frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 07 August 2024
DOI 10.3389/fenrg.2024.1440487

Check for updates

OPEN ACCESS

EDITED BY
Zhengmao Li,
Aalto University, Finland

REVIEWED BY
Gaofeng Cui,
Beijing University of Posts and Telecommunications (BUPT), China
Muhammad Tariq,
National University of Computer and Emerging Sciences, Pakistan
Zhi Liu,
Shizuoka University, Japan

*CORRESPONDENCE
Tao Lu,
Tao_lu1108@163.com

RECEIVED 29 May 2024
ACCEPTED 01 July 2024
PUBLISHED 07 August 2024

CITATION
Dang S, Zhang J, Lu T, Zhang Y, Song P, Zhang J and Liu R (2024), Cloud-edge collaborative high-frequency acquisition data processing for distribution network resilience improvement.
Front. Energy Res. 12:1440487.
doi: 10.3389/fenrg.2024.1440487

Cloud-edge collaborative high-frequency acquisition data processing for distribution network resilience improvement

Sanlei Dang, Jie Zhang, Tao Lu*, Yongwang Zhang, Peng Song, Jun Zhang and Rirong Liu

Metrology Center, Guangdong Power Grid Co., Ltd., Qingyuan, China

To realize transparent monitoring and resilience improvement of low-voltage distribution network, both the data acquisition scope and frequency have been greatly expanded. Cloud-edge collaboration leverages the edge server's real-time response capabilities and the cloud server's robust data processing power to enhance the performance of high-frequency data acquisition processing. Nonetheless, it continues to confront challenges such as the entanglement of optimization variables, the presence of uncertain information, and a lack of awareness regarding acquisition frequencies. In this paper, we propose a machine learning-based cloud-edge collaborative data processing optimization algorithm to minimize the weighted sum of data processing delay and device energy consumption for distribution network resilience improvement. The joint optimization problem is decoupled into device-edge data offloading subproblem and edge-cloud data splitting subproblem, which are solved by the proposed upper confidence bound (UCB) based frequency-aware device-edge data offloading optimization algorithm and the exponential-weight algorithm for exploration and exploitation (EXP3) based edge-cloud data splitting optimization algorithm, respectively. Simulation results show that the proposed algorithm is superior to existing algorithms in performances of energy consumption and total processing delay.

KEYWORDS

distribution network resilience improvement, edge-cloud collaboration, data offloading, data splitting, high-frequency acquisition

1 Introduction

With a high proportion of unstable distributed renewable sources, energy storage, and controllable loads connected to the low-voltage distribution network, its transparent monitoring and resilience improvement have become an indispensable requirement Zhou et al. (2022); Yang et al. (2023); Ding et al. (2024); Wang et al. (2018); Chen et al. (2021). A large number of high-frequency acquisition devices need to be deployed in the low-voltage distribution network to collect multi-dimensional operation data such as voltage and current to support continuous monitoring, unmanned control, and fault detection, improving the resilience of distribution network operation Shah et al. (2020); Li et al. (2023); Soltani et al. (2023); Tariq et al. (2020). Compared with conventional devices, both the data acquisition scope and frequency have been greatly expanded. However, due to the limited computation and energy resources of devices, it is difficult to satisfy the stringent and differentiated data processing requirements of electric services Liao et al. (2020); Liu and Cao (2021); Li et al. (2024b), Li et al. (2024c).

Cloud-edge collaboration is a new convergent distributed computing paradigm, which combines the advantages of edge computing and cloud computing Laili et al. (2023); Jiang et al. (2023); Zhang et al. (2021). High-frequency acquisition devices offload the collected data to either edge server or offload the data to cloud server for remote processing. The real-time response capability of edge server and the large data processing capability of cloud server are integrated to improve data processing performance Gao et al. (2022); Dong et al. (2021); Guo et al. (2020); Naeem et al. (2021). However, the application of cloud-edge collaborative high-frequency acquisition data processing for distribution network resilience improvement still faces several challenges.

First, cloud-edge collaborative data processing involves the joint optimization of transmission power selection, edge server selection, and data splitting Long et al. (2023); Wu et al. (2020); Lin et al. (2024). The coupling relationship among optimization variables causes difficulties in solving the joint optimization problem. Second, traditional optimization methods are based on the global state information (GSI), while it is impractical to obtain complete GSI in real-world applications Zhang et al. (2022); Wang et al. (2022); Zhang et al. (2022). Uncertain GSI leads to large deviations in the optimization of cloud-edge collaborative data processing decisions. Last but not least, the data processing performance is affected by the frequency of data acquisition. The data processing optimization without the consideration of acquisition frequency cannot satisfy differentiated data processing requirements of high-frequency acquisition, which degrades the optimization performance Xiao et al. (2022); Cui et al. (2021).

Currently, some works have explored data processing for the distribution network. In Xia et al. (2022), Xia *et al.* proposed a data processing algorithm based on the Lyapunov optimization framework and the Markov approximation method, the objective of which is to minimize the long-term energy cost while meeting the real-time data processing constraint. However, the above study does not consider the joint optimization of edge server selection, data splitting, and device data transmission power control. In Mu et al. (2019), Mu *et al.* proposed a data processing method based on the centralized Kuhn-Munkers algorithm for a binary integer linear programming problem, the objective of which is to guarantee the network stability and improve energy saving. In Li et al. (2024a), *Li et al.* proposed a data processing method based on the three-dimensional learning-matching-based joint selection algorithm of server and container, the objective of which is to reduce the delay of high-priority service. However, the above studies do not consider how to make device data processing decisions under uncertain GSI. In Zhang et al. (2022), Zhang *et al.* proposed a data processing method based on convolutional neural networks and mathematical methods to solve the problems of sampling period anomalies, sampling reference time anomalies, data noise, and data missing in low-voltage distribution substation area. However, it does not consider the joint optimization of high-frequency acquisition device energy consumption and data processing delay.

Motivated by the above challenges, we propose a machine learning-based cloud-edge collaborative data processing optimization algorithm to minimize the weighted sum of data processing delay and device energy consumption for distribution network resilience improvement. First, we formulate a joint optimization problem of transmission power selection, edge server selection, and data splitting under cloud-edge collaboration. Second, the joint optimization problem is decoupled into device-edge data offloading subproblem and edge-cloud data splitting subproblem and solved by machine learning-based cloud-edge collaborative data processing optimization algorithm. Specifically, devices and edge servers can learn the optimal data offloading and data splitting strategy by upper confidence bound (UCB) based frequency-aware device-edge data offloading optimization and exponential-weight algorithm for exploration and exploitation (EXP3) based edge-cloud data splitting optimization, respectively. Finally, the effectiveness is verified through simulations. The main contributions of this paper are summarized as follows.

- Two-stage joint optimization of transmission power selection, edge server selection, and data splitting under cloud-edge collaboration: We decompose the joint optimization of transmission power selection, edge server selection, and data splitting into two stages. In the first stage, the selection of transmission power and edge server is realized through UCB-based frequency-aware device-edge data offloading optimization algorithm, and in the second stage, the selection of data splitting ratio is realized through EXP3-based edge-cloud data splitting optimization algorithm.
- Cloud-edge collaborative high-frequency acquisition data processing under uncertain GSI: We model the cloud-edge collaborative high-frequency acquisition data processing problem as a multi-armed bandit (MAB) problem and propose a machine learning-based cloud-edge collaborative data processing optimization algorithm, which introduces the acquisition frequency weight into the confidence upper bound calculation formula to achieve frequency awareness. The proposed algorithm optimizes device data offloading and edge-cloud data splitting through historical observations of cloud-edge collaborative processing delay and device energy consumption, integrating edge server and cloud server for collaborative computing.
- Low-delay and low-energy consumption data processing: We formulate the optimization objective as the weighted sum of total data processing delay and energy consumption to achieve simultaneous reduction of delay and energy consumption. The proposed algorithm dynamically selects transmission power, edge server, and data splitting ratio based on the historical observation of weighted sum performance.

This paper is structured as follows. Section 2 formulates the system model and the cloud-edge data processing problem. The proposed machine learning-based cloud-edge collaborative data processing optimization algorithm is presented in Section 3. Simulation results are provided in Section 4. Section 5 concludes this paper.

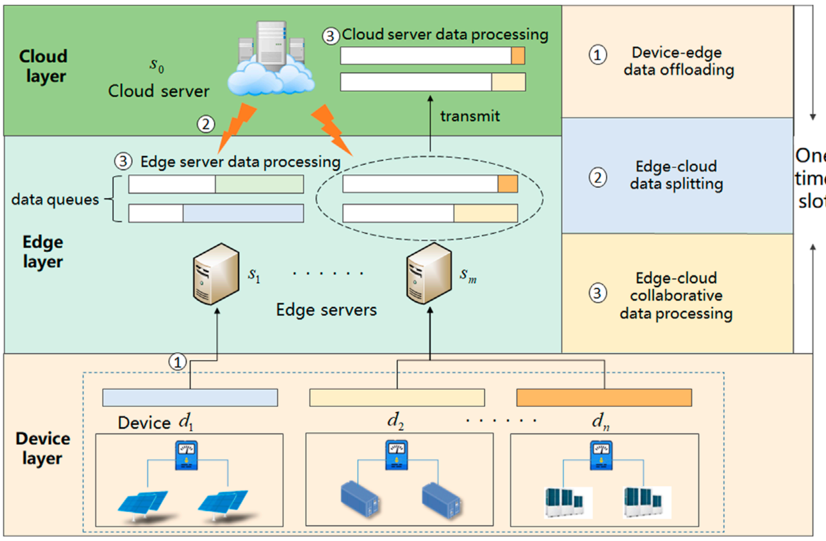

FIGURE 1
Cloud-edge collaborative high-frequency acquisition data processing architecture for distribution network resilience improvement.

2 System model

As shown in Figure 1, we consider a cloud-edge collaborative high-frequency acquisition data processing architecture for distribution network resilience improvement, which consists of the device layer, the edge layer, and the cloud layer. In the device layer, the information acquisition devices are deployed on distributed electrical equipment such as photovoltaic and distributed energy storage to collect data to support different services. There exists N devices. The set of which is denoted as $\mathcal{D} = \{d_1, \ldots, d_n, \ldots, d_N\}$. The edge layer and the cloud layer consist of M edge servers and one cloud server, the set of edge servers is denoted as $\mathcal{S} = \{s_1, \ldots, s_m, \ldots, s_M\}$ and the cloud server is denoted as s_0. Firstly, devices offload data to edge servers via 5G to reduce the data processing delay. Then, edge servers split the data offloaded from devices and transmit them to the cloud server via optical fiber communication to relieve data processing stress caused by device data growth. Through edge-cloud collaborative data processing, the processing requirements of high-frequency power distribution information acquisition can be met.

The total optimization period is divided into T time slots, and the set is $\mathcal{T} = \{1, \ldots, t, \ldots, T\}$. Each slot includes three stages: device-edge data offloading, edge-cloud data transmission, and edge-cloud collaborative data processing. The maximum time length of each stage is set as τ_1, τ_2, τ_3. Denote $x_{n,m}(t)$ as edge server selection variable, where $x_{n,m}(t) = 1$ represents that device d_n selects edge server s_m for data offloading in the t-th slot, and $x_{n,m}(t) = 0$ otherwise.

2.1 Device-edge data offloading model

The power distribution information acquisition devices have differentiated acquisition frequencies. Devices collect data with different volumes in each slot and offload the data to the selected edge server for data processing. Denoting the amount of data collected at d_n in the t-th slot as $U_n(t)$, the data stored at d_n are modeled as a data backlog queue $Q_n(t)$, which is evolved as Eq. 1:

$$Q_n(t+1) = Q_n(t) + U_n(t) - \sum_{m=1}^{M} x_{n,m}(t)\, U_{n,m}^{D,E}(t), \tag{1}$$

where $U_{n,m}^{D,E}(t)$ represents the amount of data offloaded by d_n to s_m in the t-th slot, which is given by Eq. 2:

$$U_{n,m}^{D,E}(t) = \min\{Q_n(t) + U_n(t), \tau_1 R_{n,m}(t)\}, \tag{2}$$

where $R_{n,m}(t)$ represents the rate of data transmission between d_n and s_m, which is given by Eq. 3:

$$R_{n,m}(t) = B_{n,m}(t)\log_2\left(1 + \frac{P_n(t) g_{n,m}(t)}{\delta_0 + I_{n,m}(t)}\right), \tag{3}$$

where $B_{n,m}(t)$ and $g_{n,m}(t)$ represent the bandwidth and channel gain for data transmission between d_n and s_m, respectively. δ_0 and $I_{n,m}(t)$ represent Gaussian white noise and electromagnetic interference power between d_n and s_m, respectively. $P_n(t) \in \mathcal{P}_n$ represents transmission power of d_n. $\mathcal{P}_n$ is the transmission power set, which contains L levels and can be expressed by Eq. 4:

$$\mathcal{P}_n = \left\{P_{n,min}, \ldots, P_{n,min} + \frac{(l-1)(P_{n,max} - P_{n,min})}{L-1}, \ldots, P_{n,max}\right\}, \tag{4}$$

where L is the transmission power level number. $P_{n,min}$ and $P_{n,max}$ represent the minimum transmission power and maximum transmission power of d_n, respectively.

Therefore, the transmission delay of data offloading from d_n to s_m in the t-th slot is calculated as Eq. 5:

$$\tau_{n,m}^{D,E}(t) = \min\left\{\tau_1, \frac{Q_n(t) + U_n(t)}{R_{n,m}(t)}\right\}. \tag{5}$$

The data transmission energy consumption $E_{n,m}(t)$ for data offloading from d_n to s_m in the t-th slot is represented as Eq. 6:

$$E_{n,m}(t) = P_n(t)\,\tau_{n,m}^{D,E}(t). \tag{6}$$

2.2 Edge-cloud data splitting model

Edge server s_m maintains data backlog queue $O_{n,m}^{E}(t)$ for the data offloaded from device d_n, which is dynamically evolved as Eq. 7:

$$O_{n,m}^{E}(t+1) = O_{n,m}^{E}(t) + U_{n,m}^{D,E}(t) - y_{n,m}(t)\,U_{n,m}^{D,E}(t) - U_{n,m}^{E,com}(t), \tag{7}$$

where $U_{n,m}^{E,com}(t)$ is the processed data volume of s_m, whose specific explanation is shown in (11). $y_{n,m}(t) \in \mathcal{Y}$ is data splitting ratio, which represents the data splitting ratio of d_n from s_m to s_0. $\mathcal{Y}$ is the data splitting ratio set. In order to ensure the data integrity while accomplishing ratio-based data splitting, we discretize the data splitting ratio into H levels Chen et al. (2020), which can be expressed as Eq. 8:

$$\mathcal{Y} = \left\{ y_{min}(t), \ldots, y_{min}(t) + \frac{(h-1)(y_{max}-y_{min})}{H-1}, \ldots, y_{max} \right\}. \tag{8}$$

Denote $U_{n,m}^{E,C}(t)$ as the data amount of d_n offloaded from s_m to cloud server s_0 in the t-th slot, which is given by Eq. 9:

$$U_{n,m}^{E,C}(t) = \min\left\{ y_{n,m}(t)\,U_{n,m}^{D,E}(t), \tau_2 R_m^{E,C}(t) \right\}, \tag{9}$$

where $R_m^{E,C}(t)$ represents the transmission rate between s_m and s_0, whose calculation method is as same as (3).

The transmission delay of data transmitted from s_m to s_0 in the t-th slot is calculated as Eq. 10:

$$\tau_{n,m}^{E,C}(t) = \min\left\{ \tau_2, \frac{y_{n,m}(t)\,U_{n,m}^{D,E}(t)}{R_m^{E,C}(t)} \right\}. \tag{10}$$

2.3 Edge-cloud collaborative data processing model

2.3.1 Edge server data processing delay model

Define the data amount of device d_n processed by edge server s_m in the t-th slot as $U_{n,m}^{E,com}(t)$, which is given by Eq. 11:

$$U_{n,m}^{E,com}(t) = \min\left\{ O_{n,m}^{E}(t) + \left(1 - y_{n,m}(t)\right) U_{n,m}^{D,E}(t), \tau_3 \frac{f_{n,m}^{E}(t)}{\chi_n} \right\}, \tag{11}$$

where $f_{n,m}^{E}(t)$ (cycles/s) is the computing capacity of s_m in the t-th slot, and χ_n (cycles/bit) is the data processing complexity of d_n. Therefore, the processing delay of s_m in the t-th slot is given by Eq. 12:

$$\tau_{n,m}^{E,com}(t) = \min\left\{ \tau_3, \chi_n \left[\frac{\left(O_{n,m}^{E}(t) + \left(1 - y_{n,m}(t)\right) U_{n,m}^{D,E}(t)\right)}{f_{n,m}^{E}(t)} \right] \right\}. \tag{12}$$

2.3.2 Cloud server data processing delay model

Cloud server s_0 maintains data backlog queue $Z_{n,m}^{C}(t)$ for the data of device d_n split from edge server s_m, which is dynamically evolved as Eq. 13:

$$Z_{n,m}^{C}(t+1) = Z_{n,m}^{C}(t) + U_{n,m}^{E,C}(t) - U_{n,m}^{C,com}(t), \tag{13}$$

where $U_{n,m}^{C,com}(t)$ is the cloud server processing data amount, which is given by Eq. 14:

$$U_{n,m}^{C,com}(t) = \min\left\{ Z_{n,m}^{C}(t) + U_{n,m}^{E,C}(t), \frac{\left(\tau_3 - \tau_{n,m}^{E,C}(t)\right) f_{n,m}^{C}(t)}{\chi_n} \right\}. \tag{14}$$

The processing delay of d_n's data in s_0 is given by Eq. 15:

$$\tau_{n,m}^{C,com}(t) = \min\left\{ \tau_3 - \tau_{n,m}^{E,C}(t), \frac{\chi_n\left(Z_{n,m}^{C}(t) + U_{n,m}^{E,C}(t)\right)}{f_{n,m}^{C}(t)} \right\}. \tag{15}$$

2.4 Problem formulation

The total delay of high-frequency acquisition data processing consists of the delay of device-edge data offloading, and the maximum value between edge server data processing delay and the sum of edge-cloud data splitting delay and cloud server data processing delay, which is given by Eq. 16:

$$\tau_{n,m}^{sum}(t) = \tau_{n,m}^{D,E}(t) + \max\left\{ \tau_{n,m}^{E,com}(t), \tau_{n,m}^{E,C}(t) + \tau_{n,m}^{C,com}(t) \right\} \tag{16}$$

In this paper, we aim to address the problem of low delay and low energy consumption edge-cloud collaborative high-frequency acquisition data processing for distribution network resilience improvement. The objective is to minimize the weighted sum of data processing delay and device energy consumption through the joint optimization of transmission power selection, edge server selection, and edge-cloud data splitting ratio selection. The joint optimization problem is formulated as Eq. 17:

$$\begin{aligned}
\textbf{P1:} \quad & \min_{\{P_n(t), x_{n,m}(t), y_{n,m}(t)\}} \frac{1}{N} \sum_{n=1}^{N} \sum_{m=1}^{M} x_{n,m}(t)\,\Upsilon_{n,m}(t) \\
\text{s.t. } & C_1: x_{n,m}(t) \in \{0,1\}, \forall d_n \in \mathcal{D}, \forall s_m \in \mathcal{S}, \forall t \in \mathcal{T} \\
& C_2: \sum_{m=1}^{M} x_{n,m}(t) = 1, \forall d_n \in \mathcal{D}, \forall s_m \in \mathcal{S}, \\
& C_3: \sum_{n=1}^{N} x_{n,m}(t) \le num_m^{max}, \forall d_n \in \mathcal{D}, \forall s_m \in \mathcal{S}, \\
& C_4: P_n(t) \in \mathcal{P}, \forall d_n \in \mathcal{D}, \forall s_m \in \mathcal{S}, \\
& C_5: y_{n,m}(t) \in \mathcal{Y}, \forall d_n \in \mathcal{D}, \forall s_m \in \mathcal{S},
\end{aligned} \tag{17}$$

where $\Upsilon_{n,m}(t) = \tau_{n,m}^{sum}(t) + V_E E_{n,m}(t)$, and V_E is the weight of the energy consumption. C_1, C_2 and C_3 are edge server selection constraints, i.e., each device can only select one edge server for data offloading, and the maximum number of devices can be handled by s_m is num_m^{max}. C_4 is the transmission power selection constraint and C_5 is the data splitting ratio selection constraint.

P1 involves the device-edge data offloading optimization and edge-cloud data splitting optimization, which can be solved by a time-sequential manner. Specifically, the device-edge data offloading should be first optimized and edge-cloud data splitting should be optimized based on the data offloading strategy. Therefore, **P1** is decomposed into two subproblems, i.e., **SP1**: device-edge data offloading subproblem involving $P_n(t)$ and $x_{n,m}(t)$; **SP2**: edge-cloud data splitting subproblem involving $y_{n,m}(t)$.

3 Machine learning-based cloud-edge collaborative data processing optimization algorithm

In this section, a machine learning-based cloud-edge collaborative data processing optimization algorithm is proposed to solve the optimization problem. The implementation procedure of the machine learning-based cloud-edge collaborative data processing optimization algorithm is shown in Algorithm 1.

3.1 UCB-based frequency-aware device-edge data offloading optimization

SP1 is formulated as Eq. 18:

$$\begin{aligned} &\mathbf{SP1}: \min_{\{P_n(t), x_{n,m}(t)\}} \frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M} x_{n,m}(t)\Upsilon_{n,m}(t) \\ &\text{s.t. } C_1 \sim C_4. \end{aligned} \tag{18}$$

SP1 optimizes the device-edge data offloading process. Its optimization variables involve $P_n(t)$ and $x_{n,m}(t)$. The device obtains the optimal offloading decision by minimizing the weighted sum of $\tau_{n,m}^{sum}(t)$ and $E_{n,m}(t)$.

However, the precise knowledge of global state information such as channel quality and edge server computing resources is inaccurate. It is difficult for devices to make the optimal offloading decision. Devices should optimize edge server selection and power selection based on the local state information. Multi-armed bandit (MAB) is an effective solution to solve the combinatorial optimization problem with incomplete information Hashima et al. (2020); Zhao et al. (2020). In each slot, the decision maker selects an arm. Then, the selected arm generates a reward. The goal of the decision maker is to maximize the cumulative reward.

We transform **SP1** into an MAB problem. The decision maker, arm, action, and reward are described as follows.

- Decision maker: Define the acquisition devices as the decision maker, which makes the edge server selection and power control decision for data offloading.
- Arm: The power $\mathcal{P}_n$ and edge server $\mathcal{S}$ are combined to reduce the space complexity of action. Define $\mathcal{A}^n = \{A_{1,1}^n, \ldots, A_{l,m}^n, \ldots, A_{L,M}^n\}$ as the set of arms which satisfy $|\mathcal{A}^n| = L \times M$. The arm $A_{l,m}^n$ represents the combination of the power level l and edge server s_m. Define the number of times to select $A_{l,m}^n$ as $r_{l,m}^n(t)$.
- Action: Define the arm selection action indicator variable as $a_{l,m}^n(t)$, $a_{l,m}^n(t) = 1$ represents that device d_n selects

Input: $\mathcal{D}$, $\mathcal{S}$, $\mathcal{T}$, $\mathcal{P}$, $\mathcal{Y}$, N, M, V_E.
Output: $\{x_{n,m}(t)\}$, $\{P_n(t)\}$ and $\{y_{n,m}(t)\}$.
For $t = 1, 2, \ldots, T$ **do**
Phase 1: UCB-based frequency-aware device-edge data offloading optimization
For $d_n \in \mathcal{D}$ **do**
Initialize $a_{l,m}^n(t), x_{n,m}(t)$, and $\alpha_{l,m}^n(t)$.
Sequentially select each arm and obtains the initial reward.
Calculate the confidence upper bound based on (20).
Select arm A_{l^*,m^*}^n and perform the action $a_{l^*,m^*}^n(t)$ based on (21).
Update $\alpha_{l^*,m^*}^n(t)$, $\bar{\alpha}_{l^*,m^*}^n(t)$, and $r_{l^*,m^*}^n(t)$ based on (19) and (22) respectively.
End for
Phase 2: EXP3-based edge-cloud data splitting optimization
For $s_m \in \mathcal{S}$ **do**
Initialize the uniform distribution parameter $\xi \in (0,1]$. Set the empirical performance-related distribution parameter $\lambda_h^{n,m}(t) = 1$, $\forall Y_h^{n,m} \in \mathcal{Y}$.
Calculate the probability for selecting $Y_h^{n,m}$ based on (25).
Calculate the cumulative distribution function of $prob_h^{n,m}(t)$ based on (26).
Generate a random value $prob_0^{n,m}(t) \in [0,1]$, and get the optimal splitting ratio decision based on (27).
Execute $b_1^{n,m}(t) = 1$ and get the reward $\gamma_h^{n,m}(t)$ based on (24).
Update the empirical performance-related distribution parameter $\lambda_h^{n,m}(t)$ based on (28) and (29).
End for
End for

Algorithm 1. Machine Learning-based Cloud-Edge Collaborative Data Processing Optimization Algorithm.

the transmission power $P_n(t) = P_{n,min} + \frac{(l-1)(P_{n,max} - P_{n,min})}{L-1}$ to offload data to the edge server s_m.

- Reward: In the t-th slot, d_n selects $A_{l,m}^n$ to get the reward $\alpha_{l,m}^n(t)$, which is given by Eq. 19:

$$\alpha_{l,m}^n(t) = -\Upsilon_{n,m}(t). \tag{19}$$

We propose a UCB-based frequency-aware device-edge data offloading optimization algorithm, which introduces the acquisition frequency weight into the confidence upper bound calculation formula to achieve frequency awareness, and addresses the MAB problem of device-edge data offloading. UCB is a low-complexity learning-based algorithm to balance exploitation and exploration. The proposed algorithm enables the acquisition devices to take action based on local state information such as delay. Afterward, combined with the optimization variables $\tau_{n,m}^{D,E}(t)$ and $E_{n,m}(t)$, the acquisition devices perceive the obtained reward and updated state information for the next selection Xia et al. (2020).

The implementation procedure of UCB-based frequency-aware device-edge data offloading optimization algorithm is introduced as follows.

3.1.1 Initialization

Initialize $a_{l,m}^{n}(t)=0$, $x_{n,m}(t)=0$, and $\alpha_{l,m}^{n}(t)=0$, $\forall d_n \in \mathcal{D}$, $\forall s_m \in \mathcal{S}$. When $t \le |\mathcal{A}^n|$, device $d_n \in \mathcal{D}$ sequentially selects each arm and obtains the initial reward.

3.1.2 Decision making

d_n calculates the confidence upper bound based on the selection number $r_{l,m}^{n}(t)$ of $A_{l,m}^{n}$ in the t-th slot, which is expressed by Eq. 20:

$$\alpha_{l,m}^{n,up}(t)=\bar{\alpha}_{l,m}^{n}(t-1)+\frac{1}{\beta_n\bar{\alpha}_{l,m}^{n}(t-1)}\sqrt{\frac{\ln(t)}{r_{l,m}^{n}(t-1)}}, \tag{20}$$

where $\bar{\alpha}_{l,m}^{n}(t-1)$ is the average reward before the $(t-1)$-th slot, β_n is the acquisition frequency weight of d_n, $\sqrt{\frac{\ln(t)}{r_{l,m}^{n}(t)}}$ is the confidence interval of $A_{l,m}^{n}$, and β_n is the acquisition frequency weight of d_n. If the acquisition frequency weight β_n of d_n is larger, more data will be collected at each slot. In order to ensure its transmission delay and energy consumption performance, it is necessary to utilize the best possible decision. Thus, the second term indicates that if β_n is larger and the reward value of the selected arm is higher, the confidence interval is smaller and the device tends to exploit the currently selected arm. If β_n is smaller and the reward value of the selected arm is lower, the confidence interval is larger and the device tends to explore other arms.

After obtaining $\alpha_{l,m}^{n,up}(t)$, d_n selects the arm $A_{l,m}^{n}$ with the highest confidence upper bound to perform the action, which is expressed by Eq. 21:

$$\begin{cases} A_{l^*,m^*}^{n}=\arg\max_{\{A_{l^*,m^*}^{n}\}}\left\{\alpha_{l,m}^{n,up}(t)\right\}, \\ a_{l^*,m^*}^{n}(t)=1. \end{cases} \tag{21}$$

3.1.3 Learning process

The device observes delay and energy efficiency performances. Then, gets the reward $\alpha_{l^*,m^*}^{n}(t)$ based on (19). Accordingly, $\bar{\alpha}_{l^*,m^*}^{n}(t)$ and $r_{l^*,m^*}^{n}(t)$ are updated as Eq. 22:

$$\begin{aligned} \bar{\alpha}_{l^*,m^*}^{n}(t)&=\frac{r_{l^*,m^*}^{n}(t-1)\bar{\alpha}_{l^*,m^*}^{n}(t-1)+a_{l^*,m^*}^{n}(t)\alpha_{l^*,m^*}^{n}(t)}{r_{l^*,m^*}^{n}(t-1)+a_{l^*,m^*}^{n}(t)}, \\ r_{l^*,m^*}^{n}(t)&=r_{l^*,m^*}^{n}(t-1)+a_{l^*,m^*}^{n}(t). \end{aligned} \tag{22}$$

3.2 EXP3-based edge-cloud data splitting optimization

SP2 is formulated as Eq. 23:

$$\begin{aligned} &\mathbf{SP2}: \min_{\{y_{n,m}(t)\}}\frac{1}{N}\sum_{n=1}^{N}\sum_{m=1}^{M}\Upsilon_{n,m}^{Exp}(t) \\ &\text{s.t. } C_5, \end{aligned} \tag{23}$$

where $\Upsilon_{n,m}^{Exp}(t)=\max\{\tau_{n,m}^{E,com}(t),\tau_{n,m}^{E,C}(t)+\tau_{n,m}^{C,com}(t)\}$.

Based on the offloading decision obtained by optimizing **SP1**, **SP2** optimizes the edge-cloud data splitting process. Its optimization variable involves $y_{n,m}(t)$. The edge server obtains the optimal splitting decision by minimizing the delay required to process all data. Similarly, we transform **SP2** into an MAB problem. The decision maker, arm, action, and reward are described as follows.

- Decision maker: Define the edge server as the decision maker, which makes the splitting ratio decision when conducting edge-cloud data splitting.
- Arm: Define the arm as the total splitting ratio based on $\mathcal{Y}$. The arm $Y_h^{n,m}\in\mathcal{Y}$ indicates that the data splitting level of the d_n specified by the edge server s_m is h.
- Action: Define the action indicator variable as $b_h^{n,m}(t)$, $b_h^{n,m}(t)=1$ represents that edge server s_m sets the splitting ratio of the data from d_n as $y_{n,m}(t)=y_{min}+\frac{(h-1)(y_{max}-y_{min})}{H-1}$ in the t-th slot.
- Reward: Define $\gamma_h^{n,m}$ to represent the reward obtained by s_m of selecting arm $Y_h^{n,m}$, which is given by Eq. 24:

$$\gamma_h^{n,m}(t)=\frac{1}{\Upsilon_{n,m}^{Exp}(t)}. \tag{24}$$

We propose an EXP3-based edge-cloud data splitting optimization algorithm to address the MAB problem. The core idea is to maintain the probability of a certain arm. Then, the algorithm randomly selects a certain arm each time and updates the weight of the arm based on the observed reward after selection Zhou et al. (2021). Through iteration, this algorithm can ensure that the regret value is within a certain acceptable range.

The implementation procedure of the EXP3-based edge-cloud data splitting optimization algorithm is introduced as follows.

3.2.1 Initialization

Initialize the uniform distribution parameter $\xi\in(0,1]$. Set the empirical performance-related distribution parameter $\lambda_h^{n,m}(t)=1$, $\forall Y_h^{n,m}\in\mathcal{Y}$.

3.2.2 Decision making

In the t-th slot, firstly, calculate the probability for selecting $Y_h^{n,m}$, which is given by Eq. 25:

$$prob_h^{n,m}(t)=(1-\xi)\frac{\lambda_h^{n,m}(t)}{\sum_{h=1}^{H}\lambda_h^{n,m}(t)}+\frac{\xi}{H}. \tag{25}$$

Then, calculate the cumulative distribution function of $prob_h^{n,m}(t)$, which is given by Eq. 26:

$$F^{n,m}(h)=\sum_{h=1}^{H}prob_h^{n,m}(t). \tag{26}$$

Finally, generate a random value $prob_0^{n,m}(t)\in[0,1]$, and get the optimal splitting ratio decision, which is given by Eq. 27:

$$b_h^{n,m}(t)=\begin{cases}1, & \text{if } F^{n,m}(h-1)\le prob_0^{n,m}(t)\le F^{n,m}(h), \\ 0, & \text{otherwise.}\end{cases} \tag{27}$$

Specially, if $0\le prob_0^{n,m}(t)\le F^{n,m}(1)$, $b_1^{n,m}(t)=1$.

TABLE 1 Simulation parameters.

Parameter	Value	Parameter	Value
N	10	M	3
T	100	L	6
H	5	V_E	5×10^3
$f_{n,m}^E(t)$	6×10^{10} cycles/s	$f_{n,m}^C(t)$	12×10^{10} cycles/s
β_n	[5,10] times/s	η	1
τ_1	60 ms	τ_2	40 ms
τ_3	80 ms	δ_0	−114 dBm
$B_{n,m}$	[4,6] MHz	$U_n(t)$	[1.2,1.8] Mbits
$I_{n,m}(t)$	[28 30] dBm	χ_n	10^3 cycles/bit
$\mathcal{P}_n$	[0.1, 0.2, 0.3, 0.4, 0.5] W	$\mathcal{Y}$	[0 0.2 0.4 0.6 0.8 1]

3.2.3 Learning process

The edge server executes the splitting ratio decision $b_1^{n,m}(t)$, gets the reward $\gamma_h^{n,m}(t)$ based on (24), and updates the empirical performance-related distribution parameter as Eq. 28:

$$\lambda_h^{n,m}(t+1) = \lambda_h^{n,m}(t)\exp\left(\frac{\eta\tilde{\gamma}_h^{n,m}(t)}{H}\right), \tag{28}$$

where $\eta > 0$ is the adjustment factor of empirical performance-related distribution parameter and $\tilde{\gamma}_h^{n,m}(t)$ is the estimated reward, which is given by Eq. 29:

$$\tilde{\gamma}_h^{n,m}(t) = \begin{cases} \dfrac{\gamma_h^{n,m}(t)}{prob_h^{n,m}(t)}, & \text{if } b_h^{n,m}(t) = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{29}$$

Finally, the algorithm terminates until $t > T$.

4 Simulation result

In this paper, we take a low-voltage distribution network in a certain area as the simulation scenario to verify the system model and the performance of the proposed algorithm, which includes 10 power distribution acquisition devices, 3 edge servers, and one cloud server. The amount of data collected by a device in each slot distributed within [1.2, 1.8] Mbits. The transmission power and the data splitting ratio contain 5 and 6 levels, respectively. The specific simulation parameters are shown in Table 1 Liao et al. (2022); Yang et al. (2023).

Two state-of-the-art algorithms are used for comparison. The first one is the multi-index evaluation learning-based computation offloading algorithm (MINCO), which sets the average total data processing delay minimization as the optimization objective, but lacks energy consumption control of device Lu et al. (2023). MINCO considers multiple indices in power internet of things to improve the learning performance of its algorithm, thereby achieving the low-delay computation offloading. The other one is the UCB-advantage actor-critic-based data offloading algorithm (UCB-A3C), which considers energy consumption management and transmission delay optimization Yang et al. (2022). UCB-A3C combines UCB and actor-critic algorithm to enhance the learning ability of its algorithm, and achieves the joint optimization of energy consumption and delay. Meanwhile, both comparison algorithms do not consider data splitting optimization.

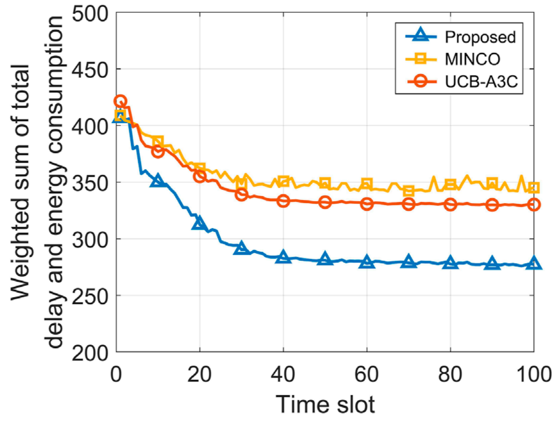

FIGURE 2
The weighted sum of total delay and energy consumption *versus* time slot.

Figure 2 shows the weighted sum of total delay and energy consumption *versus* time slot. The simulation result shows that the proposed algorithm has the lowest weighted sum among the three algorithms. Compared with MINCO and UCB-A3C, the proposed algorithm can decrease the weighted sum performance by 19.69% and 16.05%, respectively. The reason is that the proposed algorithm can coordinate the balance between total delay and energy consumption by adjusting the transmission power of devices and the data splitting ratio of edge servers, which reduces energy consumption while maintaining low delay. However, MINCO merely focuses on delay reduction while the energy consumption balance is neglected. UCB-A3C considers energy consumption management, but the utilization of cloud-edge computing resources is inadequate, resulting in poor weighted sum performance.

Figure 3 shows the total delay of data processing *versus* time slot. It can be seen that the proposed algorithm has the optimal total delay performance of data processing. Compared with MINCO and UCB-A3C, the proposed algorithm can decrease the total delay by 16.91% and 23.11%, respectively. The reason is that both MINCO and UCB-A3C adopt the traditional binary full offloading strategy, and do not take into account the optimization of the edge-cloud data splitting process, which makes them difficult to fully utilize the computing resources of cloud server and edge servers, leading to worse delay performance.

Figure 4 shows the cumulative energy consumption *versus* time slot. Compared with MINCO and UCB-A3C, the proposed algorithm can reduce the cumulative energy consumption by 15.04% and 9.52% respectively. The reason is that the proposed algorithm can coordinate the balance between total delay and energy consumption through the joint optimization of data offloading and data splitting allocation. MINCO only considers the data offloading

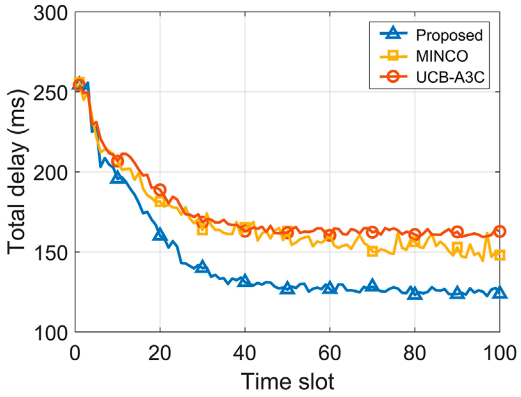

FIGURE 3
The total delay of data processing *versus* time slot.

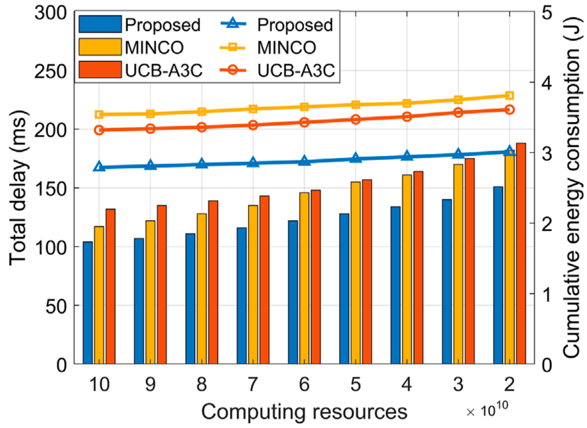

FIGURE 6
The total delay and cumulative energy consumption of different algorithms *versus* computing resources of edge servers.

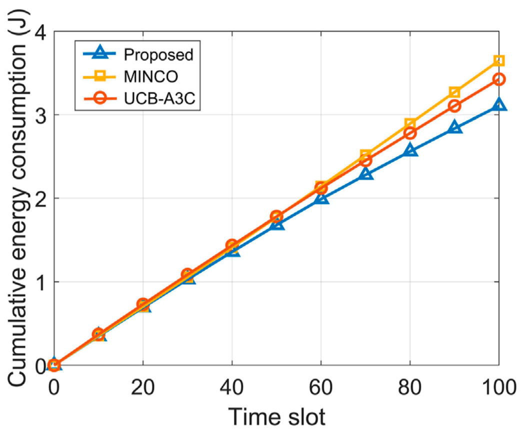

FIGURE 4
The cumulative energy consumption *versus* time slot.

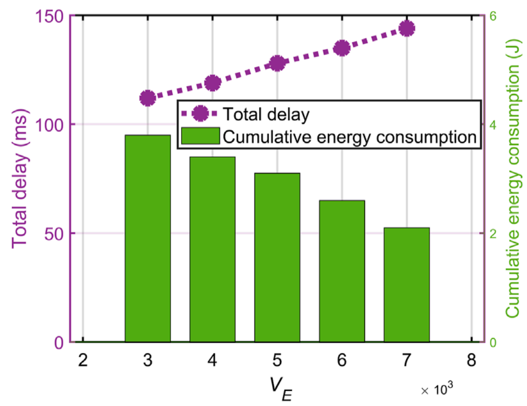

FIGURE 7
The impact of V_E on the total delay and cumulative energy consumption.

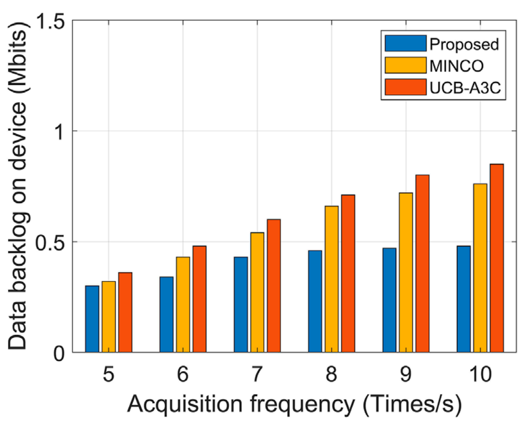

FIGURE 5
The data backlog on device *versus* acquisition frequency.

delay optimization but ignores the coupling relationship between data transmission power and data offloading delay, which leads to the highest energy consumption. UCB-A3C lacks optimization of edge-cloud data splitting process, and cannot make full use of computing resources of cloud server and edge server, resulting in serious data backlog queue, which consumes more energy for device data offloading.

Figure 5 shows the data backlog on device *versus* acquisition frequency. With the increase of acquisition frequency, the data backlogs of the three algorithms all increase, but the data backlog of the proposed algorithm increases the least. This is because the proposed algorithm can adaptively learn the server and transmission power selection strategies by adjusting the balance between exploration and exploitation through the acquisition frequency awareness. When the acquisition frequency is large, the proposed algorithm will tend to utilize the current optimal strategy to effectively reduce the data backlog. On the contrary, the proposed algorithm will tend to explore other strategies to avoid the optimization falling into local optimality.

Figure 6 shows the total delay and cumulative energy consumption of different algorithms *versus* computing resources of edge servers. When the computing resources of the edge servers decrease, the total delay of all algorithms increases due to the increase of data processing time. However, the proposed algorithm exhibits a minimal upward trend in terms of total latency and energy consumption. This is because the proposed algorithm transmits part of the data to the cloud server for processing through data splitting of the edge server, so as to relieve the processing pressure of the edge server. At the same time, more edge servers with better performance are available for devices to choose for data offloading, thus reducing data transmission power consumption. However, the binary unloading strategy is adopted in the comparison algorithm, which transmits too much data to the cloud server, and the computing resources of the edge server cannot be fully utilized, resulting in the increase of the total delay performance.

Figure 7 shows the impact of V_E on the total delay and cumulative energy consumption. With the increase of V_E, the proposed algorithm pays more attention to energy consumption and ignores the total delay performance. As a result, the total delay gradually increases, and the energy consumption gradually decreases. This result provides a reference for the setting of V_E in practical applications. Reasonable setting can achieve a compromise between the total delay of data processing and device energy consumption performance.

5 Conclusion

In this paper, we investigated the cloud-edge collaborative high-frequency acquisition data processing architecture for distribution network resilience improvement. A machine learning-based cloud-edge collaborative data processing optimization algorithm was proposed to minimize the weighted sum of data processing delay and device energy consumption by jointly optimizing transmission power selection, edge server selection, and data splitting ratio selection. Firstly, we decomposed the optimization problem into two subproblems of device-edge data offloading and edge-cloud data splitting. Then, a UCB-based frequency-aware device-edge data offloading optimization algorithm was employed to address the device-edge data offloading subproblem, and an EXP3-based edge-cloud data splitting optimization algorithm was employed to address the edge-cloud data splitting subproblem. Simulation results demonstrated that the proposed algorithm can achieve superior performance in terms of processing delay and energy consumption. Compared with MINCO and UCB-A3C, the proposed algorithm can decrease the weighted sum performance by 19.69% and 16.05%, respectively.

In the future, we will focus on the combination of high-frequency acquisition data processing with security technologies such as blockchain, encryption authentication, or differential privacy, thereby achieving the joint guarantee of low processing delay, low energy consumption, and high data security and privacy performances.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

SD: Conceptualization, Formal Analysis, Funding acquisition, Investigation, Methodology, Writing–original draft, Writing–review and editing. JiZ: Conceptualization, Formal Analysis, Investigation, Methodology, Writing–original draft, Writing–review and editing. TL: Investigation, Methodology, Writing–original draft, Writing–review and editing. YuZ: Software, Validation, Visualization, Writing–original draft. PS: Software, Validation, Writing–review and editing. JZ: Supervision, Visualization, Writing–review and editing. RL: Supervision, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by Science and Technology Project of China Southern Power Grid Corporation under Grant Number 035900KK52220012 (GDKJXM20220909).

Conflict of interest

Authors SD, JiZ, TL, YZ, PS, JuZ, and RL were employed by the Metrology Center, Guangdong Power Grid Co., Ltd.

The authors declare that this study received funding from the Science and Technology Project of China Southern Power Grid Corporation. The funder had the following involvement in the study: resources, data collection and analysis, software and methodology.

Publisher's note

References

Chen, X., Wu, C., Liu, Z., Zhang, N., and Ji, Y. (2021). Computation offloading in beyond 5G networks: a distributed learning framework and applications. *IEEE Wirel. Commun.* 28, 56–62. doi:10.1109/mwc.001.2000296

Chen, Y.-J., Jiang, W.-C., Ju, M.-Y., and Hwang, K. S. (2020). Policy sharing using aggregation trees for ${Q}$ -learning in a continuous state and action spaces. *IEEE Trans. Cognitive Dev. Syst.* 12, 474–485. doi:10.1109/tcds.2019.2926477

Cui, G., Long, Y., Xu, L., and Wang, W. (2021). Joint offloading and resource allocation for satellite assisted vehicle-to-vehicle communication. *IEEE Syst. J.* 15, 3958–3969. doi:10.1109/jsyst.2020.3017710

Ding, B., Li, Z., Li, Z., Xue, Y., Chang, X., Su, J., et al. (2024). A CCP-based distributed cooperative operation strategy for multi-agent energy systems integrated with wind, solar, and buildings. *Appl. Energy* 365, 123275. doi:10.1016/j.apenergy.2024.123275

Dong, W., Yang, Q., Li, W., and Zomaya, A. Y. (2021). Machine-learning-based real-time economic dispatch in islanding microgrids in a cloud-edge computing environment. *IEEE Internet Things J.* 8, 13703–13711. doi:10.1109/jiot.2021.3067951

Gao, H., Ma, W., He, S., Wang, L., and Liu, J. (2022). Time-segmented multi-level reconfiguration in distribution network: a novel cloud-edge collaboration framework. *IEEE Trans. Smart Grid* 13, 3319–3322. doi:10.1109/tsg.2022.3156433

Guo, S., Dai, Y., Xu, S., Qiu, X., and Qi, F. (2020). Trusted cloud-edge network resource management: DRL-driven service function chain orchestration for IoT. *IEEE Internet Things J.* 7, 6010–6022. doi:10.1109/jiot.2019.2951593

Hashima, S., Hatano, K., Takimoto, E., and Mahmoud Mohamed, E. (2020). Neighbor discovery and selection in millimeter wave D2D networks using stochastic MAB. *IEEE Commun. Lett.* 24, 1840–1844. doi:10.1109/lcomm.2020.2991535

Jiang, S., Gao, H., Wang, X., Liu, J., and Zuo, K. (2023). Deep reinforcement learning based multi-level dynamic reconfiguration for urban distribution network: a cloud-edge collaboration architecture. *Glob. Energy Interconnect.* 6, 1–14. doi:10.1016/j.gloei.2023.02.001

Laili, Y., Guo, F., Ren, L., Li, X., and Zhang, L. (2023). Parallel scheduling of large-scale tasks for industrial cloud–edge collaboration. *IEEE Internet Things J.* 10, 3231–3242. doi:10.1109/jiot.2021.3139689

Li, S., Wen, X., Wen, Y., Li, L., and Zhang, S. (2024a). Container-based cloud-edge-device collaborative multi-service data processing in distribution network. *IEEE Trans. Consumer Electron.* 70, 2323–2332. doi:10.1109/tce.2023.3344514

Li, Z., Su, S., Jin, X., et al. (2024b). Stochastic and distributed optimal energy management of active distribution networks within integrated office buildings. *CSEE J. Power Energy Syst.* 10, 504–517. doi:10.17775/CSEEJPES.2021.04510

Li, Z., Xu, Y., Wang, P., and Xiao, G. (2023). Coordinated preparation and recovery of a post-disaster multi-energy distribution system considering thermal inertia and diverse uncertainties. *Appl. Energy* 336, 120736. doi:10.1016/j.apenergy.2023.120736

Li, Z., Xu, Y., Wang, P., and Xiao, G. (2024c). Restoration of a multi-energy distribution system with joint district network reconfiguration via distributed stochastic programming. *IEEE Trans. Smart Grid* 15, 2667–2680. doi:10.1109/tsg.2023.3317780

Liao, H., Jia, Z., Zhou, Z., Wang, Y., Zhang, H., and Mumtaz, S. (2022). Cloud-edge-end collaboration in air–ground integrated power IoT: a semidistributed learning approach. *IEEE Trans. Industrial Inf.* 18, 8047–8057. doi:10.1109/tii.2022.3164395

Liao, H., Zhou, Z., Zhao, X., Zhang, L., Mumtaz, S., Jolfaei, A., et al. (2020). Learning-based context-aware resource allocation for edge-computing-empowered industrial IoT. *IEEE Internet Things J.* 7, 4260–4277. doi:10.1109/jiot.2019.2963371

Lin, N., Bai, L., Hawbani, A., Guan, Y., Mao, C., Liu, Z., et al. (2024). Deep-reinforcement-learning-based computation offloading for servicing dynamic demand in multi-UAV-assisted IoT network. *IEEE Internet Things J.* 11, 17249–17263. doi:10.1109/jiot.2024.3356725

Liu, H., and Cao, G. (2021). Deep reinforcement learning-based server selection for mobile edge computing. *IEEE Trans. Veh. Technol.* 70, 13351–13363. doi:10.1109/tvt.2021.3124127

Long, S., Zhang, Y., Deng, Q., Pei, T., Ouyang, J., and Xia, Z. (2023). An efficient task offloading approach based on multi-objective evolutionary algorithm in cloud-edge collaborative environment. *IEEE Trans. Netw. Sci. Eng.* 10, 645–657. doi:10.1109/tnse.2022.3217085

Lu, J., Shi, Z., Wang, Y., Pan, C., and Zhang, S. (2023). Multi-index evaluation learning-based computation offloading for power internet of things. *Phys. Commun.* 56, 101949. doi:10.1016/j.phycom.2022.101949

Mu, S., Zhong, Z., Zhao, D., and Ni, M. (2019). Joint job partitioning and collaborative computation offloading for internet of things. *IEEE Internet Things J.* 6, 1046–1059. doi:10.1109/jiot.2018.2866945

Naeem, F., Tariq, M., and Poor, H. V. (2021). SDN-enabled energy-efficient routing optimization framework for industrial internet of things. *IEEE Trans. Industrial Inf.* 17, 5660–5667. doi:10.1109/tii.2020.3006885

Shah, Z., Anwar, A., Mahmood, A. N., Tari, Z., and Zomaya, A. Y. (2020). A spatiotemporal data summarization approach for real-time operation of smart grid. *IEEE Trans. Big Data* 6, 624–637. doi:10.1109/tbdata.2017.2691350

Soltani, Z., Ma, S., Khorsand, M., and Vittal, V. (2023). Simultaneous robust state estimation, topology error processing, and outage detection for unbalanced distribution systems. *IEEE Trans. Power Syst.* 38, 2018–2034. doi:10.1109/tpwrs.2022.3181118

Tariq, M., Adnan, M., Srivastava, G., and Poor, H. V. (2020). Instability detection and prevention in smart grids under asymmetric faults. *IEEE Trans. Industry Appl.* 56, 1–4520. doi:10.1109/tia.2020.2964594

Wang, X., Li, N., and Sun, W. (2018). Quantitative analysis of distributed and centralized development of renewable energy. *Glob. Energy Interconnect.* 1, 576–584.

Wang, Z., Zhou, Z., Zhang, H., Zhang, G., Ding, H., and Farouk, A. (2022). AI-based cloud-edge-device collaboration in 6G space-air-ground integrated power IoT. *IEEE Wirel. Commun.* 29, 16–23. doi:10.1109/mwc.001.00254

Wu, H., Zhang, Z., Guan, C., Wolter, K., and Xu, M. (2020). Collaborate edge and cloud computing with distributed deep learning for smart city internet of things. *IEEE Internet Things J.* 7, 8099–8110. doi:10.1109/jiot.2020.2996784

Xia, F., Gu, Y., Liu, K., Song, Y., Tang, M., and Cao, T. (2022). "Joint task segmentation and server selection for real-time stream data analytics in edge-cloud environments," in 2022 IEEE 4th international conference on civil aviation safety and information Technology (ICCASIT), 12-14 October 2022, Dali, China, 60–64.

Xia, W., Quek, T. Q. S., Guo, K., Wen, W., Yang, H. H., and Zhu, H. (2020). Multi-armed bandit-based client scheduling for federated learning. *IEEE Trans. Wirel. Commun.* 19, 7108–7123. doi:10.1109/twc.2020.3008091

Xiao, K., Li, D., Wang, X., and Guo, P. (2022). Modeling and application of marketing and distribution data based on graph computing. *Glob. Energy Interconnect.* 5, 448–460. doi:10.1016/j.gloei.2022.08.010

Yang, A., Lu, C., Yu, W., Hu, J., Nakanishi, Y., and Wu, M. (2023a). Data augmentation considering distribution discrepancy for fault diagnosis of drilling process with limited samples. *IEEE Trans. Industrial Electron.* 70, 11774–11783. doi:10.1109/tie.2022.3229274

Yang, Y., Li, Z., Mandapaka, P. V., and Lo, E. Y. (2023b). Risk-averse restoration of coupled power and water systems with small pumped-hydro storage and stochastic rooftop renewables. *Appl. Energy* 339, 120953. doi:10.1016/j.apenergy.2023.120953

Yang, Y., Li, H., Shen, B., Pei, W., and Peng, D. (2022). Microgrid energy management strategy base on UCB-A3C learning. *Front. Energy Res.* 10, 1–8. doi:10.3389/fenrg.2022.858895

Zhang, G., Ni, S., and Zhao, P. (2022a). Learning-based joint optimization of energy delay and privacy in multiple-user edge-cloud collaboration MEC systems. *IEEE Internet Things J.* 9, 1491–1502. doi:10.1109/jiot.2021.3088607

Zhang, K., Dou, X., Yu, H., et al. (2022b). "Data processing technology of distribution station area combined with convolutional neural network," in 2022 12th international conference on CYBER Technology in automation, control, and intelligent systems (CYBER), 27-31 July 2022, Baishan, China, 866–870.

Zhang, S., Cui, G., Long, Y., and Wang, W. (2021). Joint computing and communication resource allocation for satellite communication networks with edge computing. *China Commun.* 18, 236–252. doi:10.23919/jcc.2021.07.019

Zhang, S., Wang, Z., Zhou, Z., Wang, Y., Zhang, H., Zhang, G., et al. (2022c). Blockchain and federated deep reinforcement learning based secure cloud-edge-end collaboration in power IoT. *IEEE Wirel. Commun.* 29, 84–91. doi:10.1109/mwc.010.2100491

Zhao, P., Tian, H., Chen, K.-C., Fan, S., and Nie, G. (2020). Context-aware TDD configuration and resource allocation for mobile edge computing. *IEEE Trans. Commun.* 68, 1118–1131. doi:10.1109/tcomm.2019.2952580

Zhou, Z., Jia, Z., Liao, H., Lu, W., Mumtaz, S., Guizani, M., et al. (2022). Secure and latency-aware digital twin assisted resource scheduling for 5G edge computing-empowered distribution grids. *IEEE Trans. Industrial Inf.* 18, 4933–4943. doi:10.1109/tii.2021.3137349

Zhou, Z., Wang, Z., Yu, H., Liao, H., Mumtaz, S., Oliveira, L., et al. (2021). Learning-based URLLC-aware task offloading for internet of health things. *IEEE J. Sel. Areas Commun.* 39, 396–410. doi:10.1109/jsac.2020.3020680