Graph Neural Networks for the Optimization of Collaborative Federated Learning Energy Efficiency

Nuocheng Yang, *Student Member, IEEE*, Sihua Wang, *Member, IEEE*, Yuchen Liu, *Member, IEEE*, Christopher G. Brinton, *Senior Member, IEEE*, Changchuan Yin, *Senior Member, IEEE*, and Mingzhe Chen, *Senior Member, IEEE*

***Abstract*—This paper delves into the design of an energy efficient collaborative federated learning (CFL) methodology using which mobile devices exchange their FL model with a subset of their neighbors without reliance on a parameter server based on the distributed graph neural network (GNN) method. Each device is unable to send its FL model to every neighboring device due to device mobility and wireless resource limitations. To reduce the energy consumption of FL model transmission, each device must choose a subset of devices with which to share its FL model. This problem is formulated as an optimization problem to meet the constraints of delay and training loss while minimizing the energy consumption for model transmission. However, the formulated problem is difficult to solve since the device mobility patterns, and the relationship between the device connection scheme and CFL performance are unknown. To address this challenge, we analytically characterize the relationship between dynamic device connections and the performance of CFL methodology. Based on the analysis, a GNN based algorithm is proposed to enable each device to select a subset of its neighbors and the transmit power in a decentralized method. Compared to standard optimization methods that must determine device connections in a centralized manner, the GNN based method enables each device to use its neighboring devices' location and connection information to individually determine a subset of devices to transmit the local model. Given the device connections, the optimal transmit power of each device can be determined by convex optimization. Simulation results show that the proposed method can reduce the energy consumption for model transmission and training loss by up to 46% and 2%, respectively.**

***Index Terms*—Collaborative federated learning, energy consumption, graph neural network.**

Received 12 December 2024; revised 6 May 2025; accepted 17 June 2025. Date of publication 24 June 2025; date of current version 3 October 2025. This work was supported in part by the National Key Research and Development Program of China under Grant 2024YFE0200300, in part by Beijing Natural Science Foundation under Grant L223027, in part by the National Natural Science Foundation of China under Grant 62471056, and in part by the 111 Project under Grant B17007. Recommended for acceptance by M. S. Hossain. *(Corresponding author: Changchuan Yin.)*

Nuocheng Yang, Sihua Wang, and Changchuan Yin are with the Beijing Laboratory of Advanced Information Network, Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: yangnuocheng@bupt.edu.cn; sihuawang@bupt.edu.cn; ccyin@bupt.edu.cn).

Yuchen Liu is with the Department of Computer Science, North Carolina State University, Raleigh, NC 27606 USA (e-mail: yuchen.liu@ncsu.edu).

Christopher G. Brinton is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907 USA (e-mail: cgb@purdue.edu).

Mingzhe Chen is with the Department of Electrical and Computer Engineering, Institute for Data Science and Computing, University of Miami, Coral Gables, FL 33146 USA (e-mail: mingzhe.chen@miami.edu).

Digital Object Identifier 10.1109/TMC.2025.3582911

I. Introduction

FEDERATED learning (FL) [1], [2], [3] allows devices to train a machine learning (ML) model collaboratively without data sharing across wireless channels thus improving data privacy. However, standard FL requires devices to upload their FL models to a parameter server, which may limit its application scenarios since standard FL requires communications between the parameter server and devices frequently. Collaborative federated learning (CFL) [4], a novel FL framework that integrates federated learning with collaborative learning principles, has been presented as a solution to this issue. CFL framework enables devices to collaboratively train an ML model via exchanging their local model with their neighboring devices without relying on a parameter server. Compared to standard FL, CFL enables more devices to join FL training. For example, devices with limited wireless capacity can connect to neighboring devices to participate in the CFL training even if they cannot establish a direct connection with the parameter server, making it suitable for deployment in large-scale systems such as mobile networks. However, the implementation of CFL over wireless networks presents several challenges including CFL learning efficiency optimization, device connection optimization [5], and distributed gradient update method design [6].

A. Related Works

Recently, several works [7], [8], [9], [10], [11], [12] have focused on the CFL performance optimization. The authors in [7] jointly considered network topology and FL model compression to improve CFL convergence speed. The authors in [8] developed a model-based deep Q-network (DQN) algorithm to optimize the quantization bit and device selection, thus improving CFL training performance. The authors in [9] designed a device connection scheme to improve CFL convergence speed. The authors in [10] investigated the impact of different model update rules on the convergence speed of CFL. The authors in [11] developed a distributed gradient descent algorithm to enhance the CFL convergence rate. The authors in [12] proposed a semisupervised CFL framework to address the data heterogeneity of modalities caused by different collection methods. However, these methods [7], [8], [9], [10], [11], [12] ignored the impact of network dynamics (i.e., devices' mobility) on CFL training. In [13], the authors integrated a hierarchical Bayesian framework to optimize CFL performance considering the varying wireless channel

state information. The authors in [14] proposed a hierarchical asynchronous aggregation mechanism to reduce communication costs and enhance the aggregation efficiency of mobile devices. However, the works in [13], [14] used traditional algorithms such as convex optimization to optimize the CFL training process. Hence, we must rerun these optimization algorithms to account for the dynamics of the devices' location in the CFL framework, leading to increased latency in each round [15]. Reinforcement learning (RL) presents a promising solution to this problem by evaluating the impacts of devices' movement on CFL training and adaptably enhancing device connections by exploring the dynamic environment.

Several existing works such as [16], [17], [18], [19], [20] have investigated the use of RL to optimize CFL performance. The work in [16] developed a soft actor-critic (SAC) reinforcement learning (RL) algorithm to improve FL communication efficiency in mobile scenarios. In [17], the authors used RL to optimize the device connection scheme under non-independent and identically distributed (non-IID) data distribution. The authors in [18] used an RL method for efficient device selection and training to improve the convergence and accuracy of FL. In [20], the authors designed a decentralized multi-agent RL (MARL) to efficiently determine device connections and manage wireless resources in industrial IoT networks. Since the size of the RL [17], [18], [19], [20] input depends on the number of participating FL devices and cannot be adjusted as the number of participating FL devices changes, these designed RL methods can only be applied to a static network where the number of FL participating devices is constant and will not vary over time.

To tackle this issue, one promising solution is to use graph neural networks (GNNs) [21] that can adapt flexibly to a dynamic network where the number of participating FL devices changes over time [22], [23]. The GNNs are extremely beneficial in solving optimization problems on an n-node graph problem (where the nodes may represent users, base stations, or antennas) due to their rotational invariance characteristic and their ability to grasp the graph relationship between nodes [24]. The authors in [25] utilized GNNs to cluster devices according to their data distribution to improve CFL performance. The authors in [26] developed a GNN framework to allocate the transmit power of each device to enhance the CFL performance. In [27], the authors proposed a GNN-based algorithm that improves communication efficiency by allowing partial model parameter sharing. However, although the GNNs used in [25], [26], [27] are proficient in handling graph data, they struggle to focus on important nodes (all nodes have the same importance). Graph attention networks (GAT) [28], [29], [30] address this shortcoming by dynamically assessing the importance of devices, facilitating a more sophisticated and efficient engagement with the dynamic information within the CFL context (i.e., locations) compared with conventional GNNs. Furthermore, these works [25], [26], [27] rely on a centralized server to run the GNN method, which is unsuitable for the CFL framework which enables devices to train an ML model collaboratively without a parameter server.

B. Contributions

The main contribution of this paper is a novel fully distributed CFL framework in which each device can use local observation to individually determine device connection and transmit power that minimizes the CFL loss and energy consumption. Our key contributions include:

- We propose a CFL framework in which wireless mobile devices collaboratively train an ML model by exchanging their FL model with their neighboring devices. In the considered model, due to the limited energy and bandwidth resources, each device must select a subset of devices per FL iteration to exchange their FL model for minimizing CFL training loss. Since the location of each device changes over time, it is necessary to optimize the device connection scheme dynamically. This problem is formulated as an optimization problem that jointly determines the device connections and the transmit power to minimize the energy consumption of CFL model transmission while satisfying the delay and the FL training loss requirements.
- To solve this problem, we first derive a closed-form expression for convergence of CFL. Our analysis shows how the device connection scheme, the learning rate, and the number of devices influence the expected convergence of CFL. Based on this relationship, we find a minimum number of neighboring devices that each device must connect to satisfy the constraints of CFL training loss. Then, a GAT based algorithm is developed, allowing each device to individually select a subset of devices to share local FL models. Compared to traditional algorithms that require iterative optimized device connections, the proposed GNN approach is capable of directly obtaining the optimal device connections. Given the device connection scheme, a closed form solution for the minimum energy consumption of FL model transmission is derived.

Simulation results show that, compared to a fully distributed algorithm where each device will directly connect to its first and the second nearest neighbors, the proposed GNN algorithm can reduce energy consumption and training loss by up to 46% and 2%, respectively.

II. System Model and Problem Formulation

As illustrated in Fig. 1, consider a wireless network consisting of a set $\mathcal{M}$ of M mobile devices that jointly train a supervised learning model. We assume that for each device i, there are N_i training data samples. Each training data sample n comprises an input feature vector $\boldsymbol{x}_{i,n} \in \mathbb{R}^{N_{\mathrm{I}} \times 1}$ and a corresponding label vector $\boldsymbol{y}_{i,n} \in \mathbb{R}^{N_{\mathrm{O}} \times 1}$. Training's goal is to reduce the global loss function over all data samples, i.e.,

$$F\left(\boldsymbol{g}\right) = \min_{\boldsymbol{g}} \frac{1}{N} \sum_{i=1}^{M} \sum_{n=1}^{N_i} f\left(\boldsymbol{g}_i, \boldsymbol{x}_{i,n}, \boldsymbol{y}_{i,n}\right), \tag{1}$$

where $\boldsymbol{g} = [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_M]$, $\boldsymbol{g}_i \in \mathbb{R}^{V \times 1}$ is a vector of the local model of device i. $N = \sum_{i=1}^{M} N_i$ is the total number of training data samples across devices. $f(\boldsymbol{g}_i, \boldsymbol{x}_{i,n}, \boldsymbol{y}_{i,n})$ is a loss function

TABLE I
LIST OF NOTATIONS

Notation	Description	Notation	Description
M	Number of devices	$(\boldsymbol{x}_{i,n}, \boldsymbol{y}_{i,n})$	Data collected by device i
N	Total number of training data samples	$p_{i,j}^t$	The transmit power of device i to device j
$\mathcal{M}$	Set of devices	$\boldsymbol{g}_i^t$	Local model parameters of device i
σ_N^2	Variance of additive white Gaussian noise	$l_{i,j,t}$	CFL model transmission delay from device i to j
S	Size of the CFL model	$E_{i,j,t}$	The energy for device i to transmit model to j
$E_{i,t}$	The energy for device i to transmit FL model	$\boldsymbol{p}_i^t$	The transmit power vector of device i at time slot t
$\boldsymbol{U}$	FL model transmission matrix	$\boldsymbol{P}$	FL model transmission power matrix
F^*	CFL training loss requirement	$r_{i,j,t}$	Transmission data rate of device i
B	Bandwidth of each device	$d_{i,j,t}$	Distance between device i and j
$\boldsymbol{\phi}_{i,t}$	The location of each device i at the time t	$\boldsymbol{u}_i^t$	Vector of FL model transmission index of device i
Γ	The maximum FL model transmission delay	$\mathcal{T}$	Set of time slot
$F\left(\boldsymbol{g}_i^t\right)$	Loss function of FL model at device i	$\boldsymbol{A}_t$	Model aggregation matrix
$\rho_{i,j,t}$	Rayleigh fading parameters	η_t	Learning rate
$\boldsymbol{g}*$	The optimal FL model	k	The nearest neighbor graph
$\boldsymbol{u}_i^*$	The optimal transmission vector of device i	$\boldsymbol{p}_i^*$	The optimal transmit power of device i
$\boldsymbol{R}^{(i,k)}$	The nearest neighbor matrix	$\mathcal{L}^1(i,k)$	The set of first hop devices of device i
$\mathcal{L}^2(i,k)$	The set of second hop devices of device i	$\mathcal{L}(i,k)$	The set contain first hop, second hop and device i
$F(\boldsymbol{g}_i^t)$	Loss function of device i	$\nabla F(\boldsymbol{g}_i^t)$	Gradient of device i
$\mathbf{Z}$	Output of the second GAT layer	$\mathbf{H}$	Output of the first GAT layer
$\alpha_{i,j}^{k_1}$	Self attention coefficients of first GAT layer	$\alpha_{i,j}^{k_2}$	Self attention coefficients of second GAT layer
V_1	Output dimension of first GAT layer	V_2	Output dimension of second GAT layer
W^{k_2}	The model parameter of the first GAT layer	W^{k_4}	The model parameter of the second GAT layer

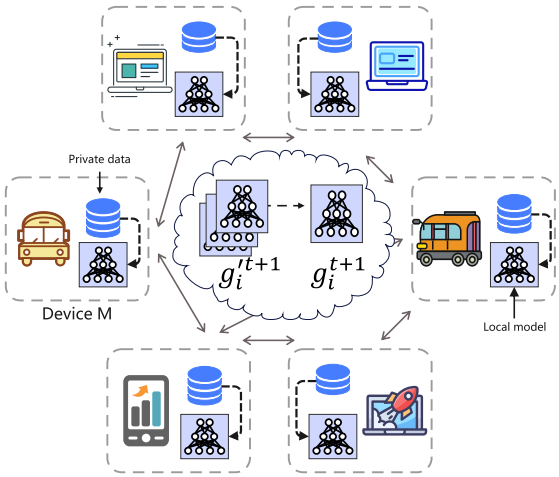


Fig. 1. Illustration of the considered CFL model.

(e.g., squared error) that measures the accuracy of the generated local model $\boldsymbol{g}_i$ in building a relationship between the input vector $\boldsymbol{x}_{i,n}$ and the output vector $\boldsymbol{y}_{i,n}$.

Different from existing works that require all devices to transmit FL model to a central server for model aggregation, we propose a novel FL that does not rely on a central controller. In particular, the designed FL training process is explained as follows:

1) Each device randomly initializes its local model.
2) Each device updates its local model using its training data samples.
3) Each device selects a subset of its neighboring devices and exchanges its updated local model with the selected devices.
4) Each device aggregates the FL model received from other devices to generate its FL model.

Steps (2)–(4) are performed until the optimal $\boldsymbol{g}$ is found. Next, we will provide a mathematical explanation of the intended FL training procedure. Table I includes a collection of important notations and associated explanations for easier comprehension.

A. CFL Training Process

Each device's local model is updated using a backward propagation (BP) approach based on stochastic gradient descent (SGD), which is given by

$$\boldsymbol{g'}_i^{t+1} = \boldsymbol{g}_i^t - \eta_t \sum_{n \in \mathcal{N}_{i,t}} \frac{\partial f\left(\boldsymbol{g}_i^t, \boldsymbol{x}_{i,n}, \boldsymbol{y}_{i,n}\right)}{\partial \boldsymbol{g}_i^t}, \tag{2}$$

where $\boldsymbol{g'}_i^{t+1}$ is the updated local model and η_t is the learning rate at iteration t. $\mathcal{N}_{i,t}$ is the subset of selected training data samples of device i at iteration t. Given the updated local model, each device i exchanges $\boldsymbol{g'}_i^{t+1}$ with its selected devices and aggregate their FL model, which can be given by [10]

$$\begin{aligned} &\boldsymbol{g}_i^{t+1}\left(\boldsymbol{u}_i^t\right) \\ &= \boldsymbol{g'}_i^{t+1} + \sum_{j \in \mathcal{M}} \frac{u_{j,i}^t}{\max\{||\boldsymbol{u}_i^t||, ||\boldsymbol{u}_j^t||\}} (\boldsymbol{g'}_j^{t+1} - \boldsymbol{g'}_i^{t+1}), \end{aligned} \tag{3}$$

where $\boldsymbol{u}_i^t = [u_{i,1}^t, \ldots, u_{i,M}^t]$ is a vector of FL transmission index of device i with $u_{i,j}^t = 1$ implying that device i will exchange its local model with device j at iteration t, and $u_{i,j}^t = 0$, otherwise. $||\boldsymbol{u}_i^t|| = \sum_{j \in \mathcal{M}} u_{i,j}^t$ is the number of devices that will transmit FL model to device i.

Based on (3), we define a model aggregation matrix $\boldsymbol{A}_t$ where each element $a_{i,j}^t$ represents the weight of $\boldsymbol{g'}_i^t$ contributing to model aggregation. Specifically, $a_{i,j}^t$ is given by

$$a_{i,j}^t = \begin{cases} 1 - \sum\limits_{j \in \mathcal{M}} \frac{u_{j,i}^t}{\max\{||\boldsymbol{u}_i^t||, ||\boldsymbol{u}_j^t||\}}, & \text{if } i = j, \\ \frac{u_{j,i}^t}{\max\{||\boldsymbol{u}_i^t||, ||\boldsymbol{u}_j^t||\}}, & \text{otherwise.} \end{cases} \tag{4}$$

Here, the model aggregation matrix $\boldsymbol{A}_t$ is a doubly stochastic matrix since $\sum_{j\in\mathcal{M}} a_{i,j}^t = 1$ and $(\boldsymbol{A}_t)^\top = \boldsymbol{A}_t$. Based on [31], the maximum eigenvalue of $\boldsymbol{A}_t$ is always 1, and all other eigenvalues are not larger than 1, i.e., $\max\{|\zeta_2(\boldsymbol{A}_t)|, |\zeta_M(\boldsymbol{A}_t)|\} \leq 1$ where $\zeta_k(\boldsymbol{A}_t)$ is the k-th largest eigenvalue of $\boldsymbol{A}_t$. The property of the doubly stochastic matrix will be used to prove theorems in Section III.

B. Mobility Model

The movement of each device is simulated using the random walk paradigm [32]. Specifically, during every CFL iteration, every device has the option to keep still or move in four directions: a) forward, b) backward, c) left, and d) right. The likelihood of each movement is represented by a vector $\boldsymbol{\xi}_i^t = [\xi_{i,0}^t, \xi_{i,1}^t, \xi_{i,2}^t, \xi_{i,3}^t, \xi_{i,4}^t]$. The location of device i at the time t is represented by $\boldsymbol{\phi}_i^t = [\phi_{i,1}^t, \phi_{i,2}^t]$ and the speed is v_i. Each time slot lasts for a duration of Δt. Given that, the location of device i at the $t+1$ is given by

$$\boldsymbol{\phi}_i^t = \begin{cases} \left[\phi_{i,1}^t, \phi_{i,2}^t\right], \text{with probability}\, \xi_{i,0}^t, \\ \left[\phi_{i,1}^t, \phi_{i,2}^t + v_i\Delta t\right], \text{with probability}\, \xi_{i,1}^t, \\ \left[\phi_{i,1}^t, \phi_{i,2}^t - v_i\Delta t\right], \text{with probability}\, \xi_{i,2}^t, \\ \left[\phi_{i,1}^t - v_i\Delta t, \phi_{i,2}^t\right], \text{with probability}\, \xi_{i,3}^t, \\ \left[\phi_{i,1}^t + v_i\Delta t, \phi_{i,2}^t\right], \text{with probability}\, \xi_{i,4}^t. \end{cases} \tag{5}$$

C. CFL Model Transmission Process

We utilize the orthogonal frequency division multiple access (OFDMA) transmission scheme to transmit CFL model. The data rate at which device i transmits the FL model to device j is

$$r_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t) = \frac{B}{||\boldsymbol{u}_i^t||}\log\left(1 + \frac{p_{i,j}^t h_{i,j,t}\left(\boldsymbol{\phi}^t\right)}{\sigma_N^2}\right), \tag{6}$$

where $h_{i,j,t}(\boldsymbol{\phi}^t) = \rho_{i,j,t} d_{i,j,t}^{-2}$ is the channel gain between device i and j with $\rho_{i,j,t}$ being the Rayleigh fading parameter, $d_{i,j,t}$ denotes the distance between them. B is the bandwidth and $p_{i,j}^t$ is the transmit power of device i to device j. $\boldsymbol{\phi}^t = [\boldsymbol{\phi}_1^t, \ldots, \boldsymbol{\phi}_M^t] \in \mathbb{R}^{M\times 2}$ contain the location of each devices. σ_N^2 represents the variance of additive white Gaussian noise. Based on (6), the delay of device i transmitting FL model to device j is

$$l_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t) = \frac{S}{r_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t)}, \tag{7}$$

where S is the size of the CFL model, which is assumed consistent across all devices.

The energy consumption of device i transmitting its FL model to device j is

$$E_{i,j,t}\left(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t\right) = p_{i,j}^t l_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t). \tag{8}$$

Then, the energy consumption of device i for model transmission during iteration t is

$$E_{i,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, \boldsymbol{p}_i^t) = \sum_{j=1}^{M} u_{i,j}^t E_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t), \tag{9}$$

where $\boldsymbol{p}_i^t = [p_{i,1}^t, \ldots, p_{i,M}^t]$ is the transmit power vector.

D. Problem Formulation

Our goal is to minimize the total energy consumption of CFL model transmission jointly considering the constraints of latency and CFL training loss. The optimization problem is formulated as

$$\min_{\boldsymbol{U},\boldsymbol{P}} \sum_{t=1}^{T}\sum_{i=1}^{M} E_{i,t}\left(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, \boldsymbol{p}_i^t\right), \tag{10}$$

$$\text{s.t. } l_{i,j,t}(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t) \leqslant \Gamma, \forall i,j \in \mathcal{M}, \forall t \in \mathcal{T}, \tag{10a}$$

$$F\left(\boldsymbol{g}_i^T\left(\boldsymbol{u}_i^{T-1}\right)\right) \leqslant F^*, \forall i \in \mathcal{M}, \tag{10b}$$

where $\boldsymbol{U} = [\boldsymbol{U}_1, \ldots, \boldsymbol{U}_T]^\top \in \mathbb{R}^{T\times M\times M}$ is the FL model transmission matrix, $\boldsymbol{P} = [\boldsymbol{P}_1, \ldots, \boldsymbol{P}_T]^\top \in \mathbb{R}^{T\times M\times M}$ is the transmit power matrix. Here, $\boldsymbol{U}_t = [\boldsymbol{u}_1^t, \ldots, \boldsymbol{u}_M^t] \in \mathbb{R}^{M\times M}$ and $\boldsymbol{P}_t = [\boldsymbol{p}_1^t, \ldots, \boldsymbol{p}_M^t]$ is the device connection matrix and transmit power matrix for each device at time slot t, respectively. (10a) is a constraint on the FL model transmission delay per iteration and Γ is the maximum transmission delay. (10b) is the CFL training loss requirement. The problem (10) is difficult to solve using typical centralized algorithms for the following reasons. First, since $\boldsymbol{U}$ and $\boldsymbol{P}$ are dependent, iterative convex optimization algorithms will result in high computational overhead. Furthermore, in problem (10), device locations will change over time. To adapt to the changes of devices locations, traditional iterative algorithms must re-execute. In addition, these iterative algorithms require each device to share its location information, which may be not feasible in practice.

III. Analysis of the CFL Convergence

To simplify problem (10), we must analyze the impact of the device connection $\boldsymbol{U}$ on CFL training loss. To this end, we first make the following assumptions, as done in [33], [34], [35].

- Frist, we assume that the gradient $\nabla F(\boldsymbol{g}_i^t)$ of $F(\boldsymbol{g}_i^t)$ is uniformly Lipschitz continuous with respect to $\boldsymbol{g}_i^t$. Thus, we have
$$||\nabla f(\boldsymbol{g}_i^{t+1}) - \nabla f(\boldsymbol{g}_i^t)|| \leqslant L||\boldsymbol{g}_i^{t+1} - \boldsymbol{g}_i^t||, \tag{11}$$
where L is a positive constant and $||\boldsymbol{g}_i^{t+1} - \boldsymbol{g}_i^t||$ is the norm of $\boldsymbol{g}_i^{t+1} - \boldsymbol{g}_i^t$.
- Second, we assume that $F(\boldsymbol{g})$ is strongly convex with a positive parameter μ, such that
$$F(\boldsymbol{g}_i^{t+1}) \geqslant F(\boldsymbol{g}_i^t) + \left(\boldsymbol{g}_i^{t+1} - \boldsymbol{g}_i^t\right)^\top \nabla F\left(\boldsymbol{g}_i^t\right) + \frac{\mu}{2}||\boldsymbol{g}_i^{t+1} - \boldsymbol{g}_i^t||^2. \tag{12}$$
- Third, we assume that $F(\boldsymbol{g}_i^t)$ is twice-continuously differentiable, based on (11) and (12), we have
$$\mu\boldsymbol{I} \preceq \nabla^2 F\left(\boldsymbol{g}_i^t\right) \preceq L\boldsymbol{I}. \tag{13}$$
where $\boldsymbol{I}$ is identity matrix.

These assumptions can be satisfied by the CFL model aggregation process [6] and several widely used loss functions.

To find the relationship between the CFL convergence performance and $\boldsymbol{U}$, we can analyze the convergence of each local model. However, since the model aggregation only depends on a

subset of neighboring devices, the local model $\boldsymbol{g}_i^t$ of each device is inconsistent. Hence, we can only analyze the convergence of the average model of all devices per FL iteration. In particular, we first analyze the upper bound of the gap between each local model $\boldsymbol{g}_i^t$ and the average model $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ of all devices. Then, we prove that the average model of all devices will converge to the optimal model $\boldsymbol{g}^*$. To this end, the upper bound of the gap between the local model $\boldsymbol{g}_i^t$ and the average model $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ is shown in Theorem 1.

Theorem 1: Given the model transmission matrix $\boldsymbol{U}_t$ at each iteration, an upper bound of the gap between each local model $\boldsymbol{g}_i^t$ of device i and the average model $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ of all devices can be given by

$$\sum_{i=1}^{M}||\boldsymbol{g}_i^t-\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t||_2^2\leqslant\left(\eta^2ML\frac{1-\lambda^t}{1-\lambda}\right)^2, \tag{14}$$

where $\lambda=\sup_{t-1\geqslant l\geqslant 0}(\zeta_2((\prod_{k=0}^{l}\boldsymbol{A}_k)^l))^t$ is the t-th power of the second largest eigenvalue of matrix $\prod_{k=0}^{t}\boldsymbol{A}_k$, t is the number of iterations. $\eta=\sup_{t\geqslant l\geqslant 0}\eta_l$ is the largest learning rate among all iterations.

Proof: See Appendix A. ■

From Theorem 1, we see that the gap $(\eta^2ML\frac{1-\lambda^t}{1-\lambda})^2$ is affected by λ which depends on the model aggregation matrix $\boldsymbol{A}_k$. From Theorem 1, we see that as λ decreases, the divergence between $\boldsymbol{g}_i^t$ and $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ decreases, which in turn accelerates the convergence of CFL. Furthermore, Theorem 1 indicates that the gap between $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ and $\boldsymbol{g}_i^t$ is bounded if and only if $\lambda<1$ (i.e. all devices are always directly or indirectly connected). Next, we will prove that $\frac{1}{M}\sum_{i=1}^{M}\boldsymbol{g}_i^t$ will finally converge to $\boldsymbol{g}^*$ by the following theorem.

Theorem 2: As $\eta_l=\frac{1}{\sqrt{T}}$, the loss function gap between the mean of average model $\frac{1}{MT}\sum_{l=1}^{T}\sum_{i=1}^{M}\boldsymbol{g}_i^l$ and the optimal model $\boldsymbol{g}^*$ is bounded by

$$f\left(\frac{1}{MT}\sum_{l=1}^{T}\sum_{i=1}^{M}\boldsymbol{g}_i^l\right)-f(\boldsymbol{g}^*)\leqslant\frac{(\boldsymbol{y}_0-\boldsymbol{g}^*)^2+L^2}{2\sqrt{T}}+\frac{L^3}{T(1-\lambda)^2}. \tag{15}$$

Proof: See Appendix B. ■

From Theorem 2, we see that the gap of loss function between $\frac{1}{MT}\sum_{l=1}^{T}\sum_{i=1}^{M}\boldsymbol{g}_i^l$ and $\boldsymbol{g}^*$ is also affected by λ. As λ decreases, the loss gap between $\frac{1}{MT}\sum_{l=1}^{T}\sum_{i=1}^{M}\boldsymbol{g}_i^l$ and $\boldsymbol{g}^*$ decreases. Meanwhile, as time elapses, the final loss gap converges to zero.

Based on Theorems 1 and 2, we can find that maintaining a connected wireless network (i.e., $\lambda<1$) can guarantee the CFL convergence. From Theorem 2, we also see that, to solve the problem (10), we can generate device connections that satisfy $\lambda<1$ with minimum energy consumption of CFL model transmission.

IV. Proposed Method

To find the connected network with minimum FL model transmission energy consumption that satisfies $\lambda<1$, a novel two-stage distributed approach is proposed. At the first stage, we propose a GNN algorithm [36], [37] which enables each device to independently determine the connections probability with its neighboring devices by utilizing their location information. Then, given the estimated connection probabilities, a fully distributed GHS algorithm is used to determine the FL model transmission matrix $\boldsymbol{U}$. Given $\boldsymbol{U}$, we use a convex optimization algorithm to obtain the transmit power matrix $\boldsymbol{P}$ that can minimize the energy consumption during the transmission of CFL, at the second stage. Next, we will delve into the constituents of the designed GNN-based algorithm and its training method. Then, we will introduce the GHS algorithm to determine the device connection matrix $\boldsymbol{U}$ and the optimization of $\boldsymbol{P}$. Finally, we examine the algorithm's overall complexity.

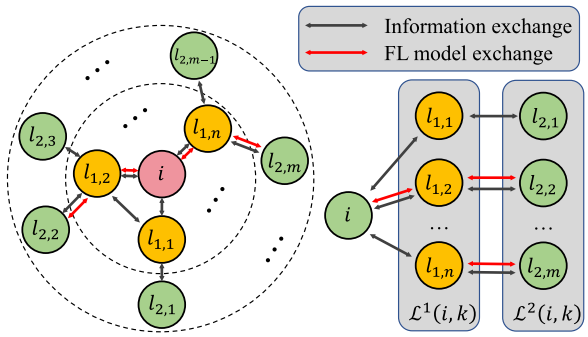


Fig. 2. Exchange information and FL model with neighbors.

A. GNN-Based Algorithm for Connection Optimization

The proposed GNN-based algorithm implemented by each device consists of three components: a) input, b) hidden layers, and c) output, which are specified as follows:

1) Input: To define the input of the GNN model at each device i, we first denote the set of k nearest neighbors of device i as $\mathcal{L}^1(i,k)$, i.e. first-hop devices. Let $\mathcal{L}^2(i,k)$ be the set of k nearest neighbors of the devices in $\mathcal{L}^1(i,k)$, i.e. second-hop devices. For example, in Fig. 2, $\mathcal{L}^1(i,k)=\{l_{1,1},l_{1,2},\ldots,l_{1,n}\}$ while $\mathcal{L}^2(i,k)=\{l_{2,1},l_{2,2},\ldots,l_{2,m}\}$. Let $\mathcal{L}(i,k)=\mathcal{L}^1(i,k)\cup\mathcal{L}^2(i,k)\cup\{i\}$ and $|\mathcal{L}(i,k)|$ is the number of devices in $\mathcal{L}(i,k)$. Then, we can define a matrix $\boldsymbol{R}^{(i,k)}\in\mathbb{R}^{|\mathcal{L}(i,k)|\times|\mathcal{L}(i,k)|}$ to represent the neighbor relationship among devices in $\mathcal{L}(i,k)$, where each element $R_{s,d}^{(i,k)}\in\{0,1\}$ in $\boldsymbol{R}^{(i,k)}$ with $R_{s,d}^{(i,k)}=1$ indicating that device d is connected to device s (i.e. $d\in\mathcal{L}^1(s,k)$) and $R_{s,d}^{(i,k)}=0$, otherwise. Here, using Top-k neighbors instead of all reachable neighbors can effectively reduce the computational complexity without reducing the accuracy of GNN, which will be explained in the subsequent complexity analysis and simulation result. The vector indicating the locations of devices in $\mathcal{L}(i,k)$ can be expressed as $\boldsymbol{\Phi}_i=[\boldsymbol{\phi}_{1,t},\ldots,\boldsymbol{\phi}_{|\mathcal{L}(i,k)|,t}]^\top\in\mathbb{R}^{|\mathcal{L}(i,k)|\times 2}$. Given the definitions, the input of a GNN model at each device i is $\boldsymbol{R}^{(i,k)}$ and $\boldsymbol{\Phi}_i$.

2) GNN Structure: The GNN hidden layers consist of a) GAT layer I, b) fully connected (FC) layer II, c) GAT layer III, and d) FC layer IV, which are expressed as

$$\boldsymbol{q}_{i,t}=\psi\left(\frac{1}{K_1}\sum_{k_1=1}^{K_1}\sum_{j\in\mathcal{L}^1(i,j)}\alpha_{i,j}^{k_1}\boldsymbol{W}^{k_1}\boldsymbol{\phi}_j^t\right), \tag{16}$$

$$\boldsymbol{z}_{i,t} = \psi\left(\boldsymbol{W}^{K_2} \times (\boldsymbol{q}_{i,t}, \boldsymbol{\phi}_i^t)\right), \tag{17}$$

$$\boldsymbol{q}'_{i,t} = \psi\left(\frac{1}{K_3}\sum_{k_3=1}^{K_3}\sum_{j\in\mathcal{L}^1(i,j)} \alpha_{i,j}^{k_3}\boldsymbol{W}^{k_3}\boldsymbol{z}_{j,t}\right), \tag{18}$$

$$\boldsymbol{z}'_{i,t} = \psi\left(\boldsymbol{W}^{K_4} \times (\boldsymbol{q}'_{i,t}, \boldsymbol{\phi}_i^t)\right), \tag{19}$$

where $\boldsymbol{q}_{i,t}$ and $\boldsymbol{q}'_{i,t}$ are the output of the GAT layer I and III, $\boldsymbol{z}_{i,t}$ and $\boldsymbol{z}'_{i,t}$ are the output of the FC layers II and FC layers IV, respectively. K_1 and K_3 are the numbers of attention heads in the first GAT layer and the second GAT layer. $\boldsymbol{W}^{k_1} \in \mathbb{R}^{2\times V_1}$ and $\boldsymbol{W}^{k_3} \in \mathbb{R}^{(V_1+2)\times V_2}$ are the GAT model parameters used to transform the input features into higher-level features. $\boldsymbol{W}^{k_2} \in \mathbb{R}^{(V_1+2)\times V_1}$ and $\boldsymbol{W}^{k_4} \in \mathbb{R}^{(V_2+2)\times V_2}$ are the two FC layers. Note that $\boldsymbol{z}_{i,t}$ and $\boldsymbol{z}'_{i,t}$ updated by device i will be shared with the devices in $\mathcal{L}(i,k)$. We ignore the overhead of sharing $\boldsymbol{z}_{i,t}$ and $\boldsymbol{z}'_{i,t}$ with neighbors since it is smaller compared to the overhead of sharing local CFL models. $\alpha_{i,j}^{k_1}$ and $\alpha_{i,j}^{k_3}$ are the self-attention coefficients of the first and second GAT layers, and $\psi(\cdot)$ is the leaky rectified linear unit function. To define the attention coefficients, $\alpha_{i,j}^{k_1}$, we need to define FC function [28] that can compute the correlation between two neighbor devices as follows:

$$s_{i,j} = \psi\left(\boldsymbol{W}^{\alpha} \times \left(\boldsymbol{W}^{k_1}\boldsymbol{\phi}_i^t, \boldsymbol{W}^{k_1}\boldsymbol{\phi}_j^t\right)\right), \tag{20}$$

where $\boldsymbol{W}^{\alpha}$ is the trainable parameter. Subsequently, we normalize the correlation across each device in $\mathcal{L}^1(i,k)$ as the attention coefficient using the softmax function to calculate the self-attention coefficient of device i and device j [30]:

$$\alpha_{i,j}^{k_1} = \frac{e^{s_{i,j}}}{\sum_{l=1}^{|\mathcal{L}^1(i,k)|} e^{s_{i,l}}}, \forall j \in \mathcal{L}^1(i,k). \tag{21}$$

3) Output: The output of the proposed GNN in each device i is $\boldsymbol{\mu}_i^t = [\mu_{i,1}^t, \ldots, \mu_{i,|\mathcal{L}^1(i,k)|}^t]$ that represents the probability of device i connecting to devices in $\mathcal{L}^1(i,k)$, which can be expressed as

$$\mu_{i,j}^t = \frac{e^{\boldsymbol{z}'_{i,t}\times\boldsymbol{z}'^{\top}_{j,t}}}{\sum_{l=1}^{|\mathcal{L}^1(i,k)|} e^{\boldsymbol{z}'_{i,t}\times\boldsymbol{z}'^{\top}_{j,t}}}, \forall j \in \mathcal{L}^1(i,k). \tag{22}$$

B. Training the Proposed GNN-Based Model

With the components clearly defined, we now present the comprehensive training process for the proposed GNNs based on supervised learning. Here, we employ the binary cross-entropy (BCE) loss function to minimize the gap between the predicted connections probability $\mu_{i,j}^t \in (0,1)$ and the optimal connections $\mu_{i,j}^{t*} \in \{0,1\}$ which satisfies $\lambda < 1$ with minimum energy consumption found by exhaustive search, with $\mu_{i,j}^{t*} = 1$ indicating that device i connects to device j and $\mu_{i,j}^{t*} = 0$, otherwise. The loss function is expressed as

$$\mathcal{J}(\boldsymbol{R}^{(i,k)}, \boldsymbol{\phi}_i^t, \boldsymbol{W}^{K_1}, \boldsymbol{W}^{K_2}, \boldsymbol{W}^{K_3}, \boldsymbol{W}^{K_4}, \boldsymbol{W}^{\alpha})$$
$$= \sum_{i=1}^{M}\sum_{j=1}^{|\mathcal{L}^1(i,k)|} -\mu_{i,j}^{t*}\log\delta(\mu_{i,j}^t) - (1-\mu_{i,j}^{t*})\log(1-\delta(\mu_{i,j}^t)), \tag{23}$$

where $\delta(\cdot)$ is the sigmoid function. To minimize the training loss in (23), we optimize parameters by employing a mini-batch SGD alongside the BP algorithm.

C. Establish Connection With GHS Algorithm

Given the estimated connection probabilities $\boldsymbol{\mu}_i^t$, we employ a modified GHS algorithm based on [38], [39] that enables each device to determine device connections independently. The use of the GHS algorithm to determine device connections via the estimated $\boldsymbol{\mu}_i^t$ of each device i is summarized as follows:

1) Each device i obtains $\boldsymbol{\mu}_i^t$ using (22) and initialize an empty vector $\boldsymbol{u}_t$.
2) Each device i send a connection request to its neighbor j with the highest connection probability $\mu_{i,j}^t \in \boldsymbol{\mu}_i^t$, and receives the request from its neighbors.
3) Each device builds a connection with the neighbors that has the highest connection probability request. Then, the connection probabilities among the devices connected by the newly established connections are set to zeros.

Steps (2)–(4) are repeated until each device i is connected with device in $\mathcal{L}^1(i,k)$ via direct or indirect links. The proposed GHS algorithms ensure local connectivity with strict algorithmic overhead.

D. Optimizing Transmit Power Vector With Fixed FL Model Transmission Matrix

Given $\boldsymbol{u}_t$, the transmit power optimization problem for device i can be rewritten as

$$\min_{\boldsymbol{p}_i^t}\sum_{j=1}^{M} E_{i,j,t}\left(\boldsymbol{u}_i^t, \boldsymbol{\phi}^t, p_{i,j}^t\right), \tag{24}$$

$$\text{s.t. } l_{i,j,t}(\boldsymbol{\phi}^t, B, p_{i,j}^t) \leqslant \Gamma, \forall j \in \mathcal{L}^1(i,k). \tag{24a}$$

Based on (24), the optimal transmit power of each device i is given by the following lemma.

Lemma 1: The optimal transmit power $p_{i,j,t}$ of device i for transmitting its FL model to device j is

$$p_{i,j,t}^* = \frac{u_{i,j,t}\sigma_N^2}{h_{i,j,t}(\boldsymbol{\phi}_t)}\left(2^{\frac{A\|\boldsymbol{u}_i\|}{B\Gamma}} - 1\right). \tag{25}$$

Proof: See Appendix C. ■

In summary, our proposed GNN-based approach estimates the connection probabilities of each device based on their neighbors' location information. Then, a GHS algorithm is used to determine $\boldsymbol{U}$. Finally, we leverage the convex optimization method to calculate the transmit power of each device as shown in Algorithm 1 and Fig. 3.

E. Complexity Analysis

Here, we examine the computational complexity of the proposed scheme to solve the problem (10). This complexity is determined by two parts:

1) Complexity of Training GNN-Based Scheme: The complexity of GNN training arises from the convergence of multiple attention mechanisms. The computational complexity depends

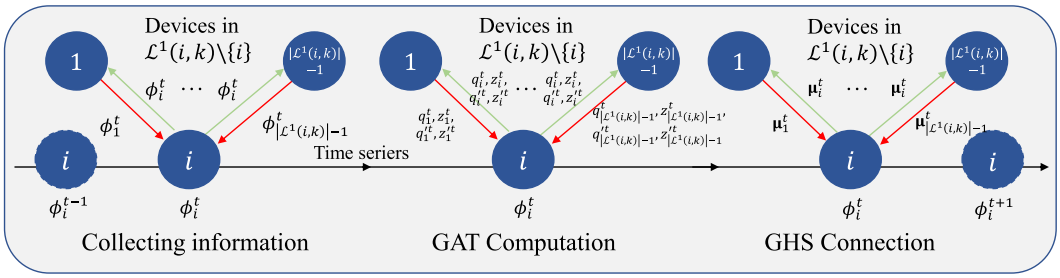


Fig. 3. Illustration of our proposed GNN-based algorithm.

Algorithm 1: GNN-Based Algorithm for Connection Optimization.

1: **Initialize** the FL model transmission matrix $\boldsymbol{U}$ as empty.
2: Devices obtain location information based on GPS data;
3: Each device i broadcasts its location information to devices in $\mathcal{L}(i,k)$;
4: Each device i establishes $\boldsymbol{R}^{(i,k)}$ and $\boldsymbol{\Phi}_i$;
5: **for** device $i \in \mathcal{M}$ **do**
6: Aggregate the neighboring devices' information based on (16) and (18);
7: Calculate the output probability distribution $\boldsymbol{\mu}_i^t$ based on (22);
8: Each device i transmits $\boldsymbol{\mu}_i^t$ to its neighboring devices;
9: **end for**
10: Each device i determines $\boldsymbol{u}_i$ based on GHS algorithm;
Output: The FL model transmission matrix $\boldsymbol{U}^*$.

on the size of neighbors k and, the dimensions of intermediate V_1, output V_2, and the number of devices M. Hence, the computational complexity is $\mathcal{O}(kM(H_1V_1^2 + H_2V_2^2))$ [30], where H_1 and H_2 is the number of the attention heads of the first layer and the second layer.

2) Complexity for Inference: The computational complexity for the GNN algorithm is $\mathcal{O}(k(H_1V_1^2 + H_2V_2^2))$. For a graph comprising M devices with Top-k neighbors, the computational complexity of GHS of each device is $\mathcal{O}(k)$. Hence, the complexity of the proposed method is independent of the number of devices participating in the CFL making it suitable for deployment in large scale systems.

We can see that the complexity of the proposed method is related to the selection of k and remains invariant as M scales up.

V. Simulation Results

For our simulations, we consider a CFL framework with a circular area having a radius r = 1000r = 1000 m and uniformly distributed devices across the region. A comprehensive list of additional parameters utilized can be found in Table II. For comparison, we utilize four baselines:

TABLE II
Simulation Parameters

Parameters	Values	Parameters	Values
B	1 MHz	f	3.3 GHz
σ_N^2	-174 dB	Γ	10 ms
V_1	5	V_2	10

- A centralized approach that enables each device to collect location information from all devices to optimize $\boldsymbol{U}$ using the greedy algorithm (labeled "Greedy search" in plots).
- A centralized approach that enables each device to collect location information from all devices to optimize $\boldsymbol{U}$ using the exhaustive algorithm based on Theorem 1 and Theorem 2 (labeled "Exhaustive search" in plots).
- A distributed approach that enables each device to collect location information from its nearest neighbors (labeled "Top1 connection" in plots).
- A distributed approach that enables each device to collect location information from its nearest and subsequent nearest neighbors (labeled "Top2 connection" in plots).

A. Dataset Introduction

1) MNIST Dataset: The first dataset is handwritten digit identification on the MNIST dataset [40]. MNIST data set consists of 70000 handwritten images in 10 classes. Each image consists of 28×28 pixels with only one channel. The dataset is divided into $N = 60000$ pieces of training data and 10000 pieces of test data. The CFL model algorithm is employed for the recognition of handwritten digits which is composed of three fully-connected neural network (FNN) layers. The total number of model parameters is $217728\,(= 28 \times 28 \times 256 + 256 \times 64 + 64 \times 10)$.

2) CIFAR10 Dataset: The second dataset is image classification on the CIFAR10 dataset [41]. The CIFAR10 dataset consists of 60000 pieces of $3 \times 32 \times 32$ color images in 10 classes. There are $N = 50000$ training images and 10000 test images. The CFL model algorithm used for image classification is Resnet18 [42]. For MNIST datasets, we consider the non-i.i.d. data distributions where each device is allocated 300 samples from only 3 of 10 labels, where the labels are randomly selected. For CIFAR10 datasets, we consider the i.i.d data distributions across devices, each device is allocated 300 samples from all labels. For both

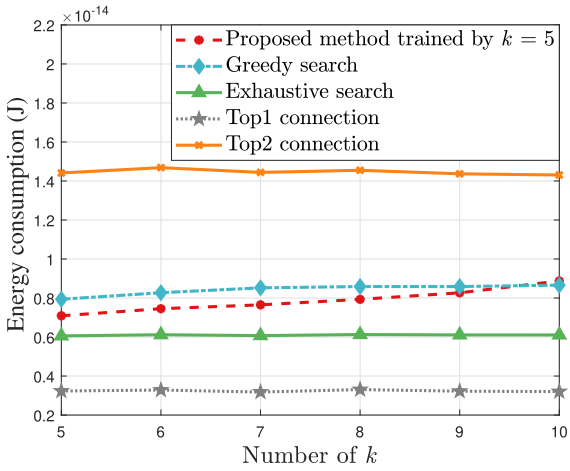


Fig. 4. Energy consumption for model transmission vs. the number of k in building $\boldsymbol{R}^{(i,k)}$.

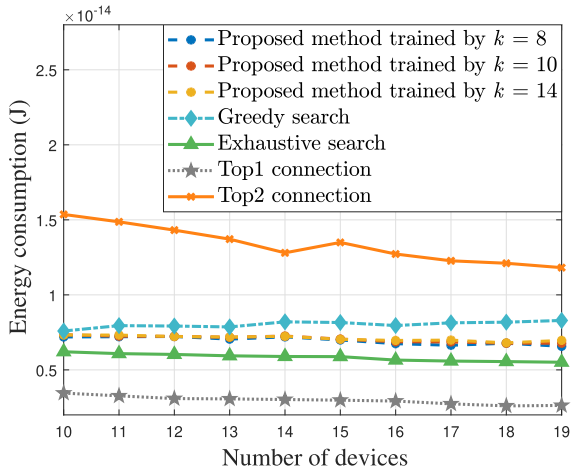


Fig. 5. Energy consumption for model transmission vs. the number of devices.

MNIST and CIFAR10, each device divides the training samples, i.e. $N_i = N_j, \forall i, j \in \mathcal{M}, N = \sum_{i=1}^{M} N_i$.

3) DACT Dataset: The third dataset we use is the DACT dataset [43] which contains 50 trajectories of vehicles. In DACT, the moving patterns of each device consist of turn, speed-up, slow-down, and etc.

B. Transmission Energy Consumption Comparisons

Fig. 4 shows how the energy consumption for model transmission changes as k varies in a network with 14 devices. Fig. 4 also shows that the proposed method can reduce the energy consumption used for CFL training by up to 15% compared to the greedy search algorithm. This is because the proposed method can select an optimal subset of devices for FL parameter exchange. From this figure, we can also see that, the Top1 connection has the lowest energy consumption than other baselines, while it has poor CFL performance in the following comparisons.

To verify the generalization performance of GNN with different sizes of networks, we show how the energy consumption used for CFL training changes as the number of devices varies in Fig. 5. In this figure, we train the GNN using different hyperparameters k over 8, 10, and 14, respectively. From Fig. 5, we can see that there is a small gap between the proposed method and the exhaustive search. This is due to the fact the proposed GNN algorithm can jointly consider the first-hop and second-hop devices' location information and find an energy-efficient connection scheme. Fig. 5 also shows that the average energy consumption for model transmission decreases slightly as the number of devices increases. This is because as the number of devices increases, the average distance among devices decreases. Fig. 5 also shows that the gap between the proposed method and the exhaustive search algorithm does not change as the number of devices varies. This characteristic reduces the computational burden and enhances the scalability of the algorithm. By leveraging the properties of permutation equivalence and invariance inherent in graph theory, the proposed method maintains a parameter dimension that is decoupled from the count of peripheral devices. This design approach not only streamlines the computational complexity but also significantly amplifies the scalability of the algorithm across various network environments.

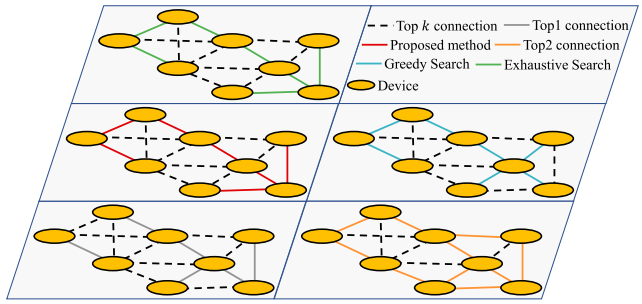


Fig. 6. Visualization of device connection as the algorithm varies.

Fig. 6 is a visualization of device connection determined by considered algorithms with $M = 8$ and $k = 3$. From Fig. 6, we can see that the proposed method has the same connection as the exhaustive search algorithm. This is due to the fact that the proposed method evaluates the potential connections for each device meticulously to establish the optimal connection with minimal transmission energy consumption. From Fig. 6, we see that the greedy algorithm enables each device to establish connections with its nearest devices, which leads to an excessive of connections on a single node, thus increasing CFL training energy consumption.

C. GNN Model Structure Comparisons

Fig. 7 shows how different network structures affect the prediction accuracy of $\boldsymbol{u}_t^i$ compared with $\boldsymbol{\mu}_i^{t*}$ as the size of networks varies when $k = 3$. The proposed method is compared with a GCN network, the proposed GAT layer without concatenation operation (i.e., (,) operation in (17), the multilayer perception, and the convolution neural network (CNN), respectively. The input of a multilayer perception network and CNN are the location of device i and its nearest neighbors' location, and

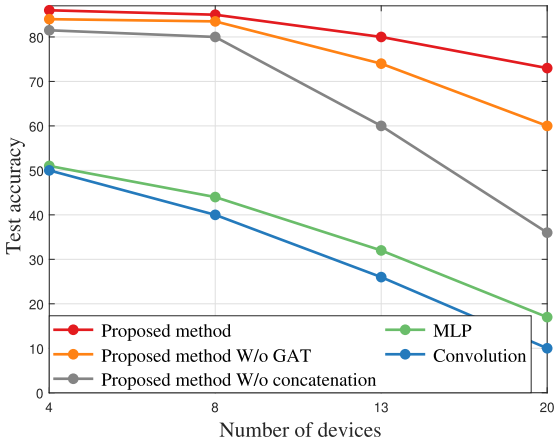


Fig. 7. Test accuracy vs. the number of devices.

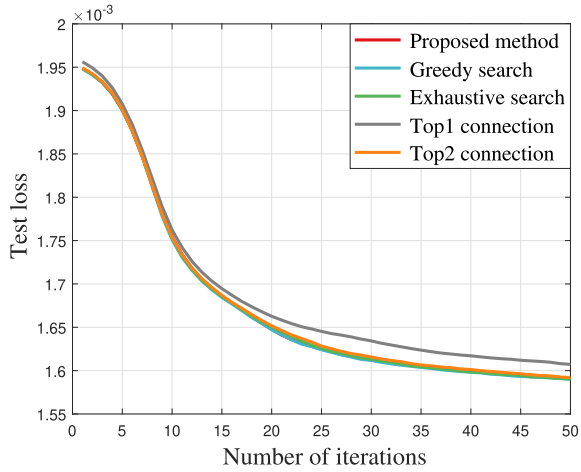


Fig. 8. Test loss vs. the number of iterations.

the output is the probability of connection $\boldsymbol{\mu}_i^t$. In this figure, we can see that the proposed method improves the connection prediction accuracy by up to 13% compared to the proposed method without GAT. The 13% improvement arises from the proposed method with only a GNN layer relying solely on location information, thus ignoring the device's neighboring information. This figure also shows that the proposed method improves the connection prediction accuracy by up to 40% compared to the model without the concatenation operation. The 40% gain stems from the fact that incorporating concatenate connections into the GNN structure ensures the efficacy and the stability of CFL training since its emphasis on the role of location proves advantageous even in scenarios with extensive neighboring devices. From this figure, we can also see that the GNN based method can increase the prediction accuracy compared with the MLP and convolution based method. This is due to the fact that the GNN based method can analyze the association relationship between devices, while the MLP and convolution based methods cannot extract this information effectively, thus reducing the prediction accuracy.

D. CFL Training Accuracy Comparisons

Fig. 8 shows how the average CFL test loss changes as the number of iterations varies. This figure is simulated using the MNIST dataset. Fig. 8 also shows that the proposed method can reduce training loss by up to 2% compared to the Top1 connection algorithm. This also demonstrates the proposed method can optimize $\boldsymbol{U}$ by analyzing the location and the device connection in the network. Although Fig. 8 shows that the Top2 connection has a similar test loss compared with the proposed method, it costs almost twice the energy consumption of the proposed method, as shown in Figs. 4 and 5.

Fig. 9 shows how the average test accuracy changes as the number of iterations varies with 12 devices and $k = 6$ on the CIFAR10 dataset. Fig. 9 shows that the proposed method achieves a similar to the greedy search and exhaustive search algorithm. This stems from the fact that the proposed method integration the neighbor's location and information exchange to

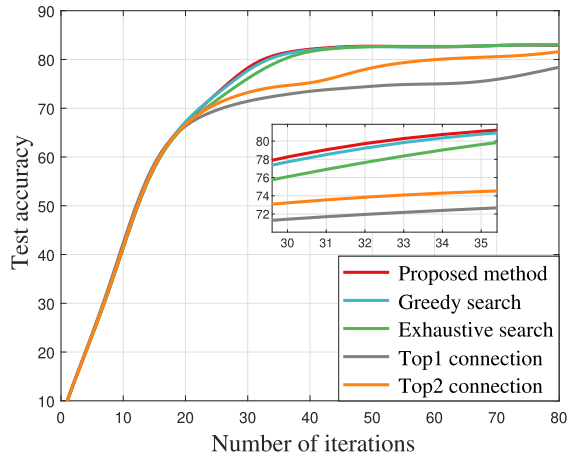


Fig. 9. Test accuracy vs. the number of iterations.

find the optimal connection. Fig. 9 also shows that the proposed method can improve the test accuracy by up to 3% compared to the Top2 connection algorithm which has more connections. The 3% gain stems from the fact that the proposed method maintained the connectivity of all devices thus improving the learning performance proved by Theorem 1 and Theorem 2.

E. Real Vehicle-Based Datasets for Proposed CFL Framework

In the previous simulation results, the devices move randomly. Here, we carried out the following experiments using the device's location collected at regular intervals in DACT dataset, as shown in Fig. 10. We randomly select the trajectories of 30 devices and normalize them within a circular region of radius $r = 1000$ m.

Fig. 11 illustrates the cumulative distribution function (CDF) curves of the energy consumption. From Fig. 11, we can see that the proposed method consistently outperforms the greedy method in most scenarios in terms of energy consumption, as the energy consumption of the proposed method is on the left of the greedy algorithm's consumption. This is due to the fact

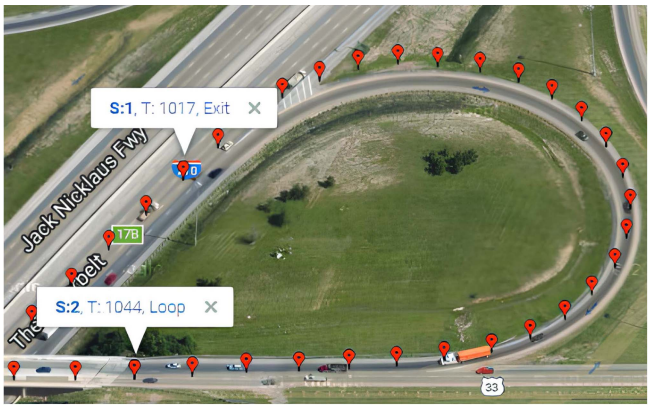


Fig. 10. Example of DACT dataset [43].

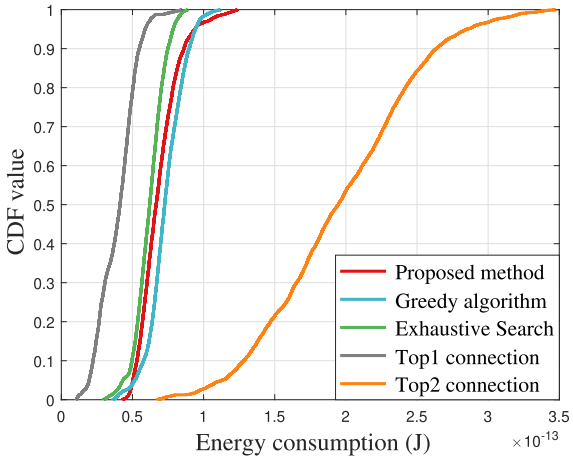


Fig. 11. Cumulative Distribution Function (CDF) of communication energy consumption based on the DACT dataset.

that the proposed method takes into account both the location information and the device connection information, allowing it to find the optimal connections. Fig. 11 also indicates the energy consumption of the Top1 connection algorithm is the lowest, while the energy consumption of the Top2 connection algorithm is the largest and has higher volatility among all baselines. This is due to the fact that the Top1 connection algorithm establishes fewer connections while the Top2 connection algorithm establishes more connections compared to the other algorithms.

Fig. 12 shows how the average test accuracy on the MNIST dataset changes with the number of iterations varies. From Fig. 12, we observe that the proposed method improves 5% test accuracy compared to the Top1 connection algorithm. The 5% gain stems from the fact that the proposed method utilizes GAT and GHS methods to maintain connectivity between devices, which improves the CFL training performance than other algorithms.

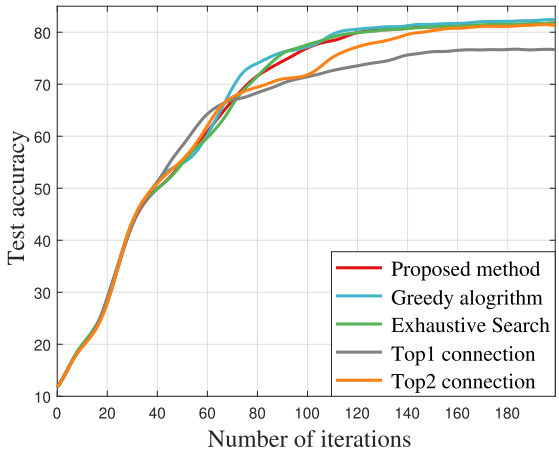


Fig. 12. Test accuracy vs. the number of iterations based on the DACT dataset.

VI. Conclusion

In this paper, we developed a novel CFL framework that enables devices to collaboratively train an ML model via exchanging their local model with their neighboring device without reliance on a centralized parameter server. We formulated an optimization problem whose goal is to minimize CFL energy consumption for model transmission while meeting the delay and CFL training loss constraints. To solve this problem, we analyzed how the dynamic device connection affects the CFL training performance. Then, we proposed a two-stage solution. At the first stage, we proposed a GNN-based algorithm that allows each device to individually determine its FL model transmission matrix based on the location and connection information of its neighbors. At the second stage, we have derived a closed-form equation of the optimal transmit power based on the optimal FL model transmission matrix. The simulation result demonstrates that compared to the algorithm in which each device would directly connect to its first and second nearest neighbors, the proposed method significantly reduces CFL energy consumption for model transmission.

Future work can extend the designed CFL algorithm to a dynamic wireless network where edge devices may not be able to participate in entire CFL training process, i.e., such that the number of participating devices per iteration is dynamic. In particular, we will first estimate devices' disconnection risks and then explore the device connection scheme design and adaptive parameter-efficiency transmission technique to accelerate CFL convergence under unstable connectivity.

References

[1] M. Chen, Z. Yang, W. Saad, C. Yin, H. V. Poor, and S. Cui, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Jan. 2021.

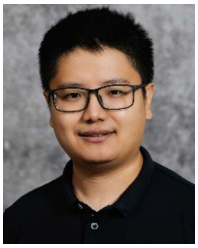
[2] Z. Chen, W. Yi, Y. Liu, and A. Nallanathan, "Robust federated learning for unreliable and resource-limited wireless networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 9793–9809, Aug. 2024.

[3] S. Wang, H. Guo, X. Zhu, C. Yin, and V. K. N. Lau, "Communication-efficient distributed Bayesian federated learning over arbitrary graphs," *IEEE Trans. Signal Process.*, vol. 73, pp. 1351–1366, 2025.

[4] M. Chen et al., "Distributed learning in wireless networks: Recent progress and future challenges," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3579–3605, Dec. 2021.
[5] B. Aurélien, K. Anne-Marie, and L. Erick, "D-cliques: Compensating for data heterogeneity with topology in decentralized federated learning," in *Proc. 2022 Int. Symp. Reliable Distrib. Syst.*, 2021, pp. 1–11.
[6] M. Chen, H. V. Poor, W. Saad, and S. Cui, "Wireless communications for collaborative federated learning," *IEEE Commun. Mag.*, vol. 58, no. 12, pp. 48–54, Dec. 2020.
[7] L. Wang, Y. Xu, H. Xu, M. Chen, and L. Huang, "Accelerating decentralized federated learning in heterogeneous edge computing," *IEEE Trans. Mobile Comput.*, vol. 22, no. 9, pp. 5001–5016, Sep. 2023.
[8] S. Wang, M. Chen, C. G. Brinton, C. Yin, W. Saad, and S. Cui, "Performance optimization for variable bitwidth federated learning in wireless networks," *IEEE Trans. Wireless Commun.*, vol. 23, no. 3, pp. 2340–2356, Mar. 2024.
[9] Y.-T. Chow, W. Shi, T. Wu, and W. Yin, "Expander graph and communication-efficient decentralized optimization," in *Proc. Asilomar Conf. Signals Syst. Comput.*, Pacific Grove, CA, USA, 2016, pp. 1715–1720.
[10] A. Nedić, A. Olshevsky, and M. G. Rabbat, "Network topology and communication-computation tradeoffs in decentralized optimization," in *Proc. IEEE*, vol. 106, no. 5, pp. 953–976, May 2018.
[11] G. Qu and N. Li, "Accelerated distributed nesterov gradient descent," *IEEE Trans. Autom. Control*, vol. 65, no. 6, pp. 2566–2581, Jun. 2020.
[12] Z. Zhang et al., "Robust semisupervised federated learning for images automatic recognition in Internet of Drones," *IEEE Internet Things J.*, vol. 10, no. 7, pp. 5733–5746, Apr. 2023.
[13] L. Barbieri, O. Simeone, and M. Nicoli, "Channel-driven decentralized bayesian federated learning for trustworthy decision making in D2D networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, 2023, pp. 1–5.
[14] Z. Yang, X. Zhang, D. Wu, R. Wang, P. Zhang, and Y. Wu, "Efficient asynchronous federated learning research in the Internet of Vehicles," *IEEE Internet Things J.*, vol. 10, no. 9, pp. 7737–7748, May 2023.
[15] E. T. Martínez Beltrán et al., "Decentralized federated learning: Fundamentals, state of the art, frameworks, trends, and challenges," *IEEE Commun. Surv. Tut.*, vol. 25, no. 4, pp. 2983–3013, Fourth Quarter, 2023.
[16] X. Yu, R. Li, C. Liang, and Z. Zhao, "Communication-efficient soft actor-critic policy collaboration via regulated segment mixture in Internet of Vehicles," Dec. 2023. Available Online: https://arxiv.org/abs/2312.10123
[17] S. Guo, B. Xiang, L. Chen, H. Yang, and D. Yu, "Multi-level federated learning mechanism with reinforcement learning optimizing in smart city," in *Proc. Int. Conf. Artif. Intell. Secur.*, 2022, pp. 441–454.
[18] W. Yang, W. Xiang, Y. Yang, and P. Cheng, "Optimizing federated learning with deep reinforcement learning for digital twin empowered industrial IoT," *IEEE Trans. Ind. Informat.*, vol. 19, no. 2, pp. 1884–1893, Feb. 2023.
[19] 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.*, Toronto, ON, Canada, 2020, pp. 1698–1707.
[20] Y. Weiting et al., "Optimizing federated learning in distributed industrial IoT: A multi-agent approach," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 12, pp. 3688–3703, Dec. 2021.
[21] W. L. Hamilton, R. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA, 2017, pp. 1025–1035.
[22] Y. Bengio, A. Lodi, and A. Prouvost, "Machine learning for combinatorial optimization: A methodological tour d'horizon," *Eur. J. Oper. Res.*, vol. 290, no. 2, pp. 405–421, Dec. 2021.
[23] M. Schlichtkrull, T. N. Kipf, P. Bloem, I. Titov, R. V. D. Berg, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proc. 15th Int. Conf. Semantic Web*, Springer, 2018, pp. 593–607.
[24] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 1, pp. 4–24, Jan. 2021.
[25] Y. Ji, K. Sun, P. Seong, and H. Choong, "GDFed: Dynamic federated learning for heterogenous device using graph neural network," in *Proc. Int. Conf. Inf. Netw.*, Bangkok, Thailand, 2023, pp. 683–685.
[26] B. Li, A. Swami, and S. Segarra, "Power allocation for wireless federated learning using graph neural networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Singapore, 2022, pp. 5243–5247.
[27] V. C. Gogineni, S. Werner, Y.-F. Huang, and A. Kuh, "Decentralized graph federated multitask learning for streaming data," in *Proc. 56th Annu. Conf. Inf. Sci. Syst.*, Princeton, NJ, USA, 2022, pp. 101–106.
[28] V. Petar, C. Guillem, C. Arantxa, R. Adriana, L. Pietro, and B. Yoshua, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, Vancouver, BC, Canada, 2018, pp. 1–12.
[29] C. Xu and W. Song, "Intelligent task allocation for mobile crowdsensing with graph attention network and deep reinforcement learning," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 2, pp. 1032–1048, Mar./Apr. 2023.
[30] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
[31] R. B. Bapat and T. E. S. Raghavan, Doubly stochastic matrices," *Encyclopedia of Mathematics and its Applications*. Cambridge, U.K.: Cambridge Univ. Press, 1997.
[32] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: Possibilities and fundamental limitations based on available wireless network measurements," *IEEE Signal Process. Mag.*, vol. 22, no. 4, pp. 41–53, Jul. 2005.
[33] W. Liu, L. Chen, and W. Zhang, "Decentralized federated learning: Balancing communication and computing costs," *IEEE Trans. Signal Inf. Process. Netw.*, vol. 8, pp. 131–143, 2022.
[34] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, Jan. 2019.
[35] P. Li et al., "Snowball: Energy efficient and accurate federated learning with coarse-to-fine compression over heterogeneous wireless edge devices," *IEEE Trans. Wireless Commun.*, vol. 22, no. 10, pp. 6778–6792, Oct. 2023.
[36] Y. Shen, J. Zhang, S. H. Song, and K. B. Letaief, "Graph neural networks for wireless communications: From theory to practice," *IEEE Trans. Wireless Commun.*, vol. 22, no. 5, pp. 3554–3569, May 2023.
[37] M. Lee, G. Yu, and G. Y. Li, "Graph embedding-based wireless link scheduling with few training samples," *IEEE Trans. Wireless Commun.*, vol. 20, no. 4, pp. 2282–2294, Apr. 2021.
[38] R. G. Gallager, P. A. Humblet, and P. M. Spira, "A distributed algorithm for minimum-weight spanning trees," *ACM Trans. Prog.. Lang. Syst.*, vol. 5, no. 1, pp. 66–77, Jan. 1983.
[39] R. Katarzyna, "GHS algorithm on a graph with random weights," *Theor. Comput. Sci.*, vol. 828-829, pp. 19–31, May 2020.
[40] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[41] A. Krizhevsky, "Learning multiple layers of features from tiny images," Apr. 2009. [Online]. Available: https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf
[42] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Las Vegas, NV, USA, 2016, pp. 770–778.
[43] M. Sobhan, O.-T. Behrooz, and R. Ramnath, "Trajectory annotation by discovering driving patterns," in *Proc. 3rd ACM SIGSPATIAL Workshop Smart Cities Urban Analytics*, New York, NY, USA, 2017, pp. 1–4.

Nuocheng Yang (Student Member, IEEE) received the BS degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2021. He is currently working toward the PhD degree with the Information and Communication Engineering Department, Beijing University of Posts and Telecommunications. His research interests include reinforcement learning, computing power networks, and machine learning in wireless networks.

Sihua Wang (Member, IEEE) received the PhD degree from the Beijing University of Posts and Telecommunications (BUPT), in 2021. He is currently an assistant professor with the School of Information and Communication Engineering, BUPT. From 2021 to 2024, he was a postdoctoral researcher with the School of Computer Science (National Pilot Software Engineering School), BUPT. From 2022 to 2024, he held a Hong Kong Scholar Fellow position with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. His research interests include mobile edge computing, resource allocation, and machine learning in wireless networks.



Yuchen Liu (Member, IEEE) received the PhD degree from the Georgia Institute of Technology, USA. He is currently an assistant professor with the Department of Computer Science, North Carolina State University, USA. His research interests include wireless networking, digital twins, generative AI, distributed learning, mobile computing, and software simulation. He has received several best paper awards at IEEE and ACM conferences. He currently serves as associate editors for *IEEE Transactions on Green Communications and Networking*, *IEEE Transactions on Machine Learning in Communications and Networking*, and *Elsevier Computer Networks*.

Christopher G. Brinton (Senior Member, IEEE) received the MS and PhD (with honors) degrees in electrical engineering from Princeton, in 2016 and 2013, respectively. He is the Elmore associate professor of electrical and computer engineering (ECE) with Purdue University. His research interest is at the intersection of networking, communications, and machine learning, specifically in fog/edge network intelligence, distributed machine learning, and AI/ML-inspired wireless network optimization. He is a recipient of four of the US top early career awards, from the National Science Foundation (CAREER), Office of Naval Research (YIP), Defense Advanced Research Projects Agency (YFA), and Air Force Office of Scientific Research (YIP). He is also a recipient of the IEEE Communication Society William Bennett Prize Best Paper Award, the Intel Rising Star Faculty Award, the Qualcomm Faculty Award, and Purdue College of Engineering Faculty Excellence Awards in Early Career Research, Early Career Teaching, and Online Learning. He currently serves as an associate editor for *IEEE/ACM Transactions on Networking*, and previously was an associate editor for *IEEE Transactions on Wireless Communications*. Prior to joining Purdue, he was the associate director of the EDGE Lab and a lecturer of electrical engineering with Princeton University. He also co-founded Zoomi Inc., a Big Data startup company that has provided learning optimization to more than one million users worldwide and holds US Patents in machine learning for education. His book *The Power of Networks: 6 Principles That Connect our Lives* and associated Massive Open Online Courses (MOOCs) have reached more than 400,000 students.

Changchuan Yin (Senior Member, IEEE) received the PhD degree in signal and information processing from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1998. In 2004, he worked as a visiting scholar with the Faculty of Science, the University of Sydney, Sydney, NSW, Australia. From 2007 to 2008, he held a visiting position with the Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. Currently, he serves as a professor with the School of Information and Communication Engineering, BUPT. His research interests include wireless networks and statistical signal processing. He was the co-recipient of the IEEE Marconi Prize Paper Award in 2023 and the IEEE International Conference on Wireless Communications and Signal Processing Best Paper Award in 2009. He has contributed as a Symposium co-chair and Technical Program Committee (TPC) member for numerous IEEE conferences.

Mingzhe Chen (Senior Member, IEEE) is currently an assistant professor with the Department of Electrical and Computer Engineering and the Knight foundation chair in Data Science and AI with the Frost Institute of Data Science and Computing, University of Miami. His research interests include federated learning, reinforcement learning, virtual reality, unmanned aerial vehicles, and Internet of Things. He has received four IEEE Communication Society journal paper awards including the IEEE Marconi Prize Paper Award in Wireless Communications in 2023, the Young Author Best Paper Award in 2021 and 2023, and the Fred W. Ellersick Prize Award in 2022, and four conference best paper awards at ICCCN in 2023, IEEE WCNC in 2021, IEEE ICC in 2020, and IEEE GLOBECOM in 2020. He currently serves as an associate editor for *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Communications*, *IEEE Wireless Communications Letters*, *IEEE Transactions on Green Communications and Networking*, and *IEEE Transactions on Machine Learning in Communications and Networking*.