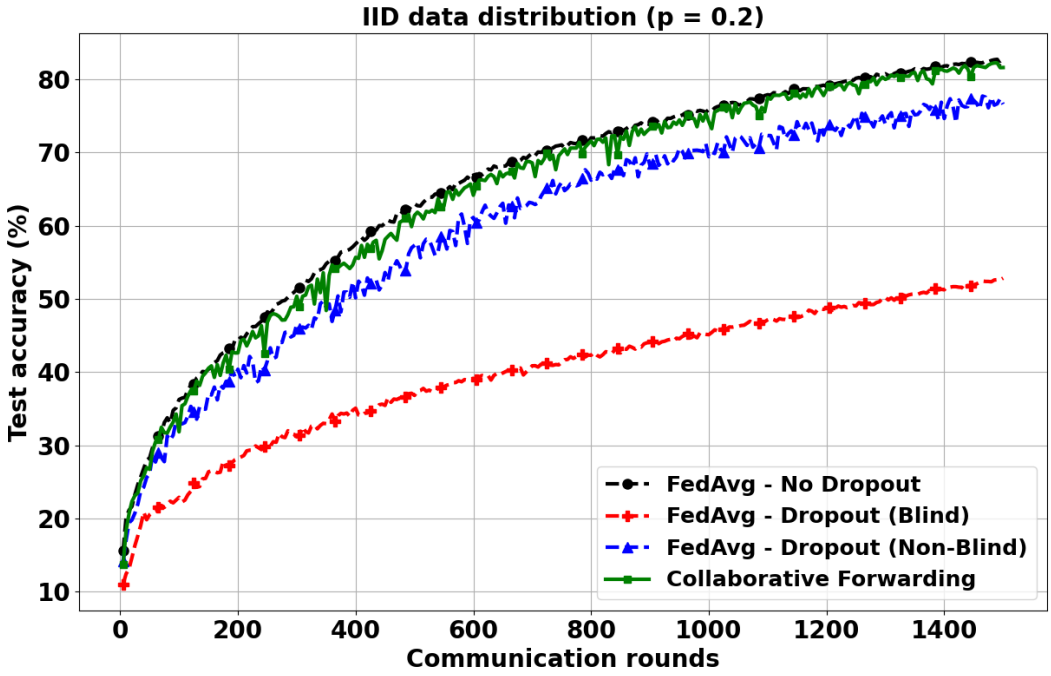

Fig. 3: Homogeneous connectivity with $p_i = 0.2, \forall i \in [n]$ and FC topology.

the term $S(\mathbf{p}, \mathbf{A})$, that consequently minimizes the variance of the iterates, subject to ensuring that the updates are unbiased according to Alg. 3. Note that explicitly optimizing the consensus weights that the clients use for their neighbors was not essential in Figs. 2 and 3, because the initial weights of Algorithms 3 and 4 are optimal for a FC topology with homogeneous connectivity, i.e., $p_i = p \; \forall i \in]n$. However this is not the case when client probabilities are different and different connection topologies are considered, namely fully-connected (Fig. 4) and ring (Fig. 5). For the ring topology, each client $i \in [n]$ is connected to clients $(i-1) \mod n$ and $(i+1) \mod n$. It is noteworthy that in Fig. 4, Collaborative Relaying with optimized weights outperforms Federated Averaging in the absence of dropouts. This is because we have not tuned the learning-rate optimally for either of these settings. The difference in performance of our proposed strategies and other naive strategies in the presence of intermittent connections to the PS is apparent in this scenario.

3) Heterogeneity in client data: Finally, in Fig. 6, we consider the setting where the training data is distributed across the clients in a non-IID fashion. For the ring topology in this plot, we

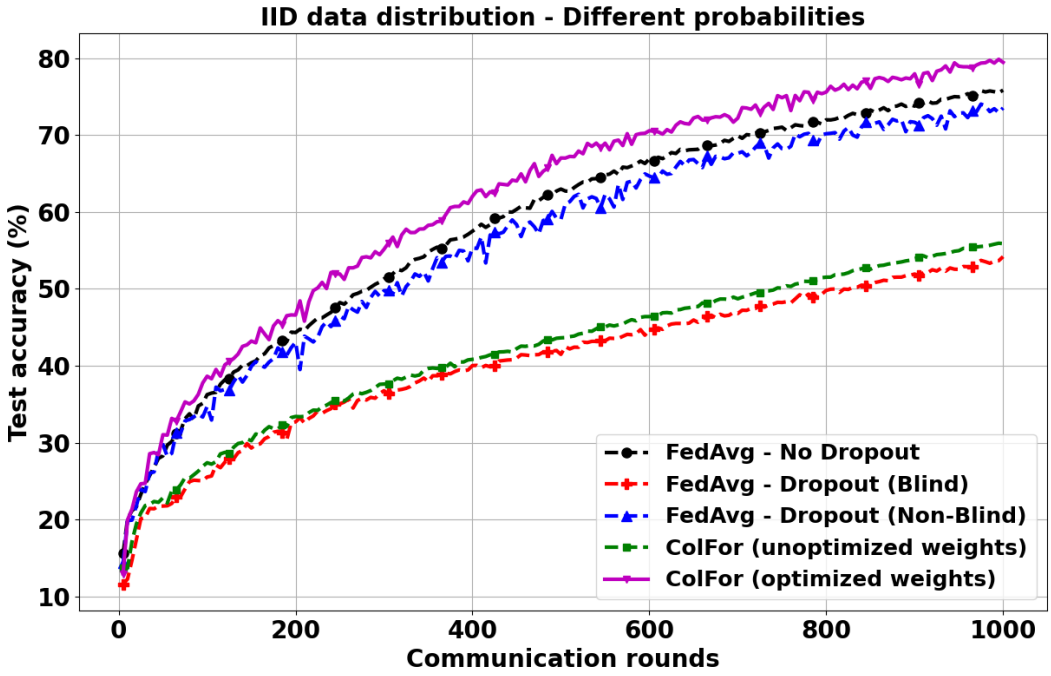

Fig. 4: Different transmission probabilities and FC topology.

have considered each client to be connected to 4 of its nearest neighbors. Remarkably, FedAvg (even with non-blind averaging) fails to converge in this setting. This is because in the absence of collaboration, clients that have important training samples that are critical for training a good model with high accuracy, may have a low probability of successful transmission and thus are rarely able to convey their updates to the PS. As a consequence, in the absence of collaboration, the global model fails to convergence, resulting in a test accuracy of 10% that is as good as a random classifier for 10 classes. Collaborative relaying ensures that the information from these critical datapoints are also conveyed to the PS even when the data owner does not have connectivity to the PS. This is the reason why the difference in performance is stark in a non-IID setting. We consider different clients have different probabilities of successfully transmitting their local update to the PS, and also simulate for fully-connected and ring topologies.

VI. Conclusion

Our goal in this paper is to mitigate the detrimental of clientss' intermittent connectivity to the PS impact on the training accuracy of FL systems. For this purpose, we proposed a

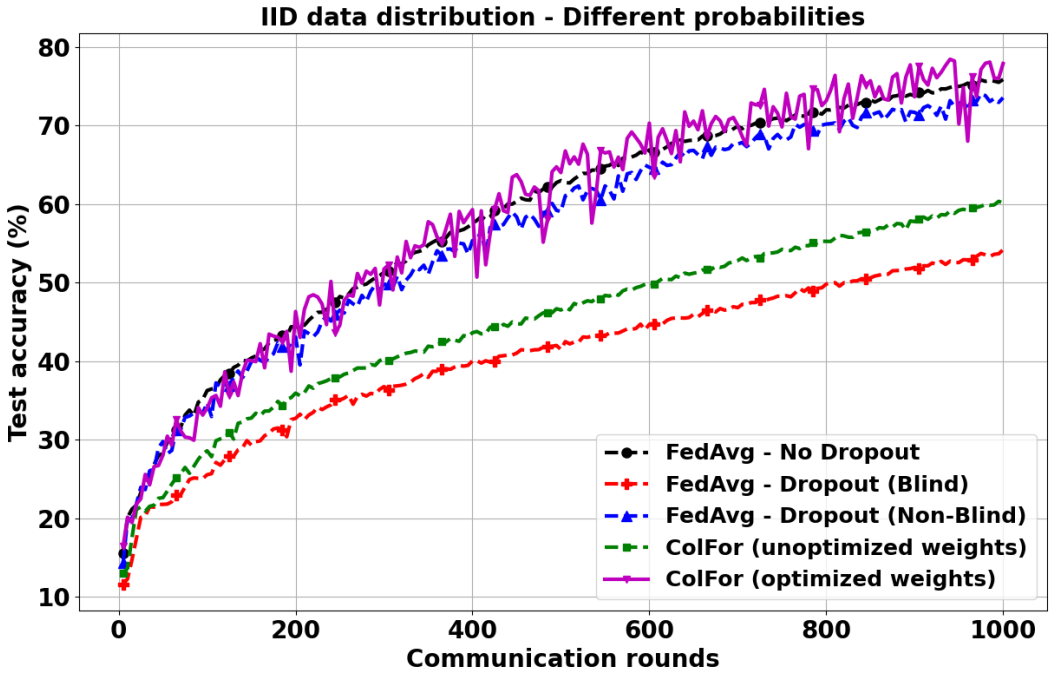

Fig. 5: Different transmission probabilities with a ring-topology.

collaborative relaying strategy, which exploits the connections between clients to relay potentially missing model updates to the PS due to blocked clients. Our algorithm allows the PS to receive an unbiased estimate of the model update, which would not be possible without relaying. We optimized the consensus weights at each client to improve the rate of convergence. Our proposed approach can be implemented even when the PS is blind to the identities of clients. Numerical results showed the improvement in performance, namely, training accuracy and convergence time, that our approach provides under various setting, including IID and non-IID data distributions, different communication graph topologies, as well as blind and non-blind PSs.

APPENDICES

APPENDIX A

PROOF OF THEOREM 1

Before proving Theorem 1, we first introduce the following notations and lemmas. Denote:

$$\overline{\mathbf{x}}^{(r+1)} = \frac{1}{n} \sum_{i \in [n]} \mathbf{x}_i^{(r,\mathcal{T})}. \tag{26}$$

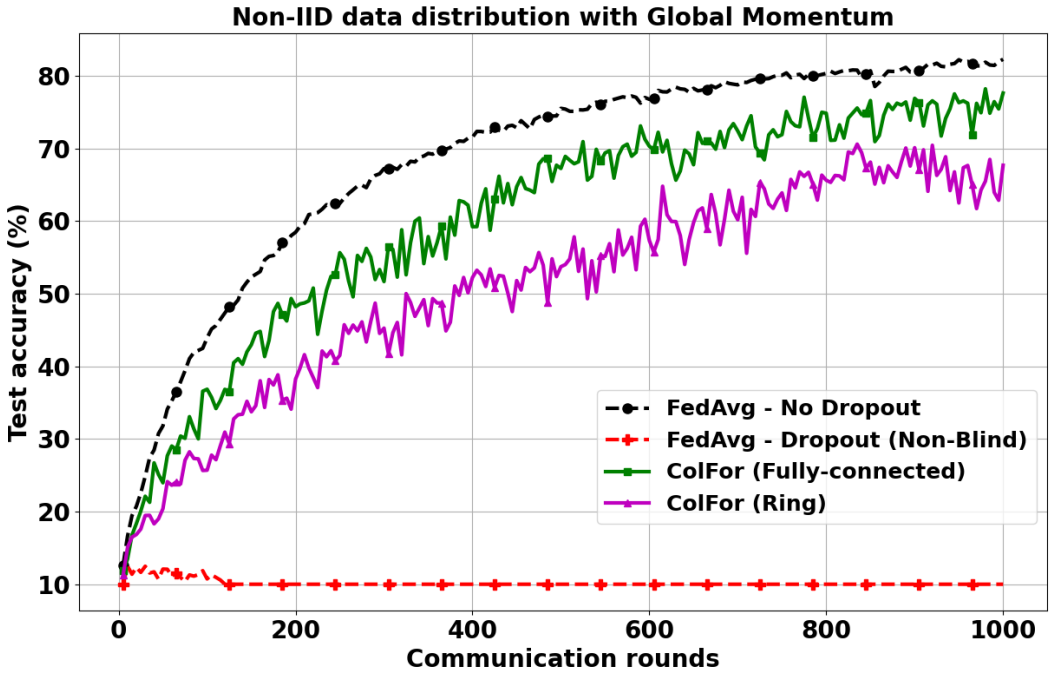

Fig. 6: Non-IID data distribution (Sort and Partition)

.

Lemma 3. *Under condition* (8)*, for every round* $r \geq 0$ *we have that:*

$$E\left\|\mathbf{x}^{(r+1)} - x^{\star}\right\|^2 = E\left\|\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}\right\|^2 + E\left\|\overline{\mathbf{x}}^{(r+1)} - \mathbf{x}^{\star}\right\|^2. \tag{27}$$

We prove Lemma 3 in Appendix B.

Lemma 4. *Under Assumptions 1-3 and condition* (8)*, for every* $r \geq 0$ *we have that*

$$E\left\|\overline{\mathbf{x}}^{(r+1)} - \mathbf{x}^{\star}\right\|^2 \leq$$

$$(1+n\eta_r^2)(1-\mu\eta_r)^{\mathcal{T}} E\|\mathbf{x}^{(r)} - \mathbf{x}^{\star}\|^2 + \mathcal{T}(\mathcal{T}-1)^2 L^2 \frac{\sigma^2}{n} e\eta_r^2 + \mathcal{T}^2 \frac{\sigma^2}{n}\eta_r^2 + \mathcal{T}^2(\mathcal{T}-1)^2 L^2 \sigma^2 e\eta_r^4, \tag{28}$$

for every positive η_r *such that* $\eta_r \leq \min\left\{\frac{\mu}{L^2}, \frac{1}{L\mathcal{T}}\right\}$.

Proof. The proof of this lemma follows directly from Lemma 2 in [58]. □

Lemma 5. *Under Assumptions 1-3 and condition* (8)*, for every* $r \geq 0$ *we have that*

$$E\left\|\mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\|^2 \leq 2\mathcal{T}^2 L^2 \eta_r^2 E\|\mathbf{x}^{(r)} - \mathbf{x}^{\star}\|^2 + 2\mathcal{T}^2\sigma^2\eta_r^2 + 2(\mathcal{T}-1)\mathcal{T}^2 L^2 \sigma^2 e\eta_r^4, \tag{29}$$

for every positive η_r *such that* $\eta_r \leq \min\left\{\frac{\mu}{L^2}, \frac{1}{L\mathcal{T}}\right\}$.

Proof. The proof of this lemma follows directly from (57) in the proof of [58, Lemma 3]. □

Lemma 6. *Under Assumptions 1-3 and condition* (8)*, for every* $r \geq 0$ *we have that*

$$E\left\|\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}\right\|^2 \leq \frac{2\mathcal{T}^2L^2\eta_r^2 E\|\mathbf{x}^{(r)} - \mathbf{x}^\star\|^2 + 2\mathcal{T}^2\sigma^2\eta_r^2 + 2(\mathcal{T}-1)\mathcal{T}^2L^2\sigma^2 e\eta_r^4}{n^2} S(\boldsymbol{p}, \mathbf{A}), \tag{30}$$

for every positive η_r *such that* $\eta_r \leq \min\left\{\frac{\mu}{L^2}, \frac{1}{L\mathcal{T}}\right\}$.

We prove Lemma 6 in Appendix C.

Proof of Theorem 1. By Lemmas 3-6 we have for every positive η_r such that $\eta_r \leq \min\left\{\frac{\mu}{L^2}, \frac{1}{L\mathcal{T}}\right\}$. the following:

$$\begin{aligned} &E\left\|\mathbf{x}^{(r+1)} - x^\star\right\|^2 \leq \\ &(1+n\eta_r^2)(1-\mu\eta_r)^{\mathcal{T}} E\|\mathbf{x}^{(r)} - \mathbf{x}^\star\|^2 + \mathcal{T}(\mathcal{T}-1)^2L^2\frac{\sigma^2}{n}e\eta_r^2 + \mathcal{T}^2\frac{\sigma^2}{n}\eta_r^2 + \mathcal{T}^2(\mathcal{T}-1)^2L^2\sigma^2 e\eta_r^4 \\ &\quad + \frac{2\mathcal{T}^2L^2\eta_r^2 E\|\mathbf{x}^{(r)} - \mathbf{x}^\star\|^2 + 2\mathcal{T}^2\sigma^2\eta_r^2 + 2(\mathcal{T}-1)\mathcal{T}^2L^2\sigma^2 e\eta_r^4}{n^2} S(\boldsymbol{p}, \mathbf{A}). \end{aligned} \tag{31}$$

Recall the notation $B(\boldsymbol{p}, \mathbf{A}) = \frac{2L^2}{n^2} S(\boldsymbol{p}, \mathbf{A})$ and denote

$$\begin{aligned} C(\eta_r, \boldsymbol{p}, \mathbf{A}) &= (1+n\eta_r^2)(1-\mu\eta_r)^{\mathcal{T}} + B(\boldsymbol{p}, \mathbf{A})\mathcal{T}^2\eta_r^2 \\ C_1(\boldsymbol{p}, \mathbf{A}) &= \frac{4^2}{\mu^2} \cdot \frac{2\sigma^2}{n^2} S(\boldsymbol{p}, \mathbf{A}) \\ C_2 &= \frac{4^2}{\mu^2} \cdot L^2\frac{\sigma^2}{n}e \\ C_3(\boldsymbol{p}, \mathbf{A}) &= \frac{4^4}{\mu^4} \cdot \left(L^2\sigma^2 e + \frac{2L^2\sigma^2 e}{n^2} S(\boldsymbol{p}, \mathbf{A})\right). \end{aligned} \tag{32}$$

Therefore,

$$\begin{aligned} &E\left\|\mathbf{x}^{(r+1)} - x^\star\right\|^2 \\ &\leq C(\eta_r, \boldsymbol{p}, \mathbf{A}) \cdot E\left\|\mathbf{x}^{(r)} - x^\star\right\|^2 + \left[\frac{\mu^2}{4^2} C_1(\boldsymbol{p}, \mathbf{A}) \cdot \mathcal{T}^2 + \frac{\mu^2}{4^2} C_2 \cdot \mathcal{T}(\mathcal{T}-1)^2\right]\eta_r^2 \\ &\qquad + \frac{\mu^4}{4^4} C_3(\boldsymbol{p}, \mathbf{A}) \cdot \mathcal{T}^2(\mathcal{T}-1)\eta_r^4. \end{aligned} \tag{33}$$

We can conclude the proof using a similar approach to that presented in [58]. First, we upper bound the term $C(\eta_r, \boldsymbol{p}, \mathbf{A})$. Since $\eta_r = \frac{4\mu^{-1}}{r\mathcal{T}+1} \leq \frac{1}{\mu}$ for every $r \geq r_0$, we can use the upper bound $\left(1 - \frac{x}{m}\right)^m \leq e^{-x}$ for every $x \leq m$:

$$(1 - \mu\eta_k)^{\mathcal{T}} = \left(1 - \frac{\mathcal{T}\mu\eta_k}{\mathcal{T}}\right) \leq e^{-\mu\mathcal{T}\eta_r}. \tag{34}$$

Now, using the bound $e^{-x} \leq 1 - x + x^2$ for every $x \geq 0$ can conclude that

$$(1 - \mu\eta_k)^{\mathcal{T}} \leq 1 - \mu\mathcal{T}\eta_r + \mu^2\mathcal{T}^2\eta_r^2. \tag{35}$$

It follows that

$$\begin{aligned} C(\eta_r, \boldsymbol{p}, \mathbf{A}) &\leq (1 + n\eta_r^2)(1 - \mu\mathcal{T}\eta_r + \mu^2\mathcal{T}^2\eta_r^2) + B(\boldsymbol{p}, \mathbf{A})\mathcal{T}^2\eta_r^2 \\ &= n\eta_r^2(1 - \mu\mathcal{T}\eta_r + \mu^2\mathcal{T}^2\eta_r^2) + 1 - \mu\mathcal{T}\eta_r + \mathcal{T}^2\eta_r^2(\mu^2 + B(\boldsymbol{p}, \mathbf{A}). \end{aligned} \tag{36}$$

Now, since $\eta_r = \frac{4\mu^{-1}}{r\mathcal{T}+1} \leq \frac{1}{\mu}$ and $r \geq r_0$ we have that

$$\mu^2 + B(\boldsymbol{p}, \mathbf{A}) \leq \frac{\mu}{4\eta_r}, \quad \eta_r \leq \frac{\mu\mathcal{T}}{4n}, \quad \text{and} \quad 0 \leq \mu\mathcal{T}\eta_r \leq 1. \tag{37}$$

The maximal value of the function $1 - x + x^2$ in the interval $x \in [0, 1]$ is 1, therefore, $1 - \mu\mathcal{T}\eta_r + \mu^2\mathcal{T}^2\eta_r^2 \leq 1$. It follows that $C(\eta_r, \boldsymbol{p}, \mathbf{A}) \leq 1 - \frac{1}{2}\mu\mathcal{T}\eta_r$ and

$$\begin{aligned} &E\left\|\mathbf{x}^{(r+1)} - x^\star\right\|^2 \\ &\leq \left(1 - \frac{1}{2}\mu\mathcal{T}\eta_r\right) \cdot E\left\|\mathbf{x}^{(r)} - x^\star\right\|^2 + \left[\frac{\mu^2}{4^2}C_1(\boldsymbol{p}, \mathbf{A}) \cdot \mathcal{T}^2\eta_r^2 + \frac{\mu^2}{4^2}C_2 \cdot \mathcal{T}(\mathcal{T}-1)^2\right]\eta_r^2] \\ &\quad + \frac{\mu^4}{4^4}C_3(\boldsymbol{p}, \mathbf{A}) \cdot \mathcal{T}^2(\mathcal{T}-1)\eta_r^4. \end{aligned} \tag{38}$$

Substituting $\eta_r = \frac{4\mu^{-1}}{r\mathcal{T}+1}$ we can conclude the proof by using [58, Lemma 5] when replacing k with r and using the constants $k_1 = \frac{1}{\mathcal{T}}$, $a = C_1(\boldsymbol{p}, \mathbf{A}) + C_2 \cdot \frac{(\mathcal{T}-1)^2}{\mathcal{T}}$ and $b = C_3 \cdot \frac{(\mathcal{T}-1)}{\mathcal{T}^2}$.

□

APPENDIX B
PROOF OF LEMMA 3

Proof of Lemma 3. Since $x \in \mathbb{R}^d$, for every $r \geq 0$ we have that

$$E\left\|\mathbf{x}^{(r+1)} - x^\star\right\|^2 = E\left\|\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}\right\|^2 + E\left\|\overline{\mathbf{x}}^{(r+1)} - \mathbf{x}^\star\right\|^2 + 2E\left(\left\langle\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}, \overline{\mathbf{x}}^{(r+1)} - \mathbf{x}^\star\right\rangle\right). \tag{39}$$

By the law of total expectation to have that

$$E\left(\langle \mathbf{x}^{(r+1)}-\overline{\mathbf{x}}^{(r+1)}, \overline{\mathbf{x}}^{(r+1)}-\mathbf{x}^{\star}\rangle\right) = E\left[E\left(\left\langle \mathbf{x}^{(r+1)}-\overline{\mathbf{x}}^{(r+1)}, \overline{\mathbf{x}}^{(r+1)}-\mathbf{x}^{\star}\right\rangle \Big| \left(\mathbf{x}_i^{(r,\mathcal{T})}\right)_{i\in[n]}, \mathbf{x}^{(r)}\right)\right]$$
$$= E\left[\left\langle E\left(\mathbf{x}^{(r+1)}\Big| \left(\mathbf{x}_i^{(r,\mathcal{T})}\right)_{i\in[n]}, \mathbf{x}^{(r)}\right) - \overline{\mathbf{x}}^{(r+1)}, \overline{x}^{(r+1)}-\mathbf{x}^{\star}\right\rangle\right], \tag{40}$$

where the second equality follows since the $\overline{\mathbf{x}}^{(r+1)}$ and $\mathbf{x}^{\star}$ are deterministic functions of $\left(\mathbf{x}_i^{(r,\mathcal{T})}\right)_{i\in[n]}$. Now we calculate the conditional expectation

$$\begin{aligned}
&E\left(\mathbf{x}^{(r+1)}\Big| \left(\mathbf{x}_i^{(r,\mathcal{T})}\right)_{i\in[n]}, \mathbf{x}^{(r)}\right)\\
&= E\left(\mathbf{x}^{(r+1)}\Big| \left(\mathbf{x}_i^{(r,\mathcal{T})}\right)_{i\in[n]}, \mathbf{x}^{(r)}\right)\\
&= E\left(\mathbf{x}_i^{(r)} + \frac{1}{n}\sum_{i\in[n]}\tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\left(\mathbf{x}_j^{(r,\mathcal{T})}-\mathbf{x}^{(r)}\right)\Big| \left(\mathbf{x}_i^{(r,\mathcal{T})},\right)_{i\in[n]}, \mathbf{x}^{(r)}\right)\\
&= \mathbf{x}^{(r)} - \frac{1}{n}\sum_{i\in[n]}E\left(\tau_i(r+1)\right)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}_j^{(r)} + \frac{1}{n}\sum_{i\in[n]}E\left(\tau_i(r+1)\right)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}_j^{(r,\mathcal{T})}\\
&= \mathbf{x}^{(r)} - \frac{1}{n}\sum_{i\in[n]}p_i\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}_j^{(r)} + \frac{1}{n}\sum_{i\in[n]}p_i\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}_j^{(r,\mathcal{T})}\\
&\overset{(a)}{=} \frac{1}{n}\sum_{i\in[n]}\mathbf{x}_i^{(r,\mathcal{T})} \triangleq \overline{\mathbf{x}}^{(r+1)},
\end{aligned} \tag{41}$$

where the equality (a) follows from (8). □

APPENDIX C
PROOF OF LEMMA 6

Proof of Lemma 6. First, we observe that

$$\begin{aligned}
&E\left\|\mathbf{x}^{(r+1)}-\overline{\mathbf{x}}^{(r+1)}\right\|^2\\
&= E\left\|\mathbf{x}^{(r)} + \frac{1}{n}\sum_{i\in[n]}\tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\left(\mathbf{x}_j^{(r,\mathcal{T})}-\mathbf{x}^{(r)}\right) - \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_j^{(r,\mathcal{T})}\right\|^2\\
&= E\left\|\mathbf{x}^{(r)} - \frac{1}{n}\sum_{i\in[n]}\tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}^{(r)} + \frac{1}{n}\sum_{i\in[n]}\tau_i(r+1)\sum_{j\in\mathcal{N}_i\cup\{i\}}\alpha_{ij}\mathbf{x}_j^{(r,\mathcal{T})} - \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^{(r,\mathcal{T})}\right\|^2\\
&= E\left\|\mathbf{x}^{(r)} - \frac{1}{n}\sum_{i\in[n]}\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\mathbf{x}^{(r)} + \frac{1}{n}\sum_{i\in[n]}\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\mathbf{x}_i^{(r,\mathcal{T})} - \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i^{(r,\mathcal{T})}\right\|^2
\end{aligned}$$

$$= E\left\|\frac{1}{n}\sum_{i\in[n]}\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)\mathbf{x}_i^{(r,\mathcal{T})}-\frac{1}{n}\sum_{i\in[n]}\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)\mathbf{x}^{(r)}\right\|^2$$

$$= \frac{1}{n^2}E\left\|\sum_{i\in[n]}\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)\left(\mathbf{x}_i^{(r,\mathcal{T})}-\mathbf{x}^{(r)}\right)\right\|^2$$

$$= \frac{1}{n^2}\sum_{i\in[n]}E\left[\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)^2\right]E\left\|\left(\mathbf{x}_i^{(r,\mathcal{T})}-\mathbf{x}^{(r)}\right)\right\|^2$$

$$+\frac{1}{n^2}\sum_{i\neq l}E\left[\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)\left(\sum_{m:m\in\mathcal{N}_l\cup\{l\}}\tau_m(r+1)\alpha_{ml}-1\right)\right]$$

$$\cdot E\left[\left\langle\mathbf{x}_i^{(r,\mathcal{T})}-\mathbf{x}^{(r)},\mathbf{x}_l^{(r,\mathcal{T})}-\mathbf{x}^{(r)}\right\rangle\right], \tag{42}$$

where the last equality follows since the random variables $\tau_i(r+1),\ i\in[n]$ are statistically independent of the random vectors $\mathbf{x}_i^{(r,\mathcal{T})},\ i\in[n]$.

Now,

$$E\left[\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)^2\right]$$

$$= \sum_{j:j\in\mathcal{N}_i\cup\{i\}}E\left(\tau_j^2(r+1)\alpha_{ji}^2\right)+\sum_{\substack{j_1\neq j_2\\ j_1,j_2\in\mathcal{N}_i\cup\{i\}}}E\left[\tau_{j_1}(r+1)\tau_{j_2}(r+1)\alpha_{j_1i}\alpha_{j_2i}\right]-2\sum_{j:j\in\mathcal{N}_i\cup\{i\}}E\left(\tau_j(r+1)\alpha_{ji}\right)+1$$

$$= \sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji}^2+\sum_{\substack{j_1\neq j_2\\ j_1,j_2\in\mathcal{N}_i\cup\{i\}}}p_{j_1}p_{j_2}\alpha_{j_1i}\alpha_{j_2i}-2\underbrace{\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji}}_{=1}+1$$

$$= \sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji}^2v+\underbrace{\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji}\right)^2}_{=1}-\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j^2\alpha_{ji}^2-1$$

$$= \sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j(1-p_j)\alpha_{ji}^2. \tag{43}$$

Similarly,

$$E\left[\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}-1\right)\left(\sum_{m:m\in\mathcal{N}_l\cup\{l\}}\tau_m(r+1)\alpha_{ml}-1\right)\right]$$

$$= E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\sum_{m:m\in\mathcal{N}_l\cup\{l\}}\tau_j(r+1)\tau_m(r+1)\alpha_{ji}\alpha_{ml}\right]-\underbrace{E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\tau_j(r+1)\alpha_{ji}\right]}_{=1}$$

$$
\begin{aligned}
&\underbrace{- E\left[\sum_{m:m\in\mathcal{N}_l\cup\{l\}} \tau_m(r+1)\alpha_{ml}\right]}_{=1} +1 \\
&= E\left[\sum_{j:j\in\mathcal{N}_{il}} \tau_j^2(r+1)\alpha_{ji}\alpha_{jl}\right] + E\left[\sum_{j:j\in\mathcal{N}_i\cup\{i\}}\sum_{\substack{m:m\in\mathcal{N}_l\cup\{l\},\\ m\neq j}} \tau_j(r+1)\tau_m(r+1)\alpha_{ji}\alpha_{ml}\right] - 1 \\
&= \sum_{j:j\in\mathcal{N}_{il}} p_j\alpha_{ji}\alpha_{jl} + \sum_{j:j\in\mathcal{N}_i\cup\{i\}}\sum_{\substack{m:m\in\mathcal{N}_l\cup\{l\},\\ m\neq j}} p_jp_m\alpha_{ji}\alpha_{ml} - 1 \\
&= \sum_{j:j\in\mathcal{N}_{il}} p_j\alpha_{ji}\alpha_{jl} + \left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}} p_j\alpha_{ji}\right)\left(\sum_{m:m\in\mathcal{N}_l\cup\{l\}} p_m\alpha_{ml}\right) - \sum_{j:j\in\mathcal{N}_{il}} p_j^2\alpha_{ji}\alpha_{jl} - 1 \\
&= \sum_{j:j\in\mathcal{N}_{il}} p_j(1-p_j)\alpha_{ji}\alpha_{jl}. \qquad (44)
\end{aligned}
$$

Therefore,

$$
\begin{aligned}
&E\left\|\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}\right\|^2 \\
&\quad= \frac{1}{n^2}\sum_{i\in[n]}\sum_{j:j\in\mathcal{N}_i\cup\{i\}} p_j(1-p_j)\alpha_{ji}^2 E\left\|\left(\mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right)\right\|^2 \\
&\quad+ \frac{1}{n^2}\sum_{i\neq l}\sum_{j:j\in\mathcal{N}_{il}} p_j(1-p_j)\alpha_{ji}\alpha_{jl}\cdot E\left[\left\langle \mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}, \mathbf{x}_l^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\rangle\right] \\
&\quad= \frac{1}{n^2}\sum_{i,l\in[n]}\sum_{j:j\in\mathcal{N}_{il}} p_j(1-p_j)\alpha_{ji}\alpha_{jl}\cdot E\left[\left\langle \mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}, \mathbf{x}_l^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\rangle\right]. \qquad (45)
\end{aligned}
$$

By Lemma 5 and the Cauchy–Schwarz inequality we have for each $i \in [n]$ that

$$
\begin{aligned}
\left|E\left[\left\langle \mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}, \mathbf{x}_l^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\rangle\right]\right| &\leq \sqrt{E\left\|\mathbf{x}_i^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\|^2}\cdot\sqrt{E\left\|\mathbf{x}_l^{(r,\mathcal{T})} - \mathbf{x}^{(r)}\right\|^2} \\
&\leq 2\mathcal{T}^2L^2\eta_r^2 E\|\mathbf{x}^{(r)} - \mathbf{x}^\star\|^2 + 2\mathcal{T}^2\sigma^2\eta_r^2 + 2(\mathcal{T}-1)\mathcal{T}^2L^2\sigma^2 e\eta_r^4. \qquad (46)
\end{aligned}
$$

Consequently, it follows from the definition of $S(\boldsymbol{p}, \mathbf{A})$ that

$$
E\left\|\mathbf{x}^{(r+1)} - \overline{\mathbf{x}}^{(r+1)}\right\|^2 \leq \frac{2\mathcal{T}^2L^2\eta_r^2 E\|\mathbf{x}^{(r)} - \mathbf{x}^\star\|^2 + 2\mathcal{T}^2\sigma^2\eta_r^2 + 2(\mathcal{T}-1)\mathcal{T}^2L^2\sigma^2 e\eta_r^4}{n^2} S(\boldsymbol{p}, \mathbf{A}). \qquad (47)
$$

□

APPENDIX D
PROOF OF LEMMA 2

To prove Lemma 2 we first present the following auxiliary result.

Lemma 7. *Let $\mathbf{y} \in \mathbb{R}^{\tilde{d}}$ where $\tilde{d}$ is some positive integer. Denote $h(\mathbf{y}) = \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)^2$, then $h(\mathbf{y})$ is convex.*

Proof of Lemma 7. We prove this by using the definition of a convex function, namely showing that

$$h(\lambda \mathbf{y} + (1-\lambda)\mathbf{z}) \leq \lambda h(\mathbf{y}) + (1-\lambda) h(\mathbf{z}),$$

for every $\mathbf{y}, \mathbf{z} \in \mathbb{R}^{\tilde{d}}$ and $\lambda \in [0,1]$:

$$\begin{aligned}
&\lambda h(\mathbf{y}) + (1-\lambda) h(\mathbf{z}) - h(\lambda \mathbf{y} + (1-\lambda)\mathbf{z}) \\
&= \lambda \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)^2 + (1-\lambda) \left(\sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right)^2 - \lambda^2 \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)^2 \\
&\qquad - 2\lambda(1-\lambda) \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)\left(\sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right) - (1-\lambda)^2 \left(\sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right)^2 \\
&= \lambda(1-\lambda) \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)^2 + \lambda(1-\lambda) \left(\sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right)^2 - 2\lambda(1-\lambda) \left(\sum_{i=1}^{\tilde{d}} \mathbf{y}_i\right)\left(\sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right) \\
&= \lambda(1-\lambda) \left[\sum_{i=1}^{\tilde{d}} \mathbf{y}_i - \sum_{i=1}^{\tilde{d}} \mathbf{z}_i\right]^2 \geq 0, \qquad (48)
\end{aligned}$$

where the last inequality follows since $\mathbf{y}, \mathbf{z} \in \mathbb{R}^{\tilde{d}}$ and $\lambda \in [0,1]$. □

Proof of Lemma 2. We prove that $S(\boldsymbol{p}, \mathbf{A})$ is convex with respect to $\mathbf{A}$ by rewriting it as a nonnegative weighted sum of convex quadratic function.

$$\begin{aligned}
\sum_{i,l \in [n]} \sum_{j : j \in \mathcal{N}_{il}} p_j (1-p_j) \alpha_{ji} \alpha_{jl} &= \sum_{j \in [n]} p_j (1-p_j) \sum_{i,l \in [n] : j \in \mathcal{N}_{il}} \alpha_{ji} \alpha_{jl} \\
&= \sum_{j \in [n]} p_j (1-p_j) \left(\sum_{i \in [n] : j \in \mathcal{N}_i \cup \{i\}} \alpha_{ji}\right)^2, \qquad (49)
\end{aligned}$$

where the equality follows since if $j \in \mathcal{N}_i \cup \{i\}$ and $j \in \mathcal{N}_l \cup \{l\}$, then $j \in \mathcal{N}_{il}$. Now, since the function $\left(\sum_{i \in [n] : j \in \mathcal{N}_i \cup \{i\}} \alpha_{ji}\right)^2$ is convex in $\mathbf{A}$ (see Lemma 7) and $p_j \in [0,1]$ for every $j \in [n]$, the function $S(\boldsymbol{p}, \mathbf{A})$ is convex. □

APPENDIX E
SOLVING THE OPTIMIZATION PROBLEM (24)

First we observe that $p_j(1-p_j)=0$ whenever $p_j=0$ or $p_j=1$. Therefore, we can set $\alpha_{ji}=0$ for every $j\notin\mathcal{N}_i\cup\{i\}$ and k such that $p_j=0$. Additionally, if $p_j=1$ then we can set $\alpha_{ji}=1/\sum_{k\in[n]}\mathbb{1}_{\{p_k=1,k\in\mathcal{N}_i\cup\{i\}\}}$. Therefore, hereafter we assume that $j\in(\mathcal{N}_i\cup\{i\})\cap\{k:p_k\in(0,1)\}$. The Lagrangian of (24) is

$$L(\mathbf{A}_i^{(\ell)},\lambda_i)=\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j(1-p_j)\alpha_{ji}^2+2\sum_{l\in[n],l\neq i}\sum_{j:j\in\mathcal{N}_{il}}p_j(1-p_j)\alpha_{ji}\alpha_{jl}^{(\ell-1)}$$
$$-\lambda_i\left(\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji}-1\right)-\mu_{ji}\alpha_{ji}. \tag{50}$$

Additionally,

$$\frac{\partial L(\mathbf{A}_i^{(\ell)},\lambda_i)}{\partial\alpha_{ji}}=2p_j(1-p_j)\alpha_{ji}+2p_j(1-p_j)\sum_{l\in L_{ji}}\alpha_{jl}^{(\ell-1)}-\lambda_i p_j+\mu_{ji} \tag{51}$$

$$\frac{\partial L(\mathbf{A}_i^{(\ell)},\lambda_i)}{\partial\lambda_i}=1-\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\alpha_{ji} \tag{52}$$

$$\frac{\partial L(\mathbf{A}_i^{(\ell)},\lambda_i)}{\partial\mu_{ji}}=-\alpha_{ji}. \tag{53}$$

Recall that $\beta_{ji}=\sum_{l\in L_{ji}}\alpha_{jl}^{(\ell-1)}$. In this case, it follows from the Karush–Kuhn–Tucker (KKT) conditions that

$$\alpha_{ji}=-\beta_{ji}+\frac{\lambda_i}{2(1-p_j)}+\frac{\mu_{ji}}{2(1-p_j)}=\left(-\beta_{ji}+\frac{\lambda_i}{2(1-p_j)}\right)^+, \tag{54}$$

and $\lambda_i\geq 0$ is set such that $\sum_{j:j\in\mathcal{N}_i\cup\{i\}}p_j\left(-\beta_{ji}+\frac{\lambda_i}{2(1-p_j)}\right)^+=1$.

REFERENCES

[1] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, vol. 54. Fort Lauderdale, FL, USA: PMLR, Apr 2017, pp. 1273–1282.

[2] M. Chen, D. Gündüz, K. Huang, W. Saad, M. Bennis, A. V. Feljan, and H. V. Poor, "Distributed learning in wireless networks: Recent progress and future challenges," arxiv:2104.02151, 2021.

[3] M. R. Akdeniz, Y. Liu, M. K. Samimi, S. Sun, S. Rangan, T. S. Rappaport, and E. Erkip, "Millimeter wave channel modeling and cellular capacity evaluation," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1164–1179, June 2014.

[4] M. Gapeyenko, A. Samuylov, M. Gerasimenko, D. Moltchanov, S. Singh, E. Aryafar, S. Yeh, N. Himayat, S. Andreev, and Y. Koucheryavy, "Analysis of human-body blockage in urban millimeter-wave cellular communications," in *2016 IEEE International Conference on Communications (ICC)*, May 2016, pp. 1–7.

[5] M. Gapeyenko, A. Samuylov, M. Gerasimenko, D. Moltchanov, S. Singh, M. R. Akdeniz, E. Aryafar, N. Himayat, S. Andreev, and Y. Koucheryavy, "On the temporal effects of mobile blockers in urban millimeter-wave cellular scenarios," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 11, pp. 10 124–10 138, Nov 2017.

[6] Y. Yan and Y. Mostofi, "Co-optimization of communication and motion planning of a robotic operation under resource constraints and in fading environments," *IEEE Trans. Wireless Commun.*, vol. 12, no. 4, pp. 1562–1572, April 2013.

[7] M. M. Zavlanos, M. B. Egerstedt, and G. J. Pappas, "Graph-theoretic connectivity control of mobile robot networks," *Proc. IEEE*, vol. 99, no. 9, pp. 1525–1540, Sep. 2011.

[8] N. Michael, M. M. Zavlanos, V. Kumar, and G. J. Pappas, "Maintaining connectivity in mobile robot networks," in *Experimental Robotics*, 2009.

[9] S. Gil, S. Kumar, D. Katabi, and D. Rus, "Adaptive communication in multi-robot systems using directionality of signal strength," *The International Journal of Robotics Research*, vol. 34, no. 7, pp. 946–968, 2015.

[10] D. Gündüz, D. B. Kurka, M. Jankowski, M. M. Amiri, E. Ozfatura, and S. Sreekumar, "Communicate to learn at the edge," *IEEE Comm. Magazine*, vol. 58, no. 12, pp. 14–19, 2020.

[11] M. E. Ozfatura, J. Zhao, and D. Gündüz, "Fast federated edge learning with overlapped communication and computation and channel-aware fair client scheduling," in *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2021, pp. 311–315.

[12] D. Liu, G. Zhu, J. Zhang, and K. Huang, "Data-importance aware user scheduling for communication-efficient edge machine learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 265–278, 2021.

[13] W. Xia, T. Q. S. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu, "Multi-armed bandit based client scheduling for federated learning," *IEEE Transactions on Wireless Communications*, pp. 1–1, 2020.

[14] H. H. Yang, A. Arafa, T. Q. S. Quek, and H. Vincent Poor, "Age-based scheduling policy for federated learning in mobile edge networks," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8743–8747.

[15] M. M. Amiri and D. Gündüz, "Federated learning over wireless fading channels," *IEEE Trans. Wireless Comms.*, vol. 19, no. 5, pp. 3546–3557, 2020.

[16] E. Ozfatura, S. Rini, and D. Gündüz, "Decentralized sgd with over-the-air computation," in *GLOBECOM 2020 - 2020 IEEE Global Communications Conference*, 2020, pp. 1–6.

[17] G. Zhu, Y. Wang, and K. Huang, "Broadband analog aggregation for low-latency federated edge learning (extended version)," arxiv:1812.11494, 2019.

[18] B. Hasircioglu and D. Gunduz, "Private wireless federated learning with anonymous over-the-air computation," arxiv:2011.08579, 2021.

[19] M. S. E. Mohamed, W.-T. Chang, and R. Tandon, "Privacy amplification for federated learning via user sampling and wireless aggregation," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3821–3835, 2021.

[20] M. Yemini, S. Gil, and A. J. Goldsmith, "Exploiting local and cloud sensor fusion in intermittently connected sensor networks," in *2020 IEEE Global Communications Conference (Globecom)*, December 2020.

[21] ——, "Cloud-cluster architecture for detection in intermittently connected sensor networks," submitted, October 2021. [Online]. Available: arXiv:2110.01119

[22] M. S. H. Abad, E. Ozfatura, D. Gündüz, and O. Ercetin, "Hierarchical federated learning across heterogeneous cellular

networks," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8866–8870.

[23] 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," in *NIPS*, Dec. 2017.

[24] H. Tang, X. Lian, M. Yan, C. Zhang, and J. Liu, "d^2: Decentralized training over decentralized data," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. Stockholmsmässan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 4848–4856.

[25] Z. Jiang, A. Balu, C. Hegde, and S. Sarkar, "Collaborative deep learning in fixed topology networks," in *NIPS*, Dec. 2017.

[26] K. Yuan, Q. Ling, and W. Yin, "On the convergence of decentralized gradient descent," *SIAM Journal on Optimization*, 2016.

[27] M. Kamp, L. Adilova, J. Sicking, F. Hüger, P. Schlicht, T. Wirtz, and S. Wrobel, "Efficient decentralized deep learning by dynamic model averaging," in *Machine Learning and Knowledge Discovery in Databases*, M. Berlingerio, F. Bonchi, T. Gärtner, N. Hurley, and G. Ifrim, Eds. Cham: Springer International Publishing, 2019, pp. 393–409.

[28] J. Zeng and W. Yin, "On nonconvex decentralized gradient descent," *IEEE Transactions on Signal Processing*, vol. 66, no. 11, pp. 2834–2848, June 2018.

[29] L. Kong, T. Lin, A. Koloskova, M. Jaggi, and S. U. Stich, "Consensus control for decentralized deep learning," arXiv:2102.04828, 2021.

[30] T. Vogels, L. He, A. Koloskova, S. P. Karimireddy, T. Lin, S. U. Stich, and M. Jaggi, "Relaysum for decentralized deep learning on heterogeneous data," in *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. [Online]. Available: https://openreview.net/forum?id=Qo6kYy4SBI-

[31] R. Saha, S. Rini, M. Rao, and A. J. Goldsmith, "Decentralized optimization over noisy, rate-constrained networks: Achieving consensus by communicating differences," *IEEE Journal on Selected Areas in Communications*, vol. 40, no. 2, pp. 449–467, 2022.

[32] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. Stich, "A unified theory of decentralized SGD with changing topology and local updates," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 5381–5393. [Online]. Available: https://proceedings.mlr.press/v119/koloskova20a.html

[33] J. Wang, A. K. Sahu, Z. Yang, G. Joshi, and S. Kar, "MATCHA: speeding up decentralized SGD via matching decomposition sampling," arxiv:1905.09435, 2019.

[34] M. Assran, N. Loizou, N. Ballas, and M. Rabbat, "Stochastic gradient push for distributed deep learning," in *Proceedings of the 36th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri and R. Salakhutdinov, Eds., vol. 97. PMLR, 09–15 Jun 2019, pp. 344–353. [Online]. Available: https://proceedings.mlr.press/v97/assran19a.html

[35] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[36] W. Y. B. Lim, J. S. Ng, Z. Xiong, J. Jin, Y. Zhang, D. Niyato, C. Leung, and C. Miao, "Decentralized edge intelligence: A dynamic resource allocation framework for hierarchical federated learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 33, no. 3, pp. 536–550, 2022.

[37] T. Castiglia, A. Das, and S. Patterson, "Multi-level local {sgd}: Distributed {sgd} for heterogeneous hierarchical networks," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=C70cp4Cn32

[38] F. P.-C. Lin, S. Hosseinalipour, S. S. Azam, C. G. Brinton, and N. Michelusi, "Semi-decentralized federated learning with cooperative d2d local model aggregations," *IEEE Journal on Selected Areas in Communications*, vol. 39, no. 12, pp. 3851–3869, 2021.

[39] Anonymous, "Hybrid local SGD for federated learning with heterogeneous communications," in *Submitted to The Tenth International Conference on Learning Representations*, 2022, under review. [Online]. Available: https://openreview.net/forum?id=H0oaWl6THa

[40] H. Yang, M. Fang, and J. Liu, "Achieving linear speedup with partial worker participation in non-IID federated learning," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=jDdzh5ul-d

[41] X. Gu, K. Huang, J. Zhang, and L. Huang, "Fast federated learning in the presence of arbitrary device unavailability," arxiv:2106.04159, 2021.

[42] T. H. Hsu, H. Qi, and M. Brown, "Measuring the effects of non-identical data distribution for federated visual classification," *CoRR*, vol. abs/1909.06335, 2019.

[43] T.-M. H. Hsu, H. Qi, and M. Brown, "Federated visual classification with real-world data distribution," *CoRR*, vol. abs/2003.08082, 2020.

[44] K. Hsieh, A. Phanishayee, O. Mutlu, and P. Gibbons, "The non-IID data quagmire of decentralized machine learning," in *Proceedings of the 37th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 119. Virtual: PMLR, 13–18 Jul 2020, pp. 4387–4398.

[45] Y. Zhao, M. Li, L. Lai, N. Suda, D. Civin, and V. Chandra, "Federated learning with non-iid data," *CoRR*, vol. abs/1806.00582, 2018.

[46] E. Ozfatura, D. Gunduz, and H. V. Poor, "Collaborative learning over wireless networks: An introductory overview," arxiv:2112.05559, 2021.

[47] H. H. Yang, Z. Liu, T. Q. S. Quek, and H. V. Poor, "Scheduling policies for federated learning in wireless networks," *IEEE Transactions on Communications*, vol. 68, no. 1, pp. 317–333, 2020.

[48] W. Shi, S. Zhou, and Z. Niu, "Device scheduling with fast convergence for wireless federated learning," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–6.

[49] M. M. Amiri, D. Gündüz, S. R. Kulkarni, and H. V. Poor, "Convergence of update aware device scheduling for federated learning at the wireless edge," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3643–3658, 2021.

[50] S. Li, M. A. Maddah-Ali, and A. S. Avestimehr, "Fundamental tradeoff between computation and communication in distributed computing," in *2016 IEEE International Symposium on Information Theory (ISIT)*, 2016, pp. 1814–1818.

[51] R. Tandon, Q. Lei, A. G. Dimakis, and N. Karampatziakis, "Gradient coding: Avoiding stragglers in distributed learning," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 70, Aug 2017, pp. 3368–3376.

[52] K. Lee, M. Lam, R. Pedarsani, D. Papailiopoulos, and K. Ramchandran, "Speeding up distributed machine learning using codes," *IEEE Transactions on Information Theory*, vol. 64, no. 3, pp. 1514–1529, 2018.

[53] E. Ozfatura, S. Ulukus, and D. Gündüz, "Straggler-aware distributed learning: Communication–computation latency trade-off," *Entropy*, vol. 22, no. 5, 2020.

[54] D. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.

[55] A. Krizhevsky, V. Nair, and G. Hinton, "Cifar-10 (canadian institute for advanced research)." [Online]. Available: http://www.cs.toronto.edu/~kriz/cifar.html

[56] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[57] J. Wang, V. Tantia, N. Ballas, and M. Rabbat, "Slowmo: Improving communication-efficient distributed sgd with slow momentum," in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=SkxJ8REYPH

[58] A. Reisizadeh, A. Mokhtari, H. Hassani, A. Jadbabaie, and R. Pedarsani, "FedPAQ: A communication-efficient federated learning method with periodic averaging and quantization," in *The 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, August 2020.