Cloud-Edge-Device Collaborative Reliable and Communication-Efficient Digital Twin For Low-Carbon Electrical Equipment Management

Haijun Liao, *Student Member, IEEE,* Zhenyu Zhou, *Senior Member, IEEE,* Nian Liu, *Member, IEEE,* Yan Zhang, *Fellow, IEEE,* Guangyuan Xu, Zhenti Wang, Shahid Mumtaz, *Senior Member, IEEE*

***Abstract*—The real-time electrical equipment management such as renewable energy, controllable loads, and storage units, plays a key role in low-carbon operation of smart industrial park. Digital twin (DT) which explores cloud-edge-device collaboration and artificial intelligence to establish accurate digital representation of physical equipment is a cutting-edge technology to realize intelligent optimization of electrical equipment management. However, the practical implementation still faces reliability and communication efficiency problems, such as adverse impact of electromagnetic interference on DT reliability, high communication cost of DT model training, and uncoordinated resource allocation among cloud, edge, and device layers. We propose a Cloud-edge-device Collaborative reliable and Communication-efficient DT for lOW-carbon electrical equipment management named C^3-FLOW. It minimizes the long-term global loss function and time-average communication cost by jointly optimizing device scheduling, channel allocation, and computational resource allocation. Simulation results verify that C^3-FLOW performs superior in loss function, communication efficiency, and carbon emission reduction.**

***Index Terms*—Electrical equipment management, digital twin, cloud-edge-device collaboration, federated learning, reliability and communication efficiency**

I. Introduction

Low-carbon smart industrial park with various types of electrical equipment including renewable energy generators, controllable loads, and energy storage units, is a key pillar for building new-type power system [1]. Based on real-time management, electrical equipment can be accurately controlled to increase renewable energy consumption, reduce carbon emission and realize carbon neutrality [2]. Digital twin (DT) is an important technology to achieve these goals. DT provides a real-time digital representation of electrical equipment as well as energy flows based on data collected by massive internet of things (IoT) devices [3], [4]. DT can be further combined with cloud-edge-device collaboration and artificial intelligence (AI) to mine the relationship between grid states and electrical equipment management optimization through model training.

To further reduce the large-volume data exchange and processing required for DT-enabled low-carbon electrical equipment management, federated learning (FL) can be explored to separate global model training from raw data exchange [5]. The large-scale electrical equipment management can be divided into several regions, where regional model is trained by edge aggregation based on device-side training models and global model is trained by global aggregation based on multiple edge-side regional models. However, despite the potential advantages of DT for supporting low-carbon electrical equipment management, the electromagnetic interference (EMI) and communication cost raise new reliability and efficiency issues, which are introduced as follows.

Adverse Impact of EMI on DT Reliability: The error data caused by EMI of electrical equipment as well as noise pose a hazardous impact on model training reliability. They significantly degrade the global loss function performance, which quantifies the deviation between the trained model and the optimal one.

High Communication Cost of DT Model Training: The frequent model distribution, training, and uploading of FL results in high communication cost in terms of delay as well as energy consumption of data transmission and computation [6], [7]. Due to the interdependency of model training among different layers, delay performance is dominated by devices with less computational resources and poor channel gains [8]. Moreover, considering the paradox between delay and energy, optimizing one metric deteriorates the other metric performance. For instance, allocating more transmission power and computational resources to reduce delay will increase energy consumption.

Uncoordinated resource allocation among cloud, edge and device layers: The joint optimization of multi-layer, multi-entity, and multi-dimensional heterogeneous resources of cloud, edge, and device layers is NP-hard and faces dimensionality curse. First, the interdependency of resource allocation strategies among different devices introduces externality. Second, considering the different time granularities of global,

Manuscript received January 12, 2022; revised June 15, 2022; accepted July 21, 2022; current version July 26, 2022. This work was supported by the Science and Technology Project of State Grid Corporation of China under Grant Number 52094021N010 (5400-202199534A-0-5-ZN). *(Corresponding author: Zhenyu Zhou.)*

H. Liao, Z. Zhou and N. Liu are with Hebei Key Laboratory of Power Internet of Things Technology, North China Electric Power University, Baoding, 071003, Beijing, 102206, China (e-mail: haijun_liao@ncepu.edu.cn, zhenyu_zhou@ncepu.edu.cn, nian_liu@163.com).

Y. Zhang is with the Department of Informatics, University of Oslo, 0316 Oslo, Norway, and also with the Simula Metropolitan Center for Digital Engineering, 0167 Oslo, Norway (e-mail: yanzhang@ieee.org).

G. Xu and Z. Wang are with the Chuzhou Power Supply Company of Smart Grid Anhui Electric Power Co. Ltd., Chuzhou, 239000, Anhui, China (e-mail:18855005943@189.com, 18955000052@189.com).

S. Mumtaz is with the Instituto de Telecomunicações, Campus Universitário de Santiago, University of Aveiro, 3810-193, Aveiro, Portugal, and also with Nottingham Trent University, Nottingham, NG1 4BU, UK (e-mail: smumtaz@av.it.pt).

regional, and device-side model training, resource allocations of cloud, edge, and device layers should also be optimized in different timescales. Last but not least, the optimization of resource allocation from a long-term perspective is coupled with short-term constraints.

DT loss function optimization has attracted intensive attention from industry and academia. In [9], Shi *et al.* proposed a DT loss function minimization solution through the joint optimization of device scheduling and bandwidth allocation. In [10], Wang *et al.* minimized the DT loss function by exploiting the tradeoff between global aggregation and local updating. However, the above works mainly focus on edge computing-based regional model training, and are not be applicable to large-scale model training performed in cloud, edge, and device layers with different timescales. A federated averaging algorithm was proposed to reduce the training time and energy cost based on a client-edge-cloud hierarchical FL-assisted DT framework [11]. However, the impact of data error caused by EMI and noise on the loss function optimization is neglected. In addition, the tradeoff between DT reliability and communication efficiency as well as the coupling of resource allocation among multiple layers, entities, and timescales are not considered.

Motivated by these challenges, we propose a Cloud-edge-device Collaborative reliable and Communication-efficient DT for lOW-carbon electrical equipment management named C^3-FLOW. First, we present a four-layer hierarchical FL framework and formulate a DT-assisted equipment management model training problem. The objective is to minimize the sum of long-term global loss function and time-average communication cost through the joint optimization of device scheduling, power control, channel allocation, and computational resource allocation. Second, the problem is decoupled and transformed into minimizing the upper bound of the expected convergence gap and the communication cost in each slot. Next, the decoupled device scheduling, channel allocation and power control, and computational resource allocation subproblems are solved by the proposed deep-actor-critic (DAC) with TopN mapping, group-swap-matching, and smooth approximation-integrated Lagrange optimization algorithms.

The main contributions are summarized as follows.

- *Reliable DT under EMI and Noise:* DT reliability is improved from two aspects. On one hand, packet error rate (PER) caused by EMI and noise is considered in the loss function minimization, and reduced through the joint optimization of power control and channel allocation. On the other hand, cyclic redundancy check (CRC) is adopted to remove uploaded models with data error from edge aggregation, which further reduces the global loss function.
- *Dynamic Tradeoff between Reliability and Communication Efficiency:* A dynamic tradeoff between reliability and communication efficiency can be achieved through adjusting the corresponding weights of loss function, delay and energy consumption. The proposed DAC with TopN mapping algorithm can further reduce the communication cost by establishing a constrained set of scheduled devices with largest scheduling probabilities learned through constant environment interaction.
- *Cloud-Edge-Device Collaborative Resource Allocation:* DT provides estimated state information such as channel gain and EMI power for cloud-edge-device collaborative resource allocation. The spectrum and energy resources at the device layer are matched with the scheduled devices by the group-swap-matching-based power control and channel allocation algorithm, which solves externality in low complexity. The computational resources at the device, edge, and cloud layers are dynamically allocated based on the smooth approximation-integrated Lagrange optimization.

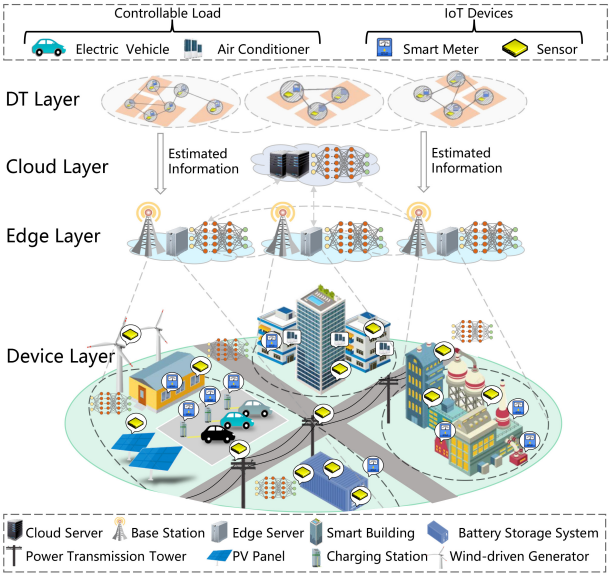

Fig. 1. A cloud-edge-device collaborative DT framework for low-carbon electrical equipment management.

The rest of this paper is organized as follows. The system model is introduced in Section II. Section III presents the problem formulation and transformation. The proposed C^3-FLOW is elaborated in Section IV. The simulation results are given in Section V. Section VI concludes this paper.

II. System Model

A cloud-edge-device collaborative DT framework for low-carbon electrical equipment management is shown in Fig. 1. FL is explored to train a low-carbon electrical equipment management model $\boldsymbol{\omega}$ through the cooperation among four layers, i.e., the device layer, the edge layer, the cloud layer, and the DT layer. In the device layer, massive IoT devices are deployed in electrical equipment, such as PV panels, power grid, loads, and energy storage units, to facilitate low-carbon management. The set of I devices is denoted as $\mathcal{U} = \{u_1, \cdots, u_i, \cdots, u_I\}$. In the edge layer, there are multiple edge servers co-located with base stations (BSs) to provide communication coverage and data processing. Each edge server schedules a portion of devices within its coverage to upload their trained models. Based on the device-side models, edge aggregation is performed to obtain a regional

model. In the cloud layer, a cloud server collects the regional models from the edge layer and performs cloud aggregation to obtain a global model. The DT layer is maintained by the cloud server, where the digital representations of the electrical equipment are continuously synchronized with the physical objects through state information exchange [12]. Edge servers dynamically optimize device scheduling and resource allocation to improve the reliability and communication efficiency of DT. Compared with the traditional edge-device DT framework and cloud-device DT framework, the cloud-edge-device collaborative DT framework has the advantages of global digitalization and cross-region resource scheduling, DT consistency improvement, and full utilization of cloud-edge-device multi-dimensional resources.

A two-timescale FL-based model training process is proposed [13]. We consider G epochs, and each epoch consists of T_0 slots. The G epochs contain $T = GT_0$ slots, the set of which is $\mathcal{T} = \{1, \cdots, t, \cdots, T\}$. The edge aggregation-based regional model training is performed in a small timescale, i.e., every slot, while the cloud aggregation-based global model training is performed in a large timescale, i.e., every epoch. The training process has four phases, i.e., model distribution, device-side model training, regional model training, and global model training, which are introduced as follows.

A. Model Distribution

The model distribution is composed of global model distribution per epoch and regional model distribution per slot. In each epoch, the cloud server distributes the global model to the edge servers through fiber networks, the delay of which can be neglected. In each slot, an edge server distributes the regional model to the scheduled devices, the delay of which is a constant $\tau_D(t)$.

Denote the set of J edge servers as $\mathcal{S} = \{s^1, \cdots, s^j, \cdots, s^J\}$, and the set of devices within the coverage of s^j as $\mathcal{U}^j = \{u_1^j, \cdots, u_i^j, \cdots, u_{U^j}^j\}$, $\forall \mathcal{U}^j \cap \forall \mathcal{U}^{j'} = \emptyset, \forall s^j, s^{j'} \in \mathcal{S}, j \neq j'$. The device scheduling strategy of s^j is denoted as $\boldsymbol{a}^j(t) = \{a_i^j(t) | \forall u_i^j\}$, where $a_i^j(t) = 1$ represents that s^j schedules u_i^j for FL training in the t-th slot, and $a_i^j(t) = 0$ otherwise.

B. Device-Side Model Training and Uploading

1) Device-Side Model Training: The scheduled devices, e.g., u_i^j, perform device-side model training based on their local dataset of electrical equipment management, i.e., $\mathcal{D}_i^j$, which contains D_i^j data samples. A device-side loss function $F(\boldsymbol{\omega}_i^j(t))$ is introduced to quantify the gap between the management performance based on the device-side model $\boldsymbol{\omega}_i^j(t)$ and the optimal one.

u_i^j updates $\boldsymbol{\omega}_i^j$ to minimize the device-side loss function as

$$\boldsymbol{\omega}_i^j(t) = \boldsymbol{\omega}_i^j(t-1) - \eta \nabla F(\boldsymbol{\omega}_i^j(t-1)), \tag{1}$$

where η is the learning rate.

Denote $f_i^j(t)$ as the CPU cycle frequencies allocated by u_i^j for device-side model training. The delay and the energy costs for training D_i^j data samples are given by

$$\tau_i^{j,Com}(t) = \frac{a_i^j(t) D_i^j \delta}{f_i^j(t)}, \ E_i^{j,Com}(t) = \alpha_i^j a_i^j(t) D_i^j \delta {f_i^j}^2(t), \tag{2}$$

where α_i^j is the capacitive constant of u_i^j, and δ is the CPU cycles required for training one sample.

2) Device-Side Model Uploading: There exist N_j orthogonal subchannels, the set of which is denoted as $\mathcal{C}^j = \{c_1^j, \cdots, c_n^j, \cdots, c_{N_j}^j\}$. Denote the channel allocation strategy of s^j as $\boldsymbol{r}^j(t) = \{r_{i,n}^j(t) | \forall u_i^j, \forall c_n^j\}$, where $r_{i,n}^j(t) = 1$ represents that s^j allocates c_n^j to u_i^j, and $r_{i,n}^j(t) = 0$ otherwise.

The model uploading delay and energy costs of u_i^j are given by

$$\begin{aligned} \tau_i^{j,Tx}(t) &= \frac{a_i^j(t) S}{\sum_{n=1}^{N} B^U r_{i,n}^j(t) \log_2 \left(1 + \frac{P_i^j(t) h_{i,n}^j(t)}{I_i^{j,U}(t) + B^U N_0}\right)}, \\ E_i^{j,Tx}(t) &= \tau_i^{j,Tx}(t) P_i(t), \end{aligned} \tag{3}$$

where S is the packet size of the device-side model $\boldsymbol{\omega}_i^j(t)$. B^U is the subchannel bandwidth, $P_i^j(t)$ is the transmission power, and $h_{i,n}^j(t)$ is the uplink channel gain of c_n^j. $I_i^{j,U}(t)$ and N_0 are the EMI power and the noise power spectral density [14].

C. Regional Model Training at Edge Layer

Based on the uploaded device-side models $\{\boldsymbol{\omega}_i^j(t)\}$, the edge server s^j updates the regional model $\boldsymbol{\omega}^j(t)$ through edge aggregation. To remove the error data caused by EMI and noise from edge aggregation, cyclic redundancy check (CRC) is adopted. Denote $\boldsymbol{q}^j(t) = \{q_i^j(t) \mid \forall u_i\}$ as the error indicator, where $q_i^j(t) = 0$ indicates that the received $\boldsymbol{\omega}_i^j(t)$ contains data error. $q_i^j(t)$ is calculated as

$$q_i^j(t) = \begin{cases} 1, \text{with probability } 1 - \sum_{n=1}^{N} r_{i,n}^j(t) q_{i,n}^j(t), \\ 0, \text{with probability } \sum_{n=1}^{N} r_{i,n}^j(t) q_{i,n}^j(t). \end{cases} \tag{4}$$

Here, $q_{i,n}^j(t)$ is the PER when u_i^j uploads $\boldsymbol{\omega}^j(t)$ to s^j on c_n^j with the presence of EMI and noise [15], which is given by

$$q_{i,n}^j(t) = 1 - \exp\left(-\frac{C(I_i^{j,U}(t) + B^U N_0)}{P_i^j(t) h_{i,n}^j(t)}\right), \tag{5}$$

where C is the waterfall threshold [15], [16].

After CRC, $\boldsymbol{\omega}^j(t)$ is updated as

$$\boldsymbol{\omega}^j(t) = \frac{\sum_{i=1}^{U^j} a_i^j(t) q_i^j(t) D_i^j \boldsymbol{\omega}_i^j(t)}{\sum_{i=1}^{U^j} a_i^j(t) q_i^j(t) D_i^j}. \tag{6}$$

Denote $f_G^j(t)$ as the CPU cycle frequencies allocated by s^j for edge aggregation. The regional model training delay and energy costs of s^j are given by

$$\begin{aligned} \tau_G^j(t) &= \frac{\left\| \boldsymbol{a}^j(t) * \boldsymbol{q}^j(t) \right\|^2 S\iota}{f_G^j(t)}, \\ E_G^j(t) &= \alpha_G^j \left\| \boldsymbol{a}^j(t) * \boldsymbol{q}^j(t) \right\|^2 S\iota {f_G^j(t)}^2, \end{aligned} \tag{7}$$

where $\|\cdot\|$ is the L^2 norm. $*$ is the Hadamard product symbol. ι is the number of CPU cycles required for aggregating one bit and α_G^j is the capacitive constant of s^j.

The edge servers upload regional models to the cloud server through fiber networks, the delay and energy costs of which can be neglected.

D. Global Model Training at Cloud Layer

Based on the uploaded regional models, i.e., $\{\boldsymbol{\omega}^j(t)\}$, $t = gT_0$, the cloud server updates the global model $\boldsymbol{\omega}(t)$ through cloud aggregation at each epoch, which is given by

$$\boldsymbol{\omega}(t) = \frac{\sum_{j=1}^{J}\sum_{i=1}^{U^j} a_i^j(t) q_i^j(t) D_i^j \boldsymbol{\omega}^j(t)}{\sum_{j=1}^{J}\sum_{i=1}^{U^j} a_i^j(t) q_i^j(t) D_i^j}. \tag{8}$$

Denote $f_C(g)$ as the CPU cycle frequencies allocated by the cloud server for cloud aggregation. The delay and energy costs of global model training are given by

$$\tau_C(g) = \frac{JS\iota}{f_C(g)}, E_C(g) = JS\iota f_C(g)^2, \tag{9}$$

where α_C is the capacitive constant of the cloud server.

E. Communication Cost of FL-Based Model Training

The total delay cost of s^j and the total energy cost of u_i^j in the t-th slot are given by

$$\begin{aligned}&\tau^j(t) = \tau_D(t) + \max\{\tau_i^{j,Tx}(t) + \tau_i^{j,Com}(t) | \forall u_i \in \mathcal{U}^j\} + \tau_G(t),\\ &E_i^j(t) = E_i^{j,Tx}(t) + E_i^{j,Com}(t).\end{aligned} \tag{10}$$

The total delay cost of FL-based model training in the g-th epoch, i.e., T_0 slots, is given by

$$\tau(g) = \max\left\{\sum_{t=T_0(g-1)+1}^{gT_0} \tau^j(t) \mid \forall s^j\right\} + \tau_C(g). \tag{11}$$

The communication cost of FL-based model training over T slots is defined as the weighted sum of the total delay cost, as well as the energy costs of the cloud server, edge servers, and devices, which is given by

$$\begin{aligned}\Phi(T) =& \gamma_\tau \sum_{g=1}^{G} \tau(g) + \gamma_C \sum_{g=1}^{G} E_C(g) + \gamma_G \sum_{t=1}^{T}\sum_{i=1}^{J} E_G^j(t)\\ &+ \sum_{t=1}^{T}\sum_{j=1}^{J}\sum_{i=1}^{U^j} \gamma_i^j E_i^j(t),\end{aligned} \tag{12}$$

where γ_τ, γ_C, γ_G, and γ_i^j are the corresponding weights.

III. Problem Formulation and Transformation

In this section, we introduce the problem formulation and transformation.

A. Problem Formulation

Define the global loss function $F(\boldsymbol{\omega}(t))$ as the gap between the electrical equipment management performance based on the global model $\boldsymbol{\omega}(t)$ and the optimal one. The FL aims to reduce the global loss function under EMI and noise, i.e., reliability improvement, and minimize the time-average communication cost simultaneously, i.e., communication efficiency improvement, by jointly optimizing device scheduling, power control, channel allocation, and computational resource allocation. The joint optimization problem is formulated as

$$\begin{aligned}\mathbf{P1}: \min_{\boldsymbol{a},\boldsymbol{P},\boldsymbol{r},\boldsymbol{f},\boldsymbol{f_G},\boldsymbol{f_C}} & \lim_{T\to\infty} F(\boldsymbol{\omega}(T)) + \frac{1}{T}\Phi(T),\\ \text{s.t. } & C_1: ||\boldsymbol{a}^j(t)||^2 \le A^j(t), \forall s^j, \forall t,\\ & C_2: \sum_{n=1}^{N} r_{i,n}^j(t) = a_i^j(t), \forall u_i, \forall s^j, \forall t,\\ & C_3: P_i^j(t) \in \{P_{i,min}^j, \cdots, P_{i,d}^j, \cdots, P_{i,max}^j\}, \forall s^j, \forall u_i, \forall t,\\ & C_4: f_i^j(t) \le f_{i,max}^j(t),\ f_e^j(t) \le f_{e,max}^j(t), \forall s^j, \forall u_i, \forall t,\\ & C_5: f_C(g) \le f_{C,max}(g), \forall g,\end{aligned} \tag{13}$$

where $\boldsymbol{a} = (\boldsymbol{a}^j(t), \forall s^j, \forall t)$, $\boldsymbol{P} = (P_i^j(t), \forall s^j, \forall u_i, \forall t)$, and $\boldsymbol{r} = (r_{i,n}^j(t), \forall u_i, \forall s^j, \forall n, \forall t)$ denote the vectors of device scheduling, power control, and channel allocation variables. $\boldsymbol{f} = (f_i^j(t), \forall s^j, \forall u_i, \forall t)$, $\boldsymbol{f_G} = (f_G^j(t), \forall s^j, \forall t)$, and $\boldsymbol{f_C} = (f_C(g), \forall g)$ denote the vectors of device-side, edge-side, and cloud-side computational resource allocation variables. C_1 is the maximum number of scheduled devices. C_2 guarantees that a scheduled device is allocated with at most one sub-channel. C_3 is the transmission power constraint. C_4 and C_5 are the constraints of device-side, edge-side, and cloud-side computational resource allocation.

B. Problem Transformation

$\mathbf{P1}$ is NP-hard due to the coupling of network states and optimization strategies among T slots. To solve $\mathbf{P1}$, we derive an upper bound of the expected convergence performance as

$$\begin{aligned}&\mathbb{E}\left[F(\boldsymbol{\omega}(t)) - F(\boldsymbol{\omega}^*)\right] \le \Delta F(t, t-1) + Z(t) F(\boldsymbol{\omega}(t-1))\\ &+ \xi_1 \sum_{j=1}^{J}\sum_{i=1}^{U^j} D_i^j \left(1 - a_i^j(t) + \sum_{n=1}^{N} a_i^j(t) r_{i,n}^j(t) q_{i,n}^j(t)\right) = B(t),\end{aligned} \tag{14}$$

where $F(\boldsymbol{\omega}^*)$ is the minimal global loss function. $Z(t)$ and $\Delta F(t, t-1)$ are given by

$$\begin{aligned}&Z(t) = 1 - \frac{\mu}{\rho} + \xi_2 \sum_{j=1}^{J}\sum_{i=1}^{I} D_i^j \left(1 - a_i^j(t) + \sum_{n=1}^{N} a_i^j(t) r_{i,n}^j(t) q_{i,n}^j(t)\right),\\ &\Delta F(t, t-1) = \frac{Z(t)}{g(g-1) G(\eta, T_0)},\end{aligned} \tag{15}$$

where $G(\eta, T_0)$ is a constant since η and T_0 are known [11]. ξ_1 and ξ_2 are the constants satisfying convergence requirement [16]. Then, based on (14), the minimization of $F(\boldsymbol{\omega}(T))$ is decoupled and transformed into minimizing the expected convergence gap, i.e., $B(t)$, in each slot. Besides, by utilizing the smooth approximation $\max\{g(x,y), h(x,y)\} = \ln[\exp(g(x,y) + h(x,y))]$, the minimization of $\Phi(T)$ is transformed into minimizing the delay and energy costs in each slot. Thus, $\mathbf{P1}$ can be transformed as

$$\begin{aligned}\mathbf{P2}: \min_{\boldsymbol{a},\boldsymbol{P},\boldsymbol{r},\boldsymbol{f},\boldsymbol{f_G}} & \Psi(t) = \Phi(t) + B(t),\\ \text{s.t. } & C_1 \sim C_4,\end{aligned} \tag{16}$$

where

$$\Phi(t) = \begin{cases}\gamma_\tau \tau_C(g) + \sum_{i=1}^{J} \gamma_G E_G^j(t) + \gamma_\tau \max\{\tau^j(t) | \forall s^j\}\\ + \gamma_C E_C(g) + \sum_{j=1}^{J}\sum_{i=1}^{U^j} \gamma_i^j E_i^j(t), & t = gT_0,\\ \gamma_\tau \max\{\tau^j(t) | \forall s^j\} + \sum_{i=1}^{J} \gamma_G E_G^j(t)\\ + \sum_{j=1}^{J}\sum_{i=1}^{U^j} \gamma_i^j E_i^j(t), & t \ne gT_0.\end{cases} \tag{17}$$

P2 can be further decomposed into five subproblems, i.e., the device scheduling subproblem **SP1**, the device-side computational resource allocation subproblem **SP2**, the device-side channel allocation and power control subproblem **SP3**, the edge-side computational resource allocation subproblem **SP4**, and the cloud-side computational resource allocation subproblem **SP5**. The corresponding solutions are introduced in the next section.

IV. Cloud-Edge-Device Collaborative Reliable and Communication-Efficient DT for Electrical Equipment Management

In this section, we propose the C^3-FLOW algorithm to sequentially solve **SP1** $\sim$ **SP5**.

A. Device Scheduling

The device scheduling subproblem **SP1** is formulated as

$$\begin{aligned}\mathbf{SP1}: \min_{\boldsymbol{a}^j(t)} \Psi^j(t) &= \frac{\Delta F(t,t-1)+Z(t)F(\boldsymbol{\omega}(t-1))}{J} \\ &+\xi_1\sum_{i=1}^{U^j} D_i^j\left(1-a_i^j(t)+\sum_{n=1}^{N} a_i^j(t) r_{i,n}^j(t) q_{i,n}^j(t)\right) \\ &+\gamma_G E_G^j(t)+\sum_{i=1}^{U^j}\gamma_i^j E_i^j(t)+\gamma_j \tau^j(t), \\ \text{s.t. } & C_1. \end{aligned} \tag{18}$$

SP1 can be modeled as a Markov decision process (MDP), and solved by DAC with TopN mapping [17], [18]. DT is utilized to estimate state information such as the device-side available computational resources $\tilde{f}_{i,max}^j(t)$, EMI $\tilde{I}_i^{j,U}(t)$, and channel gain $\tilde{h}_{i,n}^j(t)$. The state space is defined as $\boldsymbol{S}^j(t) = \{\boldsymbol{S}_1^j(t),\cdots,\boldsymbol{S}_i^j(t),\cdots,\boldsymbol{S}_{U^j}^j(t)\}$, where $\boldsymbol{S}_i^j(t) = \{\tilde{f}_{i,max}^j(t), P_{i,max}^j, \alpha_i^j, \tilde{I}_i^{j,U}(t), \tilde{h}_{i,n}^j(t)\}$. The action space is defined as $\boldsymbol{a}^j(t) = \{a_1^j(t),\cdots,a_i^j(t),\cdots,a_{U^j}^j(t)\}$, and the cost function is defined as the optimization objective of **SP1**, i.e., $\Psi^j(t)$.

The proposed algorithm leverages a policy-based actor network $\boldsymbol{\theta}^j(t)$ to draw and learn the device scheduling strategy, and a value-based critic network $\boldsymbol{\gamma}^j(t)$ to output the strategy performance evaluation. With $\boldsymbol{S}_i^j(t)$, $\boldsymbol{\theta}^j(t)$ outputs the probability that u_i^j is scheduled, which is denoted as $\pi(a_i^j(t)=1 \mid \boldsymbol{S}_i^j(t), \boldsymbol{\theta}^j(t))$ and simplified as $\pi(a_i^j(t)=1)$. Then, derive the temporary set of scheduled devices $\tilde{\mathcal{U}^j}(t)$ based on $\pi(a_i^j(t)=1)$. Next, a TopN mapping function $g(\cdot)$ obtains the set of scheduled devices with the largest $A^j(t)$ or $|\tilde{\mathcal{U}^j}(t)|$ scheduling probabilities, which is given by

$$\begin{aligned}&\mathcal{U}^j(t)=g(\tilde{\mathcal{U}^j}(t))= \\ &\begin{cases}\text{argTop}_{\text{N}}\left(A^j(t),\tilde{\mathcal{U}^j}(t),\pi(a_i^j(t)=1),\text{descent}\right), |\tilde{\mathcal{U}^j}(t)|>A^j(t), \\ \tilde{\mathcal{U}^j}(t), \quad |\tilde{\mathcal{U}^j}(t)|\le A^j(t).\end{cases}\end{aligned} \tag{19}$$

Therefore, the device scheduling strategy is obtained as

$$\boldsymbol{a}^j(t)=\{a_i^j(t) \mid a_i^j(t)=1 \text{ if } u_i^j\in\mathcal{U}^j(t); \text{else } a_i^j(t)=0\}. \tag{20}$$

B. Device-Side Computational Resource Allocation

The device-side computational resource allocation subproblem **SP2** is formulated as

$$\begin{aligned}\mathbf{SP2}: &\min_{\boldsymbol{f}^j(t)} \sum_{u_i\in\mathcal{U}^j(t)} \gamma_i\alpha_i D_i^j \delta (f_i^j(t))^2+\gamma_\tau \max_i\left\{\frac{D_i^j\delta}{f_i^j(t)}\right\}, \\ &\text{s.t. } C_4. \end{aligned} \tag{21}$$

where $\boldsymbol{f}^j(t)=\{f_i^j(t)|\forall u_i^j\}$.

By replacing $\max\{g(x,y),h(x,y)\}$ with $\ln[\exp(g(x,y)+h(x,y))]$ based on the smooth approximation, and $f_{i,max}^j(t)$ in C_4 with $\tilde{f}_{i,max}^j(t)$ estimated by DT, **SP2** is rewritten as

$$\begin{aligned}\tilde{\mathbf{SP2}}: &\min_{\boldsymbol{f}^j(t)} \sum_{u_i\in\mathcal{U}^j(t)} \gamma_i\alpha_i D_i^j\delta(f_i^j(t))^2 \\ &+\gamma_\tau \ln\left(\sum_{u_i\in\mathcal{U}^j(t)}\exp\left(\frac{D_i^j\delta}{f_i^j(t)}\right)\right), \\ &\text{s.t. } \hat{C}_4: f_i^j(t)\le\tilde{f}_{i,max}^j(t),\ \forall u_i\in\mathcal{U}^j(t). \end{aligned} \tag{22}$$

$\tilde{\mathbf{SP2}}$ is a convex problem and can be solved by Lagrange optimization.

C. Device-Side Channel Allocation and Power Control

The device-side channel allocation and power control subproblem **SP3** is formulated as

$$\begin{aligned}\mathbf{SP3}: &\min_{\boldsymbol{P}^j(t),\boldsymbol{r}^j(t)} \Lambda^j(t)=Z(t)(F(\boldsymbol{\omega}(t-1)+\frac{1}{g(g-1)G(\eta,T_0)}) \\ &+\xi_1\sum_{u_i^j\in\mathcal{U}^j(t)}\sum_{n=1}^{N^j} D_i^j r_{i,n}^j(t) q_{i,n}^j(t) \\ &+\max_{\boldsymbol{P}^j(t),\boldsymbol{r}^j(t)}\left\{\frac{\gamma_\tau S}{\sum_{n=1}^{N^j} B^U r_{i,n}^j(t)\log_2\left(1+\frac{P_i^j(t)\tilde{h}_{i,n}^j(t)}{\tilde{I}_i^{j,U}(t)+B^U N_0}\right)} \mid u_i^j\in\mathcal{U}^j(t)\right\} \\ &+\sum_{u_i^j\in\mathcal{U}^j(t)}\gamma_i\frac{P_i^j(t)S}{\sum_{n=1}^{N^j} B^U r_{i,n}^j(t)\log_2\left(1+\frac{P_i^j(t)\tilde{h}_{i,n}^j(t)}{\tilde{I}_i^{j,U}(t)+B^U N_0}\right)}, \\ &\text{s.t.}\tilde{C}_2: \sum_{n=1}^{N^j} r_{i,n}^j(t)=1,\ \forall u_i^j\in\mathcal{U}^j(t), \\ &\tilde{C}_3: P_i^j(t)\in\{P_{i,min}^j,\cdots,P_{i,d}^j,\cdots,P_{i,max}^j\},\forall u_i^j\in\mathcal{U}^j(t). \end{aligned} \tag{23}$$

where $\boldsymbol{P}^j(t)=\{P_i^j(t)|\forall u_i^j\}$. **SP3** can be transformed into a one-to-one matching problem with externality, which is caused by the interdependency of channel allocation strategies among devices. A group-swap-matching-based channel allocation and power control joint optimization algorithm is proposed to achieve stable matching between devices and channels in low complexity.

First, divide the devices and channels into L equal groups, i.e., $\mathcal{U}^j(t) = \{\mathcal{U}_1^j(t),\cdots,\mathcal{U}_l^j(t),\cdots,\mathcal{U}_L^j(t)\}$ and $\mathcal{N}^j = \{\mathcal{N}_1^j,\cdots,\mathcal{N}_l^j,\cdots,\mathcal{N}_L^j\}$. The l-th channel group in $\mathcal{N}^j$ is allocated to the l-th device group in $\mathcal{U}^j(t)$. Second, both

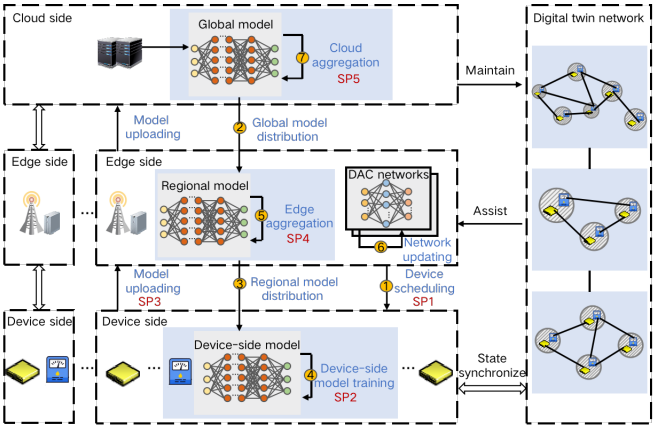

Fig. 2. The framework of C^3-FLOW.

devices and channels establish their matching preferences. The preference of u_i^j towards c_n^j is defined as $\varpi_i^j(n)$, which is obtained by solving the following power control subproblem

$$\tilde{\mathbf{SP3}}: \varpi_{i\mapsto n}^j(t) = \max_{P_i^j(t)} \left\{ \frac{-\gamma_\tau S(\boldsymbol{\omega}_i^j(t)) - \gamma_i^j P_i^j(t) S(\boldsymbol{\omega}_i^j(t))}{B^U \log_2\left(1+\frac{P_i^j(t)\tilde{h}_{i,n}^j(t)}{\tilde{I}_i^{j,U}(t)+B^U N_0}\right)} - D_i^j q_{i,n}^j(t)\left[\xi_1 F(\boldsymbol{\omega}(t-1)+\frac{\xi_1}{g(g-1)G(\eta,T_0)}+\xi_2\right]\right\},$$
$$\text{s.t. } \tilde{C}_3. \tag{24}$$

Based on the objective of **SP3**, the preference of c_n^j towards u_i^j is defined as $\varpi_{n\mapsto i}^j(t) = -\Lambda^j(t)$.

The definition of swap matching is introduced as follows.

Definition 1. *Given a matching set $\boldsymbol{\eta}^j$ and two device-channel pairs $(i,n),(v,l) \in \boldsymbol{\eta}^j$, if they satisfy*

$$\varpi_{i\mapsto l}^j(t) \geq \varpi_{i\mapsto n}^j(t) \text{ and } \varpi_{v\mapsto n}^j(t) \geq \varpi_{v\mapsto l}^j(t),$$
$$\varpi_{n\mapsto v}^j(t) > \varpi_{n\mapsto i}^j(t) \text{ and } \varpi_{l\mapsto i}^j(t) > \varpi_{n\mapsto v}^j(t), \tag{25}$$

$(\boldsymbol{\eta}^j)_{il}^{vn} = \{\boldsymbol{\eta^j} \setminus (i,n),(v,l)\} \cup \{(i,l),(v,n)\}$ is defined as a swap matching of $(\boldsymbol{\eta}^j)_{in}^{vl}$, and $(\boldsymbol{\eta}^j)_{il}^{vn} \succ (\boldsymbol{\eta}^j)_{in}^{vl}$.

Third, the devices and channels in each group perform the swap matching. Given a matching set $\boldsymbol{\eta}^j$ and two device-channel pairs $(i,n),(v,l) \in \boldsymbol{\eta}^j$, if there exists a swap matching $(\boldsymbol{\eta}^j)_{il}^{vn} \succ (\boldsymbol{\eta}^j)_{in}^{vl}$, replace $(\boldsymbol{\eta}^j)_{in}^{vl}$ with $(\boldsymbol{\eta}^j)_{il}^{vn}$. Otherwise, $\boldsymbol{\eta}^j$ remains unchanged. Continue this process until no swap matching exists. Finally, obtain $\boldsymbol{r}^j(t)$ based on the final $\boldsymbol{\eta}^j$.

D. Edge-side and Cloud-side Computational Resource Allocation

The edge-side and cloud-side computational resource allocation subproblems **SP4** and **SP5** are formulated as

$$\mathbf{SP4}: \min_{f_G^j(t)} \gamma_\tau \tau_G^j(t) + \gamma_G E_G^j(t),$$
$$\text{s.t. } \hat{C}_4: f_G^j(t) \leq f_{G,max}^j(t), \tag{26}$$
$$\mathbf{SP5}: \min_{f_C(g)} \gamma_\tau \tau_C(g) + \gamma_C E_C(g),$$
$$\text{s.t. } \check{C}_5: f_C(g) \leq f_{C,max}(g). \tag{27}$$

SP4 and **SP5** are convex and the optimal strategies are given by

$$f_G^{j,*}(t) = \min\left\{\sqrt[3]{\frac{\gamma_\tau}{2\alpha_G\gamma_G}}, f_{G,max}^j(t)\right\}, \tag{28}$$
$$f_C^*(g) = \min\left\{\sqrt[3]{\frac{\gamma_\tau}{2\alpha_C\gamma_C}}, f_{C,max}(g)\right\}. \tag{29}$$

E. C^3-FLOW

Algorithm 1 C^3-FLOW

Input: $\{\boldsymbol{S}^j(t)\}$, $\{\mathcal{N}^j\}$, $\{\tilde{f}_{i,max}^j(t)\}$, $\{f_{G,max}^j(t)\}$, $\{f_{C,max}(g)\}$, $\{\tilde{I}_i^{j,U}(t)\}$, $\{\tilde{h}_{i,n}^j(t)\}$.
Output: $\{\boldsymbol{a}^j\}, \{\boldsymbol{f}^j\}, \{\boldsymbol{r}^j\}, \{\boldsymbol{P}^j\}, \{\boldsymbol{f}^j\}, \{\boldsymbol{f_G^j}\}, \{\boldsymbol{f_C}\}, \boldsymbol{\omega}$.
For $g = 1, \cdots, G$ **do**
  Phase 1: Global Model Distribution
  Distribute $\omega(g-1)$ to edge servers.
  For $t = (g-1)T_0+1, \cdots, gT_0$ **do**
    For $j = 1, \cdots, J$ **do**
    Phase 2: Device Scheduling
    Obtain $\mathcal{U}^j(t)$ as (19) and $\boldsymbol{a}^j(t)$ as (20).
    Phase 3: Regional Model Distribution
    Distribute $\boldsymbol{\omega}^j(t-1)$ to u_i^j, $\forall u_i^j \in \mathcal{U}^j(t)$.
    Phase 4: Device-Side Model Training
    Solve **SP2** to obtain $\boldsymbol{f}^j(t)$, and perform device-side model training.
    Phase 5: Edge Aggregation and DAC Network Updating
    Solve **SP3** to obtain $\boldsymbol{r}^j(t)$ and $\boldsymbol{P}^j(t)$ based on the proposed group-swap-matching-based algorithm.
    Solve **SP4** to obtain $f_G^j(t)$ as (28), and perform edge aggregation as (6).
    Calculate $\Psi^j(t)$ as (18).
    Update $\boldsymbol{\theta}^j(t+1)$ and $\boldsymbol{\gamma}^j(t+1)$ based on gradient descent.
    end for
  end for
  Phase 6: Cloud Aggregation
  Solve **SP5** to obtain $f_C(g)$ as (29), and perform cloud aggregation as (8).
end for

The framework of C^3-FLOW is shown in Fig. 2, which consists of six phases, i.e., global model distribution, device scheduling, regional model distribution, device-side model training, edge aggregation and DAC network updating, and cloud aggregation. The implementation procedures of C^3-FLOW are presented in Algorithm 1.

First, at the beginning of each epoch, i.e., $t = (g-1)T_0+1$, the cloud server distributes the global model $\omega(g-1)$ to edge servers. Second, each edge server, e.g., s^j, obtains the set of scheduled devices $\mathcal{U}^j(t)$ and device scheduling strategy $\boldsymbol{a}^j(t)$ as (19) and (20). Third, in each slot t, each edge server, e.g., s^j, distributes $\boldsymbol{\omega}^j(t-1)$ to the scheduled devices, e.g., u_i^j, $\forall u_i^j \in \mathcal{U}^j(t)$. Fourth, s^j obtains $\boldsymbol{f}^j$ by solving **SP2** with Lagrange optimization, and each $u_i^j \in \mathcal{U}^j(t)$ performs device-side model training. Fifth, s^j obtains $f_G^j(t)$ as (28) and performs edge

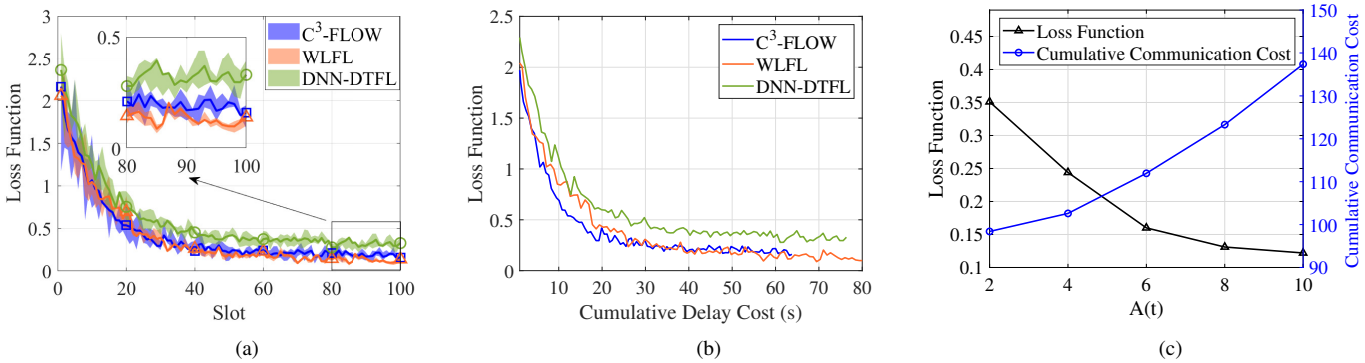

Fig. 3. Reliability of C^3-FLOW: (a) Loss function; (b) Loss function versus cumulative delay cost; (c) The tradeoff between loss function and cumulative communication cost.

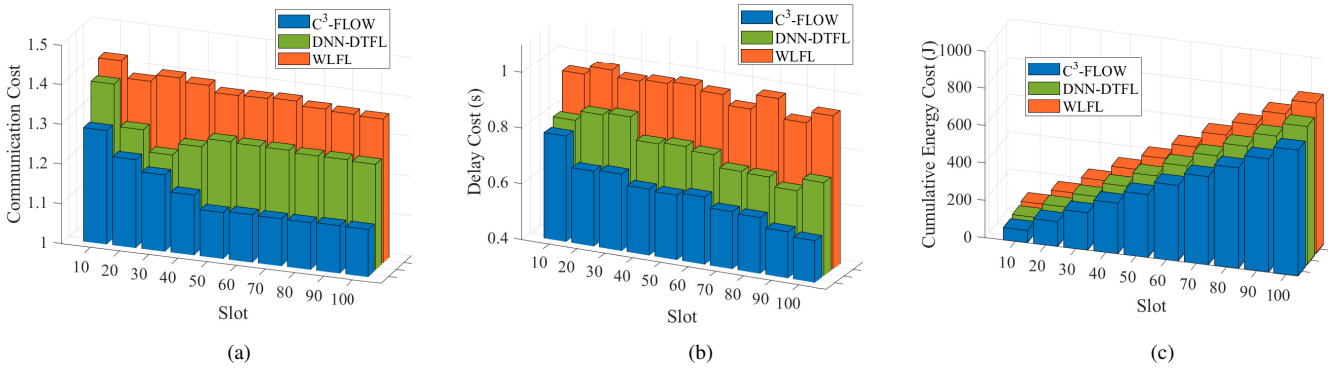

Fig. 4. Communication performance of C^3-FLOW: (a) Communication cost; (b) Delay cost; (c) Cumulative energy cost.

aggregation as (6). Then, s^j calculates the cost function $\Psi^j(t)$ as (18), and updates the DAC networks $\boldsymbol{\theta}^j(t+1)$ and $\boldsymbol{\gamma}^j(t+1)$ based on gradient descent. Finally, at the end of each epoch, i.e., $t = gT_0$, the cloud server obtains $f_C(g)$ as (29), and performs cloud aggregation as (8). Repeat the previous phases until $g > G$.

TABLE I
SIMULATION PARAMETERS

Parameter	Value	Parameter	Value
T	100	J, N^j	$3, 20$
D_i^j	500	N_0	-180 dBm/Hz
S	5 Kbits	$\alpha_i^j, \alpha_C, \alpha_G^j$	10^{-27} Watt·s^3/cycle^3
$f_{G,max}^j(t)$	$[20, 30]$ GHz	$f_{i,max}^j(t)$	$[1.5, 2]$ GHz
$f_{C,max}(t)$	$[50, 100]$ GHz	$A^j(t)$	6
C	0.023dB	P_i^j	$[0.1 : 0.1 : 0.5]$ W
B^U	0.1 MHz	γ_τ	1
δ	10^6 cycles/sample	$\gamma_i^j, \gamma_G, \gamma_C$	2×10^{-3}
ι	10^4 cycles/bit	η	0.001

V. SIMULATION RESULTS

We consider an electric equipment management scenario in a low-carbon smart industrial park, which consists of three disjoint regions. In each region, there is an edge server co-located with a BS to provide coverage and data processing. Devices are evenly distributed within the region. MNIST dataset [19] is utilized to evaluate the reliability and communication efficiency performances, while the vehicle-to-grid (V2G) dataset [20] is utilized to evaluate the energy scheduling performance. The simulation parameters are summarized in Table I [16], [21]. Two state-of-the-art algorithms are compared. The first one is the wireless FL algorithm (WLFL) [16], which considers EMI in loss function and jointly optimizes power control and channel allocation. DT assistance is not considered in WLFL. The second one is the DNN-based DT-assisted user and resource scheduling algorithm for FL (DNN-DTFL) [22], which minimizes the communication cost by optimizing power control, device-side computational resource allocation and device scheduling. The impact of EMI on loss function is ignored. Both WLFL and DNN-DTFL have not considered the cloud-edge-device collaboration.

Figs. 3(a)-(b) show the loss function versus slots and cumulative delay cost. The shaded bar represents the loss function fluctuation and the solid line represents the average value. WLFL performs best in loss function but it has a large communication cost as shown in Fig. 4(a). When $t = 100$, WLFL reduces the loss function by 13.45% at the cost of 14.71% communication cost increment. Fig. 3(b) indicates that C^3-FLOW achieves the minimum loss function under the same cumulative delay cost. Given a cumulative delay cost of 40s, C^3-FLOW outperforms WLFL and DNN-DTFL in loss function by 9.78% and 49.28%, respectively. The reason is that C^3-FLOW leverages DT-assisted power control and channel allocation as well as CRC to improve the DT reliability, and facilitates cloud-edge-device collaborative resource allocation to reduce communication cost.

Fig. 3(c) depicts the tradeoff between loss function and communication cost versus the maximum number of scheduled

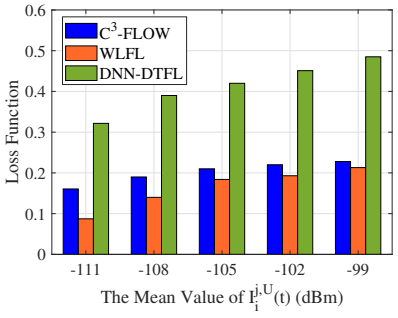

Fig. 5. Loss function versus EMI power.

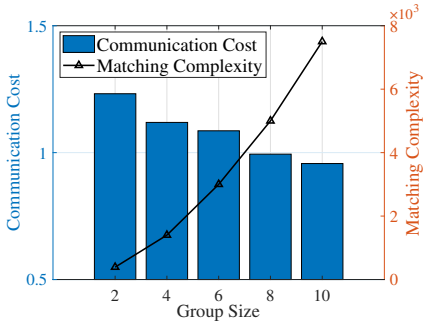

Fig. 6. The tradeoff between communication cost and matching complexity.

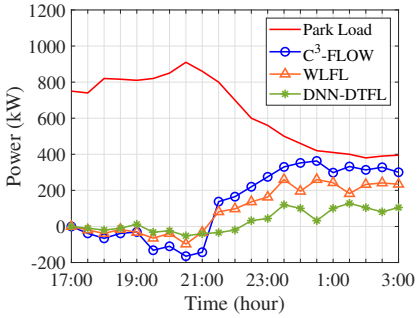

Fig. 7. Energy scheduling performance based on equipment life-cycle management.

TABLE II
CARBON EMISSION REDUCTION

Time	17 : 00 ∼ 18 : 00	18 : 00 ∼ 19 : 00	19 : 00 ∼ 20 : 00	20 : 00 ∼ 21 : 00
C^3-FLOW	13.74 kg	37.29 kg	58.09 kg	89.13 kg
WLFL	11.78 kg	21.69kg	38.29kg	53.58kg
DNN-DTFL	9.81 kg	12.77kg	18.27 kg	29.83kg

devices $A^j(t)$. When $A^j(t)$ increases from 2 to 10, the loss function decreases by 65.24% while the communication cost increases by 28.38%. The reason is that C^3-FLOW can well exploit the diversity gain provided by $A^j(t)$ increment to reduce loss function.

Figs. 4(a)-(c) show the communication cost, delay cost, and cumulative energy cost versus slots. Compared with WLFL, C^3-FLOW reduces the communication cost, delay cost, and cumulative energy cost by 14.71%, 32.48%, and 19.41%. The reasons are two folds. On one hand, C^3-FLOW facilitates the cloud-edge-device collaborative resource allocation, thereby making the delay cost substantially decreasing. On the other hand, C^3-FLOW exploits the key information estimated by DT to reduce delay through device scheduling, i.e., a few straggler devices with less computational resources and inferior channel gains are avoided. Compared with DNN-DTFL, C^3-FLOW reduces the communication cost, delay cost, and cumulative energy cost by 8.59%, 19.5%, and 8.56% since the resources of edge and cloud layers are not collaboratively allocated with that of device layer. Besides, device mapping and constrained scheduling are not leveraged to reduce communication cost.

Fig. 5 shows the loss function versus the mean value of EMI power $I_i^{j,U}(t)$. When $I_i^{j,U}(t)$ increases from -111dBm to -99dBm, compared with WLFL and DNN-DTFL, C^3-FLOW reduces the loss function increment by 49.14% and 52.99% by dynamically adjusting device scheduling to avoid devices with inferior channel gains and jointly optimizing power control and channel allocation to reduce PER.

Fig. 6 depicts the tradeoff between communication cost and matching complexity versus group size. With group size decreasing from 10 to 2, matching complexity decreases by 94.28% and communication cost increases by 22.32%. The proposed group-swap-matching-based algorithm efficiently solves dimensionality curse with limited communication cost increment.

Fig. 7 and Table II shows the energy scheduling and carbon emission reduction performances. The red line represents the load profile. The positive power represents charging vehicles to absorb renewable energy, and the negative power represents discharging vehicles to increase power supply. Compared with WLFL and DNN-DTFL, C^3-FLOW reduces the carbon emission by 36.78% and 64.35% during the peak time, i.e., 17 : 00–21 : 00. During the off-peak time, i.e., 23 : 00–3 : 00, C^3-FLOW increases renewable energy absorption by 29.41% and 69.78%. Numerical results verify that C^3-FLOW can achieve intelligent energy scheduling through proactive peak-shaving and improved renewable energy utilization.

VI. CONCLUSIONS

In this paper, we addressed the problem of DT unreliability and low communication efficiency for low-carbon electric equipment management. C^3-FLOW was proposed to achieve reliable and communication-efficient DT-assisted cloud-edge-device collaborative resource allocation. Compared with WLFL and DNN-DTFL, C^3-FLOW reduces loss function by 9.78% and 49.28% and reduces communication cost by 8.59% and 14.71%. Besides, the V2G energy scheduling case verifies that C^3-FLOW can reduce the carbon emission by 36.78% and 64.35% during the peak time, and increase renewable energy absorption by 29.41% and 69.78% during the off-peak time. In the future, the heterogeneities of device-side communication and computational resources will be studied to further improve the electrical equipment management performance.

REFERENCES

[1] J. Pan *et al.*, "An Internet of Things framework for smart energy in buildings: Designs, prototype, and experiments," *IEEE Internet of Things J.*, vol. 2, no. 6, pp. 527–537, Dec. 2015.

[2] J. Qi, L. Liu, Z. Shen, B. Xu, K.-S. Leung, and Y. Sun, "Low-carbon community adaptive energy management optimization toward smart services," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3587–3596, Oct. 2020.

[3] Z. Yang, Y. Fang, G. Han, and K. M. S. Huq, "Spatially coupled protograph LDPC-coded hierarchical modulated BICM-ID systems: A promising transmission technique for 6G-enabled internet of things," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5149–5163, 2021.

[4] H. Zhou, X. Wang, Z. Liu, Y. Ji, and S. Yamada, "Resource allocation for SVC streaming over cooperative vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 9, pp. 7924–7936, Jun. 2018.

[5] S. Yu, X. Chen, Z. Zhou, X. Gong, and D. Wu, "When deep reinforcement learning meets federated learning: Intelligent multitimescale resource management for multiaccess edge computing in 5G ultradense network," *IEEE Internet of Things J.*, vol. 8, no. 4, pp. 2238–2251, Feb. 2021.

[6] J. Yao and N. Ansari, "Enhancing federated learning in fog-aided IoT by CPU frequency and wireless power control," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3438–3445, Sep. 2021.

[7] Y. Gao, X. Liu, J. Li, Z. Fang, X. Jiang, and K. M. S. Huq, "LFT-Net: Local feature transformer network for point clouds analysis," *IEEE Trans Intell. Transp. Syst.*, vol. 6, no. 3, pp. 1–11, 2022.

[8] H. Zhou, Y. Ji, X. Wang, and S. Yamada, "EICIC configuration algorithm with service scalability in heterogeneous cellular networks," *IEEE/ACM Trans. Networking*, vol. 25, no. 1, pp. 520–535, Jul. 2017.

[9] W. Shi, S. Zhou, Z. Niu, J. Miao, and G. Lu, "Joint device scheduling and resource allocation for latency constrained wireless federated learning," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 453–467, Jan. 2021.

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

[11] L. Liu, J. Zhang, S. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dublin, Ireland, Jul. 2020, pp. 1–6.

[12] W. Sun, N. Xu, L. Wang, H. Zhang, and Y. Zhang, "Dynamic digital twin and federated learning with incentives for air-ground networks," *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 1, pp. 321–333, Jan. 2022.

[13] L. Zhao *et al.*, "A novel prediction-based temporal graph routing algorithm for software-defined vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. PP, no. 99, pp. 1–16, Nov. 2021.

[14] L. Zhao, T. Zheng, M. Lin, A. Hawbani, J. Shang, and C. Fan, "SPIDER: A social computing inspired predictive routing scheme for softwarized vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. PP, no. 99, pp. 1–12, Oct. 2021.

[15] Y. Xi, A. Burr, J. Wei, and D. Grace, "A general upper bound to evaluate packet error rate over quasi-static fading channels," *IEEE Trans. Wireless Commun.*, vol. 10, no. 5, pp. 1373–1377, Jan. 2011.

[16] M. Chen *et al.*, "A joint learning and communications framework for federated learning over wireless networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 269–283, Oct. 2021.

[17] G. Cui, Y. Long, L. Xu, and W. Wang, "Joint offloading and resource allocation for satellite assisted vehicle-to-vehicle communication," *IEEE Syst. J.*, vol. 15, no. 3, pp. 3958–3969, Sept. 2021.

[18] L. Zhao, H. Chai, Y. Han, K. Yu, and S. Mumtaz, "A collaborative V2X data correction method for road safety," *IEEE Trans. Reliab.*, vol. 71, no. 2, pp. 951–962, Jun. 2022.

[19] V.-D. Nguyen, S. K. Sharma, T. X. Vu, S. Chatzinotas, and B. Ottersten, "Efficient federated learning algorithm for resource allocation in wireless IoT networks," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3394–3409, Sept. 2021.

[20] Z. Zhou *et al.*, "Robust energy scheduling in vehicle-to-grid networks," *IEEE Netw.*, vol. 31, no. 2, pp. 30–37, Mar. 2017.

[21] W. Xia *et al.*, "Multi-armed bandit-based client scheduling for federated learning," *IEEE Trans. Wireless Commun.*, vol. 19, no. 11, pp. 7108–7123, Nov. 2020.

[22] Y. Lu, X. Huang, K. Zhang, S.Maharjan, and Y. Zhang, "Communication-efficient federated learning and permissioned blockchain for digital twin edge networks," *IEEE Internet Things J.*, vol. 8, no. 4, pp. 2276–2288, 2021.

Haijun Liao is working toward the Ph.D degree in Electrical Engineering at School of Electrical and Electronic Engineering, North China Electric Power University, China. She was the recipient of the IEEE IWCMC 2019 Best Paper Award, the IEEE VTC-2020 Spring Best Student Paper Award, and the IEEE CAMAD 2021 Best Paper Award. Her research interests mainly focus on cloud-edge-end collaboration in smart grid communications and Internet of Things (IoT).

Zhenyu Zhou (Senior Member, IEEE) received the M.E. and Ph.D. degrees in international information and communication studies from Waseda University, Tokyo, Japan, in 2008 and 2011, respectively.

From September 2012 to April 2019, he was an Associate Professor with the School of Electrical and Electronic Engineering, North China Electric Power University, China, where he has been a Full Professor since April 2019. His research interests mainly focus on resource allocation in device-to-device communications, machine-to-machine communications, smart grid communications, and Internet of Things. He was a recipient of the IET Premium Award in 2017, the IEEE Globecom 2018 Best Paper Award, the IEEE International Wireless Communications and Mobile Computing Conference 2019 Best Paper Award, and the IEEE Communications Society Asia–Pacific Board Outstanding Young Researcher. He served as an Associate Editor for IEEE Internet of Things Journal, IET Quantum Communication, IEEE Access, EURASIP Journal on Wireless Communications and Networking and a Guest Editor for IEEE Communications Magazine, IEEE Transactions on Industrial Informatics, and Transactions on Emerging Telecommunications Technologies. He is a Senior Member of the Chinese Institute of Electronics and the China Institute of Communications.

Nian Liu (Member, IEEE) received the B.S. and M.S. degrees in electric engineering from Xiangtan University, Hunan, China, in 2003 and 2006, respectively, and the Ph.D. degree in electrical engineering from North China Electric Power University, Beijing, China, in 2009.

He is currently a Professor with the School of Electrical and Electronic Engineering, North China Electric Power University, and is also the leader of research direction 4, multi-information fusion and integrated energy system optimization of the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, and is a member of the Standardization Committee of Power Supply and Consumption in Power Industry of China. He was a Visiting Research Fellow with RMIT University, Melbourne, Australia, from 2015 to 2016. He has authored or coauthored more than 160 journal and conference publications and has been granted for more than ten patents of China. His research interests include multienergy system integration, microgrids, cyber-physical energy system, and renewable energy integration. Prof. Liu was the recipient of the 2020 Highly Cited Chinese Researcher (Elsevier). He is an Editor of the IEEE TRANSACTIONS ON SMART GRID, IEEE TRANSACTIONS ON SUSTAINABLE ENERGY, IEEE POWER ENGINEERING LETTERS, and Journal of Modern Power Systems and Clean Energy.

Yan Zhang (Fellow, IEEE) received the Ph.D. degree in electrical and electronic engineering from the School of Electrical and Electronics Engineering, Nanyang Technological University, Singapore, in 2005.

He is currently a Full Professor with the Department of Informatics, University of Oslo, Oslo, Norway. His research interests include next-generation wireless networks leading to 5G beyond/6G, green, and secure cyber-physical systems (e.g., smart grid and transport).

Prof. Zhang is a Fellow of IET, an elected member of Academia Europaea, and an elected member of Norwegian Academy of Technological Sciences. He is an Editor (or Area Editor, Associate Editor) for several IEEE publications, including IEEE Communications Magazine, IEEE Network Magazine, IEEE TRANSACTIONS ON NETWORK SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE COMMUNICATIONS SURVEY AND TUTORIALS, IEEE INTERNET OF THINGS JOURNAL, IEEE SYSTEMS JOURNAL, IEEE Vehicular Technology Magazine, and IEEE BLOCKCHAIN TECHNICAL BRIEFS. He is a symposium/track chair in a number of conferences, including IEEE ICC 2021, IEEE Globecom 2017, IEEE PIMRC 2016, and IEEE SmartGridComm 2015. He is an IEEE Vehicular Technology Society Distinguished Lecturer during 2016–2020 and he is named as CCF 2019 Distinguished Speaker. He is the Chair of IEEE Communications Society Technical Committee on Green Communications and Computing. He is an elected member of CCF Technical Committee of Blockchain. In both 2018 and 2019, he was a recipient of the global Highly Cited Researcher Award.

Guangyuan Xu received the B.S. degree in Electrical Engineering and Automation from Beijing Institute of Petro-Chemical Technology in 2004 and received the M.E. degree in Electrical Machines and Apparatus from North China Electric Power University in 2007, respectively. He is currently a Senior Engineer with Chuzhou Power Supply Company of Smart Grid Anhui Electric Power Co. Ltd., Anhui, China. His research interest includes information and communication technology.

Zhenti Wang received the B.S. degree in Computer Science and Technology from University of Science and Technology of China in 1997 and received the M.E. degree in Computer Science and Technology from Southeast University in 2005, respectively. He is currently a Senior Engineer with Chuzhou Power Supply Company of Smart Grid Anhui Electric Power Co. Ltd., Anhui, China. His research interest includes information and communication technology.

Shahid Mumtaz (Senior Member, IEEE) received the M.Sc. degree in electrical and electronic engineering from the Blekinge Institute of Technology, Karlskrona, Sweden, in 2006, and the Ph.D. degree in electrical and electronic engineering from the University of Aveiro, Portugal, in 2011. He is currently a Senior Research Scientist with the Instituto de Telecomunicações, Aveiro, Portugal. Prior to his current position, he was a Research Intern at Ericsson and Huawei Research Labs, Karlskrona, Sweden, in 2005. His research interests mainly focus on wireless communication, and Internet of Things. He was a recipient of the Alain Bensoussan Fellowship by ERCIM to pursue research in communication networks for one year at the VTT Technical Research Centre of Finland in 2012. He was nominated as the Vice Chair for the IEEE new standardization on P1932.1: Standard for Licensed/Unlicensed Spectrum Interoperability in Wireless Mobile Networks.