IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 20, 2020, accepted March 4, 2020, date of publication March 9, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2979323

Federated Cooperation and Augmentation for Power Allocation in Decentralized Wireless Networks

MU YAN, BOLUN CHEN, GANG FENG, (Senior Member, IEEE), AND SHUANG QIN, (Member, IEEE)

National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China, Chengdu 611731, China

Corresponding author: Gang Feng (fenggang@uestc.edu.cn)

This work was supported in part by the National Science Foundation of China under Grant 61871099 and Grant 61631005, and in part by the Research and Development Program in Key Areas of Guangdong Province under Grant 2018B010114001.

ABSTRACT Emerging mobile edge techniques and applications such as Augmented Reality (AR)/Virtual Reality (VR), Internet of Things (IoT), and vehicle networking, result in an explosive growth of power and computing resource consumptions. In the meantime, the volume of data generated at the edge networks is also increasing rapidly. Under this circumstance, building energy-efficient and privacy-protected communications is imperative for 5G and beyond wireless communication systems. The recent emerging distributed learning methods such as federated learning (FL) perform well in improving resource efficiency while protecting user privacy with low communication overhead. Specifically, FL enables edge devices to learn a shared network model by aggregating local updates while keeping all the training processes on local devices. This paper investigates distributed power allocation for edge users in decentralized wireless networks with aim to maximize energy/spectrum efficiency while preventing privacy leakage based on a FL framework. Due to the dynamics and complexity of wireless networks, we adopt an on-line Actor-Critic (AC) architecture as the local training model, and FL performs cooperation for edge users by sharing the gradients and weightages generated in the Actor network. Moreover, in order to resolve the over-fitting problem caused by data leakages in Non-independent and identically distributed (Non-i.i.d) data environment, we propose a federated augmentation mechanism with Wasserstein Generative Adversarial Networks (WGANs) algorithm for data augmentation. Federated augmentation empowers each device to replenish the data buffer using a generative model of WGANs until accomplishing an i.i.d training dataset, which significantly reduces the communication overhead in distributed learning compared to direct data sample exchange method. Numerical results reveal that the proposed federated learning based cooperation and augmentation (FL-CA) algorithm possesses a good convergence property, high robustness and achieves better accuracy of power allocation strategy than other three benchmark algorithms.

INDEX TERMS Federated learning, power allocation, wireless networks, federated cooperation, federated augmentation.

I. INTRODUCTION

Currently, there are nearly 7 billion connected Internet of Things (IoT) devices and 3 billion Smart-phones at the network edge [1], where power and computing resource consumptions from the information and communication technology sector are expected to increase significantly [2]. Moreover, over 90% of the data will be stored and processed locally [3], and most of the data is privacy-sensitive. Most mobile devices such as iPhone 11 and HUAWEI MATE 30 are equipped with advanced sensors and Central Processing Unit (CPU). Therefore, local data storing and processing can be empowered by the emerging mobile edge computing (MEC) [4], [5] and/or user devices, which offloads the burden of central controller or cloud server by pushing the computation/storage resources to the edge users. This brings intelligence closer to the edge users, which enables most tasks to be performed locally at the edge user equipments.

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

Energy/Spectrum efficiency as an important metric for 5G and the forthcoming 6G [6] systems has been extensively investigated in edge mobile networks [7], [8]. As the network architecture is gradually evolving into an intelligent autonomous network paradigm, where the telecom operators need to automate their networks in a plug-and-play manner thus to reduce the amount of manual intervention. Under this circumstance, optimizing power allocation in a distributed manner is an effective way to improve the energy/spectrum efficiency for autonomous networks. In most traditional cloud-centric approaches, the data collected by mobile devices is uploaded and processed centrally in a cloud server/central controller, which may result in unacceptable latency, communication inefficiency, and privacy leakage. To maintain the privacy of user data, it is necessary to adopt distributed training that shifts the computation to the edge network where data samples can be locally trained without sharing [9]. Moreover, distributed training is able to stimulate the self-learning and self-configuration with reduced signaling interactions [10].

In decentralized networks, the network states are usually time-varying and imponderable due to the changes of user behavior, channel status and external weather environment. Therefore, it is ineffective to use static optimization based algorithms to solve the dynamic power allocation problem in an autonomous network, due to its poor adaptability and generalization for dynamic environment [11]. This inspires us to perform resource allocation in an adaptive and intelligent way. Fortunately, recent emerging machine learning (ML) algorithms have been widely used for optimizing mobile network edges [12]–[14]. In particular, the state-of-the-art ML tools such as neural networks based deep learning, Markov decision process (MDP) based reinforcement learning and zero-sum game based Wasserstein Generative Adversarial Networks (WGANs) have been proven to possess strong data processing and decision-making capabilities in the wireless networks [15], [16]. Nevertheless, most conventional ML algorithms at mobile edge networks still require personal data to be shared with cloud servers [17], [18]. Recently, as data privacy regulations have become increasingly stringent and privacy issues are becoming severer, the concept of federated learning (FL) has been introduced and drawn great attentions. FL is a distributed machine learning algorithm that enables users to collaboratively learn a shared prediction model while keeping their datasets on their local devices [19]–[21]. In FL, edge devices locally train their data samples by using a specific ML model, and then send the model updates such as gradients, weightages rather than raw data to the cloud server for information aggregation.

In large-scale and complex distributed mobile edge networks, numerous user devices with varying quality of service (QoS) requirements are involved. This raises significant challenges of communication overhead, resource scheduling, and privacy security in the implementation of FL at scale [22]. Moreover, due to Non-i.i.d data distributions at different user devices, the datasets between users have a certain degree of similarity, resulting in data leakage [23] which causes over-fitting problem easily in the training process. This brings great challenges in applying FL to solve resource allocation problem in decentralized wireless networks.

This paper investigates distributed power allocation in decentralized wireless networks based on a generic horizontal FL framework. We propose a federated learning framework based cooperation and augmentation (FL-CA) algorithm for solving the power allocation problem in decentralized networks. In more detail, each edge device locally obtains the power allocation strategy by training a local Actor-Critic (AC) model, and then uploads the gradients and weightages generated by the Actor network to the base station for information aggregation periodically. Moreover, we propose to implement information aggregation and model updating by using Stochastic Variance Reduced Gradient (SVRG) [24] and Distributed Approximate Newton (DAN) [25]. In particular, SVRG is a stochastic method with explicit variance reduction, and DAN is used for distributed optimization. Furthermore, to tackle the over-fitting problem caused by data leakages, we employ a federated augmentation (FAu) algorithm which uses WGANs for data augmentation. FAu empowers individual devices to replenish the data buffer using a generative model of WGANs until accomplishing an i.i.d training dataset, which significantly reduces the communication overhead compared to direct data sample exchange method.

The main contributions of this work can be summarized as follows:

1) To facilitate the power allocation in decentralized wireless networks while protecting user privacy, we employ FL scheme to transfer the control and responsibility from the centralized controller to individual user devices. The distributed control stimulates user devices' abilities of self-learning and self-configuring with reduced signaling interactions.
2) We propose to incorporate the state-of-the-art WGANs in the federated augmentation with the aim of improving the sample diversity and reducing the correlation between data samples. This can further overcome the over-fitting problem caused by data leakage.
3) We adopt Actor Critic algorithm at local user devices to solve the power allocation problem. AC algorithm performs well in solving problems with continuous action space, and can provide fully on-line solutions. This proposed solution is feasible for network environment changing at (transmission time interval) TTI/ms level.

The rest of the paper is organized as follows. The system model and problem formulation are presented in Section III. In Section IV, we propose a federated learning framework and provide a federated cooperation solution to implement the distributed power allocation. Next, in Section V, we design federated augmentation mechanism and introduce an on-line power control algorithm as the local learning algorithm which can provide gradients and weightages for the federated cooperation. In Section VI, we present the numerical results as well as discussions, and finally conclude the paper in Section VII.

II. RELATED WORK

Power allocation as an effective way to improve the energy/spectrum efficiency in edge mobile networks has recently spurred extensive investigations from different perspectives with various design objectives.

Existing solutions for power allocation such as static optimizing [26]–[28] and game theory [8], [29] are widely used for deriving the global optimal power allocation strategies based on precisely modeling the network. However, in real uncertain and time-varying network environments, it is very difficult to accurately model the environments, resulting in that the optimization models can hardly be built inaccurately. On the other hand, frequently performing power allocation to maintain optimality in a dynamic environment apparently incur heavy signaling overhead and high computing cost. Fortunately, recent emerging machine learning technologies [30], [31] such as deep reinforcement learning and WGANs have been proven effective for addressing a wide range of model-free problems and have strong adaptability to dynamic and complex environments.

From the perspective of network deployment, power allocation can be accomplished in either a distributed or a centralized manner. The authors of [30], [32], [33] aim to optimize the network performance with the assistance of a central controller. The central controller is deployed to collect global information (i.e., network states and the strategies of all user devices) and help make decisions towards the direction of improving the overall network performance. Although centralized optimizing/learning can theoretically achieve optimal system performance, it is usually inapplicable to large-scale wireless networks due to its high computational complexity and signaling overhead. The authors of [34]–[36] focus on decentralized HetNets where most of the decisions are made locally without information sharing. Distributed learning which offloads the computation from the central controller to individual user devices can effectively decrease the communication overhead and computation complexity, and is thus more efficient for solving large-scale network problems. However, due to the lack of information interactions, distributed decision-making is hard to achieve the global optimal solution without knowing other users' strategies.

On the other hand, most existing work on power allocation by using machine learning algorithms, such as reinforcement learning, has not considered computation cost, and training data privacy. In [30] and [31], we also exploit reinforcement learning algorithms for solving online access control and resource scheduling problems, where we did not consider training data constraints.

The concept of FL was first proposed in [37] which advocates an alternative that leaves the training data distributed on the local user devices, and learns a shared model by aggregating local updates (i.e., gradients and weightages). FL has been widely used in privacy protecting and information sharing across different devices [13], [19], [20], which can overcome the weakness of traditional distributed learning algorithms. However, most of the FL applications are designed under i.i.d data environment. Training i.i.d data in the neural network of FL causes data leakage, which easily leads to over-fitting problem.

Fortunately, the recent emerging machine learning algorithms such as Wasserstein generative adversarial networks (WGANs) [38] provide an effective tool for data augmentation. In this paper, we propose to incorporate the WGANs in FL framework, which empowers individual devices to replenish the data buffer using a generative model until fulfilling an i.i.d training dataset, which significantly reduces the communication overhead compared to using direct data sample exchange method.

III. SYSTEM MODEL AND PROBLEM FORMULATION

This paper focuses on Orthogonal Frequency Division Multiple Access (OFDMA) based downlink cellular network consisting of one BS and a set of $\mathcal{U} = \{1, \cdots, u, \cdots, U\}$ edge user equipments (UEs) (i.e., tablets, mobile phone, etc.). Resource block (RB) or subcarrier is defined as the minimum transmission spectrum unit in OFDMA systems with identical bandwidth w, and we define $\mathcal{K} = \{1, \cdots, k, \cdots, K\}$ as the set of the subcarriers. In order to capture the network dynamics at small time granularity, we consider to implement TTI-level's decision-making process. We define $\mathcal{T} = \{1, \cdots, t, \cdots, T\}$ as the decision-time horizon, where we assume that the decision interval is unit of length called decision time intervals (DTIs).

A. NETWORK THROUGHPUT AND SERVICE CONSTRAINTS

Define $p_{u,k}^{(t)}$ as the transmit power allocated to UE u on the kth subcarrier at time t. Let $g_{u,k}^{(t)}$ be the channel gain between the BS and UE u on the kth subcarrier. Let N_0 be the spectral density of additive white Gaussian noise (AWGN). The Signal to Interference plus Noise Ratio (SINR) which measures the signal quality is defined as the ratio of the received sum power of the desired signal over the sum power of the interfering signals and the background noise. Therefore, the SINR of UE u on the kth subcarrier at time t is given by

$$\gamma_{u,k}^{(t)} = \frac{p_{u,k}^{(t)} g_{u,k}^{(t)}}{\sum_{i \in \mathcal{U} \setminus \{u\}} p_{i,k}^{(t)} g_{i,k}^{(t)} + wN_0} \tag{1}$$

Let $c_u^{(t)}$ be the transmission rate of UE u at time t. We use a binary variable $a_{u,k} \in \{0, 1\}$ to denote if the kth subcarrier is allocated to UE u. Specifically, $a_{u,k} = 1$ indicates the subcarrier k is assigned to UE u, and $a_{u,k} = 0$, otherwise. Shannon capacity formula can be used to describe the transmission rate. Therefore, the transmission rate for UE u at time t is given by

$$c_u^{(t)} = \sum_{k \in \mathcal{K}} a_{u,k} w \log_2(1 + \gamma_{u,k}^{(t)}) \tag{2}$$

The total transmit power consumed by the BS is given by $\sum_{u\in\mathcal{U}}\sum_{k\in\mathcal{K}} p_{u,k}^{(t)}$, and the power constraint for the BS at time t is represented as

$$\sum_{u\in\mathcal{U}}\sum_{k\in\mathcal{K}} p_{u,k}^{(t)} \leq \hat{p}, \tag{3}$$

where $\hat{p}$ is the maximum transmit power which can be managed by each SBS.

Assume each UE selects subcarriers with the best SINR by sensing the surrounding network environment. Then, we let $\check{\gamma}_u^{(t)}$ be the SINR threshold of user u at time t, and the SINR constraint for a given user u on the kth subcarrier is represented as

$$\gamma_{u,k}^{(t)} \geq \check{\gamma}_u^{(t)}. \tag{4}$$

Moreover, we also take user privacy as a constraint in this decentralized network. We roughly define the set of user's privacy information as $\mathcal{I}_u$. Then the constraint on privacy protection is defined as

$$\mathcal{I}_u \cap \mathcal{I}_v = \emptyset, \forall u, v \in \mathcal{U}, u \neq v. \tag{5}$$

B. PROBLEM FORMULATION

In this decentralized wireless network, we are interested in minimizing the sum of long-term transmit power for each user device in a distributed manner. Note that we consider a fully decentralized power allocation optimization at mobile edge. Therefore, there is no central controller deployed at BS and each user makes power allocation decision locally by observing surrounding environment without information sharing and cooperation with other user devices. Note that a power allocation decision at a specific device inexplicitly affects the SINR for other devices, and thus their decisions. Obviously, minimizing the transmit power requires accurate power allocation strategy to mitigate the interferences between user devices, which can significantly decrease energy consumption and improve spectrum efficiency in wireless networks. Specifically, the long-term power allocation problem for UE u ($u \in \mathcal{U}$) at time slot t is formulated as

$$\min_{\mathcal{P}_u} \sum_{t\in\mathcal{T}} P_u^{(t)} \tag{6}$$

$$\text{s.t. } \langle (3), (4), (5) \rangle, \tag{7}$$

where $\mathcal{P}_u = \{P_u^{(1)}, \cdots, P_u^{(t)}, \cdots, P_u^{(T)}\}$ is the sequential power allocation strategies made by user u from the current time to the termination time T. Problem (6) is non-convex and has been proved as NP-hard [32], [39]. Furthermore, without knowing other users' behaviors (i.e., movements and strategies), the network environment becomes uncertain and more constrained, resulting in that the performance metrics such as SINR and throughput cannot be calculated before perception. This implies that the optimal power allocation strategy is unable to be directly derived by solving Problem (6) with simple static optimization or game theory. Therefore, adopting an efficient algorithm from a distributed algorithm to solve the power allocation problem is indeed necessary.

IV. FEDERATED LEARNING SOLUTIONS: A DISTRIBUTED LEARNING PERSPECTIVE

To implement the distributed power allocation, we employ an FL framework/system (FLF) at the mobile edge. The FL framework enables cooperation and augmentation for data training of edge users, which is effective in solving the distributed power allocation with certain information sharing ability, and no user privacy leakage. Fig. 1 shows the FL based cooperation and augmentation (FL-CA) framework.

1) Cooperation: federated cooperation enables edge devices to collaboratively maintain a shared model by aggregating local updates while keeping all the training data on local device without privacy leakage.
2) Augmentation: federated augmentation is used for data augmentation which empowers each user device to replenish the data buffer using a generative model of WGANs until reaching an i.i.d training dataset.

Considering that the network is dynamic and complex, we adopt fully on-line Actor-Critic (AC) networks as the local training model, and federated cooperation for edge users is implemented by sharing the gradients and weightages generated from the local Actor networks. Moreover, under the constrained network environment, user-generated data samples are likely to become non-i.i.d across devices, which usually degrades the efficiency of learning a shared network model [40]. Therefore, federated augmentation is an effective way for data augmentation and can significantly reduce the communication overhead compared to direct data sample exchanges.

A. FEDERATED LEARNING BASED COOPERATION

To implement the federated learning based cooperation in the FL framework, we introduce the loss functions as the distributed optimization objective. Note that Loss functions such as linear regression, logistic regression and support vector machines are widely used in the federated learning models to capture the error of the neural network model [19].

Assume that each participating UE u collects a local dataset $\mathcal{D}_u = \{D_1, \cdots, D_{N_u}\}$ with batch size of N_u, and D_i is a set of input-output pairs $\{x_i, y_i\}$. Moreover, we define $X_u = [x_{u,1}, \cdots, x_{u,N_u}]$ as the input dataset of UE u, and $Y_u = [y_{i,u}, \cdots, y_{i,N_u}]$ as the output dataset. In order to capture the relationship between the input dataset X_u and the output Y_u, we introduce the loss function $f(\mathbf{w}, D_i)$ which is parameterized by vector $\mathbf{w}$ with input training dataset $D_i \in \mathcal{D}_u$. Then, the local loss function $F_u(\mathbf{w})$ approximated by $\mathbf{w}$ is given by

$$F_u(\mathbf{w}) = \frac{1}{N_u}\sum_{D_i\in\mathcal{D}_u} f(\mathbf{w}, D_i) \tag{8}$$

Then, we give the learning model of global loss function minimization problem as

$$\min_{\mathbf{w}\in\mathbb{R}^d} F(\mathbf{w}) = \sum_{u\in\mathcal{U}} \frac{N_u}{\sum_{u\in\mathcal{U}} N_u} F_u(\mathbf{w}) \tag{9}$$

We make an assumption that the loss function satisfies the following assumption.

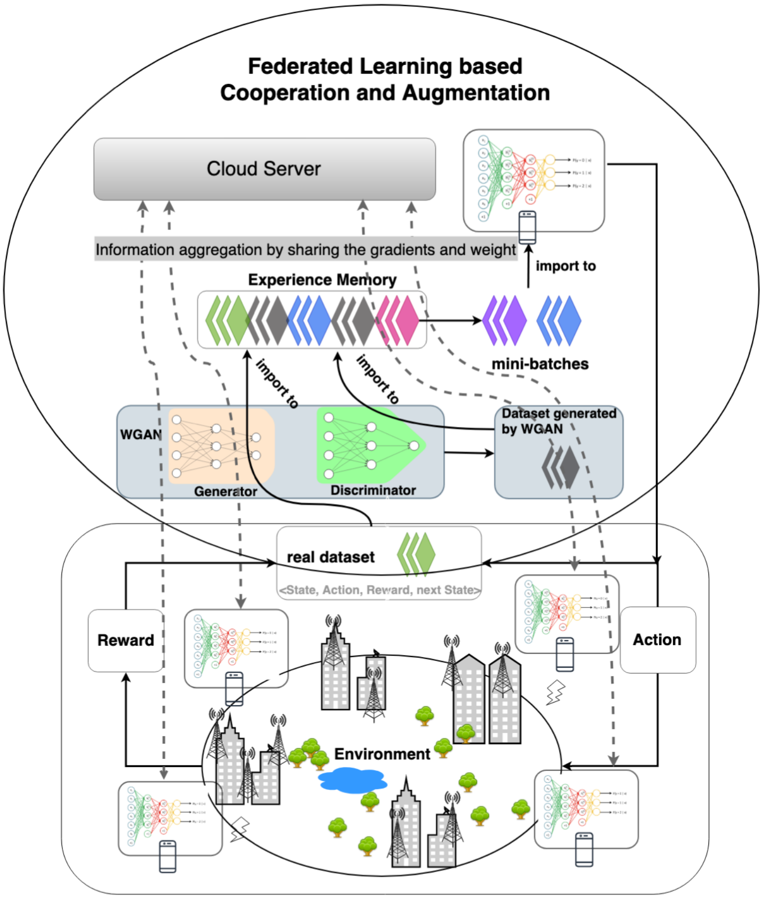


FIGURE 1. Federated learning framework based cooperation and augmentation.

Assumption 1: $F_i(\cdot)$ is η-strongly convex and σ-smooth, $\forall i \in \{1, \cdots, U\}$, and $\forall \mathbf{w}, \mathbf{w}' \in \mathbb{R}^d$:

1) $F_i(\mathbf{w})$ is η-strongly convex, i.e., $F_i(\mathbf{w}) \geq F_i(\mathbf{w}') + \langle \nabla F_i(\mathbf{w}'), \Delta\mathbf{w} \rangle + \frac{\eta}{2}\|\Delta\mathbf{w}\|$,
2) $F_i(\mathbf{w})$ is σ-smooth, i.e., $F_i(\mathbf{w}) \leq F_i(\mathbf{w}') + \langle \nabla F_i(\mathbf{w}'), \Delta\mathbf{w} \rangle + \frac{\sigma}{2}\|\Delta\mathbf{w}\|$,

where $\langle \mathbf{a}, \mathbf{b} \rangle$ denotes the inner product of vectors $\mathbf{a}$ and $\mathbf{b}$, $\nabla F(\cdot)$ denotes the gradient of the loss function $F(\cdot)$, and $\Delta\mathbf{w} = \mathbf{w} - \mathbf{w}'$.

This paper considers linear regression (i.e., Temporal Difference (TD)-errors) as the loss function in the local training model. Assume there is a set of input-output pairs $\{x_i, y_i\}_{i=1}^{N_i}$, and thus the loss function is represented as $F_i(\mathbf{w}) = \frac{1}{2}\|y_i - \mathbf{w}^{\mathrm{T}} \cdot x_i\|^2$. Note that the strong convexity and smoothness in Assumption 1 are proved to be satisfied for the linear regression which are widely used in [19], [21].

Theorem 1: The global loss function $F(\mathbf{w})$ is a convex combination of the local loss functions $f(\mathbf{w})$.

Proof: We substitute the local function in (8) for (9), and rephrase the objective (9) as

$$\begin{aligned} F(\mathbf{w}) &= \sum_{u\in\mathcal{U}} \frac{N_u}{\sum_{u\in\mathcal{U}} N_u} F_u(\mathbf{w}) \\ &= \sum_{u\in\mathcal{U}} \frac{N_u}{\sum_{u\in\mathcal{U}} N_u} \cdot \frac{1}{N_u} \sum_{D_i\in\mathcal{D}_u} f(\mathbf{w}, D_i) \\ &= \sum_{u\in\mathcal{U}} \frac{1}{\sum_{u\in\mathcal{U}} N_u} \sum_{D_i\in\mathcal{D}_u} f(\mathbf{w}, D_i). \end{aligned} \tag{10}$$

From (10) we can see that the global function $F(\mathbf{w})$ is the linear addition of the local loss functions $f(\mathbf{w})$, and we can draw the conclusion. □

Therefore, we consider to implement the power allocation in the FL systems which enables information sharing without privacy leakage. Then we formulate an optimization problem whose objective is to minimize the FL loss function, while factoring in the wireless network parameters. This minimization problem is optimizing transmit power allocation for each UE, which is given by

$$\min_{\mathbf{w}} F(\mathbf{w}) \tag{11}$$

$$\text{s.t.} \ \langle (3), (4), (5) \rangle . \tag{12}$$

However, $F(\mathbf{w})$ cannot be directly computed without sharing information of the local loss function $F_u(\mathbf{w})$ which carries edge user's privacy information.

B. DISTRIBUTED GRADIENT-DESCENT ALGORITHM

Distributed gradient-descent algorithm is widely used in the state-of-the-art FL systems [21], [41], [42]. According to (11), the objective is to minimize $F(\mathbf{w})$, which means to find the optimal vector $\mathbf{w}^*$ that satisfies

$$\mathbf{w}^* = \arg\min F(\mathbf{w}) \tag{13}$$

Note that without knowing the input training dataset, the vector $\mathbf{w}$ in each local FL system carriers little information. On this account, we consider to directly derive the optimal vector $\mathbf{w}^*$ by using the FL algorithm to solve (27). Specifically, by using a simple gradient descent (GD) [19], the local parameter $\mathbf{w}_u$ of UE u at time t is updated by

$$\mathbf{w}_u^{(t)} = \mathbf{w}_u^{(t-1)} - \alpha_u \nabla F_u(\mathbf{w}_u^{(t-1)}) \tag{14}$$

where α_u is the step size of UE u.

Note τ is the length of the DTI which is specifically the interval between every two global aggregations, and T is the termination steps. Then, with every τ time steps, the global parameter $\mathbf{w}$ is updated by

$$\mathbf{w}^\tau = \frac{\sum_{u \in \mathcal{U}} N_u \mathbf{w}_u^{(t)}}{\sum_{u \in \mathcal{U}} N_u} \tag{15}$$

Specifically, after each global aggregation, all of the distributed local parameters $\mathbf{w}_{u \in \mathcal{U}}$ are replaced by the global parameter $\mathbf{w}^\tau$. Moreover, we define $\mathbf{w}^T$ as the final model parameter after T iteration epochs.

Remark 1: In the federated learning architecture, to avoid UE privacy leakage

1) The local loss functions $F_u(\mathbf{w})$, $u \in \mathcal{U}$ shown in (8) cannot be used for sharing among multiple nodes.
2) The local parameters $\mathbf{w}_u$, $u \in \mathcal{U}$ and global parameter $\mathbf{w}^\tau$ can be used for sharing among multiple nodes.
3) Gradients of the local loss functions $\nabla F_u(\mathbf{w}_u^{(t)})$, $u \in \mathcal{U}$ and the global loss function $\nabla F(\mathbf{w}^\tau)$ can be used for sharing among multiple nodes.

In order to derive the final model parameter $\mathbf{w}^T$, we first introduce two important algorithms which are Stochastic Variance Reduced Gradient (SVRG) [24] and Distributed Approximate Newton (DAN) [25]. In more detail, SVRG is a stochastic method with explicit variance reduction, and DAN is used for distributed optimization. In this work, we implement a Federated SVRG (F-SVRG) algorithm incorporating the algorithms of SVRG and DAN. The main idea of F-SVRG is to avoid directly using the GD to estimate the gradient $\nabla F_u(\mathbf{w})$. If the global parameter $\mathbf{w}^\tau$ is close to the local parameter $\mathbf{w}_u^{(t)}$, the variance of the result of $\nabla F_u(\mathbf{w}^\tau) - \nabla F_u(\mathbf{w}_u^{(t)})$ should be small as well. Therefore, in the F-SVRG, the local parameters are updated by

$$\mathbf{w}_u^{(t)} = \mathbf{w}_u^{(t-1)} - \alpha_u^f(\nabla F_u(\mathbf{w}_u^{(t)}) - \nabla F_u(\mathbf{w}^\tau) + \nabla F(\mathbf{w}^\tau)), \tag{16}$$

where α_u^f is the step-size used for the gradient descent.

C. CONVERGENCE ANALYSIS

Iterations through the aforementioned steps continue until the loss function converges, thus completing the entire training process. Thus, convergence is very important for ML based solutions. Indeed, FL architecture is independent of specific machine-learning algorithms (logistic regression, DNN, etc.) and all participants will share the final model parameters. In detail, the training process contains the following four steps.

1) Edge UEs locally compute training gradients, and update the weightages of the neural networks.
2) The server (BS) performs secure aggregation without learning information about any edge UE.
3) The server sends back the aggregated results to edge UEs.
4) Edge UEs update their respective model with the feedback weightages and gradients.

For clarity, we summarize the training process of the distributed gradient descent in Algorithm 1.

Algorithm 1 Federated SVRG (F-SVRG)

```
Input: Global communication interval $\tau$; Termination iteration time steps $T$
Output:
for $t = 0, 1, \cdots, T$ do
    if $t$ mod $\tau = 0$ then
        Communication and Compute
        $\nabla F(\mathbf{w}^\tau) = \frac{1}{U}\sum_{u=1}^{U} \nabla F_u(\mathbf{w}_u^{(t)})$ and
        $\mathbf{w}^\tau = \mathbf{w}^\tau(old) + \frac{1}{U}\sum_{u \in \mathcal{U}}(\mathbf{w}_u^{(t)} - \mathbf{w}^\tau(old))$
    end if
    for $u = 1, 2, \cdots, U$ do
        Initialize $\mathbf{w}_u^{(t)} = \mathbf{w}^\tau$
        for $t_0 = 1, 2, \cdots, \tau$ do
            $\mathbf{w}_u^{(t+1)} = \mathbf{w}_u^{(t)} - \alpha_u^f(\nabla F_u(\mathbf{w}_u^{(t)}) - \nabla F_u(\mathbf{w}^\tau) + \nabla F(\mathbf{w}^\tau))$
        end for
    end for
end for
```

In the following, we provide a quantitative analysis of the convergence property of the F-SVRG algorithm and find an upper bond of the divergence between the FL derived loss function and the global optimal loss function $F(\mathbf{w}^T) - F(\mathbf{w}^*)$. We define the following convergence condition to determine if $\mathbf{w}^T$ achieves the global optimal parameter $\mathbf{w}^*$.

Definition 1: For an arbitrary small constant $\epsilon > 0$, the iteration result of the FL algorithm achieves the global optimality if it satisfies

$$|F(\mathbf{w}^T) - F(\mathbf{w}^*)| \leq \epsilon. \tag{17}$$

Theorem 2: For η-strongly convex and σ-smooth functions $F(\cdot)$, the upper bond of $\mathbb{E}[F(\mathbf{w}^T) - F(\mathbf{w}^*)]$ is given by

$$\mathbb{E}[F(\mathbf{w}^T) - F(\mathbf{w}^*)] \leq c^T[F(\mathbf{w}^0) - F(\mathbf{w}^*)], \tag{18}$$

where $c = \Theta(\frac{1}{mh}) + \Theta(h)$, m is the number of stochastic steps per epoch and h is the step size.

Proof: cf. [25] for proofs □

For appropriate choice of parameters m and h, the convergence rate (18) translates to the need of $(n + \mathcal{O}(L/\lambda))\log(1/\epsilon)$ evaluations of ∇F_i for some i to satisfy (17).

V. ON-LINE LOCAL POWER CONTROL ALGORITHM WITH FEDERATED AUGMENTATION

In this section, we seek for communication-efficient on-device machine learning approaches under non-i.i.d private data. On this account, an on-line closed-loop control algorithm based on the reinforcement learning is adopted. Prior to operating federated cooperation, we implement federated augmentation by using Wasserstein generative adversarial networks (WGANs) [38] which empowers each device to locally reproduce the data samples of all devices, so as to reduce the communication overhead and guarantee that the local solutions do not diverge from the global model.

A. FEDERATED AUGMENTATION BY WGANs

Let the expert policy distribution which represents the real dataset distribution be denoted by $\Pi^e = [\pi_1^e, \cdots, \pi_U^e]$, and the generated policy distribution imitated by the proposed W-GANs algorithm be denoted by $\Pi^g = [\pi_1^g, \cdots, \pi_U^g]$. W-GANs aim to optimize the earth-movement (EM) distance between Π^e and Π^g. In detail, the EM distance is the "cost" of the transport plan which transforms the distributions Π^g into the distribution Π^e. By considering the Kantorovich-Rubinstein duality [43], the Wasserstein estimation for a given UE u is given by

$$W(\pi_u^e, \pi_u^g) = \sup_{\|D\|_L \le 1} \mathbb{E}_{x \sim \pi_u^e}[D(x)] - \mathbb{E}_{y \sim \pi_u^g}[D(y)], \quad (19)$$

where the supremum is over all the 1-Lipschitz functions $D(\cdot)$ (*i.e.,* the gradient of $D(\cdot)$ is not bigger than 1). Moreover, we define $D_\theta(\cdot)$ and $G_\phi(\cdot)$ as a discriminator and a generator which are respectively represented by neural networks with parameters θ and ϕ. To learn the generator's distribution π_u^g over real data $x \sim \pi_u^e$, we define a generated policy on input noise variables $z \sim \pi_u^z$ which represents a mapping to generated data space as $G_\phi(z)$. Moreover, $D_\theta(G_\phi(z))$ outputs a scalar within [0, 1] which represents the probability that x comes from the real data rather than π_u^g. Then, we consider solving the problem

$$W(\pi_u^e, \pi_u^g) = \max_{\theta: \|D_\theta\|_L \le 1} \mathbb{E}_{x \sim \pi_u^e}[D_\theta(x)] - \mathbb{E}_{z \sim \pi_u^z}[D_\theta(G_\phi(z))]. \quad (20)$$

In detail, parameters θ and ϕ are respectively updated by m-batched gradient descend. The detailed updating process of the two parameters are given by

$$\nabla_\theta W(\pi_u^e, \pi_u^g) = \nabla_\theta[\frac{1}{m}\sum_{i=1}^{m}[D_\theta(x^{(i)}) - D_\theta(G_\phi(z^{(i)}))]], \quad (21)$$

and

$$\nabla_\phi W(\pi_u^e, \pi_u^g) = -\nabla_\phi[\frac{1}{m}\sum_{i=1}^{m} D_\theta(G_\phi(z^{(i)}))]. \quad (22)$$

It is obvious that (21) and (22) respectively update the parameters of θ and ϕ towards opposite directions. In particular, the objective of discriminator D is to discriminate the generated dataset and the real one to the greatest extent, while the generator G tends to minimize the possibility of being discriminated by the discriminator D.

We summarize the process of W-GANs in Algorithm 1.

Algorithm 2 Federated Augmentation

```
Input: $\alpha_g$, the learning rate of generator; $\alpha_d$, the learning rate of discriminator. $m$, the batch size; $N_d$, the number of training steps of the discriminator; $N_g$, the number of training steps until convergence of the generator; $\theta$, initial parameters of the discriminator. $\phi$, initial parameters of the generator.
Output: Generator $G_\phi$
while $n_g \le N_g$ do
    for $n_d = 1, \cdots, N_d$ do
        Sample $\{x^{(i)}\}_{i=1}^m \sim \pi_u^e$ a batch from the expert dataset.
        Sample $\{z^{(i)}\}_{i=1}^m \sim \pi_u^z$ a batch from the prior noise samples.
        $\dot{\theta} \leftarrow \nabla_\theta[\frac{1}{m}\sum_{i=1}^m[D_\theta(x^{(i)}) - D_\theta(G_\phi(z^{(i)}))]]$
        $\theta \leftarrow \theta + \alpha_d \cdot RMSProp(\theta, \dot{\theta})$
    end for
    Sample $\{z^{(i)}\}_{i=1}^m \sim \pi_z$ a batch of the prior noise samples.
    $\dot{\phi} \leftarrow -\nabla_\phi[\frac{1}{m}\sum_{i=1}^m D_\theta(G_\phi(z^{(i)}))]$
    $\phi \leftarrow \phi - \alpha_g \cdot RMSProp(\phi, \dot{\phi})$
    $n_g = n_g + 1$
end while
```

B. ACTOR-CRITIC ALGORITHM FOR POWER CONTROL IN A LOCAL USER DEVICE

To assist the distributed edge UEs in making local decisions, we introduce the deep reinforcement learning (DRL) based Actor and Critic neural networks [44] as the local decision-making algorithm which makes decisions by timely interacting with the dynamic network environment. Specifically, the decision-making process of each UE is formulated as a non-cooperative partially observable Markov decision process (POMDP), where UE makes decisions individually by observing its surrounding network information.

We consider deterministic state transition in the POMDP $\mathcal{M}$, which can be modeled as a four-tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{R}, \gamma)$, and we respectively elaborate these parameters below.

- $\mathcal{S}^{(t)}$ represents the partially observable network state of SBSs. Let the observed information consist of the allocated transmit power, the SINR on the assigned RBs, and the reward of the previous training.
- $\mathcal{A}^{(t)}$ represents the actions set at time t. In the POMDP, the network states change with the actions which is defined as the power allocation strategies.
- $\mathcal{R}^{(t)} : \mathcal{S}^{(t)} \times \mathcal{A}^{(t)} \times \mathcal{S}^{(t+1)} \rightarrow R^{(t)}$ is a family of reward functions which guides UEs make decisions towards the expected direction. Recall that the SINR is defined as

$$\gamma_{u,k}^{(t)} = \frac{p_{u,k}^{(t)} g_{u,k}^{(t)}}{N^{env}}, \quad (23)$$

where $N^{env} = \sum_{i \in \mathcal{U} \setminus \{u\}} p_{i,k}^{(t)} g_{i,k}^{(t)} + wN_0$ is the environment noise consisting of interferences from other UEs and the AWGN. Obviously, increasing the transmit power can improve the SINR and thus the throughput of (2).

To derive the reward function, we first take the logarithm on both sides of (23), and replace the logarithm results of SINR, power, pass loss, and environmental noise with $\Gamma, \mathcal{P}, \mathcal{G}$ and $\mathcal{N}$, respectively. Therefore, the SINR of (23) is rewritten as

$$\Gamma_{u,k}^{(t)} = \mathcal{P}_{u,k}^{(t)} - \mathcal{G}_{u,k}^{(t)} - \mathcal{N}, \tag{24}$$

where the SINR is decomposed into three parts which are respectively transmit power, path loss, and environment noise. Obviously, the path loss $\mathcal{G}_{u,k}^{(t)}$ is determined once the location of the base station is determined. Moreover, the environment noise is not controllable in this decentralized network. As shown in (6), the objective is to minimize the transmit power while satisfying the traffic QoS requirements. Therefore, we design the local cost function $\check{R}$ for UE u at time t as

$$\check{R}_u^{(t)} = \mathcal{P}_{u,k}^{(t)} - \mathbb{P}(\Gamma_{u,k}^{(t)}), \tag{25}$$

where the penalty $\mathbb{P}(\gamma_{u,k}^{(t)})$ is defined as

$$\mathbb{P}(\gamma_{u,k}^{(t)}) = -|10\lg(\gamma_{u,k}^{(t)} - \check{\gamma}_u^{(t)} + 1)|, \tag{26}$$

where $\gamma_{u,k}^{(t)} = \begin{cases} \frac{\gamma_{u,k}^{(t)}}{\check{\gamma}_u^{(t)}} + \check{\gamma}_u^{(t)} - 1, & \text{if } \gamma_{u,k}^{(t)} \leq \check{\gamma}_u^{(t)} \\ \gamma_{u,k}^{(t)}, & \text{otherwise.} \end{cases}$

- γ ($\gamma \in (0, 1)$) denotes the reward discount factor in the Markov chain. Discount factors are important in infinite-horizon MDPs, in which they determine how the reward is counted.

The objective of the POMDP is to find policies $\pi(\ell)$ which optimize reward $\check{R}_u^{(t)}$ from time t to $t + T$, following a strategies' trajectory $\ell \sim \{s_0, a_0, s_1, a_1, \cdots, s_T, a_T\}$. We formulate the objective function $O(\pi_\vartheta)$ as:

$$\text{Min} \quad O(\pi(\ell)) = E_{\ell \sim \pi(\ell)}[\mathcal{R}_u(\ell)], \tag{27}$$

where $\mathcal{R}_u(\ell) = \sum_{k=0}^{T} \gamma^k \check{R}_u^{(t+k)}$ is the discounted cumulative reward starting from time t and increasingly discounted at subsequent steps by factor $\gamma \in (0, 1]$.

Note that the POMDP problem has continues state and action space as the wireless channel state and the amount of assigned resources are continuous variables, and thus it is infeasible to compute and save all value functions for every particular state-action pair. With respect to continuous or infinite state and action problems, the objective of (27) for a given edge UE is rewritten as

$$O(\pi(\ell)) = E_{\ell \sim \pi(\ell)}[\mathcal{R}(\ell)] = \int_{\ell \sim \pi(\ell)} \pi(\ell)\mathcal{R}(\ell)d\ell. \tag{28}$$

In the following, we adopt the DRL based Actor-Critic algorithm to optimize the (27) from both of the Actor and Critic processes. In particular, the Actor process can respond to the network state quickly and provide corresponding strategies, and the Critic process is used to modify the parameters of the neural networks afterwards. Therefore, the collaboration of Actor and Critic is able to guide the strategies towards the expected direction in an efficient way.

1) ACTOR PROCESS

The Actor process works with a family of parameterized policies, which guides agents to make decisions towards the expected directions. We use Gaussian probability distribution to approximate the policy distributions. By using Gaussian probability distribution, *exploration* (searching for more better strategies) and *exploitation* (exploiting the current best strategies) can be dynamically balanced in the action selection process. Therefore, $\pi_\vartheta(a|s) \sim \mathcal{N}(\mu_\vartheta(s), \sigma_\xi^2(s))$, where $\mu(s)$ is the mean value and $\sigma(s)$ is the standard deviation. Obviously, $\mu(s)$ is indeed the action that has the largest probability to be chosen at state s, and $\sigma(s)$ indicates the extent of exploration over all actions at state s.

In the following, we use network state as the input feature vector $\Phi(s^{(t)})$. Moreover, a non-linear feature-based function is used to approximate the $\mu_\vartheta(s^{(t)})$ which is given by

$$\mu_\vartheta(s^{(t)}) = \text{NN}(\vartheta^{\text{T}}, \Phi(s^{(t)})), \tag{29}$$

where NN$(\cdot)$ represents the Neural Network applied in the non-linear approximation. Specifically, the Neural Network outputs the results with the input feature vector $\Phi(s^{(t)})$ and weight parameters ϑ of the dense layer. Similarily, the variance σ is updated by

$$\sigma_\xi(s^{(t)}) = \text{NN}(\xi^{\text{T}}, \Phi(s^{(t)})). \tag{30}$$

The parameters ϑ are optimized towards the direction of improving the objective (27). We define the gradient of the objective function with respect to the parameters ϑ and ξ as $\nabla_\vartheta O(\pi_\vartheta)$ and $\nabla_\xi O(\pi_\xi)$ which are respectively updated by

$$\nabla_\vartheta O(\pi_\vartheta) = \nabla_\vartheta \log \pi_\vartheta(a|s)\mathcal{R}(\ell) + \alpha_h \mathcal{H}, \tag{31}$$

and

$$\nabla_\xi O(\pi_\xi) = \nabla_\xi \log \pi_\xi(a|s)\mathcal{R}(\ell) + \alpha_h \mathcal{H}, \tag{32}$$

where $\mathcal{H}$ is the cross entropy cost used to encourage exploration, and α_h is the step-size.

Then, parameters of ϑ and ξ are respectively updated by $\Delta\vartheta = \alpha_\vartheta^{(t)} \nabla_\vartheta O(\pi_\vartheta)$ and $\Delta\xi = \alpha_\xi^{(t)} \nabla_\xi O(\pi_\xi)$, where $\alpha_\xi^{(t)}$ and $\alpha_\vartheta^{(t)} > 0$ are the step-size used for the policy update and ∇ denotes the back propagation through time.

2) CRITIC PROCESS

The Critic relies exclusively on the value function approximation aiming at approximating the Bellman equation [45]. Define the approximated state-action value as $V_\varphi(s^{(t)})$ which is parameterized by vector φ. We choose the non-linear feature-based function to approximate $V_\varphi(s^{(t)})$. Thus,

$$V_\varphi(s^{(t)}) = \text{NN}(\varphi^{\text{T}}, \Phi(s^{(t)})). \tag{33}$$

Next, we update the parameter vector φ in the critic process. We introduce temporal difference (TD) error as the loss function, therefore

$$f(\varphi^{(t)}) = V(s^{(t)}) - V_\varphi(s^{(t)}), \tag{34}$$

where $V(s^{(t)}) = R^{(t+1)} + \gamma V_\varphi(s^{(t+1)})$. Therefore, the objective of the critic is $\arg\min_\varphi \frac{1}{2} f(\varphi^{(t)})^2$, and the gradient of this quadratic error with respect to $f(\varphi^{(t)})$ is derived as $|f(\varphi^{(t)})| \cdot \nabla_\varphi V_\varphi(s^{(t)})$. Then we use the gradient descent method [46] to update the approximation towards the gradient, and thus the parameter vector φ can be updated by $\Delta\varphi = \alpha_c^{(t)} |f(\varphi^{(t)})| \nabla_\varphi V_\varphi(s^{(t)})$, where $\alpha_c^{(t)}$ is the step-size used for the gradient descent of the critic process.

As the variance of convergence in Actor-Critic algorithm could be very large, we introduce the advantage function $A(s^{(t)})$ as the bias to decrease the variance [46]. Moreover, we choose the widely used TD-error $f(\varphi^{(t)})$ as the advantage function $A(s^{(t)})$. Then, (31) can be rewritten as

$$\nabla_\vartheta O(\pi_\vartheta) = \nabla_\vartheta \log \pi_\vartheta(a|s) A(s^{(t)}) + \alpha_h \mathcal{H}. \tag{35}$$

Definition 2: Accordingly, the parameter $\mathbf{w}$ of the FL system is defined as $\mathbf{w}^{\mathrm{T}} = [\vartheta, \xi]^{\mathrm{T}}$, which is locally updated by the Actor-Critic algorithm, and globally updated by the FL learning.

In summary, we elaborate the algorithm of the fully on-line local AC with data augmentation in Algorithm 3.

VI. SIMULATION RESULTS

In this section, we evaluate the performance of our proposed FL-CA algorithm by extensive simulations. We use the Multi-Wall-and-Floor (MWF) model [47] as the propagation and penetration loss model between BS and UEs in our indoor scenario. MWF takes into account the decreasing penetration loss of walls and floors of the same category as the number of traversed walls/floors increase, which is given by $PL(d)[\text{dB}] = L_0[\text{dB}] + 20\,log_{10}(d) + n_w L_w$, where L_0 is the reference loss [dB] taken at one meter of distance between the transmitter and the receiver, d is the distance between the BS and UEs in meters, $L_w = 6dB$ is the penetration loss of the concrete wall, n_w is the number of walls. Other parameters used are listed in TABLE 1.

A. COMPARISON REFERENCES IN THE SIMULATION

Under the FL framework, we consider the following four algorithms as the comparison algorithms.

1) Federated Learning (FL): the traditional FL helps agents make decisions under FL system.
2) Federated Learning with Cooperation and Augmentation (FL-CA) is the algorithm proposed in this paper, which collaboratively utilize the federated augmentation and federated learning in the decision-making process.
3) Non-Cooperation Power Allocation (Non-CPA) or local AC algorithm: agents adopting the Non-CPA algorithm make decisions locally without any information sharing.

Algorithm 3 On-Line Actor-Critic Algorithm in a Local User Device With Augmentation

```
Input: $\Phi(s)$, feature vector at state $s$; $\vartheta$ and $\xi$, initial parameters of Actor network for $\mu$ and $\sigma$; $\varphi$, initial parameters of Critic network; $\alpha_\vartheta$ and $\alpha_\xi$, step-size of actor network; $\alpha_c$, step-size of critic network; $\alpha_h$, step-size of cross entropy; $t$ and $t_l$, counter; $T$ and $T_l$, length of the decision trajectory and training times of Actor-Critic.
Initialize time step counter $t \leftarrow 0$.
Initialize all of the parameters of Actor network and Critic network, and get initial state $s$.
while $t < T$ do
    Output power allocation strategies by inputting feature vector $\Phi(s)$ to Actor Network.
    for $t_l = 1, 2, \cdots, T_l$ do
        Data augmentation by generator $G_\phi(\Phi(s^{(t)}))$
        Compute the TD-error
            $f(\varphi^{(t)}) = R^{(t+1)} + \gamma V_\varphi(s^{(t+1)}) - V_\varphi(s^{(t)})$.
        Update the Critic Network:
            $\varphi = \varphi - \alpha_c^{(t)} |f(\varphi^{(t)})| \nabla_\varphi V_\varphi(s^{(t)})$.
        Compute the gradients $\vartheta$ and $\xi$:
            $\nabla_\vartheta O(\pi_\vartheta) = \nabla_\vartheta \log \pi_\vartheta(a|s) A(s) + \alpha_h \mathcal{H}$,
            $\nabla_\xi O(\pi_\xi) = \nabla_\xi \log \pi_\xi(a|s) A(s) + \alpha_h \mathcal{H}$.
        Update the Actor Network by gradient descent:
            $\vartheta = \vartheta - \alpha_\vartheta^{(t)} \nabla_\vartheta O(\pi_\vartheta)$,
            $\xi = \xi - \alpha_\xi^{(t)} \nabla_\xi O(\pi_\xi)$.
    end for
    $t \leftarrow t + 1$
end while
```

4) Greedy Algorithm (GA): agents choose power allocation strategies by observing the current network states.

B. NUMERICAL RESULTS AND DISCUSSION

We first examine the convergence and the performance of the generator and discriminator of WGANs. Fig. 2 shows the Wasserstein estimation with training epochs. We can see that both the two training curves converge at about 2000 epochs. Finally, the Wasserstein estimation of the generator converges to 1 and the discriminator D converges to 0, which means that the adversarial dataset generated by the generator G cannot be discriminated by the discriminator D, and D is unable to distinguish between the generated dataset and the real dataset correctly.

Next we examine the convergence of local AC algorithm with different decision trajectory lengths. As aforementioned, the DTI determines the decision trajectory length, which

TABLE 1. Simulation parameters.

Parameter Description	Value
System Bandwidth	20 MHz
Number of edge UEs	10
RB Bandwidth	180 KHz
Noise power spectral density	-174 dBm/Hz
Maximum SBS Transmit Power	23 dBm
Number of RBs	30
Reward Discount γ	0.90
Learning rate of Actor process α_a	0.001
Learning rate of Critic process α_c	0.001
Learning rate of generator α_g	0.001
Leanring rate of discriminator α_d	0.001
Batch size in WGANs m	64

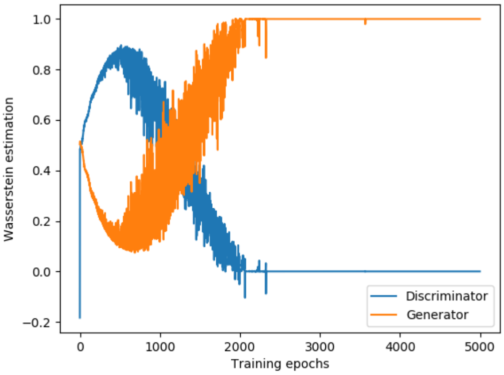


FIGURE 2. Training process of WGANs.

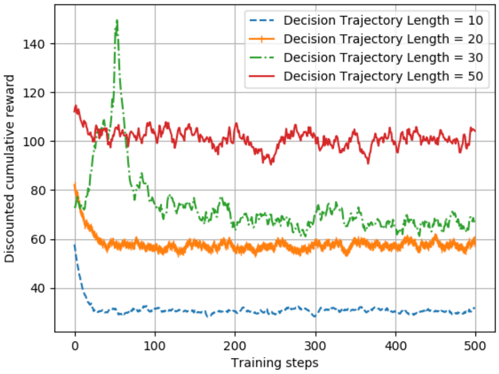


FIGURE 3. Convergence of local AC with different decision trajectory lengths.

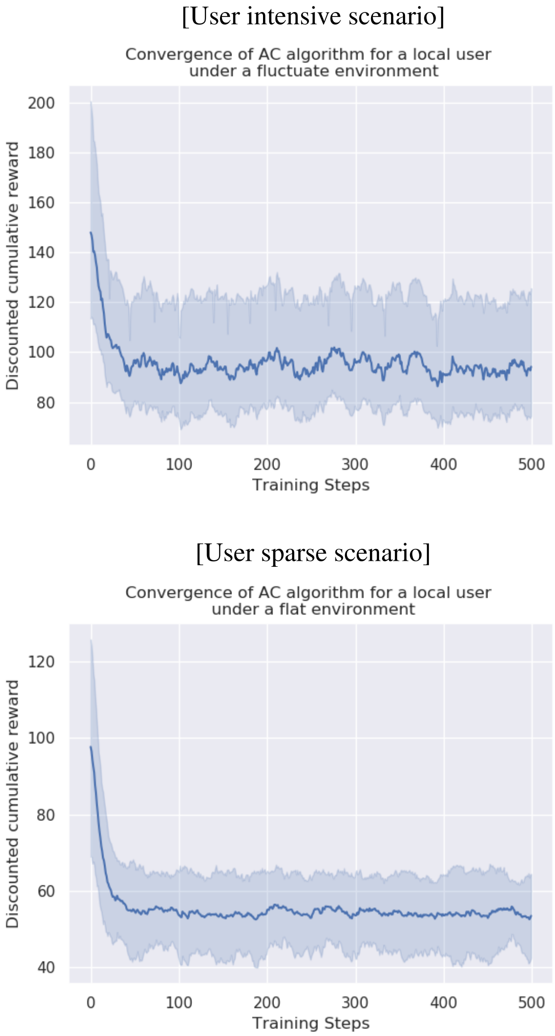
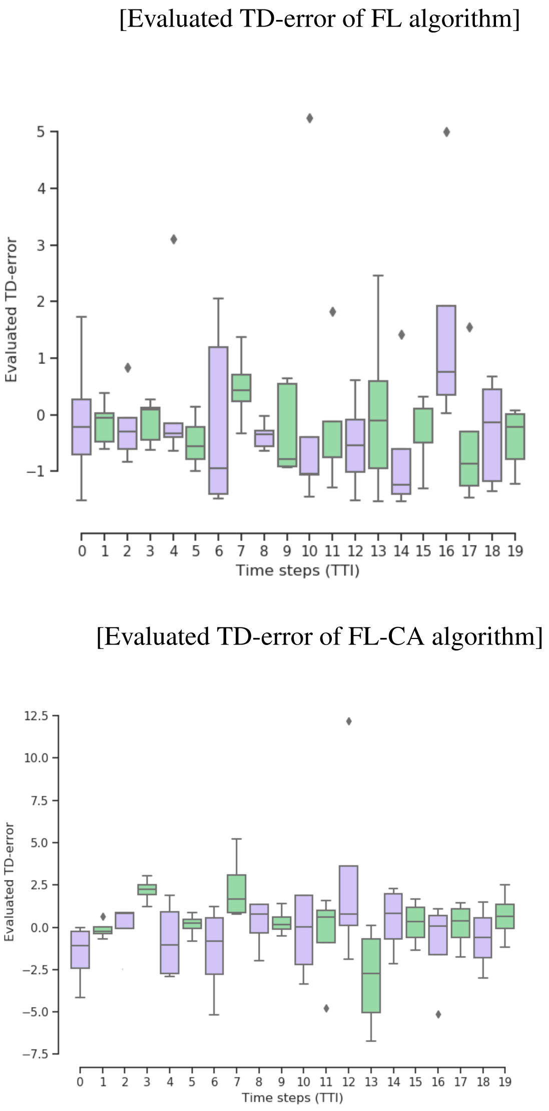


FIGURE 4. Convergence of local AC algorithm under different network deployments.

controls the cycle of the long-term optimization in the MDP. As shown in Fig. 3, we respectively set the decision trajectory length (DTL) as 10, 20, 30, and 50. Obviously, we can see that with the increase of DTL, the discounted cumulative reward increases, and so dose the jitter of the convergence. This is because with the increase of the decision-time period, the decision-depth also increases, which makes the network states space increases exponentially, resulting in that the convergence becomes more difficult. Therefore, in the following experiments, we choose the DTL as 20 for more stable simulation results.

Then, we investigate the convergence of the AC algorithm in a local user device under different network deployments. Fig. 4(a) and Fig. 4(b) show the discounted cumulative reward with training steps. Specifically, we fix the DTL as 20, and consider to investigate the convergence properties under user intensive network scenario and user sparse network scenario. In Fig. 4(a) and Fig. 4(b), we can see that the AC algorithm converges in less than 100 training steps, which fully satisfies the on-line learning requirement. Moreover, from the area of the shadow part, we can see that the variance of the convergence in user intensive scenario is much smaller. This is because the network states in user intensive scenario are more dynamic and difficult to be captured. Moreover, we can see that the discounted cumulative reward of Fig. 4(b) is smaller than that of Fig. 4(a). This is because the amount of power resources in user intensive environment are more scarce, which implies that the network environment is an important factor of the local AC algorithm's performance.


[Evaluated TD-error of FL algorithm]

[Evaluated TD-error of FL-CA algorithm]

FIGURE 5. Evaluated TD-error with 20 decision trajectory length.

In the next experiment, we evaluate the TD-error of traditional FL and the proposed FL-CA. Note that the evaluation is used to evaluate algorithm without extra training, which can verify the generalization and robustness of an algorithm. As defined in (34), TD-error shows the error between the approximated state value $V_\varphi(s^{(t)})$ and the average state value $V(s)$. Therefore, the more TD-error deviates from zero, the more inaccurate the approximated strategy is. Fig. 5 shows the box plot of the TD-error with evaluation time steps. In Fig. 5(a), we can see that the number of bad points of FL account for 35% of the total 20 evaluation samples, and in Fig. 5(b), the number of bad points of FL-CA account for 15% of the evaluation samples. This means the power allocation strategies derived from the FL-CA are more reliable than traditional FL algorithm. Moreover, from Fig. 6 we can directly see that the jitter of the TD-error of FL-CA is much smaller that of the FL. This implies that the robustness/generalization of FL-CA is better than that of the traditional FL.

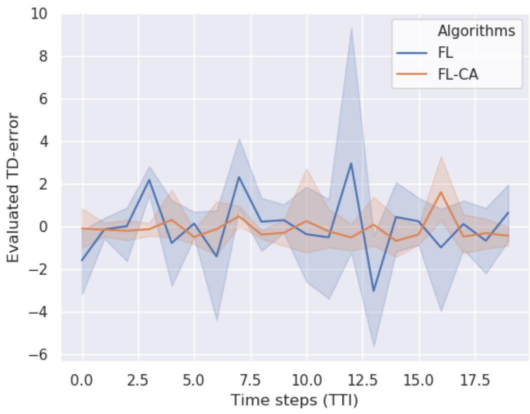


FIGURE 6. Comparison of evaluated TD-error.

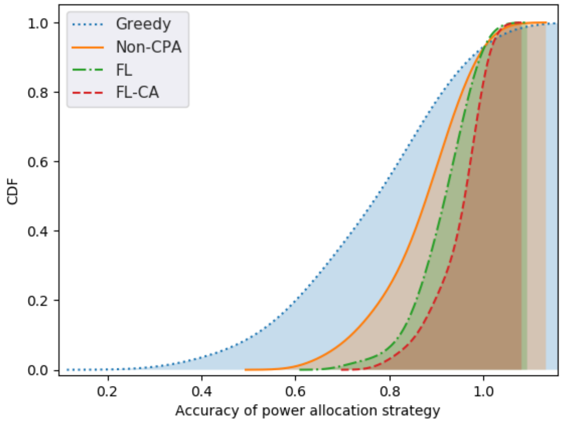


FIGURE 7. Accuracy of power allocation strategy.

In the last experiment, we compare the accuracy of power allocation strategy of the proposed four algorithms. As defined in (6), we aim to minimize a sum of the long-term transmit power objective function while satisfying the QoS requirements of user devices. Moreover, it is obvious that minimizing the transmit power requires accurate power allocation to mitigate the interferences between user devices. Therefore, we investigate the accuracy of the power allocation strategies to achieve the minimum power requirement. We statistically analyze the experiment results of 50 DTIs, and Fig. 7 shows the cumulative distribution function (CDF) of power allocation strategy accuracy. It is obvious that the accuracy of power allocation becomes higher with the narrower of the CDF graph. From Fig. 7, we can sort the accuracy of these four algorithms from small to large as Greedy, Non-CPA, FL, and FL-CA. Therefore, our proposed FL-CA algorithm in this paper performs the best in power allocation and can provides the most accurate power allocation strategies when compared with other benchmark algorithms.

VII. CONCLUSION

In this paper, we have proposed a federated learning framework based cooperation and augmentation (FL-CA) for solving the power allocation in decentralized networks. FL-CA aims at minimizing the power consumption while satisfying the user QoS requirement and protecting user privacy. In the FL framework, edge devices locally make decisions on power allocation through training a local Actor-Critic (AC) model, and then send the gradients and weightages generated by the Actor network to BS for information aggregation at regular intervals. Furthermore, to overcome the over-fitting problem caused by data leakages, we adopt federated augmentation (FAu) algorithm which uses WGANs for data augmentation. FAu empowers each device to replenish the data buffer using a generative model of WGANs until reaching an i.i.d training dataset, which significantly reduces the communication overhead compared to direct data sample exchanges. Significant performance improvements in terms of algorithm robustness and the power allocation accuracy when compared with other three benchmark algorithms.

REFERENCES

[1] C. V. N. Index, "Global mobile data traffic forecast update, 2017–2022," Cisco, San Jose, CA, USA, White Paper, 2019.

[2] N. Ahmed Malik and M. Ur-Rehman, "Green communications: Techniques and challenges," *EAI Endorsed Trans. Energy Web*, vol. 4, no. 14, 2017, Art. no. 153162.

[3] R. Kelly, "Internet of Things data to top 1.6 zettabytes by 2022," *Campus Technol.*, vol. 9, pp. 1233–1536, Apr. 2016.

[4] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," *ETSI White Paper*, vol. 11, no. 11, pp. 1–16, 2015.

[5] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[6] N. Kato, B. Mao, F. Tang, Y. Kawamoto, and J. Liu, "Ten challenges in advancing machine learning technologies towards 6G," *IEEE Wireless Commun. Mag.*, to be published, doi: 10.1109/MNET.001.1900476.

[7] Y. Mao, J. Zhang, and K. B. Letaief, "Joint task offloading scheduling and transmit power allocation for mobile-edge computing systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2017, pp. 1–6.

[8] S. Fu, B. Wu, H. Wen, P.-H. Ho, and G. Feng, "Transmission scheduling and game theoretical power allocation for interference coordination in CoMP," *IEEE Trans. Wireless Commun.*, vol. 13, no. 1, pp. 112–123, Jan. 2014.

[9] W. House, *Consumer Data Privacy in a Networked World: A Framework for Protecting Privacy and Promoting Innovation in the Global Digital Economy*. Washington, DC, USA: White House, 2012, pp. 1–62.

[10] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "A collaborative training algorithm for distributed learning," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1856–1871, Apr. 2009.

[11] Y. Cheng and M. Pesavento, "Joint optimization of source power allocation and distributed relay beamforming in multiuser Peer-to-Peer relay networks," *IEEE Trans. Signal Process.*, vol. 60, no. 6, pp. 2962–2973, Jun. 2012.

[12] G. Zhu, D. Liu, Y. Du, C. You, J. Zhang, and K. Huang, "Towards an intelligent edge: Wireless communication meets machine learning," 2018, *arXiv:1809.00343*. [Online]. Available: http://arxiv.org/abs/1809.00343

[13] X. Wang, Y. Han, C. Wang, Q. Zhao, X. Chen, and M. Chen, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep. 2019.

[14] M. Liu, T. Song, and G. Gui, "Deep cognitive perspective: Resource allocation for NOMA-based heterogeneous IoT with imperfect SIC," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2885–2894, Apr. 2019.

[15] B. Cao, L. Zhang, Y. Li, D. Feng, and W. Cao, "Intelligent offloading in multi-access edge computing: A state-of-the-art review and framework," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 56–62, Mar. 2019.

[16] M. Liu, T. Song, J. Hu, J. Yang, and G. Gui, "Deep learning-inspired message passing algorithm for efficient resource allocation in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 641–653, Jan. 2019.

[17] L. Xiao, Y. Li, C. Dai, H. Dai, and H. V. Poor, "Reinforcement learning-based NOMA power allocation in the presence of smart jamming," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3377–3389, Apr. 2018.

[18] Z. Xu, Y. Wang, J. Tang, J. Wang, and M. C. Gursoy, "A deep reinforcement learning based framework for power-efficient resource allocation in cloud RANs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–6.

[19] J. Konečný, H. B. McMahan, F. X. Yu, P. Richtárik, A. Theertha Suresh, and D. Bacon, "Federated learning: Strategies for improving communication efficiency," 2016, *arXiv:1610.05492*. [Online]. Available: http://arxiv.org/abs/1610.05492

[20] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S. Talwalkar, "Federated multi-task learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4424–4434.

[21] S. Wang, T. Tuor, T. Salonidis, K. K. Leung, C. Makaya, T. He, and K. Chan, "Adaptive federated learning in resource constrained edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 6, pp. 1205–1221, Jun. 2019.

[22] W. Yang Bryan Lim, N. Cong Luong, D. Thai Hoang, Y. Jiao, Y.-C. Liang, Q. Yang, D. Niyato, and C. Miao, "Federated learning in mobile edge networks: A comprehensive survey," 2019, *arXiv:1909.11875*. [Online]. Available: http://arxiv.org/abs/1909.11875

[23] P. Papadimitriou and H. Garcia-Molina, "Data leakage detection," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 1, pp. 51–63, Jan. 2011.

[24] R. Johnson and T. Zhang, "Accelerating stochastic gradient descent using predictive variance reduction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 315–323.

[25] O. Shamir, N. Srebro, and T. Zhang, "Communication-efficient distributed optimization using an approximate Newton-type method," in *Proc. Int. Conf. Mach. Learn.*, 2014, pp. 1000–1008.

[26] Y. Tachwali, B. F. Lo, I. F. Akyildiz, and R. Agusti, "Multiuser resource allocation optimization using bandwidth-power product in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 3, pp. 451–463, Mar. 2013.

[27] W.-D. Gao, W.-B. Wang, G. Yuan, and M.-G. Peng, "Joint relay selection and power allocation optimization in cooperative communications," *J. Beijing Univ. Posts Telecommun.*, vol. 31, no. 2, pp. 68–72, 2008.

[28] F. Fang, H. Zhang, J. Cheng, S. Roy, and V. C. M. Leung, "Joint user scheduling and power allocation optimization for energy-efficient NOMA systems with imperfect CSI," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 12, pp. 2874–2885, Dec. 2017.

[29] B. Cao, S. Xia, J. Han, and Y. Li, "A distributed game methodology for crowdsensing in uncertain wireless scenario," *IEEE Trans. Mobile Comput.*, vol. 19, no. 1, pp. 15–28, Jan. 2020.

[30] M. Yan, G. Feng, J. Zhou, and S. Qin, "Smart multi-RAT access based on multiagent reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4539–4551, May 2018.

[31] M. Yan, G. Feng, J. Zhou, Y. Sun, and Y.-C. Liang, "Intelligent resource scheduling for 5G radio access network slicing," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7691–7703, Aug. 2019.

[32] Y. Li, M. Sheng, Y. Sun, and Y. Shi, "Joint optimization of BS operation, user association, subcarrier assignment, and power allocation for energy-efficient HetNets," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3339–3353, Dec. 2016.

[33] C. Xiong, G. Ye Li, S. Zhang, Y. Chen, and S. Xu, "Energy- and spectral-efficiency tradeoff in downlink OFDMA networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 11, pp. 3874–3886, Nov. 2011.

[34] V. Sciancalepore, I. Filippini, V. Mancuso, A. Capone, and A. Banchs, "A multi-traffic inter-cell interference coordination scheme in dense cellular networks," *IEEE/ACM Trans. Netw.*, vol. 26, no. 5, pp. 2361–2375, Oct. 2018.

[35] F. Ahmed, A. A. Dowhuszko, and O. Tirkkonen, "Self-organizing algorithms for interference coordination in small cell networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 8333–8346, Sep. 2017.

[36] M. Rasti, A. R. Sharafat, and J. Zander, "Pareto and energy-efficient distributed power control with feasibility check in wireless networks," *IEEE Trans. Inf. Theory*, vol. 57, no. 1, pp. 245–255, Jan. 2011.

[37] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," 2016, *arXiv:1602.05629*. [Online]. Available: http://arxiv.org/abs/1602.05629

[38] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[39] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.

[40] E. Jeong, S. Oh, H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Communication-efficient on-device machine learning: Federated distillation and augmentation under non-IID private data," 2018, *arXiv:1811.11479*. [Online]. Available: http://arxiv.org/abs/1811.11479

[41] C. Dinh, N. H. Tran, M. N. H. Nguyen, C. Seon Hong, W. Bao, A. Y. Zomaya, and V. Gramoli, "Federated learning over wireless networks: Convergence analysis and resource allocation," 2019, *arXiv:1910.13067*. [Online]. Available: http://arxiv.org/abs/1910.13067

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

[43] J. L. Lions, *Optimal Control of Systems Governed by Partial Differential Equations (Grundlehren der mathematischen Wissenschaften)*, vol. 170. Berlin, Germany: Springer, 1971.

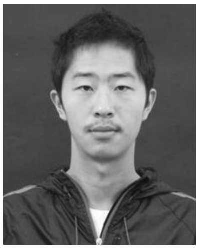
[44] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing Atari with deep reinforcement learning," 2013, *arXiv:1312.5602*. [Online]. Available: http://arxiv.org/abs/1312.5602

[45] I. Grondman, L. Busoniu, G. A. D. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 42, no. 6, pp. 1291–1307, Nov. 2012.

[46] R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Proc. Adv. Neural Inf. Process. Syst.*, 1999, pp. 1057–1063.

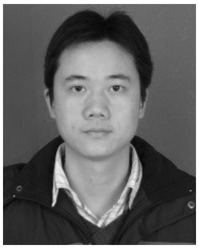
[47] M. Lott and I. Forkel, "A multi-wall-and-floor model for indoor radio propagation," in *Proc. IEEE VTS 53rd Veh. Technol. Conf., Spring*, vol. 1, May 2001, pp. 464–468.

MU YAN received the B.S. degree in electronic engineering from Beijing Jiaotong University, in 2014. He is currently pursuing the Ph.D. degree with the National Key Laboratory of Science and Technology on Communications, University of Electronic Science and Technology of China (UESTC). His research interests include next generation cellular networks, access control, multirate transmission, and machine learning.



BOLUN CHEN received the B.S. degree in electronic engineering from the University of Electronic Science and Technology of China (UESTC), in 2018, where he is currently pursuing the master's degree with the National Key Laboratory of Science and Technology on Communications. His research interests include mobile ad hoc networks, virtual network embedding, and machine learning.

GANG FENG (Senior Member, IEEE) received the B.Eng. and M.Eng. degrees in electronic engineering from the University of Electronic Science and Technology of China (UESTC), in 1986 and 1989, respectively, and the Ph.D. degree in information engineering from The Chinese University of Hong Kong, in 1998. He joined the School of Electrical and Electronic Engineering, Nanyang Technological University, in December 2000, as an Assistant Professor and became an Associate Professor, in October 2005. He is currently a Professor with the National Key Laboratory of Science and Technology on Communications, UESTC. He has extensive research experience and has published widely in wireless networking research. A number of his articles have been highly cited. His research interests include next generation mobile networks, mobile cloud computing, and AI-enabled wireless networking. He received the IEEE ComSoc TAOS Best Paper Award and the ICC Best Paper Award, in 2019.



SHUANG QIN (Member, IEEE) received the B.E. degree in electronic information science and technology and the Ph.D. degree in communication and information system from the University of Electronic Science and Technology of China (UESTC), in 2006 and 2012, respectively. He is currently an Associate Professor with the National Key Laboratory of Science and Technology on Communications, UESTC. His research interests include cooperative communication in wireless networks, data transmission in opportunistic networks, and green communication in heterogeneous networks.