# Resource-Constrained Decentralized Federated Learning via Personalized Event-Triggering

Shahryar Zehtabi, *Graduate Student Member, IEEE*, Seyyedali Hosseinalipour, *Senior Member, IEEE*, and Christopher G. Brinton, *Senior Member, IEEE*

***Abstract*—Federated learning (FL) is a popular technique for distributing machine learning (ML) across a set of edge devices. In this paper, we study fully decentralized FL, where in addition to devices conducting training locally, they carry out model aggregations via cooperative consensus formation over device-to-device (D2D) networks. We introduce asynchronous, event-triggered communications among the devices to handle settings where access to a central server is not feasible. To account for the inherent resource heterogeneity and statistical diversity challenges in FL, we define personalized communication triggering conditions at each device that weigh the change in local model parameters against the available local network resources. We theoretically recover the $\mathcal{O}(\ln k/\sqrt{k})$ convergence rate to the globally optimal model of decentralized gradient descent (DGD) methods in the setup of our methodology. We provide our convergence guarantees for the last iterates of models, under relaxed graph connectivity and data heterogeneity assumptions compared with the existing literature. To do so, we demonstrate a $B$-connected information flow guarantee in the presence of sporadic communications over the time-varying D2D graph. Our subsequent numerical evaluations demonstrate that our methodology obtains substantial improvements in convergence speed and/or communication savings compared to existing decentralized FL baselines.**

***Index Terms*—Federated learning, decentralized learning, event-triggered communications.**

## I. Introduction

FEDERATED learning (FL) has emerged as a popular technique to distribute machine learning (ML) model training across a network of devices [2]. With initial deployments including next-word prediction across mobile devices [3], FL is envisioned to serve many intelligence applications in edge/fog computing and the Internet of Things (IoT) [4].

Received 15 January 2025; revised 11 October 2025; accepted 17 November 2025; approved by IEEE TRANSACTIONS ON NETWORKING Editor Y. Liang. This work was supported in part by the National Science Foundation (NSF) under Grant CPS-2313109 and Grant ECCS-2512911, in part by the Defense Advanced Research Projects Agency (DARPA) under Grant D22AP00168, and in part by the Office of Naval Research (ONR) under Grant N00014-22-1-2305. A preliminary version of this work [DOI: 10.1109/CDC51059.2022.9993258] appeared in the 2022 IEEE Conference on Decision and Control (CDC). *(Corresponding author: Shahryar Zehtabi.)*

Shahryar Zehtabi and Christopher G. Brinton are with the Elmore Family School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: szehtabi@purdue.edu; cgb@purdue.edu).

Seyyedali Hosseinalipour is with the Department of Electrical Engineering, State University of New York at Buffalo, Buffalo, NY 14260 USA (e-mail: alipour@buffalo.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/TON.2025.3635484, provided by the authors.

Digital Object Identifier 10.1109/TON.2025.3635484

In the conventional FL architecture, a set of devices is connected to a central server (e.g., at a base station) in a star topology configuration [3]. Devices conduct model updates locally based on their individual datasets, and the server periodically aggregates these local models into a global model, synchronizing devices with this global model to begin the next round of training. Several works in the past few years have built functionality into this architecture to manage different dimensions of heterogeneity that manifest in fog, edge, and IoT networks, including varying communication and computation abilities of devices [5], [6] and varying statistical properties of local device datasets [7], [8].

However, access to a central server is not practical in some cases; For example, when conducting online training over dynamic intelligent systems, such as smart vehicles that move across a city, they can rely on vehicle-to-vehicle (V2V) links rather than vehicle-to-edge/server communications [9]. In addition, the model aggregation step in FL can be resource intensive when it requires frequent uplink transmission of large models [10]. In particular, it can lead to longer delays and larger bandwidth utilization for network links since there are multiple layers of network devices between the devices and the server [11]. In wireless networks where device-to-server connectivity is energy intensive or unavailable, ad hoc structures formed through device-to-device (D2D) links serve as an efficient alternative [4]. The proliferation of such settings motivates *fully decentralized FL* [12], [13], [14], where the model aggregation step is distributed across devices (in addition to the data processing step).

In this paper, we propose a novel methodology to facilitate fully decentralized FL and analyze its convergence characteristics. Our methodology has two key components. The first component involves combining updates based on stochastic gradient descent of local ML models with *cooperative consensus formation over the D2D graph topology* available across devices. Compared with more traditional decentralized optimization problems, the FL context introduces new challenges for these procedures due to (i) heterogeneity in device resources and local ML objectives, due to diversity in local datasets, and (ii) heterogeneity in wireless communication resources, which impacts the ability of devices to carry out consensus iterations. We consider these unique properties of decentralized FL in our algorithm design and analysis.

The central server in FL is also commonly employed for timing synchronization, i.e., to determine the time between global aggregations [11]. To overcome the lack of a cen-

TABLE I
COMPARISON OF OUR WORK AGAINST REPRESENTATIVE WORKS IN THE DECENTRALIZED FL LITERATURE

| Paper | Time-Varying Graph | Sporadic Comm. | Smooth. Assump. | Constant & Diminishing Step Size | Convex & Non-Convex Analysis | Last Iterates Conv. (Convex) |
|---|---|---|---|---|---|---|
| [13] | | ✓ | ✓ | | ✓ | |
| [15], [16] | ✓ | | | | | |
| [17] | ✓ | | | ✓ | | |
| [18], [19] | | | ✓ | | | ✓ |
| **Ours** | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

tral timing mechanism in decentralized FL, and to alleviate resource utilization, the second component of our methodology is an *asynchronous, event-triggered communication framework* for distributed ML consensus. Event-triggered communications offer several benefits in our setting. One, redundant communications can be reduced by defining event triggering conditions based on the variation of each device's model parameters. Also, eliminating the assumption that devices communicate in every iteration opens up the possibility of alleviating straggler problems, which is a prevalent concern in FL [10]. Third, we can eliminate redundant computations at each device by limiting the aggregation to only when new parameters are received.

### A. Related Work

We discuss related work in (i) distributed learning through consensus on graphs and (ii) FL over heterogeneous network systems. Our work lies at the intersection of these areas, and Table I illustrates the contributions of our work compared to existing literature.

*1) Consensus-Based Distributed Optimization:* There is a rich literature on distributed optimization on graphs using consensus algorithms, for example, [13], [15], [20], [21], [22], and [23]. For connected, undirected graph topologies, symmetric and doubly-stochastic transition matrices can be constructed for consensus iterations. In typical approaches [15], [20], each device maintains a local gradient of the target system objective (e.g., minimizing the consensus error across nodes), with the consensus matrices designed to satisfy additional convergence criteria outlined in [24] and [25]. More recently, gradient tracking optimization techniques have been developed in which the global gradient is learned simultaneously along with local parameters [21]. Also, [26] and [27] present a variation of gradient tracking algorithms where devices conduct multiple local gradient steps at each iteration. Other works have considered the distributed optimization problem over directed graphs, which is harder since constructing doubly-stochastic transition matrices is not a straightforward task for general directed graphs [28]. To resolve this issue, methods such as the push-sum protocol [22] have been proposed, where an extra optimizable parameter is introduced at each device in order to independently learn the right (or left) eigenvector corresponding to the eigenvalue of 1 of the transition matrices [23]. More recently, dual transition matrices have been studied, where two distinct transition matrices are designed to exchange model parameters and gradients separately, one column-stochastic and the other row-stochastic [23], [29], [30]. Moreover, asynchronous communications have also been researched in the literature [31], [32], [33].

On the other hand, event-triggered methods have received significant research attention in the conventional distributed optimization literature [34]. However, in federated learning (FL), there are only a handful of papers thus far which have studied event-triggered communication mechanisms, e.g., [35]. Event-triggering for inter-device communications poses unique challenges in the FL context due to different types of heterogeneity which become extremely pronounced in these setups, where a model is being trained over real-world wireless devices [2]: (i) diversity in local dataset statistics [36], which can have significant impacts on convergence behavior, since gradient iterations on these local datasets will tend to pull models apart [10], [37]; and (ii) heterogeneity in device resources [3], [7], [38]. Our methodology incorporates both of these factors, and our theoretical results reveal the impact of non-IID local data distributions on convergence characteristics of model training in decentralized FL setups. In doing so, it is important for us to bound convergence directly in terms of the statistical heterogeneity of the datasets, rather than on the (sub)-gradients as is done in existing decentralized optimization works like [15] and [17].

*2) Resource-Efficient Federated Learning:* Several recent works in FL have investigated techniques for improving the communication and computation efficiency across devices. A popular line of research has aimed to adaptively control the FL process based on device capabilities, e.g., [6], [39], [40], [41], and [42]. In [6], the authors studied FL convergence under a total network resource budget, in which the server adapts the frequency of global aggregations. Others [39], [40], [42] have considered FL under partial device participation, where the communication and processing capabilities of devices are taken into account when assessing which devices will participate in each training round. Reference [41] removed the necessity that every local device has to optimize the full model as the server, allowing weaker devices to take smaller subsets of the model to optimize. Furthermore, techniques such as quantization [43], [44] and sparsification [45] have also been studied to reduce the communication and computation overhead of the FL algorithms.

Unlike these works, we focus on novel learning topologies for decentralized FL. In this respect, some recent work [11], [12], [46], [47] has proposed D2D communication approaches for collaborative learning over local device graphs. References [11], [47], and [48] investigated a semi-decentralized FL methodology across hierarchical networks, where local model aggregations are conducted via D2D-based cooperative consensus formation to reduce the frequency of global aggregations by the coordinating node. In our work, we consider the fully decentralized setting, where a central node is not available, as in [12], [46], and [49]: along with local model updates, devices conduct consensus iterations with their neighbors in order to gradually minimize the global machine learning loss in a distributed manner. However, such techniques are not sensitive to the presence of heterogeneous

communication resources across devices. Different from [49], our methodology incorporates asynchronous event-triggered communications, where local resource levels are factored into event thresholds to account for device heterogeneity. This introduces a key challenge solved in our analysis, as it must obtain and leverage connectivity guarantees on the information flow graph (rather than the physical D2D graph). We will see that this approach leads to substantial improvements in model convergence time compared with non-heterogeneous/non-personalized thresholding.

### B. Outline and Summary of Contributions

- We develop a novel methodology for fully decentralized FL, with model aggregations occurring via cooperative model consensus iterations (Sec. II). In our methodology, communications are asynchronous and event-driven. With event thresholds defined to incorporate local model evolution and resource availability, our methodology adapts to the two salient heterogeneity dimensions in decentralized FL: limited resource availability and non-IID local dataset statistics across devices.
- We provide a detailed convergence analysis of our methodology, showing that using a diminishing step size, each device arrives at the globally optimal model over a time-varying consensus graph at an $\mathcal{O}(\ln k/\sqrt{k})$ rate (Sec. III). Our results are obtained based on statistical heterogeneity across local datasets, rather than a more restrictive bounded gradients assumption common in literature. Moreover, they do not impose overly restrictive connectivity requirements on the underlying D2D communication graph, so long as it satisfies a connectivity assumption over any $B$-consecutive iterations.
- To obtain these results, we demonstrate information flow guarantees in the presence of sporadic communications, making a distinction between physical connectivity of the underlying network graph and the information flow graph of the exchanged parameters among the devices (Proposition 1). Moreover, we lay out constraints on the local gradient step size to ensure a necessary spectral radius on this graph (Proposition 2). This allows us to derive the convergence rate in Theorems 1 and 2 for the model itself and not its cumulative average,[1] contrary to the existing trend in decentralized FL.
- We conducted numerical experiments to compare our methodology with baselines in decentralized FL, as well as a randomized gossip algorithm using two real-world machine learning task datasets (Sec. IV). We show that our method is capable of reducing the model training communication time compared to decentralized FL baselines. Also, we find that the convergence rate of our method scales well with consensus graph connectivity.

This paper is an extension of our conference version of this work [50]. Compared to [50], we make the following additional contributions: (1) connectivity of the information flow graph between devices is theoretically proven in Proposition 1, i.e., we guarantee that all devices will benefit from every other device in the federated learning system, despite the communications being sporadic, and the underlying physical network being time-varying, (2) the traditional assumption of bounded gradients has been replaced with two assumptions which are less strict in Assumptions 2: Lipschitz gradient continuity and bounded gradient diversity, (3 we obtain a non-asymptotic rate of convergence for our method, by recovering the well-known $\mathcal{O}(\ln k/\sqrt{k})$ sub-linear rate for distributed gradient descent like algorithms when using a diminishing learning rate in Theorem 2 and (4) we conduct new experiments on an additional dataset, more graph topologies and an additional distribution for bandwidth sampling, to further validate the advantages of our proposed methodology.

[1]The cumulative average of the model is defined as $(1/T)\sum_{t=0}^{T-1}\mathbf{w}^{(t)}$.

### C. Notations

Arguments for functions are denoted with parentheses, e.g., $f(x)$ implies $x$ is an argument for function $f$. The iteration index for a parameter is indicated via superscripts, e.g., $h^{(k)}$ is the value of the parameter $h$ at iteration $k$. Device indices are given via subscripts, e.g., $h_i^{(k)}$ refers to parameter belonging to device $i$. We write a graph $\mathcal{G}$ with a set of nodes (devices) $\mathcal{V}$ and a set of edges (links) $\mathcal{E}$ as $\mathcal{G}=(\mathcal{V},\mathcal{E})$.

We denote vectors with lowercase boldface, e.g., $\mathbf{x}$, and matrices with uppercase boldface, e.g., $\mathbf{X}$. All vectors $\mathbf{x}\in\mathbb{R}^{d\times 1}$ are column vectors, except in certain cases where average vectors $\bar{\mathbf{x}}\in\mathbb{R}^{1\times d}$ and optimal vectors $\mathbf{w}^\star\in\mathbb{R}^{1\times d}$ are row vectors. $\langle\mathbf{x},\mathbf{x}'\rangle$ and $\langle\mathbf{X},\mathbf{X}'\rangle$ denote the inner product of two vectors $\mathbf{x},\mathbf{x}'$ of equal dimensions and the Frobenius inner product of two matrices $\mathbf{X},\mathbf{X}'$ of equal dimensions, respectively. Moreover, $\|\mathbf{x}\|$ and $\|\mathbf{X}\|$ denote the 2-norm of the vector $\mathbf{x}$, and the Frobenius norm of the matrix $\mathbf{X}$, respectively. The spectral norm of the matrix $\mathbf{X}$ is written as $\rho(\mathbf{X})$.

Note that for brevity, all mathematical proofs have been deferred to the Appendices at the end of the manuscript.

## II. Methodology and Algorithm

In this section, we develop our methodology for decentralized FL with event-triggered communications. After discussing preliminaries of the model in FL (Sec. II-A), we present our cooperative consensus algorithm for distributed model aggregations (Sec. II-B). We then present the events in our event-triggered algorithm as iterative relations, which enables our theoretical analysis (Sec. II-C).

### A. Device and Learning Model

We consider a network of $m$ devices/nodes, collected by set $\mathcal{M}$, $m=|\mathcal{M}|$, which are engaged in distributed training of a machine model. Under the FL framework, each device $i\in\mathcal{M}$ trains a local model $\mathbf{w}_i$ using its own generated dataset $\mathcal{D}_i$. Each data point $\xi\triangleq(\mathbf{x}_\xi,y_\xi)\in\mathcal{D}_i$ consists of a feature vector $\mathbf{x}_\xi$ and a target label $y_\xi$. The performance of the local model is measured via the local loss $F_i(.)$ as

$$F_i(\mathbf{w})=\sum_{\xi\in\mathcal{D}_i}\ell_\xi(\mathbf{w}), \tag{1}$$

where $\ell_\xi(\mathbf{w})$ is the loss of the model at the data point $\xi$ (e.g., squared prediction error) under parameter realization $\mathbf{w}\in\mathbb{R}^n$,

with $n$ denoting the dimension of the target model. The global loss is defined in terms of these local losses as

$$F(\mathbf{w}) = \frac{1}{m} \sum_{i \in \mathcal{M}} F_i(\mathbf{w}). \tag{2}$$

The goal of the training process is to find an optimal parameter vector $\mathbf{w}^\star$ that minimizes the global loss function, that is, $\mathbf{w}^\star = \arg\min_{\mathbf{w} \in \mathbb{R}^n} F(\mathbf{w})$. In the distributed setting, we desire $\mathbf{w}_1 = \cdots = \mathbf{w}_m = \mathbf{w}^*$ at the end of the training process, which requires a synchronization mechanism. In conventional FL, as discussed in Sec. I, synchronization is conducted periodically by a central coordinator globally aggregating the local models. However, in this work, we are interested in a fully decentralized setting where no such central node exists. Thus, in addition to using optimization techniques to minimize local loss functions, we must develop a technique to reach consensus over the parameters in a distributed manner.

To achieve this, we propose *Event-triggered Federated learning with Heterogeneous Communication thresholds* (EF−HC). In EF−HC, devices conduct D2D communications during the model training period to synchronize their locally trained models and avoid overfitting to their local datasets. The overall EF−HC algorithm executed by each device is given in Alg. 1. Two vectors of model parameters are kept on each device $i$: (i) its instantaneous *main* model parameters $\mathbf{w}_i$, and (ii) the *auxiliary* model parameters $\widehat{\mathbf{w}}_i$, which is the outdated version of its main parameters that had been broadcast to neighbors. Decentralized ML is conducted over the (time-varying, undirected) device graph through a sequence of four events detailed in Sec. II-B. Although in our distributed setup there is no physical notion of a global iteration, we introduce the universal iteration variable $k$ for analysis purposes [51]. In other words, event-triggering in our paper implies that not every iteration $k$ includes full participation of devices in inter-device communications, which is different from how synchronous DFL works [15], [16], [17] model iterate updates.

### B. Network Model and Event-Triggering

We consider the *physical network* graph $\mathcal{G}^{(k)} = (\mathcal{M}, \mathcal{E}^{(k)})$ among devices, where $\mathcal{E}^{(k)}$ is the set of edges available at iteration $k$ in the underlying time-varying communication graph. We assume that link availability varies over time according to the underlying device-to-device communication protocol in place [10]. In each iteration, some of the edges are used for the transmission/reception of model parameters between devices. To represent this process, we define the *information flow* graph $\mathcal{G}'^{(k)} = (\mathcal{M}, \mathcal{E}'^{(k)})$, which is a subgraph of $\mathcal{G}^{(k)}$. $\mathcal{E}'^{(k)}$ only contains the links in $\mathcal{E}^{(k)}$ that are being used at iteration $k$ to exchange parameters. Based on this, we denote the neighbors of device $i$ in iteration $k$ as $\mathcal{N}_i^{(k)} = \{j : (i,j) \in \mathcal{E}^{(k)}, j \in \mathcal{M}\}$, with node degree $d_i^{(k)} = |\mathcal{N}_i^{(k)}|$. We also denote neighbors of $i$ that communicate directly with it in iteration $k$ as $\mathcal{N}_i'^{(k)} = \{j : (i,j) \in \mathcal{E}'^{(k)}, j \in \mathcal{M}\}$. Additionally, the aggregation weights associated with the link $(i,j) \in \mathcal{E}^{(k)}$ and $(i,j) \in \mathcal{E}'^{(k)}$ are defined as $\beta_{ij}^{(k)}$ and $p_{ij}^{(k)}$, respectively, with $p_{ij}^{(k)} = \beta_{ij}^{(k)}$ if the link $(i,j)$ is used for aggregation at iteration $k$, and $p_{ij}^{(k)} = 0$ otherwise.

**Algorithm 1** EF−HC procedure for device $i$.

**Input:** $K$
Initialize $k = 0$, $\mathbf{w}_i^{(0)} = \widehat{\mathbf{w}}_i^{(0)}$
1: **while** $k \leq K$ **do**
▷**Event 1.** Neighbor Connection Event
2: **if** device $j$ is connected to device $i$ **then**
3: device $i$ appends device $j$ to its list of neighbors
4: device $i$ sends $\mathbf{w}_i^{(k)}$ and $d_i^{(k)}$ to device $j$
5: device $i$ receives $\mathbf{w}_j^{(k)}$ and $d_j^{(k)}$ from device $j$
6: **else if** device $j$ is disconnected from device $i$ **then**
7: device $i$ removes device $j$ from its list of neighbors
▷**Event 2.** Broadcast Event
8: **if** $(1/n)^{\frac{1}{2}} \|\mathbf{w}_i^{(k)} - \widehat{\mathbf{w}}_i^{(k)}\|_2 \geq r\rho_i\gamma^{(k)}$ **then**
9: device $i$ broadcasts $\mathbf{w}_i^{(k)}$, $d_i^{(k)}$ to all neighbors $j \in \mathcal{N}_i^{(k)}$
10: device $i$ receives $\mathbf{w}_j^{(k)}$, $d_j^{(k)}$ from all neighbors $j \in \mathcal{N}_i^{(k)}$
11: $\widehat{\mathbf{w}}_i^{(k+1)} = \mathbf{w}_i^{(k)}$
▷**Event 3.** Aggregation Event
12: **if** Parameters $\mathbf{w}_j^{(k)}$, $d_j^{(k)}$ received from neighbor $j$ **then**
13: $\mathbf{w}_i^{(k+1)} = \mathbf{w}_i^{(k)} + \sum_{j \in \mathcal{N}_i'^{(k)}} \beta_{ij}^{(k)} (\mathbf{w}_j^{(k)} - \mathbf{w}_i^{(k)})$
▷**Event 4.** Gradient Descent Event
14: device $i$ conducts SGD iteration $\mathbf{w}_i^{(k+1)} = \mathbf{w}_i^{(k)} - \alpha^{(k)} \mathbf{g}_i^{(k)}$
15: $k \leftarrow k + 1$

In EF−HC, there are four types of events:

**Event 1: Neighbor connection**

The first event (lines 2-7 of Alg. 1) is triggered at device $i$ if new devices connect to it or existing devices disconnect from it due to the time-varying nature of the graph. Model parameters $\mathbf{w}_i^{(k)}$ and the degree of the device $i$ at that time $d_i^{(k)}$ are exchanged with this new neighbor. Consequently, this results in an aggregation event (Event 3) on both devices.

**Event 2: Broadcast**

If the normalized difference between $\mathbf{w}_i^{(k)}$ and $\widehat{\mathbf{w}}_i^{(k)}$ at device $i$ is greater than a *threshold* value $r\rho_i\gamma^{(k)}$, i.e.,

$$(1/n)^{\frac{1}{2}} \|\mathbf{w}_i^{(k)} - \widehat{\mathbf{w}}_i^{(k)}\|_2 \geq r\rho_i\gamma^{(k)}, \tag{3}$$

then a broadcast event is triggered at that device (lines 8-11 of Alg. 1). In other words, communication at a device is triggered once the instantaneous local model is sufficiently different from the outdated one. When this event triggers, device $i$ broadcasts $\mathbf{w}_i^{(k)}$ and its degree $d_i^{(k)}$ to all its neighbors and receives the same information from them. Note that if a neighbor device $j$ is not available for communication with device $i$ at a certain iteration $k$, then $j$ would not be considered inside the neighbor set of $i$ in that iteration, i.e., $\mathcal{N}_i^{(k)}$.

The threshold $r\rho_i\gamma^{(k)}$ is treated as personalized/heterogeneous across devices $i \in \mathcal{M}$, to assess whether the gain from a consensus iteration on the instantaneous main models at the devices will be worth the induced utilization of network resources. Specifically, (i) $r > 0$ is a scaling hyperparameter value; (ii) $\gamma^{(k)} > 0$ is a decaying factor that accounts for smaller expected variations in local models

over time, and $\lim_{k\to\infty}\gamma^{(k)}=0$; and (iii) $\rho_i$ quantifies the availability of resources of device $i$. See [1] for some remarks on $(1/n)^{\frac{1}{2}}$, $r$ and $\gamma^{(k)}$.

The development of $(1/n)^{\frac{1}{2}}\|\mathbf{w}_i^{(k)}-\widehat{\mathbf{w}}_i^{(k)}\|_2$ and the condition $r\rho_i\gamma^{(k)}$ is one of our contributions relative to existing event-triggered schemes [35]. For example, in a bandwidth-limited environment, the transmission delay of the model transfer will be inversely proportional to the bandwidth among two devices. Thus, to decrease the latency of model training, $\rho_i$ can be defined inversely proportional to the bandwidth, promoting a lower frequency of communication on devices with less available bandwidth. In `EF-HC`, we set $\rho_i\propto\frac{1}{b_i}$, where $b_i$ is the average bandwidth on the outgoing links of the device $i$.

**Event 3: Aggregation**

Following a broadcast event (Event 2) or a neighbor connection event (Event 1) on device $i$, an aggregation event (lines 12-13 of Alg. 1) is triggered on device $i$ and all its neighbors. This aggregation is carried out through a distributed weighted averaging consensus method [25] as

$$\mathbf{w}_i^{(k+1)}=\mathbf{w}_i^{(k)}+\sum_{j\in\mathcal{N}_i'^{(k)}}\beta_{ij}^{(k)}(\mathbf{w}_j^{(k)}-\mathbf{w}_i^{(k)}),\tag{4}$$

where $\beta_{ij}^{(k)}$ is the aggregation weight that device $i$ assigns to parameters received from device $j$ in iteration $k$. The aggregation weights $\{\beta_{ij}^{(k)}\}$ for graph $\mathcal{G}^{(k)}$ can be selected based on the degree of neighbors, as will be discussed in Sec. III-A.

**Event 4: Gradient descent**

Each device $i$ conducts stochastic gradient descent (SGD) iterations for local model training (lines 14-15 of Alg. 1). Formally, device $i$ obtains $\mathbf{w}_i^{(k+1)}=\mathbf{w}_i^{(k)}-\alpha^{(k)}\mathbf{g}_i^{(k)}$, where $\alpha^{(k)}$ is the step size, and $\mathbf{g}_i^{(k)}$ is the stochastic gradient approximation defined as $\mathbf{g}_i^{(k)}=(1/|\mathcal{S}_i^{(k)}|)\sum_{\boldsymbol{\xi}\in\mathcal{S}_i^{(k)}}\nabla\ell_\xi(\mathbf{w}_i^{(k)})$. Here, $\mathcal{S}_i^{(k)}$ denotes the set of data points (mini-batch) used to compute the gradient, chosen uniformly at random from the local dataset. In our analysis, we define

$$\mathbf{g}_i^{(k)}=\nabla F_i(\mathbf{w}_i^{(k)})+\boldsymbol{\epsilon}_i^{(k)},\tag{5}$$

in which $\nabla F_i(\mathbf{w}_i^{(k)})$ is the gradient of $F_i$ at $\mathbf{w}_i^{(k)}$, and $\boldsymbol{\epsilon}_i^{(k)}$ is the error due to the stochastic gradient approximation.

### C. Iterate Relations

We now express the model updates conducted in Alg. 1 in an iterative format, which will be useful in our subsequent theoretical analysis. Rewriting the event-based updates of Alg. 1 into one line of iterative model update, we get

$$\mathbf{w}_i^{(k+1)}=\mathbf{w}_i^{(k)}+\sum_{j\in\mathcal{N}_i'^{(k)}}\beta_{ij}^{(k)}(\mathbf{w}_j^{(k)}-\mathbf{w}_i^{(k)})v_{ij}^{(k)}-\alpha^{(k)}\mathbf{g}_i^{(k)},\tag{6}$$

where $v_{ij}^{(k)}$ indicates whether device $i$ aggregates its model with device $j$ at iteration $k$. Its value depends on $v_i^{(k)}$, which is an indicator of a broadcast event at device $i$ at iteration $k$ defined as

$$v_i^{(k)}=\begin{cases}1 & (1/n)^{\frac{1}{2}}\left\|\mathbf{w}_i^{(k)}-\widehat{\mathbf{w}}_i^{(k)}\right\|_2>r\rho_i\gamma^{(k)}\\ 0 & \text{o.w.}\end{cases},$$
$$v_{ij}^{(k)}=\begin{cases}\max\{v_i^{(k)},v_j^{(k)}\} & j\in\mathcal{N}_i^{(k)}\\ 0 & \text{o.w.}\end{cases},\tag{7}$$

with $\rho_i=1/b_i$. Also, note that the stale model parameters $\hat{\mathbf{w}}_i^{(k)}$ Rearranging the relations in (6), we have

$$\mathbf{w}_i^{(k+1)}=\left(1-\sum_{j=1}^m\beta_{ij}^{(k)}v_{ij}^{(k)}\right)\mathbf{w}_i^{(k)}+\sum_{j=1}^m\beta_{ij}^{(k)}v_{ij}^{(k)}\mathbf{w}_j^{(k)}$$
$$-\alpha^{(k)}\mathbf{g}_i^{(k)}=\sum_{j=1}^m p_{ij}^{(k)}\mathbf{w}_j^{(k)}-\alpha^{(k)}\mathbf{g}_i^{(k)},\tag{8}$$

where $p_{ij}^{(k)}$ is the transition weight that device $i$ uses to aggregate device $j$'s parameters at iteration $k$:

$$p_{ij}^{(k)}=\begin{cases}\beta_{ij}^{(k)}v_{ij}^{(k)} & i\neq j\\ 1-\sum_{j=1}^m\beta_{ij}^{(k)}v_{ij}^{(k)} & i=j\end{cases}.\tag{9}$$

Note that the aggregation and transition weights, i.e., $\beta_{ij}^{(k)}$ and $p_{ij}^{(k)}$, distinguish the two sources of time variation in the information flow graph of our method: (i) the underlying physical network being time-varying, resulting in varying number of neighbors for each device at each iteration, and (ii) the event-triggering mechanism, adding another overlay time-varying component on top of the network graph.

Next, we collect the parameter vectors of all devices that were previously introduced in matrix form as follows: $\mathbf{W}^{(k)}=[\mathbf{w}_1^{(k)}\ \ldots\ \mathbf{w}_m^{(k)}]^T$, $\mathbf{G}^{(k)}=[\mathbf{g}_1^{(k)}\ \ldots\ \mathbf{g}_m^{(k)}]^T$, $\mathbf{P}^{(k)}=[p_{ij}^{(k)}]_{1\le i,j\le m}$. Now, we transform the recursive update rules of (8) into matrix form to obtain the following relationship

$$\mathbf{W}^{(k+1)}=\mathbf{P}^{(k)}\mathbf{W}^{(k)}-\alpha^{(k)}\mathbf{G}^{(k)}.\tag{10}$$

The recursive expression in (10) has been investigated before [15] and [17]. However, the existing literature on decentralized stochastic gradient descent does not account for data heterogeneity, and this motivates us to use different analytical tools to derive convergence bounds.

As a conclusion to this section on iterate relations, we introduce two quantities, which will be frequently used in our analysis. We first derive an explicit relationship of (10). Starting from iteration $s$, where $s\le k$, we have

$$\mathbf{W}^{(k+1)}=\mathbf{P}^{(k:s)}\mathbf{W}^{(s)}-\sum_{r=s+1}^k\alpha^{(r-1)}\mathbf{P}^{(k:r)}\mathbf{G}^{(r-1)}-\alpha^{(k)}\mathbf{G}^{(k)},$$
$$\mathbf{P}^{(k:s)}=\mathbf{P}^{(k)}\mathbf{P}^{(k-1)}\cdots\mathbf{P}^{(s+1)}\mathbf{P}^{(s)}.\tag{11}$$

Second, to analyze the consensus of local models, we define the average model as $\bar{\mathbf{w}}^{(k)}=(1/m)\sum_{i=1}^m\mathbf{w}_i^{(k)}$. The recursive relation for $\bar{\mathbf{w}}^{(\mathbf{k})}$ using (8) and the stochasticity of $\mathbf{P}^{(k)}$ is

$$\bar{\mathbf{w}}^{(k+1)}=\bar{\mathbf{w}}^{(k)}-(\alpha^{(k)}/m)\sum_{i=1}^m\mathbf{g}_i^{(k)}.\tag{12}$$

Also, an explicit relationship between iteration $\bar{\mathbf{w}}^{(k+1)}$ and $\bar{\mathbf{w}}^{(s)}$, where $s \leq k$, easily follows from (12) as

$$\bar{\mathbf{w}}^{(k+1)} = \bar{\mathbf{w}}^{(s)} - (1/m) \sum_{r=s}^{k} \alpha^{(r)} \sum_{i=1}^{m} \mathbf{g}_i^{(r)}. \tag{13}$$

## III. Convergence Analysis

In this section, we first detail the assumptions used in our paper (Sec. III-A), and then provide the main connectivity result of our event-trigerred approach (Sec. III-B). Finally, we present the lemmas which are used to prove our main Theorems (Sec. III-C), and then present the main theoretical contributions themselves (Sec. III-D).

### A. Assumptions

*Assumption 1:* [Transition weights] Let $\{p_{ij}^{(k)}\}$ be the set of aggregation weights in the information graph $\mathcal{G}'(k)$. The following conditions must be met:

(a) (Non-negative weights) $\forall i \in \mathcal{M}$, we have
  (i) $p_{ii}^{(k)} > 0$ and $p_{ij}^{(k)} > 0$ for all $k \geq 0$ and all neighboring devices $j \in \mathcal{N}_i'^{(k)}$.
  (ii) $p_{ij}^{(k)} = 0$, if $j \notin \mathcal{N}_i'^{(k)}$.

(b) (Doubly-stochastic weights) The rows and columns of matrix $\mathbf{P}^{(k)} = [p_{ij}^{(k)}]$ are both stochastic, i.e., $\sum_{j=1}^{m} p_{ij}^{(k)} = 1$, $\forall i$, and $\sum_{i=1}^{m} p_{ij}^{(k)} = 1$, $\forall j$.

(c) (Symmetric weights) $p_{ij}^{(k)} = p_{ji}^{(k)}$, $\forall i, k$ and $p_{ii}^{(k)} = 1 - \sum_{j \neq i} p_{ij}^{(k)}$.

Taking into account the conditions mentioned in Assumption 1, and the definition of $p_{ij}^{(k)}$ in (9), a choice of parameters $\beta_{ij}^{(k)}$ that satisfy these assumptions are as follows

$$\beta_{ij}^{(k)} = \min \left\{ \frac{1}{1 + d_i^{(k)}}, \frac{1}{1 + d_j^{(k)}} \right\}, \tag{14}$$

which is inspired by the Metropolis-Hastings algorithm [24]. Note that $p_{ij}^{(k)}$ also depends on $v_{ij}^{(k)}$, which was defined in (7).

*Assumption 2:* [Smoothness, Strong convexity, and Data heterogeneity] The local objective function at each device $i \in \mathcal{M}$, i.e., $F_i$, satisfies the following

(a) $L_i$-Lipschitz continuous gradients: $\|\nabla F_i(\mathbf{w}) - \nabla F_i(\mathbf{w}')\| \leq L_i \|\mathbf{w} - \mathbf{w}'\|$,

(b) $\mu_i$-strong convexity: $\langle \nabla F_i(\mathbf{w}) - \nabla F_i(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle \geq \mu_i \|\mathbf{w} - \mathbf{w}'\|^2$,

(c) The data heterogeneity across the devices is measured via $\delta_i > 0$ as $\|\nabla F_i(\mathbf{w}) - \nabla F(\mathbf{w})\| \leq \delta_i$,

$\forall (\mathbf{w}', \mathbf{w}) \in \mathbb{R}^n \times \mathbb{R}^n$, where we also define $L = \max_{i \in \mathcal{M}} L_i$, $\mu = \min_{i \in \mathcal{M}} \mu_i$ and $\delta = \max_{i \in \mathcal{M}} \delta_i$.

Note that the global objective function $F(\mathbf{w})$, which is a convex combination of local objective functions, will also be strongly convex, thus having a unique minimizer, denoted by $\mathbf{w}^\star = \arg\min_{\mathbf{w} \in \mathbb{R}^n} F(\mathbf{w})$. Additionally, following Assumptions 2-(a) and 2-(b), we have $\mu \leq \mu_i \leq L_i \leq L$, for all $i \in \mathcal{M}$. Finally, note that the assumption about data heterogeneity (Assumption 2-(c)) was inspired by [6] and [47].

Works like [15] and [17] make a much stricter assumption than the statistical heterogeneity assumption we make in our paper: the bounded (sub)-gradients assumption, i.e., $\|\nabla F_i(\mathbf{w})\| \leq L_i$. The reason our Assumption 2-(c) is less strict is that by letting $\mathbf{w} = \mathbf{w}^\star$, we get that local gradients should be bounded only at the optimal point, contrary to those other papers which make the assumption that they should be bounded over the full space. The implication is that the data heterogeneity assumption in our paper only requires the difference between the global gradient at a certain point and the local gradient at that same point to be upper bounded by a constant scalar. This means the norm of local and global gradients can have arbitrarily large values in our paper (unlike the bounded gradients assumption of [15] and [17]), so long as their difference is bounded for every point in $\mathbb{R}^n$.

*Assumption 3:* [Gradient approximation errors] We make the following assumptions on the gradient approximation errors $\boldsymbol{\epsilon}_i^{(k)}$ for all $i \in \mathcal{M}$ and all $k \geq 0$:

(a) Zero mean, i.e., $\mathbb{E}[\boldsymbol{\epsilon}_i^{(k)}] = 0$.

(b) Bounded mean square, i.e., there is a scalar $\sigma_i^2$ such that $\mathbb{E}[\|\boldsymbol{\epsilon}_i^{(k)}\|_2^2] \leq \sigma_i^2 \leq \sigma^2$, where $\sigma = \max_{i \in \mathcal{M}} \sigma_i$.

(c) Each random vector $\epsilon_i^{(k)}$ is independent from $\epsilon_j^{(k)}$ for $j \neq i$.

*Assumption 4:* [Step sizes] All devices use the same step size for model training. We study the behavior of our algorithm under two policies for the step size:

(a) Constant step size, having $\alpha^{(k)} = \alpha$ where $\alpha > 0$.

(b) Diminishing step size, in which the step size decays over time, satisfying the following conditions

$$\lim_{k \to \infty} \alpha^{(k)} = 0, \quad \sum_{k=0}^{\infty} \alpha^{(k)} = \infty, \quad \sum_{k=0}^{\infty} \left(\alpha^{(k)}\right)^2 < \infty.$$

In particular, setting $\alpha^{(k)} = \alpha^{(0)}/(1 + k/\eta)^\theta$ meets the criteria of Assumption 4-(b) for $\alpha^{(0)}, \eta > 0$, and $\theta \in (0.5, 1]$.

The previous assumptions are common in the literature [6], [11]. In the next assumption, we introduce a relaxed version of graph connectivity requirements relative to existing work in distributed learning, which underscores the difference of our decentralized event-triggered FL method compared with traditional distributed optimization algorithms.

*Assumption 5:* [Network graph connectivity]

The underlying communication graph satisfies the following properties:

(a) There exists an integer $B_1 \geq 1$ such that the graph union of the physical network graph $\mathcal{G}^{(k)} = (\mathcal{M}, \mathcal{E}^{(k)})$ from any arbitrary iteration $k$ to $k + B_1 - 1$, i.e., $\mathcal{G}^{(k:k+B_1-1)} = (\mathcal{M}, \cup_{s=0}^{B_1-1} \mathcal{E}^{(k+s)})$, is connected for any $k \geq 0$.

(b) There exists an integer $B_2 \geq 1$ such that for every device $i$, triggering conditions for the broadcast event occur at least once every $B_2$ consecutive iterations $\forall k \geq 0$. This is equivalent to the following condition:

$$\exists B_2 \geq 1, \forall i : \max\left\{v_i^{(k)}, v_i^{(k+1)}, \cdots, v_i^{(k+B_2-1)}\right\} = 1.$$

Existing works in the literature either assume (i) a static physical graph ($B_1 = 1$) with sporadic communications [13], or (ii) a time-varying physical graph with communications at every round ($B_2 = 1$) [15]. Our paper is the first to combine

these two sources of time-variation (physical graph and the communication graph) for event-triggered decentralized learning methods, and provide a generalized theoretical analysis when neither of them are static. We will use Assumption 5 in the proof of Proposition 1 to analyze the connectivity behavior of information flow graphs, i.e., $\mathcal{G}'^{(k)}$.

### B. Main Connectivity Result

in Proposition 1 below, we provide a connectivity guarantee of the information flow graph in the presence of (i) the underlying physical network connecting the agents being time-varying and (ii) the event-triggering communication mechanism adding another layer of temporal variation for inter-device communication, i.e., as in Assumption 5. In prior works like [52], only the underlying physical graph is time-varying, and consensus operations are carried out whenever possible, i.e., at every iteration. As a result, obtaining an information graph connectivity guarantee has not traditionally been a key challenge. In contrast, in our paper, consensus operations do not occur at every iteration due to the unique heterogeneity challenges in decentralized federated learning discussed in Sec. I, especially due to inter-device links having varying bandwidth availability. Thus, even if the underlying physical graph is connected, more elaborate considerations are required to establish connectivity results on the information flow graphs.

Additionally, note that papers like [35], which focus on event-triggered methods, consider a simplified static underlying physical graph. Thus, the analysis in both [35] and [52] contains only one of the two connectivity criteria we consider in our analysis.

*Proposition 1:* Let Assumption 5 hold. Under the `EF-HC` algorithm (Alg. 1), the information flow graph $\mathcal{G}'^{(k)}$ is $B$-connected, i.e., $\mathcal{G}'^{(k:k+B-1)} = (\mathcal{M}, \cup_{s=0}^{B-1} \mathcal{E}'^{(k+s)})$ is connected for any $k \geq 0$, where $B = (\tilde{l}+2)B_1$ and $\tilde{l}$ are determined via $\tilde{l}B_1 \leq B_2 \leq (\tilde{l}+1)B_1 - 1$. Note that $B_1$ and $B_2$ are, respectively, the connectivity bound of the physical network graph and the bound for the occurrence of communication events of Assumption 5-(a) and 5-(b).

*Proof:* See Appendix A in the supplementary material. The high-level idea behind the proof is (i) carefully keeping track of all devices that a certain agent has communicated with at every iteration, and then (ii) finding an upper bound on the number of iterations until all devices have communicated with at least one of their neighbors. ■

It is important to note that we use the $B$ parameter introduced in Proposition 1 only for convergence analysis. It can have an arbitrarily large value. Therefore, we are not making strict connectivity assumptions on the underlying graph.[2]

### C. Intermediate Lemmas for Convergence

In this section, we provide some lemmas which are useful in the proofs of Theorems 1 and 2 of Sec. III-D. These lemmas also provide additional characteristics of our methodology.

[2]If our algorithms required the devices to exchange parameters with their neighbors upon disconnection, we would have $B = \max\{B_1, B_2\}$.

Our first lemma gives a bound on the consensus error over the course of multiple iterations, i.e., $\|\mathbf{P}^{(k:s)}\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\|$, using the spectral norm of $\mathbf{P}^{(k:s)} - (1/m)\mathbf{1}_m\mathbf{1}_m^T$, which we show that depending on iteration $s$, this bound can be made tighter.

*Lemma 1:* Let Assumption 1 hold, and let $B$ be the connectivity bound of Proposition 1. Then the following is true

(a) From iteration $k$ to $k+B-r$, where $r = 2, \cdots, B$, we have

$$\begin{aligned}&\left\|\mathbf{P}^{(k+B-r:k)}\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\| \\ &\leq \rho^{(k+B-r:k)}\left\|\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\| \\ &\leq \left\|\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|.\end{aligned}$$

(b) From iteration $k$ to $k+B-1$, we have the following

$$\begin{aligned}&\left\|\mathbf{P}^{(k+B-1:k)}\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\| \\ &\leq \rho^{(k+B-1:k)}\left\|\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\| \\ &\leq \rho\left\|\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|,\end{aligned}$$

in which $\rho^{(k+B-r:k)} = \rho\big(\mathbf{P}^{(k+B-r:k)} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T\big)$, $\rho = \sup_{k=0,1,\cdots}\rho\big(\mathbf{P}^{(k+B-1:k)} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T\big)$, and $0 < \rho < 1$.

*Proof:* Since the graph is time-varying, we can only guarantee the connectivity of $\mathbf{P}^{(k+B-1:k)}$. Therefore, $0 < \rho\big(\mathbf{P}^{(k+B-r:k)} - \frac{1}{m}\mathbf{1}_m\mathbf{1}_m^T\big) \leq 1$ for all $r = 2, \cdots, B$, but $0 < \rho < 1$. The rest of the proof follows from Sec. II-B of [53]. ■

Lemma 1 is essential in the analysis of our method and helps us to prove the convergence under time-varying communication graphs with an arbitrary connectivity bound.

*Definition 1:* We define the following gradient matrices: $\nabla^{(k)} = [\nabla F_1(\mathbf{w}_1^{(k)}), \cdots, \nabla F_m(\mathbf{w}_m^{(k)})]^T$, $\bar{\nabla}^{(k)} = \frac{1}{m}\mathbf{1}_m^T\nabla^{(k)}$ and $\nabla F^{(k)} = [\nabla F(\mathbf{w}_1^{(k)}), \cdots, \nabla F(\mathbf{w}_m^{(k)})]^T$. Furthermore, note that $\nabla F(\bar{\mathbf{w}}^{(k)}) \in \mathbb{R}^{1\times n}$, is the gradient of global objective function evaluated at $\bar{\mathbf{w}}^{(k)}$.

Using the previous definition, we next provide two inequalities which help us bound the expressions involving the gradients in any iteration $k$, via the values of the model parameters, scaled by a constant factor.

*Lemma 2:* Under Assumptions 2-(a) and 2-(b), the following holds for all $k \geq 0$:

$$\left\|\nabla F\left(\bar{\mathbf{w}}^{(k)}\right) - \bar{\nabla}^{(k)}\right\| \leq \frac{L}{\sqrt{m}}\left\|\mathbf{W}^{(k)} - \mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|.$$

Also, if $\alpha^{(k)} < \frac{2}{\mu+L}$, then

$$\left\|\bar{\mathbf{w}}^{(k)} - \alpha^{(k)}\nabla F\left(\bar{\mathbf{w}}^{(k)}\right) - \mathbf{w}^\star\right\| \leq \left(1 - \mu\alpha^{(k)}\right)\left\|\bar{\mathbf{w}}^{(k)} - \mathbf{w}^\star\right\|.$$

*Proof:* Follows from Lemmas 1-(c) and 10 of [53]. ■

Next, we obtain the following bounds for the average of gradient approximation errors, which are used to obtain the results of several subsequent lemmas.

*Lemma 3:* Let Assumption 3 hold. Provided the definitions $\bar{\epsilon}^{(k)} = \frac{1}{m}\sum_{i=1}^m \boldsymbol{\epsilon}_i^{(k)}$ and $\boldsymbol{\epsilon}^{(k)} = \left[\boldsymbol{\epsilon}_1^{(k)}, \cdots, \boldsymbol{\epsilon}_m^{(k)}\right]^T$, we have

$$\mathbb{E}\left[\left\|\bar{\epsilon}^{(k)}\right\|^2\right] \leq \frac{\sigma^2}{m}, \quad \mathbb{E}\left[\left\|\boldsymbol{\epsilon}^{(k)} - \mathbf{1}_m\bar{\epsilon}^{(k)}\right\|^2\right] \leq m\sigma^2.$$

*Proof:* The first inequality follows from Lemma 2 of [21]. For the proof of the second bound, see Appendix C in the supplementary material. ■

Traditional analysis of distributed gradient descent involves making the assumption of bounded gradient (see [15], [20]). However, since we have replaced such an assumption with two different but more general assumptions, namely smoothness and data heterogeneity (Assumptions 2-(a) and 2-(c)), our analysis is different compared to the current literature. Inspired by the gradient tracking literature in distributed learning [21], [53], in the following lemma, we look at the behavior of $\|\mathbf{W}^{(k+1)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k+1)}\|^2$ and $\|\bar{\mathbf{w}}^{(k+1)}-\mathbf{w}^\star\|^2$, and bound them simultaneously via a system of inequalities.

*Lemma 4:* Assumptions 2 and 3 yield the following bounds:

(a) Consensus error on local gradients:

$$\left\|\nabla^{(k)}-\mathbf{1}_m\bar{\nabla}^{(k)}\right\|^2\leq 2\,m\delta^2+8L^2\left\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|^2.$$

(b) Optimization error, assuming $\alpha^{(k)}<\frac{2}{\mu+L}$:

$$\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{(k+1)}-\mathbf{w}^\star\right\|^2\right]\leq a_{11}^{(k)}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{(k)}-\mathbf{w}^\star\right\|^2\right]+a_{12}^{(k)}\mathbb{E}\left[\left\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|^2\right]+c_1^{(k)},$$

where $a_{11}^{(k)}=1-\mu\alpha^{(k)}$, $a_{12}^{(k)}=(1+\mu\alpha^{(k)})\alpha^{(k)}L^2/(\mu m)$, $c_1^{(k)}=(\alpha^{(k)})^2\sigma^2/m$.

(c) Let Assumption 1 also hold. The Expected consensus error of model weights is bounded as:

$$\mathbb{E}\left[\left\|\mathbf{W}^{(k+1)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k+1)}\right\|^2\right]\leq a_{22}^{(k)}\mathbb{E}\left[\left\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|^2\right]+c_2^{(k)},$$

where we define $a_{21}^{(k)}=0$, and obtain $a_{22}^{(k)}=(1+2\sqrt{2}\alpha^{(k)}L)^2$ and $c_2^{(k)}=m(\alpha^{(k)})^2(2(1+2\sqrt{2}\alpha^{(k)}L)\delta^2/(2\sqrt{2}\alpha^{(k)}L)+\sigma^2)$.

*Proof:* See Appendix D in the supplementary material. ■

We make two observations from the above lemma. First, consider the term $2m\delta^2$ in Lemma 4-(a). This term reveals that even if consensus is reached among the devices, i.e., $\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\|^2=0$, the local gradients would always be different from each other due to the data heterogeneity assumption of 2-(c). Second, the system of inequalities is semi-coupled, as we can bound $\|\mathbf{W}^{(k+1)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\|^2$ at each iteration only by its own value at the previous iterations.

Next, we give the the following definition of the parameters to be analyzed:

*Definition 2:* The optimization error, i.e., the distance between the average model and the optimal solution, is defined as $\|\bar{\mathbf{w}}^{(k)}-\mathbf{w}^\star\|^2$ at iteration $k$. Also, the consensus error, i.e., the distance between model parameters of all devices $i\in\mathcal{M}$ with the average model, is defined as $\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\|^2$ at iteration $k$. We also define the following vector as the expected value of these error terms:

$$\boldsymbol{\Xi}^{(k)}=\begin{bmatrix}\mathbb{E}\left[\left\|\bar{\mathbf{w}}^{(k)}-\mathbf{w}^\star\right\|^2\right]\\ \mathbb{E}\left[\left\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|^2\right]\end{bmatrix}. \tag{15}$$

Using this definition, we write the iterate relations defined in parts (c)&(b) of Lemma 4 as a system of recursive inequalities:

$$\boldsymbol{\Xi}^{(k+1)}\leq\mathbf{A}^{(k)}\boldsymbol{\Xi}^{(k)}+\mathbf{C}^{(k)}, \tag{16}$$

in which $\mathbf{A}^{(k)}=[a_{ij}^{(k)}]_{1\leq i,j\leq 2}$ and $\mathbf{C}^{(k)}=[c_1^{(k)},c_2^{(k)}]^T$, and the $a_{ij}^{(k)}$ and $c_i^{(k)}$ values were defined in Lemmas 4-(b) and 4-(c) for $1\leq i,j\leq 2$.

Next, we derive an explicit relation for the system of inequalities of (16), starting from an arbitrary iteration $s$ as

$$\boldsymbol{\Xi}^{(k+1)}\leq\mathbf{A}^{(k:s)}\boldsymbol{\Xi}^{(s)}+\sum_{r=s+1}^{k}\mathbf{A}^{(k:r)}\mathbf{C}^{(r-1)}+\mathbf{C}^{(k)}, \tag{17}$$

in which $\mathbf{A}^{(k:s)}=\mathbf{A}^{(k)}\cdots\mathbf{A}^{(s)}$. Since $a_{21}^{(k)}=0$, we can easily compute the following entries of $\mathbf{A}^{(k:s)}$ which will be frequently used in our analysis. We have

$$a_{11}^{(k:s)}=a_{11}^{(k)}\cdots a_{11}^{(s)},a_{21}^{(k:s)}=0,a_{22}^{(k:s)}=a_{22}^{(k)}.\cdots a_{22}^{(s)}. \tag{18}$$

As mentioned before Lemma 4, similar analysis to our paper is common in the gradient tracking literature. But current research has only shown convergence guarantees over static communication graphs. Next, we move on to an important lemma of our paper, which is the key to proving the convergence for time-varying graphs with arbitrary connectivity $B$.

*Lemma 5:* Let Assumptions 1-3 hold. Then, using Lemma 4, we can get the following inequality on the expected consensus error of the model weights at iteration $k+B$ for any $k\geq 0$:

$$\mathbb{E}\left[\left\|\mathbf{W}^{(k+B)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k+B)}\right\|^2\right]\leq\phi_{22}^{(k)}\mathbb{E}\left[\left\|\mathbf{W}^{(k)}-\mathbf{1}_m\bar{\mathbf{w}}^{(k)}\right\|^2\right]+\psi_2^{(k)},$$

in which we have defined $\phi_{21}^{(k)}=0$, and obtained $\phi_{22}^{(k)}=\frac{1+\rho^2}{2}+16\frac{BL^2}{1-\rho^2}\sum_{r=k+1}^{k+B}(\alpha^{(r-1)})^2a_{22}^{(r-2:k)}$ and $\psi_2^{(k)}=2B\sum_{r=kB+1}^{(k+1)B}(\alpha^{(r-1)})^2\left\{\frac{2}{1-\rho^2}\left[m\delta^2+4L^2\left(\sum_{l=kB+1}^{r-2}a_{22}^{(r-2:l)}c_2^{(l-1)}+c_2^{(r-2)}\right)\right]+m\sigma^2\right\}$. Further note that $0<\rho^{(k+B-1:k)}\leq\rho<1$ for any $k\geq 0$, and $B$ is the connectivity bound of Proposition 1.

*Proof:* See Appendix E in the supplementary material. ■

We next derive a system of inequalities for the iterate relations of Lemmas 4-(c) and 4-(b), but instead of writing a recursive relation between iteration $k+1$ and $k$ as in (16), we use Lemma 5 to obtain a recursive relation between $k+B$ and $k$ as

$$\boldsymbol{\Xi}^{((k+1)B)}\leq\boldsymbol{\Phi}^{(k)}\boldsymbol{\Xi}^{(kB)}+\boldsymbol{\Psi}^{(k)}, \tag{19}$$

in which $\boldsymbol{\Phi}^{(k)}=[\phi_{ij}^{(k)}]_{1\leq i,j\leq 2}$, and $\boldsymbol{\Psi}^{(k)}=[\psi_1^{(k)},\psi_2^{(k)}]^T$. Next, note that while $\phi_{21}^{(k)}$, $\phi_{22}^{(k)}$ and $\psi_2^{(k)}$ come from Lemma 5, the remaining entries of these matrices are obtained as

$$\phi_{11}^{(k)}=a_{11}^{((k+1)B-1:kB)},\quad\phi_{12}^{(k)}=a_{12}^{((k+1)B-1:kB)}, \tag{20}$$

$$\psi_1^{(k)}=\sum_{r=kB+1}^{(k+1)B-1}\left(\alpha^{(r-1)}\right)^2\left[a_{11}^{((k+1)B-1:r)}\frac{\sigma^2}{m}\right.$$

$$+ 2ma_{12}^{((k+1)B-1:r)}\left(\frac{1+2\sqrt{2}\alpha^{(r-1)}L}{2\sqrt{2}\alpha^{(r-1)}L}\delta^2+2\sigma^2\right)\Bigg] + \frac{\left(\alpha^{((k+1)B-1)}\right)^2\sigma^2}{m}. \tag{21}$$

Comparing (17) and (19), note that we have used $\mathbf{\Phi}^{(k)} = \mathbf{A}^{((k+1)B-1:kB)}$ and $\mathbf{\Psi}^{(k)} = \sum_{r=kB+1}^{(k+1)B-1}\mathbf{A}^{((k+1)B-1:r)}\mathbf{C}^{(r-1)} + \mathbf{C}^{((k+1)B-1)}$, except for two modifications, where we have replaced $\phi_{22}^{(k)}$ and $\psi_2^{(k)}$ with the values derived in Lemma 5.

Next, we derive an explicit equation for the system of inequalities of (19), starting from an arbitrary iteration $s$, as follows

$$\mathbf{\Xi}^{((k+1)B)} \leq \mathbf{\Phi}^{(k:s)}\mathbf{\Xi}^{(sB)} + \sum_{r=s+1}^{k}\mathbf{\Phi}^{(k:r)}\mathbf{\Psi}^{(r-1)} + \mathbf{\Psi}^{(k)}, \tag{22}$$

in which $\mathbf{\Phi}^{(k:s)} = \mathbf{\Phi}^{(k)}\cdots\mathbf{\Phi}^{(s)}$. Since $\phi_{21}^{(k)} = 0$ (see Lemma 5), we can easily compute the following entries of $\mathbf{\Phi}^{(k:s)}$ that will be used in our analysis

$$\phi_{11}^{(k:s)} = \phi_{11}^{(k)}\cdots\phi_{11}^{(s)}, \phi_{21}^{(k:s)} = 0, \qquad \phi_{22}^{(k:s)} = \phi_{22}^{(k)}.\cdots\phi_{22}^{(s)}. \tag{23}$$

In the subsequent proposition, we build on the results of (22), and obtain the conditions under which the spectral norm of $\mathbf{\Phi}^{(k)}$ would be less than one. Then, we use that to bound the system of inequalities of (22).

*Proposition 2:* Let Assumptions 1-3 and 5 hold, and a non-increasing step size be used such that $\alpha^{(k+1)} \leq \alpha^{(k)}$ for all $k \geq 0$.[3] Using the definitions of $\mathbf{\Phi}^{(k:s)}$ and $\mathbf{\Psi}^{(k:s)}$ in (22), if the step size satisfies $\alpha^{(0)} \leq (1-\rho^2)/(8BL(1+\Gamma_1)^{B-1})$ where $\Gamma_1 = (1-\rho^2)/(2\sqrt{2}B)$, then $\rho(\mathbf{\Phi}^{(k)}) < 1$ and the following bound holds

$$\begin{aligned}\mathbf{\Xi}^{(kB)} \leq{}& \mathcal{O}\left(\phi_{11}^{(k-1:K)}\right)\mathcal{O}\left(\phi_{22}^{(K-1:0)}\right)\mathbf{\Xi}^{(0)} \\ &+ \mathcal{O}\left(\phi_{11}^{(k-1:K)}\right)\sum_{r=1}^{K-1}\mathcal{O}\left(\phi_{22}^{(K-1:r)}\right)\mathbf{\Psi}^{(r-1)} \\ &+ \sum_{r=K}^{k-1}\mathcal{O}\left(\phi_{11}^{(k-1:r)}\right)\mathbf{\Psi}^{(r-1)} + \mathbf{\Psi}^{(k-1)},\end{aligned} \tag{24}$$

where iteration $K$ is determined by $\phi_{11}^{(k)} < \phi_{22}^{(k)}$ for all $k = 0,\cdots,K-1$, and $\phi_{11}^{(k)} \geq \phi_{22}^{(k)}$ for $k \geq K$.

Furthermore, the matrix $\mathbf{\Psi}^{(k)}$ can be bounded as

$$\mathbf{\Psi}^{(k)} \leq \left(\alpha^{(kB)}\right)^2 B\left[2mB\left\{2\frac{\delta^2+2\mu\alpha^{(kB)}\nu^{(kB)}}{1-\rho^2}+\sigma^2\right\}\right], \tag{25}$$

where $\nu^{(k)} = (2/\mu)(B-1)L^2(1+\Gamma_1)^{2(B-1)}((1+\Gamma_1)\delta^2/(\sqrt{2}L) + \alpha^{(k)}\sigma^2)$.

*Proof:* See Appendix F in the supplementary material. ■

Proposition 2 lays out the constraints on the step size that have to be satisfied in order for the spectral radius of the state transition matrix, i.e., $\mathbf{\Phi}^{(k)}$ to be less than 1. Our novel analytical approach provided in Appendix F (see the supplementary material) helps us prove that the spectral radius of the aforementioned matrix is less than 1 despite looking at its product over $B$ iterations. Furthermore, it implies that if the spectral norm of $\mathbf{\Phi}^{(k)}$ satisfies $\rho(\mathbf{\Phi}^{(k)}) = \max\{\phi_{11}^{(k)}, \phi_{22}^{(k)}\} < 1$, then we can bound the system of inequalities in terms of the spectral norm. Moreover, the matrix $\mathbf{\Phi}^{(k-1:0)}$ is separated into the product of two terms $\mathbf{\Phi}^{(k-1:K)}\mathbf{\Phi}^{(K-1:0)}$, and this is done because we have $\rho(\mathbf{\Phi}^{(K-1:0)}) = \phi_{22}^{(K-1:0)}$ and $\rho(\mathbf{\Phi}^{(k-1:K-1)}) = \phi_{11}^{(k-1:K-1)}$ (see Appendix F (the supplementary material) for more discussion). This lemma is used directly to prove Theorems 1 and 2.

Next, we analyze the dependence of the constraint on $\alpha^{(0)}$ from Proposition 2 on $B$, which is the connectivity parameter of Proposition 1. We have

$$\begin{aligned}\alpha^{(0)} &\leq \mathcal{O}\left(1/\left[B(1+\Gamma_1)^{B-1}\right]\right) \\ &= \mathcal{O}\left(1/\left[B\left(1+(1-\rho^2)/(2\sqrt{2}B)\right)^{B-1}\right]\right) \\ &= \mathcal{O}\left(1/\left[B(1+1/B)^B\right]\right) = \mathcal{O}(1/B).\end{aligned}$$

We can see that, as expected, the constraint on $\alpha^{(0)}$ is inversely proportional to $B$, meaning that more frequent communications over a well-connected graph (lower $B$) allows us to choose larger step sizes.

### D. Main Convergence Results

We present our most central results, which obtain the convergence characteristics of EF-HC under the step size policies of Assumption 4 in Theorems 1 and 2. In Theorem 2, we reveal that using the diminishing step size of Assumption 4-(b), (a) all devices reach consensus asymptotically, i.e., each device $i$'s model $\mathbf{w}_i^{(k)}$ converges to $\bar{\mathbf{w}}^{(k)} = (1/m)\sum_{i=1}^m \mathbf{w}_i^{(k)}$ as $k \to \infty$, and (b) the final model across the devices (i.e., $\bar{\mathbf{w}}^{(k)}, k \to \infty$) minimizes the global loss. Using the constant step size of assumption 4-(a), we also show in Theorem 1 that the same results for consensus and optimization hold but with an optimality gap which is proportional to the step size.

We further show in Theorem 2 that the convergence rate with a diminishing step size is $\mathcal{O}(\ln k/\sqrt{k})$, which is a desirable property for decentralized gradient descent algorithms [47].

*Theorem 1:* Let Assumptions 1-3 and 5 hold, and the constant step size policy of Assumption 4-(a) be used. Since using a constant step size will make $\mathbf{\Phi}^{(k)}$ and $\mathbf{\Psi}^{(k)}$ of (22) time-invariant, we denote these matrices as $\mathbf{\Phi}$ and $\mathbf{\Psi}$. If the step size $\alpha$ satisfies $\alpha \leq \frac{1-\rho^2}{8BL(1+\Gamma_1)^{B-1}}$ where $\Gamma_1 = \frac{1-\rho^2}{2\sqrt{2}B}$, the following bound holds:

$$\mathbf{\Xi}^{(kB)} \leq \mathcal{O}\left(\rho(\mathbf{\Phi})^k\right)\mathbf{\Xi}^{(0)} + \left(\sum_{r=1}^{k-1}\mathcal{O}\left(\rho(\mathbf{\Phi})^{k-r-1}\right)+1\right)\mathbf{\Psi}. \tag{26}$$

Letting $k \to \infty$, we will get

$$\limsup_{k\to\infty}\mathbf{\Xi}^{(kB)} \leq \frac{\alpha^2 B}{\mathcal{O}(1-\rho(\mathbf{\Phi}))}\left[2mB\left\{2\frac{\frac{\sigma^2}{m}+\nu}{1-\rho^2}+\sigma^2\right\}\right], \tag{27}$$

[3]This satisfies the step size policies of both Assumption 4-(a) and 4-(b).

where $\nu = (2/\mu)(B-1)L^2(1+\Gamma_1)^{2(B-1)}((1+\Gamma_1)\delta^2/(\sqrt{2}L)+\alpha\sigma^2)$.

*Proof:* See Appendix G in the supplementary material. ■

**Discussion on Theorem 1.** This theorem indicates that, using a constant step size, linear convergence is achieved due to the term $\mathcal{O}(\rho(\mathbf{\Phi})^k)$ in Eq. (26). However, we observe in Eq. (27) that using a constant step size will result in an asymptotic optimality gap of $(\alpha^2 B/\mathcal{O}(1-\rho(\mathbf{\Phi})))[\psi_1,\psi_2]^T$. This gap is proportional to the step size $\alpha$ and the connectivity bound of the information flow graph $B$ (see Proposition 1). Thus, choosing a smaller $\alpha$ and employing a strategy that conducts communication rounds more frequently (decreasing $B$), results in the optimality gap getting smaller. Additionally, note that the entries of the optimality gap vector in Eq. (27), i.e., $\psi_1$ and $\psi_2$, depend on the data heterogeneity bound $\delta$ (formalized in Assumption 2-(c)) through $\nu$, and the gradient approximation errors $\sigma$ (formalized in Assumption 3-(b)). As expected, we see that a higher $\delta$ results in a higher value for the upper bound. This also implies that in a federated learning setup where data distributions among devices are non-IID – that is, $\delta \neq 0$ – the optimality gap cannot be made zero when a constant step size is employed, even if full batch sizes are used for the gradient updates, i.e., $\sigma = 0$.

Before presenting Theorem 2, we provide a supplementary lemma as a better alternative to Lemma 4 in [54]. We will later use this lemma in the proof of Theorem 2 in Appendix H (see the supplementary material). This key mathematical result helps us obtain exact convergence rates on last iterates in Theorem 2, when the diminishing step size policy of Assumption 4-(b) is used.

*Lemma 6:* Let $\{\zeta_r\}_{r=0}^{\infty}$ be a scalar sequence where $0 < \zeta_r \leq 1$, $\forall r \geq 0$. For any $p \geq 1$, we have

$$\prod_{r=s}^{k}(1-\zeta_r)^p \leq \frac{1}{p\sum_{r=s}^{k}\zeta_r}.$$

*Proof:* See Appendix B in the supplementary material. ■

*Theorem 2:* Let Assumptions 1-3 and 5 hold, and the diminishing step size policy of Assumption 4-(b) be used with $\alpha^{(k)} = \frac{\alpha^{(0)}}{\sqrt{1+k/\eta}}$. If the step size satisfies

$$\alpha^{(0)} \leq \frac{1-\rho^2}{4\sqrt{2}\sqrt{5-3\rho^2}BL(1+\Gamma_1)^{B-1}},$$

where $\Gamma_1 = \frac{1-\rho^2}{2\sqrt{2}B}$, the following bound holds:

$$\begin{aligned}\mathbf{\Xi}^{(kB)} \leq{}& \frac{1}{2\mu\alpha^{(0)}}\mathcal{O}\left(\frac{1}{\sqrt{k}}\right)\left(\frac{3+\rho^2}{4}\right)^K \mathbf{\Xi}^{(0)} \\ &+\left\{\frac{K-1}{2\mu\alpha^{(0)}}\mathcal{O}\left(\frac{1}{\sqrt{k}}\right)\left(\frac{1+\rho^2}{2}\right)+\frac{\alpha^{(0)}}{2}\mathcal{O}\left(\frac{\ln k}{\sqrt{k}}\right)\right. \\ &\left.+(\alpha^{(0)})^2\mathcal{O}\left(\frac{1}{k}\right)\right\}B\begin{bmatrix}\widehat{\psi}_1\\ \widehat{\psi}_2\end{bmatrix}, \end{aligned} \tag{28}$$

in which we have $\widehat{\psi}_1 = \sigma^2/m + \nu^{(0)}$, $\widehat{\psi}_2 = 2mB\left\{2(\delta^2+2\mu\alpha^{(0)}\nu^{(0)})/(1-\rho^2)+\sigma^2\right\}$, and $\nu^{(k)} = (2/\mu)(B-1)L^2(1+\Gamma_1)^{2(B-1)}\left((1+\Gamma_1)\delta^2/(\sqrt{2}L)+\alpha^{(k)}\sigma^2\right)$

Letting $k \to \infty$, we will get

$$\limsup_{k\to\infty}\mathbf{\Xi}^{(kB)} = 0. \tag{29}$$

*Proof:* See Appendix H in the supplementary material. ■

**Discussion on Theorem 2.** Theorem 2 implies that using the diminishing step size of $\alpha^{(k)} = \frac{\alpha^{(0)}}{\sqrt{1+k/\eta}}$, a sub-linear rate of convergence $\mathcal{O}(\ln k/\sqrt{k})$ can be achieved, and that the models of all devices asymptotically converge to the global optimum point. For the more general setup that the diminishing step size is chosen at $\alpha^{(k)} = \frac{\alpha^{(0)}}{(1+k/\eta)^\theta}$ with $\theta \in (0.5, 1]$, see Appendix I (the supplementary material). Also, the upper bound in Eq. (28) captures the effect of data heterogeneity level $\delta$ through the vector values $\widehat{\psi}_1$ and $\widehat{\psi}_2$ (since $\nu^{(0)}$ is a function of $\delta$), where a higher value of $\delta$ results in a larger value for the upper bound.

**Effect of communication sparsity.** Note that sparse communications would affect the transition spectral radius $\rho(\mathbf{\Phi}^{(k)})$ of Eq. (19) through the information flow graph connectivity parameter $B$ (see Proposition 1, where we show how $B$ captures the effects of both physical connectivity level $B_1$ and communication interval $B_2$ introduced in Assumption 5). This is because $\rho(\mathbf{\Phi}^{(k)}) = \max\{\phi_{11}^{(k)}, \phi_{22}^{(k)}\}$ (see Appendix F) (the supplementary material), and the definitions of $\phi_{11}^{(k)}$ and $\phi_{22}^{(k)}$ given in Lemmas 5 and Eq. (20) illustrate the dependence of these terms on the parameter $B$. However, although a larger $B$ would result in a higher value of $\rho(\mathbf{\Phi}^{(k)})$, thus slowing the convergence rate (see Eq. (26) in Theorem 1), our Proposition 1 lays out the constraints on the step size $\alpha^{(k)}$ such that we would still get a spectral radius of $\rho(\mathbf{\Phi}^{(k)}) < 1$. We would essentially achieve this by using a smaller step size, as we see an inverse dependence of step size $\alpha^{(k)}$ on the communication level parameter $B$ in Proposition 1.

**Significance of theoretical results.** Importantly, note that in both Theorems 1 and 2, we analyze the convergence behavior of our methodology for the last iterates of model parameters, while seminal papers have focused on average iterates [15], [17], i.e., $(1/T)\sum_{t=0}^{T-1}\mathbf{w}^{(t)}$. We provide exact convergence rates for the last iterates of model parameters in spite of the thresholds in the event-triggering mechanism being different from device to device and the underlying physical network connecting the agents being time-varying, i.e., based on Assumption 5. Thus, this is a stronger result that complements those provided in [15] and [17], and more generally has not been shown in the literature to date. Furthermore, while papers such as [13], [16], and [55] also assume strong convexity as we have done in Assumption 2-(b), they are still confined to showing convergence for average iterates.

Furthermore, the analysis of existing research was based on the (sub)-gradient bound assumption [1], while we have replaced that with two more general assumptions, namely smoothness (Assumption 2-(a)) and statistical heterogeneity of data (Assumption 2-(c)). In both Theorems 1 and 2, we show convergence for a $B$-connected time-varying graph (see Proposition 1), in spite of not making the more restrictive bounded (sub)gradients assumption [35].

**Summary of analysis.** To summarize, the goals of Proposition 2 and Theorems 1 and 2 of our paper are to (i) show that convergence to a globally optimal solution of FL can be achieved even when the event-triggering thresholds are personalized for the devices, and (ii) obtain bounds on the rate of model training convergence under such conditions, i.e., for general time-varying consensus graphs.

## IV. Numerical Results

In this section, we conduct numerical evaluations to assess the effectiveness of our methodology. We explain the setup of our experiments in Sec. IV-A and provide the results and discussion in Sec. IV-B. For further experiments and ablation studies, please see Appendix J (the supplementary material).

### A. Simulation Setup

**Datasets and models.** We evaluate our proposed methodology using two image classification tasks: Fashion-MNIST (FMNIST) [56], and Federated Extended MNIST (FEMNIST) [57]. Note that FMNIST contains data belonging to 10 labels, while FEMNIST contains data points with 62 different labels. We employ two models as classifiers, a support vector machine (SVM) and a 5-layer convolutional neural network (CNN). The loss function $\ell_\xi(\mathbf{w})$ in (1) is chosen as the multi-margin loss for the SVM model, and the cross-entropy loss for the CNN. Note that SVM satisfies the convexity assumption (see 2-(b)), while the CNN does not; thus, we will numerically evaluate the efficacy of `EF-HC` using both convex and non-convex models, although our theoretical analysis only covers convex loss functions.

**Graph topology and data distribution.** In the simulations for the FMNIST and FEMNIST datasets, a network of devices $m = 10$ and $m = 30$ is used, respectively, in which the underlying communications topology is generated according to random graphs. We conduct evaluations on two types of graphs: (i) random geometric graph, resembling local wireless networks [11], [58], with radius 0.4 by default; and (ii) the Internet graph of autonomous systems (AS) from [59]. In the Internet graph, AS are categorized into four types: tier-1, mid-level, customer and content providers, which are divided into different regions to model geographical constraints [59]. We treat each AS as a node in our system. Further, to generate non-IID data distributions across devices, each device only contains samples of the dataset from a subset of the labels. For FMNIST and FEMNIST, we consider 1 and 3 labels/device, respectively.

**Resource heterogeneity.** Link bandwidths $b_i$ are randomly chosen for each device $i$ from a probability distribution. For completeness, we have run our experiments using two different distributions; (i) uniform distribution $\mathcal{U}((1-\sigma_N)b_M, (1+\sigma_N)b_M)$, with a mean of $b_M = 5000$ and a normalized standard deviation of $\sigma_N = 0.9$, and (ii) the beta distribution $\text{Beta}(\alpha,\beta)\cdot b_M$ with $\alpha = \beta = 0.5$ for a inverted bell-shaped distribution. For the uniform distribution, we define $\sigma_N = \sigma\sqrt{3}/b_M$, in which $\sigma$ is the standard deviation of the uniform distribution. For uniform distribution, the heterogeneity of the system resources is controlled by the standard deviation, since the value of $\sigma_N = 0$ means that all devices are homogeneous in terms of resource capabilities, and $\sigma \to 1$ means choosing $b_i$ values from the range $\mathcal{U}(0, 2b_M)$. After randomly choosing $b_i$ for each device, we assign the same value as the bandwidth of all outgoing links of the device $i$ for simplicity, i.e., we do not assign different values for each outgoing link of a device $i$.

In each simulation, the diminishing step size is selected as $\alpha^{(k)} = 0.1/\sqrt{1+k}$, and the threshold decay rate is set to $\gamma^{(k)} = \alpha^{(k)}$. Also, in (3), we set the threshold $r = b_M \times 5 \times 10^{-2}$ for FMNIST, and $r = b_M \times 10^{-1}$ for FEMNIST.

**Metrics.** At iteration $k$, we define a resource utilization score as $(1/m)\sum_{i=1}^m \sum_{j=1}^m (v_{ij}^{(k)}/d_i^{(k)})\rho_i n$, which for our proposed method where $\rho_i = 1/b_i$, this score is the same as the average transmission time, that is, $(1/m)\sum_{i=1}^m \left(\sum_{j=1}^m v_{ij}^{(k)}/d_i^{(k)}\right)(n/b_i)$. The term $\sum_{j=1}^m v_{ij}^{(k)}/d_i^{(k)}$ is the utilization of the outgoing links for device $i$, making this score the weighted average of link utilization, penalizing devices with larger $\rho_i$.

### B. Results and Discussion

We compare the performance of our method `EF-HC` against three baselines: (1) Distributed learning with aggregations at every iteration, i.e., using zero thresholds (denoted by *ZT*), which is a foundational synchronous method in contrast to our event-triggered approach; (2) Decentralized event-triggered FL, with the same global threshold $r\rho\gamma^{(k)}$ across all devices (denoted by *GT*), where $\rho = 1/b_M$ is chosen as the average of personalized thresholds of `EF-HC` for a fair comparison; (3) Randomized gossip, where each device engages in broadcast communication with probability of $1/m$ at each iteration [21] (denoted by *RG*). The parameter $r$ in both `EF-HC` and GT as discussed in Sec. IV-A of the paper is chosen so that their frequency of communications would be comparable to RG, with the difference that RG does not conduct communications in an intelligent event-triggered way and instead does them randomly. We illustrate the performance of our method against these baselines in Fig. 2.

**Communication resource usage.** We first illustrate the average transmission time units each algorithm requires per training iteration in Figs. 2a-(i), 2b-(i), 3a-(i), 3b-(i), 4a-(i) and 4b-(i). As we can see, `EF-HC` results in a shorter transmission delay compared to *ZT* and *GT*, significantly helping to resolve the impact of stragglers by not requiring the same amount of communications from devices with less available bandwidth. However, it is important to note that although a shorter transmission delay per iteration is beneficial for a decentralized optimization algorithm, it can negatively impact the performance of the classification task. Hence, a better comparison between multiple decentralized algorithms is to measure the accuracy reached per transmission time units. In this regard, although *RG* achieves less transmission delay per iteration compared to our method in most cases, Figs. 2a-(iii), 2b-(iii), 3a-(iii), 3b-(iii), 4a-(iii) and 4b-(iii) reveal that it achieves substantially lower model performance, indicating that our method strikes an effective balance between these objectives.

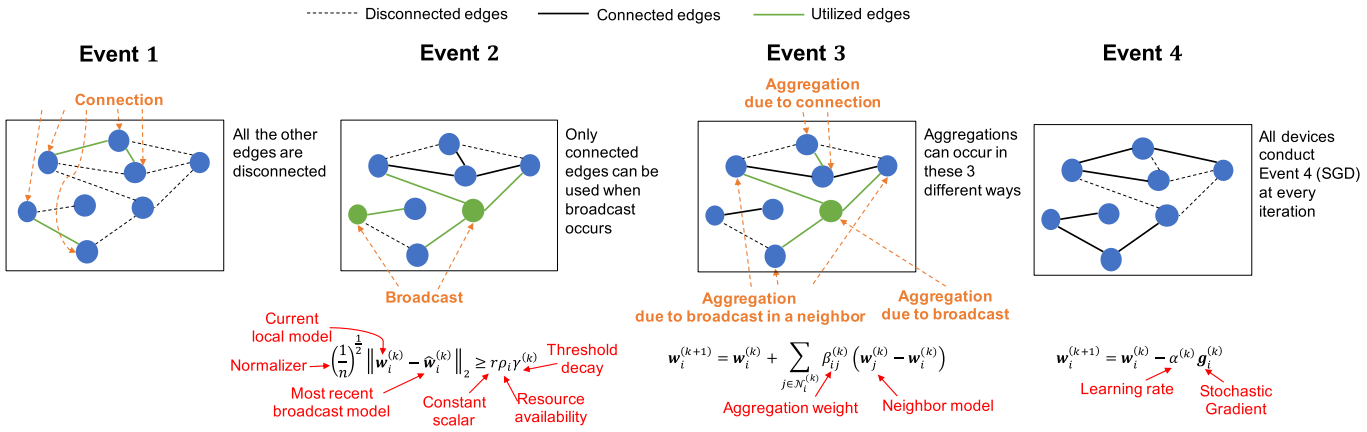
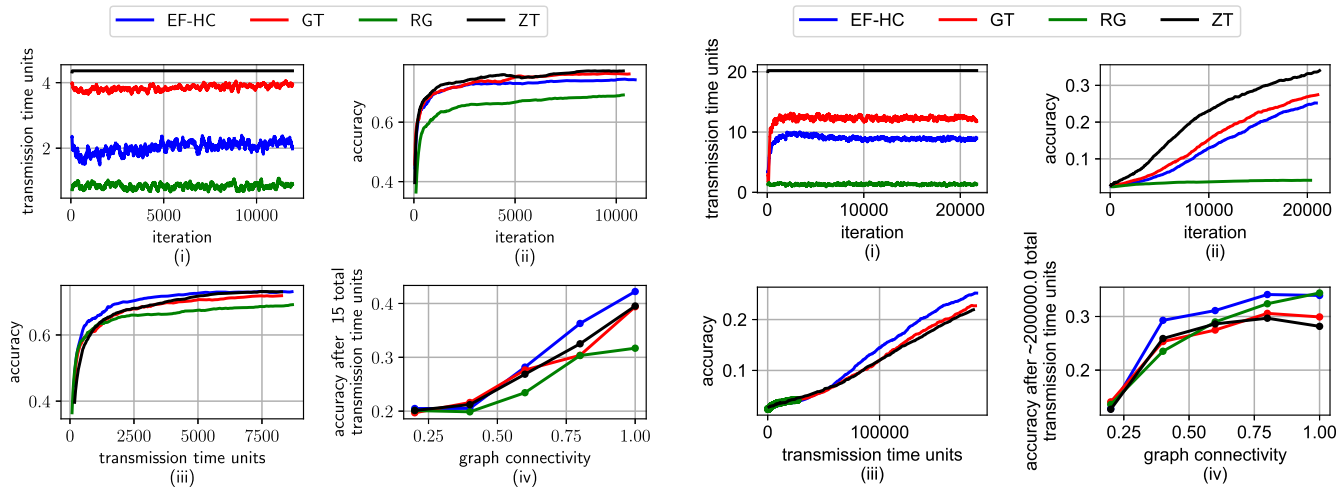



Fig. 1. System diagram of a time-varying decentralized system, illustrating the four events of Alg. 1, namely (i) neighbor connection, (ii) model broadcast, (iii) model aggregation, and (iv) stochastic gradient descent.



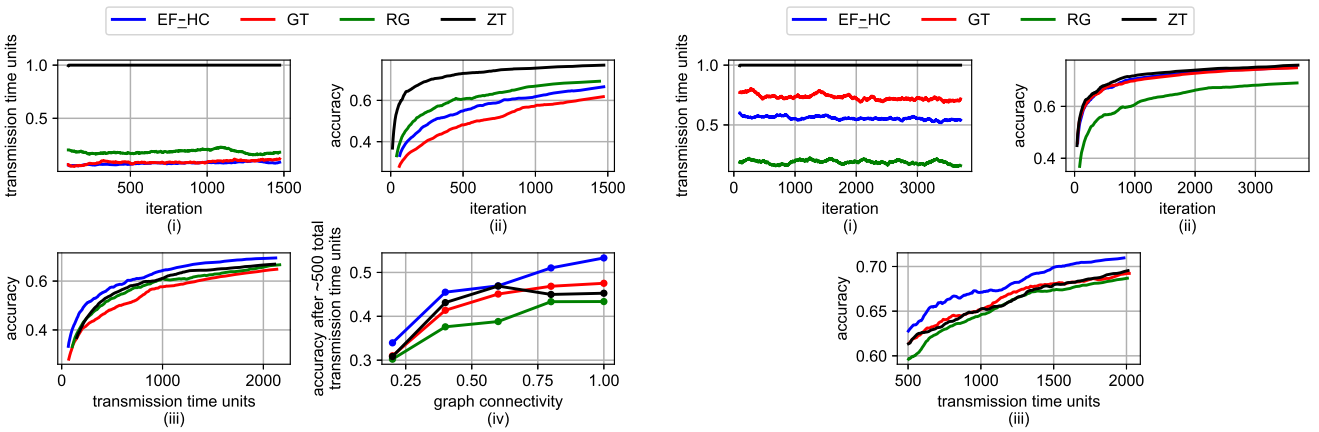

(a) FMNIST dataset (b) FEMNIST dataset

Fig. 2. Performance comparison between our method (EF−HC), global threshold (GT), zero threshold (ZT), and randomized gossip (RG) algorithms on (a) FMNIST and (b) FEMNIST datasets using an SVM model. The resources are allocated to devices using a uniform distribution $\mathcal{U}((1-\sigma_N)b_M, (1+\sigma_N)b_M)$ under a random geometric graph. The plots show (i) transmission time per iteration, (ii) accuracy per iteration, (iii) accuracy per transmission time, and (iv) accuracy after a certain number of transmissions with respect to graph connectivity. For this figure, the link bandwidths among devices are generated using a uniform distribution $\mathcal{U}((1-\sigma_N)b_M, (1+\sigma_N)b_M)$. The devices themselves are connected to each other via random geometric graph. We see how our EF−HC algorithm achieves higher accuracies with less transmission time passed in Figs. 2a-(iii) and 2b-(iii), and also how its advantage remains consistent across different graph connectivities in Figs. 2a-(iv) and 2b-(iv).

**Accuracy achieved per iteration of training.** Figs. 2a-(ii), 2b-(ii), 3a-(ii), 3b-(ii), 4a-(ii) and 4b-(ii) depict the average accuracy of the devices per iteration. These plots are indicative of processing efficiency since they evaluate the accuracy of algorithms per number of gradient descent computations. As expected, the baseline *ZT* is able to achieve the highest accuracy per iteration, since it does not take into account resource efficiency and thus sacrifices network resources to achieve better accuracy. In other words, the value of $B$ explained in Proposition 1 for *ZT* has the minimum possible value compared to other algorithms, since $B_2$ of Assumption 5-(b) has the value of $B_2 = 1$ for it. In most of these plots, that is, Figs. 2a-(ii), 2b-(ii), 3b-(ii), 4a-(ii) and 4b-(ii), we show that unlike *RG*, the performance of our proposed method EF−HC as well as *GT*, which is also event-triggered, does not degrade considerably although they use less communication resources, as will be discussed next.

**Accuracy achieved per total delay.** Figs. 2a-(iii), 2b-(iii), 3a-(iii), 3b-(iii), 4a-(iii) and 4b-(iii) are the most critical results, as they assess the accuracy vs. communication time trade-off. We see that our algorithm EF−HC can achieve higher accuracy while using less transmission time compared to all baselines. These plots reveal that our method can adapt to non-IID data distributions across devices, which is an important characteristic of FL algorithms [2], and achieve better accuracy compared to baselines given a fixed transmission time, that is, under a fixed network resource consumption.

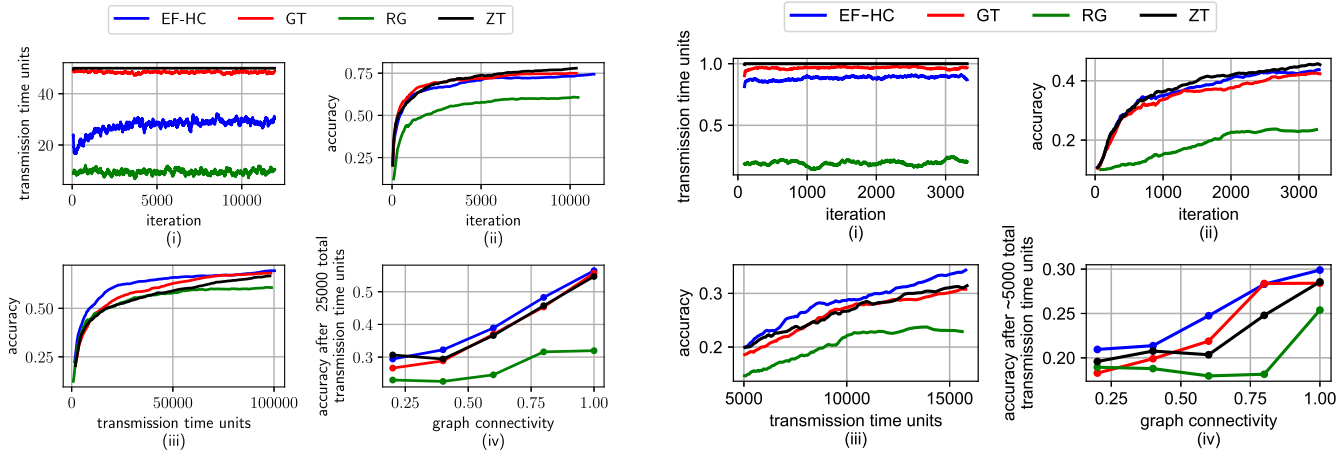
**Effect of graph connectivity.** Furthermore, we evaluated the effect of network connectivity on our method and baselines in Figs. 2a-(iv), 2b-(iv), 3a-(iv), 4a-(iv) and 4b-(iv). Since the graphs are generated randomly in our simulations, we




(a) Random geometric graph.

(b) Internet AS network graph.

Fig. 3. Performance comparison between our method (EF−HC) and the baselines using the FMNIST dataset. For this figure, the link bandwidths among devices are generated using a beta distribution $\text{Beta}(0.5, 0.5) \cdot b_M$. The devices in Figs. 3a and 3b are connected to each other via a random geometric graph and the Internet graph, respectively. We observe that regardless of the network topology, our EF−HC algorithm achieves higher accuracies faster in terms of the total transmission time passed. Also, comparing Fig. 2a to Fig. 3a, we observe that our proposed methodology outperforms the baselines for both uniform and beta distribution, which are used to sample the link bandwidths. (Note that the connectivity of the Internet graph is fixed and cannot be varied as for the random geometric graph).




(a) Uniformly distributed link bandwidths.

(b) Beta distributed link bandwidths.

Fig. 4. Performance comparison between our method (EF−HC) and the baselines using a CNN classifier on the FMNIST dataset. We use a random geometric graph as the network topology, and sample the link bandwidths in two different ways: (a) uniform distribution $\mathcal{U}((1-\sigma_N)b_M, (1+\sigma_N)b_M)$ and (b) beta distribution $\text{Beta}(0.5, 0.5) \cdot b_M$. We observe that the superiority of our EF−HC algorithm holds when using a non-convex model as well.

have taken the average performance of all four algorithms over 5 Monte Carlo instances to reduce the effect of random initialization. It can be seen that higher network connectivity improves the convergence speed of our method and most of the baselines. Importantly, however, we see that our method has the highest improvement per increase in connectivity. This becomes more pronounced in Figs. 3a-(iv) and 4b-(iv), as the degree of resource heterogeneity between devices is higher because they are sampled from a beta distribution (vs. uniform in Figs. 2a-(iv), 2b-(iv) and 4a-(iv)), and EF−HC is best suited for highly heterogeneous scenarios.

**Non-convex model.** To generate non-IID data distributions across devices, each device only contains samples of the dataset from a subset of labels, specifically 2 labels/device in these experiments. Note that in Figs. 4a-(iv) and 4b-(iv), we change the simulation setup and set $r = b_M \times 10^{-3}$, and let the devices have samples of only 1 labels/device. Looking at Fig. 4a, we can see that results similar to those of the SVM classifier (see Fig. 2 and Sec. IV-B) can be achieved with a CNN classifier as well, i.e., the results hold with and without the model convexity assumption used in our convergence analysis. Furthermore, the gap between the accuracy achieved by EF−HC in Fig. 4a-(iii) per a given delay compared to other baselines is more significant than its linear SVM counterpart.

**Summary of improvements.** Finally, we note that Figs. 2a, 2b, 3a, 3b, 4a and 4b collectively demonstrate that our EF−HC algorithm's improvements hold under various settings. First, by comparing Figs. 2a and 2b we can see that EF−HC outperforms all baselines under different datasets that the model is being trained, i.e., FMNIST and FEMNIST, respectively. Second, a comparison of Figs. 2a and 3a (or Figs. 4a and 4b)

illustrates that the improvements of EF−HC hold when different probability distributions are employed for sampling link bandwidths, i.e., uniform and beta distributions, respectively. Third, Figs. 2a and 3b show that EF−HC maintains its effectiveness in accuracy vs. resource utilization trade-off for different graph topologies connecting the devices together, i.e., random geometric graph and internet AS graph, respectively. Finally, we can compare Figs. 2a and 4a (or Figs. 3a and 4b) that EF−HC maintains its advantage gap from the baselines for both convex and non-convex models, i.e., linear SVM and a CNN architecture, respectively.

## V. Conclusion and Future Work

In this paper, we develop a novel methodology for decentralized FL, in which model aggregations are performed through D2D communications among devices. We proposed an asynchronous, event-triggered communications mechanism in which each device decides itself when to broadcast its model parameters to its neighbors. Furthermore, to alleviate straggler effects, we developed personalized thresholds for event-triggering conditions in which each device determines its communication frequency according to its available bandwidth. Through theoretical analysis, we demonstrated that our algorithm converges to the global optimal model with a $\mathcal{O}(\ln k/\sqrt{k})$ rate for appropriate step sizes. Our analysis holds for the last iterates under relaxed graph connectivity and data heterogeneity assumptions. To do so, we showed that the graph of information flow among devices is connected under our method, despite the fact that sporadic communications are conducted over a time-varying network graph.

Our work also gives rise to various future directions. For instance, it is promising to extend our methodology to consider event-triggering in gradient computations as well, complementing the event-triggered communications framework established in this work.

## References


[1] S. Zehtabi, S. Hosseinalipour, and C. G. Brinton, "Decentralized event-triggered federated learning with heterogeneous communication thresholds," in *Proc. IEEE 61st Conf. Decis. Control (CDC)*, Dec. 2022, pp. 4680–4687.

[2] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends ML*, vol. 14, nos. 1–2, pp. 1–210, 2021.

[3] J. J. Konečnỳ, H. B. McMahan, F. X. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*.

[4] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.

[5] S. Wang, Y. Ruan, Y. Tu, S. Wagle, C. G. Brinton, and C. Joe-Wong, "Network-aware optimization of distributed learning for fog computing," *IEEE/ACM Trans. Netw.*, vol. 29, no. 5, pp. 2019–2032, Oct. 2021.

[6] S. Wang et al., "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[7] K. Bonawitz et al., "Towards federated learning at scale: System design," in *Proc. Mach. Learn. Sys.*, vol. 1, 2019, pp. 374–388.

[8] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[9] E. T. M. Beltrán et al., "Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 25, no. 4, pp. 2983–3013, 4th Quart., 2023.

[10] S. Hosseinalipour, C. G. Brinton, V. Aggarwal, H. Dai, and M. Chiang, "From federated to fog learning: Distributed machine learning over heterogeneous wireless networks," *IEEE Commun. Mag.*, vol. 58, no. 12, pp. 41–47, Dec. 2020.

[11] S. Hosseinalipour et al., "Multi-stage hybrid federated learning over large-scale D2D-enabled fog networks," *IEEE/ACM Trans. Netw.*, vol. 30, no. 4, pp. 1569–1584, Aug. 2022.

[12] A. Lalitha, O. C. Kilinc, T. Javidi, and F. Koushanfar, "Peer-to-peer federated learning on graphs," 2019, *arXiv:1901.11173*.

[13] A. Koloskova, N. Loizou, S. Boreiri, M. Jaggi, and S. U. Stich, "A unified theory of decentralized SGD with changing topology and local updates," in *Proc. Int. Conf. Mach. Learn.*, vol. 1, 2020, pp. 5381–5393.

[14] J. Liu, J. Liu, H. Xu, Y. Liao, Z. Wang, and Q. Ma, "YOGA: Adaptive layer-wise model aggregation for decentralized federated learning," *IEEE/ACM Trans. Netw.*, vol. 32, no. 2, pp. 1768–1780, Apr. 2024.

[15] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[16] A. Nedic and A. Olshevsky, "Distributed optimization of strongly convex functions on directed time-varying graphs," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2013, pp. 329–332.

[17] S. Sundhar Ram, A. Nedić, and V. V. Veeravalli, "Distributed stochastic subgradient projection algorithms for convex optimization," *J. Optim. Theory Appl.*, vol. 147, no. 3, pp. 516–545, Dec. 2010.

[18] T. Sun, D. Li, and B. Wang, "Decentralized federated averaging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 4, pp. 4289–4301, Apr. 2023.

[19] K. Mishchenko, G. Malinovsky, S. U. Stich, and P. Richtárik, "ProxSkip: Yes! Local gradient steps provably lead to communication acceleration! Finally!," in *Proc. Int. Conf. Mach. Learn.*, Mali, 2022, pp. 15750–15769.

[20] A. Nedic, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, Apr. 2010.

[21] S. Pu and A. Nedic, "Distributed stochastic gradient tracking methods," *Math. Program.*, vol. 187, nos. 1–2, pp. 409–457, May 2021.

[22] A. Nedic and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 601–615, Mar. 2015.

[23] R. Xin and U. A. Khan, "A linear algorithm for optimization over directed graphs with geometric convergence," *IEEE Control Syst. Lett.*, vol. 2, no. 3, pp. 315–320, Jul. 2018.

[24] S. Boyd, P. Diaconis, and L. Xiao, "Fastest mixing Markov chain on a graph," *SIAM Rev.*, vol. 46, no. 4, pp. 667–689, Jan. 2004.

[25] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, Sep. 2004.

[26] Y. Liu, T. Lin, A. Koloskova, and S. U. Stich, "Decentralized gradient tracking with local steps," *Optim. Methods Softw.*, vol. 40, no. 5, pp. 1–28, Sep. 2025.

[27] E. D. H. Nguyen, S. A. Alghunaim, K. Yuan, and C. A. Uribe, "On the performance of gradient tracking with local updates," in *Proc. 62nd IEEE Conf. Decis. Control (CDC)*, Dec. 2023, pp. 4309–4313.

[28] B. Gharesifard and J. Cortés, "When does a digraph admit a doubly stochastic adjacency matrix," in *Proc. Amer. Control Conf. (ACC)*, 2010, pp. 2440–2445.

[29] F. Saadatniaki, R. Xin, and U. A. Khan, "Decentralized optimization over time-varying directed graphs with row and column-stochastic matrices," *IEEE Trans. Autom. Control*, vol. 65, no. 11, pp. 4769–4780, Nov. 2020.

[30] S. Pu, W. Shi, J. Xu, and A. Nedic, "Push–pull gradient methods for distributed optimization in networks," *IEEE Trans. Autom. Control*, vol. 66, no. 1, pp. 1–16, Jan. 2021.

[31] M. S. Assran and M. G. Rabbat, "Asynchronous gradient push," *IEEE Trans. Autom. Control*, vol. 66, no. 1, pp. 168–183, Jan. 2021.

[32] N. Bof, R. Carli, G. Notarstefano, L. Schenato, and D. Varagnolo, "Multiagent Newton–Raphson optimization over lossy networks," *IEEE Trans. Autom. Control*, vol. 64, no. 7, pp. 2983–2990, Jul. 2019.

[33] Y. Liao, Y. Xu, H. Xu, M. Chen, L. Wang, and C. Qiao, "Asynchronous decentralized federated learning for heterogeneous devices," *IEEE/ACM Trans. Netw.*, vol. 32, no. 5, pp. 4535–4550, Oct. 2024.

[34] T. Yang et al., "A survey of distributed optimization," *Annu. Rev. Control*, vol. 47, pp. 278–305, Jun. 2019.

[35] J. George and P. Gurram, "Distributed stochastic gradient descent with event-triggered communication," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 5, pp. 7169–7178.

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

[37] H. Wang, Z. Kaplan, D. Niu, and B. Li, "Optimizing federated learning on non-IID data with reinforcement learning," in *Proc. IEEE Conf. Comput. Commun. (IEEE INFOCOM)*, Jul. 2020, pp. 1698–1707.
[38] M. Chen, N. Shlezinger, H. V. Poor, Y. C. Eldar, and S. Cui, "Communication-efficient federated learning," *Proc. Nat. Acad. Sci. USA*, vol. 118, no. 17, 2021, Art. no. 2024789118.
[39] T. Nishio and R. Yonetani, "Client selection for federated learning with heterogeneous resources in mobile edge," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–7.
[40] H. T. Nguyen, V. Sehwag, S. Hosseinalipour, C. G. Brinton, M. Chiang, and H. V. Poor, "Fast-convergent federated learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 1, pp. 201–218, Jan. 2021.
[41] E. Diao, J. Ding, and V. Tarokh, "HeteroFL: Computation and communication efficient federated learning for heterogeneous clients," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020.
[42] X. Gu, K. Huang, J. Zhang, and L. Huang, "Fast federated learning in the presence of arbitrary device unavailability," in *Proc. Adv. Neur. Inf. Process. Sys. (NeurIPS)*, 2021.
[43] N. Shlezinger, M. Chen, Y. C. Eldar, H. V. Poor, and S. Cui, "Federated learning with quantization constraints," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 8851–8855.
[44] G. Lan, X.-Y. Liu, Y. Zhang, and X. Wang, "Communication-efficient federated learning for resource-constrained edge devices," *IEEE Trans. Mach. Learn. Commun. Netw.*, vol. 1, pp. 210–224, 2023.
[45] C. Renggli, S. Ashkboos, M. Aghagolzadeh, D. Alistarh, and T. Hoefler, "SparCML: High-performance sparse communication for machine learning," in *Proc. Int. Conf. High Perform. Comput., Netw., Storage Anal.*, Nov. 2019, pp. 1–15.
[46] S. Savazzi, M. Nicoli, and V. Rampa, "Federated learning with cooperating devices: A consensus approach for massive IoT networks," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4641–4654, May 2020.
[47] 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 J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3851–3869, Dec. 2021.
[48] Y. Sun, J. Shao, Y. Mao, J. H. Wang, and J. Zhang, "Semi-decentralized federated edge learning with data and device heterogeneity," *IEEE Trans. Netw. Service Manage.*, vol. 20, no. 2, pp. 1487–1501, Jun. 2023.
[49] S. Zehtabi, D.-J. Han, R. Parasnis, S. Hosseinalipour, and C. G. Brinton, "Decentralized sporadic federated learning: A unified algorithmic framework with convergence guarantees," in *Proc. ICLR*, 2024.
[50] S. Zehtabi, S. Hosseinalipour, and C. G. Brinton, "Event-triggered decentralized federated learning over resource-constrained edge devices," 2022, *arXiv:2211.12640*.
[51] J. Tsitsiklis, D. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. AC-31, no. 9, pp. 803–812, Sep. 1986.
[52] A. Nedić, A. Olshevsky, and W. Shi, "Achieving geometric convergence for distributed optimization over time-varying graphs," *SIAM J. Optim.*, vol. 27, no. 4, pp. 2597–2633, Jan. 2017.
[53] G. Qu and N. Li, "Harnessing smoothness to accelerate distributed optimization," *IEEE Trans. Control Netw. Syst.*, vol. 5, no. 3, pp. 1245–1260, Sep. 2018.
[54] S. Pu, A. Olshevsky, and I. C. Paschalidis, "A sharp estimate on the transient time of distributed stochastic gradient descent," *IEEE Trans. Autom. Control*, vol. 67, no. 11, pp. 5900–5915, Nov. 2022.
[55] K. I. Tsianos and M. G. Rabbat, "Distributed strongly convex optimization," in *Proc. 50th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Oct. 2012, pp. 593–600.
[56] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017, *arXiv:1708.07747*.
[57] S. Caldas et al., "LEAF: A benchmark for federated settings," 2018, *arXiv:1812.01097*.
[58] Y. Hmamouche, M. Benjillali, S. Saoudi, H. Yanikomeroglu, and M. D. Renzo, "New trends in stochastic geometry for wireless networks: A tutorial and survey," *Proc. IEEE*, vol. 109, no. 7, pp. 1200–1252, Jul. 2021.
[59] A. Elmokashfi, A. Kvalbein, and C. Dovrolis, "On the scalability of BGP: The role of topology growth," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 8, pp. 1250–1261, Oct. 2010.

**Shahryar Zehtabi** (Graduate Student Member, IEEE) received the B.Sc. degrees in EE and CE from the Amirkabir University of Technology in 2020 and 2021, respectively, and the M.Sc. degree in ECE from Purdue University in 2024, where he is currently pursuing the Ph.D. degree in ECE. He was a recipient of the 2024 Magoon Award for Graduate Teaching Assistants.

**Seyyedali Hosseinalipour** (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in EE from NCSU in 2017 and 2020, respectively. He was a Post-Doctoral Researcher at Purdue University from 2020 to 2022. He is currently an Assistant Professor with the EE Department, State University of New York at Buffalo. He has won the 2020 ECE Doctoral Scholar of the Year Award and the 2021 ECE Distinguished Dissertation Award at NCSU.

**Christopher G. Brinton** (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in EE from Princeton University in 2013 and 2016, respectively. He is currently an Elmore Associate Professor of ECE at Purdue University. He was a recipient of the four U.S. Top Early Career Awards from the National Science Foundation (CAREER), the Office of Naval Research (YIP), the Defense Advanced Research Projects Agency (YFA), and the Air Force Office of Scientific Research (YIP). He was a recipient of the Intel Rising Star Faculty Award and the Qualcomm Faculty Award.