IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2024EAP1070

Publicized:2024/07/29

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER

Task Offloading and Resource Allocation for Wireless Powered Multi-AP Mobile Edge Computing

Guanqun SHEN[†], **Kaikai CHI**[†a)], **Osama ALFARRAJ**[††], ***and*** **Amr TOLBA**[††], ***Nonmembers***

SUMMARY IoT devices, which possess limited battery capacity and computing capabilities, are unable to meet many applications' demands. The integration of wireless power transfer and edge computing has emerged as a promising solution for this problem. Nevertheless, efficiently making offloading decisions and allocating resources pose significant challenges, particularly in the scenarios of multiple access points (APs). This paper focuses on optimizing the sum computation rate (SCR) in a wireless powered network having multiple APs. The devices work in binary offloading, operating under frequency-division multiple access (FDMA) and time-division multiple access (TDMA), respectively. To efficiently address these two mixed-integer nonlinear programming problems, a deep reinforcement learning based algorithm is employed to determine the near-optimal offloading decisions. Additionally, under the given offloading decision, we present an algorithm using the golden section search for FDMA to obtain the subsequent optimal time allocation, and apply convex optimization algorithm to obtain the optimal time allocation for TDMA. Our algorithms achieve over 95 percent of the maximum SCR with low complexity. In comparison to the baseline algorithms, our proposed algorithms exhibit advantages in terms of convergence speed and attained SCR.

key words: *Edge computing, wireless power transfer, deep reinforcement learning, computation rate maximization*

1. Introduction

The IoT devices is steadily growing, playing crucial roles in industrial production and enhancing daily convenience. They encompass various domains such as intelligent plants, autonomous driving, and smart home [1]. However, the pursuit of heightened intelligence poses fresh challenges for conventional IoT nodes, which grapple with limitations in battery capacity and computing capability [2].

To address the challenges of energy and computation, an effective solution for IoT networks is using mobile edge computing (MEC) and wireless power transfer (WPT), called wireless powered mobile edge computing (WP-MEC) [3]. This paradigm involves the utilization of WPT, where hybrid access points (H-AP) facilitates the charging of IoT nodes during the energy transmission phase [4]. Additionally, edge server provides MEC service for the nodes [5–10].

Deep reinforcement learning (DRL) shows great promise for integration with MECs to enable effective and intelligent control of IoT nodes. The combination of WPT, MEC, and DRL has garnered significant attention in recent years.

This work aims to maximize the sum computation rate (SCR) based on DRL. So far, some research efforts have been devoted on this issue [11–20]. For example, [11] studied the WP-MEC using the time-division multiple access (TDMA) based binary offloading and presented an algorithm using the traditional optimization techniques to maximize the SCR. Following [11], Huang et al. [12] proposed a DRL-based algorithm called DROO for WP-MEC with TDMA-based binary offloading. It utilizes a deep neural network (DNN) to generate near-optimal offloading decisions and employs an efficient algorithm to allocate time resources, thereby maximizing the SCR. [13] investigated the WP-MEC using the non-orthogonal multiple access (NOMA) based binary offloading and proposed a greedy algorithm based on wireless channel gain. [14] proposed a resource allocation scheme for the WP-MEC using the TDMA-based binary offloading to maximize the minimal computation rate among the nodes.

It should be noted that the above work on the SCR maximization all considered the WP-MEC with one single HAP. So far, there is a scarcity of literature specifically addressing the problems of WP-MEC with multiple HAPs (i.e., multiple edge servers), which is practical in the large-scale IoT networks. The scenario involving multiple HAPs introduces greater complexity in decision-making processes related to offloading and resource allocation, making it a highly challenging problem to tackle. [15] focused on WP-MEC with multiple HAPs and proposed one DRL-based algorithm to maximize the SCR when using the binary offloading. In [15], the offloading wireless devices (WDs) associated to the same HAP use the same communication band to offload date by using TDMA mode. The offloading WDs associated to different HAPs use the orthogonal channels.

In this paper, we focus on WP-MEC consisting of several H-APs and WNs and aim to maximize the SCR. We first study the frequency-division multiple access (FDMA) based scenario where all offloading WNs (even those associated to the same H-AP) use different

[†]The authors are with School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, 310023, China.

[††]The author are with Computer Science Department, Community College, King Saud University, Saudi Arabia.

a) E-mail: kkchi@zjut.edu.cn

orthogonal channels. Then we also study the TDMA-based scenario where the offloading WNs associated to the same H-AP use the same communication band to offload date to H-AP by using TDMA mode. However, different from [15], each H-AP has its own WPT duration which is more flexible for WPT but more challenging to address.

The primary contributions can be summarized as follows:

- The SCR maximizations under FDMA and TDMA are formulated as optimization problems. Due to the large number of parameters to be optimized, we decouple the parameters into two sets: the offloading decision (OD) parameters and the time allocation parameters. Then, they are decomposed into a master problem of determining the OD and a subproblem of optimizing time allocation under the predetermined OD.
- A policy-based DRL framework is utilized to effectively generates good ODs, which takes the subproblem's optimal solution to calculate the SCR (i.e., the reward) of a given OD. The DNN within the framework demonstrates fast convergence during the online training process.
- In order to tackle the subproblem, we utilize a golden section search based method to achieve the optimal time allocation of FDMA and apply convex optimization algorithm for finding the optimal time allocation of TDMA.
- Our algorithms achieve the SCR exceeding 95 percent of the maximum achievable rate while maintaining low computational complexity.

This paper is structured as follows: Section II introduces the related works in the field. Section III presents the system model, providing a comprehensive understanding of the underlying architecture. In Section IV, we formulate the problems and introduce the algorithms. Section V presents the simulation results. In Section VI, we conclude the paper by summarizing the main findings.

2. Related Works

In WP-MEC networks, the task offloading modes of WNs can be categorized into two types: partial offloading and binary offloading. This classification is based on whether the tasks of WNs can be separated into offloading and local computation parts.

2.1 Partial Offloading

Partial offloading enables distributed processing by offloading a portion of tasks to edge servers. This approach enhances computational efficiency and resource utilization in the system.

For TDMA, Zhang et al. applied a DRL-based algorithm to maximize the SCR [16]. Meanwhile, Zhou et al. [17] utilized unmanned aerial vehicles (UAVs) for WPT and maximized the weighted SCR. Additionally, a Soft Actor-Critic-based algorithm was proposed by Zhou et al. [18] for UAV based WP-MEC, maximizing the SCR of WNs. Intelligent Reflecting Surfaces (IRS) were leveraged by Mao et al. [19] to enhance WET and task offloading, thereby maximizing the SCR. Moreover, Chen et al. [20] employed multiple IRSs, energy beamforming, and multiple-user detection to maximize the SCR.

In the context of NOMA, [21] focused on maximizing the energy efficiency of the network. Considering TDMA, Wang et al. [22] utilized beamforming technology for uninterrupted charging of WNs by AP, minimizing the total energy consumption of AP while ensuring timely task completion. Additionally, in [23], dynamically arriving tasks within a fixed period were processed with the objective of minimizing the overall energy consumption of the system. For a scenario where WNs are with battery and collect RF energy for energy supplementation, [24] addressed tasks with latency constraints, aiming to minimize transmission energy consumption. [25] proposed an AoI-aware algorithm to optimize communication and computing resource allocation, achieving maximum system utility while considering the Age of Information constraint in IoT data analysis scenarios. [26] proposed an approach that combines MEC and WPT to maximize residual energy through multi-user dynamic offloading. By employing FDMA and convex optimization techniques, the approach achieves superior performance.

2.2 Binary Offloading

Binary offloading facilitates centralized task processing by fully offloading tasks to edge servers [11, 27–29]. This approach leverages the high-performance computing capabilities of edge servers.

In the context of TDMA, both [11] and [12] focused on maximizing the SCR. For a multiple H-AP scenario, Zhang et al. proposed a DRL algorithm in [15], aiming to maximize the SCR.

Liu et al. studied a wireless powered layered fog-cloud computing network in [30,31], where multiple energy constrained users first capture energy from H-AP, and then use the captured energy to offload computing tasks to fog/cloud servers through H-AP or perform tasks locally. An optimization problem was proposed to maximize the minimum remaining energy among multiple users. On the basis of [30], [31] further optimized the max-min fairness considering energy balance, aiming to achieve fairness in the remaining energy of multiple nodes after energy collection and calculation operations.

In addition, [13, 28] studied the design of binary

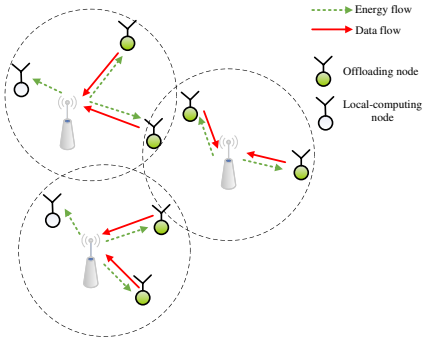

Fig. 1 WP-MEC with multiple H-AP.

offloading schemes based on NOMA. In [13], a NOMA based WP-MEC network is considered, where H-AP first provides wireless energy to nodes in each time slot, and then some nodes use NOMA to offload tasks, while other nodes choose to perform calculations locally. The goal is to maximize the SCR.

[32] presents a joint resource allocation and task offloading scheme in WP-MEC networks, achieving improved residual energy through a mixed-offloading paradigm and an efficient dual-layer optimization algorithm.

3. System Model

3.1 Network Model

Fig. 1 illustrates the network configuration considered in this study, which comprises N WNs and M H-APs. Let WN_i be the i-th WN and let AP_j be the j-th H-AP . let $\mathcal{N} = \{1, 2, ..., N\}$ and $\mathcal{M} = \{1, 2, ..., M\}$. Each H-AP is equipped with a server. This setup enables H-APs to broadcast RF energy to WNs. The WNs, in turn, possess the capability to harvest the transmitted RF energy from the H-APs. This stored energy can be utilized to power subsequent operations.

For data processing, WNs have two options: they can either perform the calculation locally or select a specific H-AP to offload the data.

In our scenario, the time slot is T seconds, and within each time slot, the channel gain remains constant but may vary across different time slots. The channel gain of wireless node WN_i and AP_j is

$$h_{ij} = |g_{ij}|^2 \alpha_{ij}, \tag{1}$$

where α_{ij} represents large-scale fading component and g_{ij} is the independent random channel fading factor. Several strategies exist to furnish the transmitter with highly accurate Channel State Information (CSI). For example, efficient analog CSI feedback strategies were introduced in [33] and [34]. Furthermore, in [35] and [36], CSI quantization feedback strategies were proposed to achieve precise CSI through limited feedback.

This work intends to maximize the SCR under both FDMA and TDMA modes. The processing time for data at the H-AP is neglected due to the powerful computing capabilities available on H-AP. Additionally, the time spent on the feedback of small-size results is also ignored. As a result, at each H-AP, a time slot is primarily divided into two parts: WPT and local/edge computing time.

In each time slot, each WN is associated to some H-AP according to the current network state. If one WN is arranged to conduct local computation in the current time slot, it harvests the RF energy from its H-AP and then conduct local computation. If one WN is arranged to offload data, it harvests the RF energy from its H-AP and then offloads data to its H-AP.

3.1.1 FDMA protocol

In FDMA communication, each offloading WN has its own communication band, and there is no interference between them. The communication bandwidth for each offloading WN is: $B/\overline{N}$, where B denotes the total available bandwidth, and $\overline{N}(\overline{N} \leq N)$ represents the total number of offloading WNs.

3.1.2 TDMA protocol

In TDMA communication, the offloading WNs associated with the same H-AP use the same communication band and communicate with the H-AP using the TDMA protocol. On the other hand, the offloading WNs associated with different H-APs utilize orthogonal channels to ensure interference-free communication. The communication bandwidth for each offloading WN is $B/\overline{M}$, where $\overline{M}$ denotes the number of H-APs that have at least one associated WN.

Let $x_{i,0} = 1$ means that WN_i processes its data locally, while $x_{i,j} = 1$ $(j \neq 0)$ indicates that WN_i offloads its data to AP_j. Additionally, if $x_{i,0} = 1$, we let WN_i harvest the RF energy form its nearest H-AP. Let i' denote the index of WN_i's nearest H-AP.

3.2 Energy Harvesting Model

During the WPT phase of $AP_j (j = 1, 2, ..., M)$, its associated offloading WNs and local-computing WNs whose nearest H-AP is AP_j collect RF energy from AP_j. The collected energy of WN_i can be calculated as follows:

$$E_i = \mu P T (x_{i0} h_{ii'} a_{i'} + \sum_{k=1}^{M} x_{ik} h_{ik} a_k), \tag{2}$$

In the equation, μ is the energy capture efficiency, P denotes H-AP's energy transmit power, and a_j is the WPT time ratio of AP_j inside one time slot.

3.3 Local Computing Model

If a WN opts to local computation, it can concurrently

engage in computation while harvesting RF energy. ϕ is the number of clock cycles for processing one bit. The CPU calculation speed of WN_i is denoted as f_i and is measured in cycles per second, with its value adjustable accordingly. Therefore, the amount of processed data is $f_i t_i/\phi$. t_i represents the computation time of WN_i. The energy consumption of the CPU is $cf_i^3 t_i$. It is crucial to ensure that the consumed energy is less than the harvested energy, i.e., $cf_i^3 t_i \leq E_i$. By maximizing the computation rate, we can determine optimal t_i and f_i. $t_i^* = T$ and $f_i^* = (\frac{E_i}{cT})^{1/3}$. Then, WN_i's local computing rate is given by:

$$\begin{aligned} r_{l,i} &= \frac{f_i t_i}{\phi T} \\ &= \frac{[\mu P(x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k)/c]^{1/3}}{\phi} \\ &= \eta_1 \left[x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k \right]^{1/3}, i \in \mathcal{N}, \end{aligned} \quad (3)$$

where $\eta_1 \triangleq \frac{(\mu P/c)^{\frac{1}{3}}}{\phi}$.

3.4 Edge Computing Model

For TDMA mode, let $\tau_{ij}T$ be WN_i's transmission time associated to AP_j. The following condition must be satisfied:

$$\sum_{i=1}^{N} \tau_{ij} + a_j \leq 1, \forall j \in \mathcal{M}. \quad (4)$$

Let b_{ij} be the amount of data for WN_i offloaded to AP_j. Then

$$b_{ij} = \frac{B_i \tau_{ij} T}{v_u} \log_2 \left(1 + \frac{P_{ij}h_{ij}}{N_0}\right), \quad (5)$$

where the term $v_u \geq 1$ represents a factor accounting for the communication overhead in data offloading, which includes costs such as encryption and data headers. The variable P_{ij} represents the transmit power of WN_i during data offloading to AP_j. Additionally, N_0 represents the power of the additive white Gaussian noise (AWGN) present in the communication channel.

To maximize the data offloading capability of WN_i, it is desirable to utilize all the harvested energy from the WPT phase. This implies that P_{ij} should be set to

$$P_{ij} = \frac{E_i}{\tau_{ij}T} = \frac{\mu P(x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k)}{\tau_{ij}}, \quad (6)$$

ensuring that all the harvested energy is utilized for data offloading. Consequently, the computation rate of WN_i, can be expressed as:

$$\begin{aligned} & r_{o,ij} \\ =& \frac{b_{ij}}{T} \\ =& \frac{B_i \tau_{ij}}{v_u} \log_2 \left(1 + \frac{\mu P h_{ij}(x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k)}{\tau_{ij}N_0}\right) \\ =& \epsilon_1 \tau_{ij} \ln \left(1 + \frac{\eta_2 h_{ij}(x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k)}{\tau_{ij}}\right), \end{aligned} \quad (7)$$

where $\epsilon_1 \triangleq \frac{B}{Mv_u \ln 2}$ and $\eta_2 \triangleq \frac{\mu P}{N_0}$.

Similarly, for FDMA mode, we have

$$\tau_{ij} + a_j \leq 1, \forall i \in \mathcal{N}, j \in \mathcal{M}, \quad (8)$$

and

$$r_{o,ij} = \epsilon_2 \tau_{ij} \ln \left(1 + \frac{\eta_2 h_{ij}(x_{i0}h_{ii'}a_{i'} + \sum_{k=1}^{M} x_{ik}h_{ik}a_k)}{\tau_{ij}}\right) \quad (9)$$

where $\epsilon_2 \triangleq \frac{B}{Nv_u \ln 2}$.

4. Problem Formulation and Efficient Algorithm

The SCR for a single time slot can be expressed as:

$$Q(\boldsymbol{h}, \boldsymbol{x}, \boldsymbol{\tau}, \boldsymbol{a}) \triangleq \sum_{i=1}^{N} \boldsymbol{x_i} \cdot [r_{l,i}, r_{o,i1}, r_{o,i2}, \ldots, r_{o,iM}]^{\mathrm{T}}, \quad (10)$$

where $\boldsymbol{h}$ is the vector of channel gains, $\boldsymbol{x}$ is the vector of offloading decisions, $\boldsymbol{\tau}$ is the vector of transmission time allocation, and $\boldsymbol{a} = [a_1, a_2, ..., a_M]$.

Without loss of generality, assume $T = 1$ below.

4.1 Problem Formulation of TDMA

The formulation of SCR maximization problem under TDMA is:

$$(P1): Q^*(\boldsymbol{h}) = \max_{\boldsymbol{x},\boldsymbol{\tau},\boldsymbol{a}} \quad Q(\boldsymbol{h}, \boldsymbol{x}, \boldsymbol{\tau}, \boldsymbol{a}) \quad (11a)$$

$$\text{s.t.} \quad 0 \leq \tau_{ij} \leq 1, \forall i \in \mathcal{N}, \forall j \in \mathcal{M}, \quad (11b)$$

$$0 \leq a_j \leq 1, \forall j \in \mathcal{M}, \quad (11c)$$

$$\sum_{i=1}^{N} \tau_{ij} + a_j \leq 1, \forall j \in \mathcal{M}, \quad (11d)$$

$$\sum_{j=0}^{M} x_{i,j} = 1, \forall i \in \mathcal{N}, \quad (11e)$$

$$x_{i,j} \in \{0, 1\}, \forall i \in \mathcal{N}, \forall j \in \mathcal{M} \cup \{0\}. \quad (11f)$$

(11b), (11c), and (11d) impose limitations on the offloading duration allocated to each WN, WPT duration, and the available time of each H-AP, respectively. Constraint (11e) ensures that every WN can only be linked to one H-AP or conducts computation locally. Note that, different from [15], each H-AP has its own WPT duration which is more flexible for WPT but more challenging to address.

The above problem is mixed-integer nonlinear programming (MINLP). Fortunately, given $\boldsymbol{x}$, it is transformed into the following M independent subproblem (P2) with respective to $AP_j (j = 1, 2, ..., M)$. Let $\mathcal{N}_j$ be the set of offloading WNs' indices which are associated to AP_j. Let $\mathcal{N}_j'$ be the set of local-computing WNs' indices whose nearest H-AP is AP_j.

$$(P2): Q_j^*(\boldsymbol{h}, \boldsymbol{x}) = \max_{\tau_{i,j}: i \in \mathcal{N}_j' \cup \mathcal{N}_j, a_j} \sum_{i \in \mathcal{N}_j'} r_{l,i} + \sum_{i \in \mathcal{N}_j} r_{o,ij} \tag{12a}$$

$$\text{s.t.} \quad 0 \le \tau_{ij} \le 1, \forall i \in \mathcal{N}_j, \tag{12b}$$

$$\sum_{i \in \mathcal{N}_j} \tau_{ij} + a_j \le 1, \tag{12c}$$

$$0 \le a_j \le 1. \tag{12d}$$

For any $i \in \mathcal{N}_j'$,

$$r_{l,i} = \eta_1 \left[h_{ij} a_j\right]^{1/3}. \tag{13}$$

For any $i \in \mathcal{N}_j$,

$$r_{o,ij} = \epsilon_1 \tau_{ij} \ln\left(1 + \frac{\eta_2 h_{ij}^2}{\tau_{ij}} a_j\right). \tag{14}$$

4.2 Problem Formulation of FDMA

The SCR maximization problem under FDMA can be formulated as (P3):

$$(P3): Q^*(\boldsymbol{h}) = \max_{\boldsymbol{x}, \boldsymbol{\tau}, \boldsymbol{a}} \quad Q(\boldsymbol{h}, \boldsymbol{x}, \boldsymbol{\tau}, \boldsymbol{a}) \tag{15a}$$

$$\text{s.t.} \quad \tau_{ij} + a_j \le 1, \forall j \in \mathcal{M}, \tag{15b}$$

$$(11b), (11c), (11e), (11f)$$

Problem (P3) is also a MINLP problem. Given $\boldsymbol{x}$, (P3) becomes M independent subproblem (P4) with respective to AP_j.

$$(P4): Q_j^*(\boldsymbol{h}, \boldsymbol{x}) = \underset{\tau_{i,j}: i \in \mathcal{N}_j' \cup \mathcal{N}_j, a_j}{\text{maximize}} \sum_{i \in \mathcal{N}_j'} r_{l,i} + \sum_{i \in \mathcal{N}_j} r_{o,ij} \tag{16a}$$

$$\text{s.t.} \quad 0 \le \tau_{ij} \le 1, \forall i \in \mathcal{N}_j, \tag{16b}$$

$$\tau_{ij} + a_j \le 1, \forall i \in \mathcal{N}_j, \tag{16c}$$

$$0 \le a_j \le 1. \tag{16d}$$

Here, $r_{l,i}$ is given in (13) and for any $i \in \mathcal{N}_j$,

$$r_{o,ij} = \epsilon_2 \tau_{ij} \ln\left(1 + \frac{\eta_2 h_{ij}^2}{\tau_{ij}} a_j\right). \tag{17}$$

4.3 The DRL Algorithm for Offloading Decision

For both two problems, Figure 2 shows the DRL framework which was proposed in our work [15]. Taking $\boldsymbol{h}$ as the input, the DNN outputs vector $\boldsymbol{o}$, where $\boldsymbol{o} = [o_{10}, ..., o_{1M}, ..., o_{N0}, ...o_{NM}]$. In the DRL algorithm, the state is $\boldsymbol{h} = [h_{11}, ..., h_{1M}, ..., h_{N1}, ...h_{NM}]$ and all possible states constitute the state space $\boldsymbol{H}$ where $\boldsymbol{h} \in \boldsymbol{H}$. The action is $\boldsymbol{o}$, and all possible actions constitute the action space $\boldsymbol{O}$ where $\boldsymbol{o} \in \boldsymbol{O}$. As our algorithm uses the DNN-based DRL approach just for obtaining the offloading decision, and the subproblem's optimal solution of $\boldsymbol{\tau}$ and $\boldsymbol{a}$ under a given offloading decision is obtained by our designed efficient approaches, the action dimension has been greatly reduced.

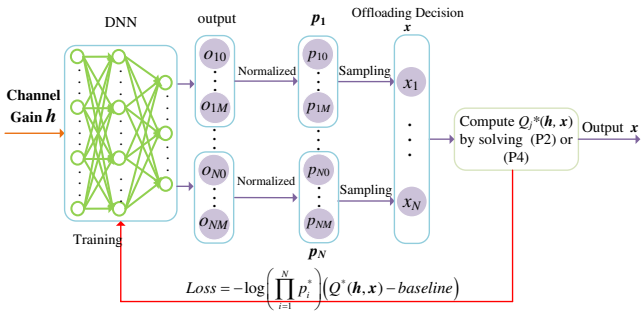

Fig. 2 The framework of the proposed algorithm.

Subsequently, taking every $M + 1$ continuous elements of $\boldsymbol{o}$ as a subset, we normalize each element in a subset. Then we have

$$\boldsymbol{p} = \begin{bmatrix} p_{10} & p_{20} & \cdots & p_{N0} \\ p_{11} & p_{21} & \cdots & p_{N1} \\ \cdots & \cdots & \cdots & \cdots \\ p_{1M} & p_{2M} & \cdots & p_{NM} \end{bmatrix}, \tag{18}$$

where $p_{ij} = \frac{o_{ij}}{\sum_{k=0}^{M} o_{ik}}$.

$\boldsymbol{p_i}$ denotes the offloading policy for WN_i, i.e., the probabilities of offloading data to different H-APs. The element p_{i0} denotes the probability that WN_i performs local computing, and $p_{ij} (j \neq 0)$ denotes the probability that WN_i offloads data to AP_j. Subsequently, we randomly sample the OD based on probability distribution $\boldsymbol{p_i}$, and obtain the sampling result $\boldsymbol{x}_i$ of WN_i.

$\boldsymbol{p_i}$ represents the offloading policy specifically tailored for WN_i. p_{i0} signifies the likelihood of WN_i opting for local computing. Conversely, $p_{ij} (j \neq 0)$ indicates the probability of WN_i offloading its data to AP_j. Following this, we conduct a random sampling of the OD guided by the probability distribution outlined in $\boldsymbol{p_i}$. This process yields $\boldsymbol{x}_i$, which serves as the sampling outcome for WN_i.

Let $p_i^* = p_{ij}$ which satisfy $x_{i,j} = 1$ in the sampling result. The value of p_i^* will be utilized to compute the loss for training the DNN based on the obtained OD.

The loss function is

$$Loss = -\log\left(\prod_{i=1}^{N} p_i^*\right) \times \left(Q^*\left(\boldsymbol{h}, \boldsymbol{x}\right) - baseline\right), \tag{19}$$

where the baseline is recent average SCR.

4.4 Algorithms for Subproblems (P2) and (P4)

4.4.1 Algorithms for Subproblems (P2) in TDMA

Lemma 1. *subproblem (P2) is convex.*

Proof. According to (13), it is clear that $r_{l,i}$ is a concave function of a_j. Additionally, $\epsilon_1 \ln\left(1+\eta_2 h_{ij}^2 a_j\right)$ is concave with respective to a_j. Since $r_{o,ij}$ in (14) is perspective of $\epsilon_1 \ln\left(1+\eta_2 h_{ij}^2 a_j\right)$, it is also concave with respect to τ_{ij} and a_j [37]. Note that the sum of concave functions is still concave. Additionally, all constraint functions are linear. This completes the proof. □

Since subproblem (P2) is convex, the available algorithms (like interior-point method) can be used to efficiently solve it. However, we present a more efficient algorithm for solving it. Below we introduce the approach for obtaining the optimal τ_{ij} for a given a_j. Having this approach, the optimal a_j can be obtained by the golden section search because (P2) is convex and its objective function is concave with respect to a_j [37].

For a given a_j, as $r_{l,i}$ becomes a constant, (P2) is simplified to

$$(P2.2): Q_j^*(\boldsymbol{h},\boldsymbol{x},a_j) = \max_{\tau_{i,j}:\mathcal{N}_j} \quad \sum_{i\in\mathcal{N}_j} r_{o,ij} \tag{20a}$$

$$\text{s.t.} \quad 0 \le \tau_{ij} \le 1, \forall i \in \mathcal{N}_j, \tag{20b}$$

$$\sum_{i\in\mathcal{N}_j} \tau_{ij} + a_j \le 1. \tag{20c}$$

Lemma 2. *The optimal time allocation of (P2.2) is*

$$\tau_{i,j}^* = \frac{\eta_2 h_{ij}^2}{\sum\limits_{i\in\mathcal{N}_j} \eta_2 h_{ij}^2}(1-a_j). \tag{21}$$

Proof. The Lagrangian of (P2.2) is

$$L(\boldsymbol{\tau}, v) = \sum_{i\in\mathcal{N}_j} r_{o,ij} - v\left(\sum_{i\in\mathcal{N}_j} \tau_{i,j} + a_j - 1\right). \tag{22}$$

Here, $v \ge 0$ is the Lagrange multiplier.

Then the dual function of (P2.2) is

$$G(v) = \min_{\boldsymbol{\tau}} L(\boldsymbol{\tau}, v). \tag{23}$$

Since (P2.2) is convex, the optimal solution satisfies the KKT conditions which are given by

$$\sum_{i\in\mathcal{N}_j} \tau_{i,j}^* + a_j \le 1, \tag{24}$$

$$v^*\left(\sum_{i\in\mathcal{N}_j} \tau_{i,j}^* + a_j - 1\right) = 0, \tag{25}$$

$$\frac{\partial L(\boldsymbol{\tau}, v)}{\partial \tau_{i,j}} = \frac{\partial \sum\limits_{i\in\mathcal{N}_j} r_{o,ij}}{\partial \tau_{i,j}} - v^*|_{\tau_{i,j}=\tau_{i,j}^*} = 0, i \in \mathcal{N}_j. \tag{26}$$

From (26) we have

$$t\left(\frac{\eta_2 h_{ij}^2}{\tau_{i,j}^*} a_j\right) = \frac{v^*}{\varepsilon_1}, i \in \mathcal{N}_j. \tag{27}$$

where

$$t(x) = \ln(1+x) - \frac{x}{1+x}. \tag{28}$$

As $t(x)$ is monotonically increasing, $\frac{\eta_2 h_{ij}^2}{\tau_{i,j}^*} a_j$ are equal for any $i \in \mathcal{N}_j$. Let $\frac{\eta_2 h_{ij}^2}{\tau_{i,j}^*} a_j = c$ and then $\tau_{i,j}^* = \frac{\eta_2 h_{ij}^2}{c} a_j$.

As $\sum\limits_{i\in\mathcal{N}_j} \tau_{i,j}^* + a_j = 1$ holds for the optimal solution, we have $\sum\limits_{i\in\mathcal{N}_j} \frac{\eta_2 h_{ij}^2}{c} a_j + a_j = 1$ and $c = \frac{a_j}{1-a_j} \sum\limits_{i\in\mathcal{N}_j} \eta_2 h_{ij}^2$. So we get the conclusion. □

4.4.2 Algorithms for Subproblems (P4) in FDMA

To efficiently solve subproblem (P4) in FDMA, we propose an algorithm based on the golden section search.

Let us analyze the property of (P4) firstly. According to (17), it is clear that for a given WPT duration a_j of AP_j, the offloading rate $r_{o,ij}$ increases with τ_{ij}. Therefore, the optimal $\tau_{ij}^* = 1 - a_j$.

So we have

$$r_{o,ij} = \epsilon_2(1-a_j)\ln\left(1 + \frac{\eta_2 h_{ij}^2}{(1-a_j)} a_j\right). \tag{29}$$

Lemma 3. $\sum_{i\in\mathcal{N}_j'} r_{l,i} + \sum_{i\in\mathcal{N}_j} r_{o,ij}$ *is a convex function of* a_j.

Proof. According to (13), it is clear that $r_{l,i}$ is a concave function of a_j.

Taking the second derivative of $r_{o,ij}$ in (29) with respect to a_j, we obtain:

$$\begin{aligned}\frac{\partial^2 r_{o,ij}}{\partial a_j^2} &= \frac{-\epsilon\eta_2 h_{ij}^2(\eta_2 h_{ij}^2 - 1)}{\left[1 + (\eta_2 h_{ij}^2 - 1)a_j\right]^2} - \\ &\quad \frac{\epsilon\eta_2 h_{ij}^2}{(1-a_j)^2 + \eta_2 h_{ij}^2 a_j(1-a_j)} \\ &= -\epsilon\eta_2 h_{ij}^2 \frac{\eta_2 h_{ij}^2}{(1-a_j)[1+(\eta_2 h_{ij}^2 - 1)a_j]^2} \\ &< 0. \end{aligned} \tag{30}$$

So $r_{o,ij}$ is a concave function of a_j. This completes the proof. □

To solve subproblem (P4), we propose Algorithm

Algorithm 1 The golden section search based optimization algorithm for (P4)

Require: $\boldsymbol{h}$, $\boldsymbol{x}$;
1: Initialize $a^{min}=0$, $a^{max}=1$, error tolerance $\gamma=10^{-8}$.
2: **while** $a^{max}-a^{min}\geq\gamma$ **do**
3: Initialize $\lambda=a^{min}+0.382(a^{max}-a^{min})$, $\mu=a^{min}+0.618(a^{max}-a^{min})$
4: For $a_j=\lambda$, substituting it into (13) and (29) and calculating the SCR Q; let $T_1=Q$;
5: For $a_j=\mu$, substituting it into (13) and (29) and calculating the SCR Q; let $T_2=Q$;
6: **if** $T_1>T_2$ **then**
7: $a^{max}=\mu$
8: **else**
9: $a^{min}=\lambda$
10: **end if**
11: **end while**
12: $a_j^*=(\lambda+\mu)/2$.

1, which applies the golden section search method to find the value of a_j^* under a given $\boldsymbol{x}$ and $\boldsymbol{h}$.

4.5 Computational Complexity

In the proposed algorithm, the offloading decision is determined by a DNN and subsequently undergoes normalization and sampling processes. The DNN has NM input nodes, a fixed number of nodes in each hidden layer, and NM output nodes. So the overall complexity of the operations in DNN is $O(NM)$. Also, the normalization and sampling processes require time $O(NM)$.

4.5.1 TDMA mode

We use the expression (21) of optimal time allocation for sub-problem (P2). The complexity is $O(N)$. As there are M sub-problems, the total complexity is $O(NM)$. Finally, the overall complexity is $O(NM)$.

4.5.2 FDMA mode

In Algorithm 1, the number of iterations of golden section search used for each sub-problem (P4) is $\mathcal{O}(\log_2(\frac{1}{\gamma}))$, which is a constant value. In each iteration, the calculation time of (13) and (29) for each node is constant. There are N nodes. So the complexity of each iteration from step 3 to step 10 is $O(N)$. As there are M sub-problems, the total complexity is $O(NM)$. Finally, the overall complexity is $O(NM)$.

5. Numerical Results

In the simulations, $P=3$ Watts, $\mu=0.8$, $B=2$ MHz, $N_0=10^{-10}$ Watts, $v_u=1.1$, $k=10^{-26}$, and $\phi=100$. WNs and H-APs are randomly distributed in an area of 12m*12m. $\alpha_{ij}=A_d\left(\frac{3\cdot10^8}{4\pi f_c d_{ij}}\right)^{d_e}$, where $A_d=4.11$, $f_c=915$ MHz, and $d_e=2.3$ [12]. $|g_{ij}|^2$ has an exponential distribution with unit mean. DNN utilized in our algorithm has 5 fully connected layers with 350, 450, 600, 700, and 800 neurons, respectively.

The exhaustive search (ES) is utilized for finding the optimal binary OD. ES algorithm traverses all possible ODs and uses our algorithm for solving the sub-problem, outputting the maximum SCR. Then we compare our DRL-based algorithm with ES by using the normalized computation rate $\overline{Q}(\boldsymbol{h})$ given below:

$$\overline{Q}(\boldsymbol{h})=\frac{Q^*(\boldsymbol{h})}{Q^*_{exhaustive}(\boldsymbol{h})}. \tag{31}$$

$Q^*(\boldsymbol{h})$ represents our algorithm's SCR, and $Q^*_{exhaustive}(\boldsymbol{h})$ represents ES'S SCR. Due to ES's large running time, only the cases of 3 H-APs and 5 WNs are investigated.

5.1 FDMA protocol

We first investigate the convergence speed of the proposed algorithm when using FDMA. Fig. 3 illustrates that during the training's beginning stage, the loss is relatively high. However, as the neural network continues to be trained, the loss value gradually decreases, indicating that the network is converging. Upon reaching the later stages of training, specifically after 5000 time slots, we observe that the loss value approaches 0.

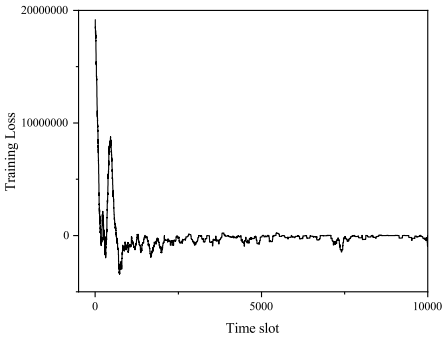

Fig. 3 The training loss for algorithm under learning rate 0.00095 when using FDMA.

Fig. 4 evaluates our algorithm under five different learning rates. The results demonstrate that when the learning rate is excessively high, such as $9.5{*}10^{-4}$, $\overline{Q}(\boldsymbol{h})$ is approximately 0.90. Conversely, when the learning rate is too low, such as $9.5*10^{-8}$, $\overline{Q}(\boldsymbol{h})$ is small and may need long time for learning. Therefore, we select $9.5*10^{-5}$ as the learning rate for our algorithm, so as to achieve a good balance between convergence speed and SCR, reaching approximately 96% of the maximum computation rate.

In our simulations, the baseline is the average of most recent K $Q^*(\boldsymbol{h})$. Fig. 5 investigates our algorithm's performance under different K. When $K=0$, $\overline{Q}(\boldsymbol{h})$ is approximately 0.8, which is much lower than

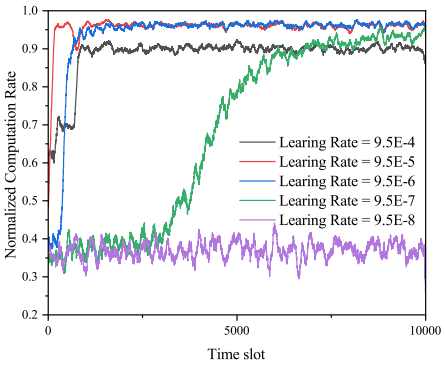

Fig. 4 $\overline{Q}(\boldsymbol{h})$ versus learning rate when using FDMA.

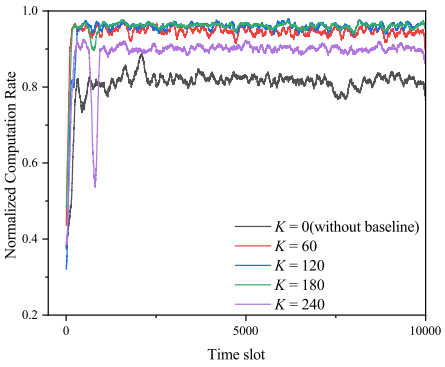

Fig. 5 $\overline{Q}(\boldsymbol{h})$ versus baseline when using FDMA.

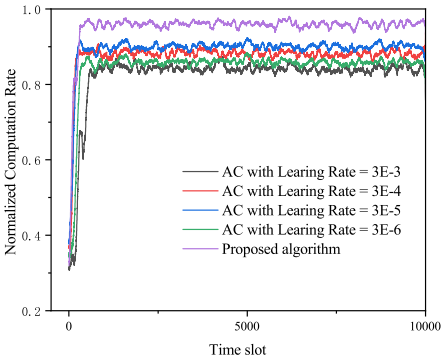

Fig. 6 Our algorithm versus AC algorithm when using FDMA.

the other curves. Based on Fig. 5, let $K = 120$.

Fig. 6 compares our algorithm with the classic Actor-Critic (AC) algorithm. We can see that, whatever the learning rate is, AC's SCR is consistently smaller than our algorithm. Moreover, our proposed algorithm can converge faster than AC.

Fig. 7 evaluates various offloading algorithms. If all WNs perform the task locally, the SCR is smallest, as the H-APs have much greater computation ability than the WNs. On the contrary, if all WNs offload data to their nearest H-APs, the achieved average SCR is still lower than that of our proposed algorithm. This is because, offloading tasks solely based on distance does

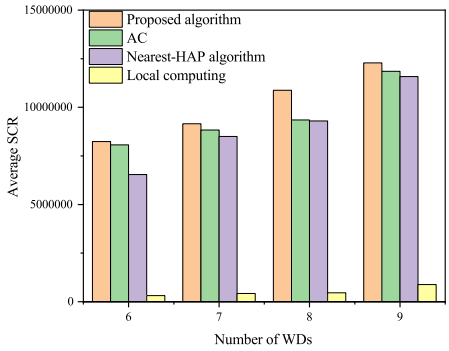

Fig. 7 Comparison of different algorithms when using FDMA.

not consider the workload at the nearest H-AP. Furthermore, there is still a performance gap between our algorithm and AC. This confirms that our proposed algorithm has an advantage over classic DRL algorithms.

5.2 TDMA protocol

We conducted an analysis of the convergence speed of proposed algorithm when using TDMA. As depicted in Fig. 8, as the training progresses, the loss value gradually diminishes, indicating DNN's convergence. After enough training, particularly after surpassing 5000 time slots, we observe the loss value approaching zero.

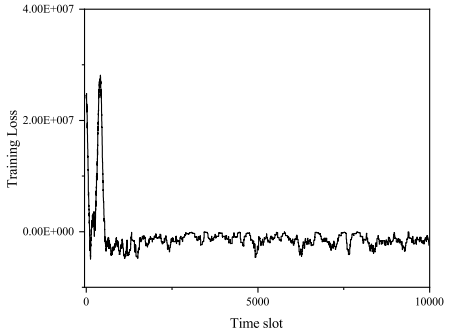

Fig. 8 The training loss for algorithm under different learning rate 0.00095 when using TDMA.

Fig. 9 presents a comparison of the proposed algorithm's performance using four different learning rates. The results indicate that an excessively high learning rate, such as $9.5 * 10^{-4}$, leads to the algorithm converging to a normalized computation rate of approximately 0.90. On the other hand, a very low learning rate, such as $9.5 * 10^{-7}$, may not significantly improve the algorithm's performance even with a large number of time slots. While a learning rate of $9.5 * 10^{-6}$ can achieve near-optimal convergence, the slow convergence speed limits the algorithm's performance. Based on these observations, we select a learning rate of $9.5 * 10^{-5}$ for our algorithm. This choice allows for faster convergence while maintaining a high level of performance, reaching

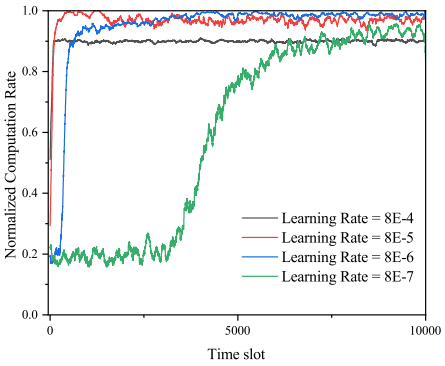

Fig. 9 $\overline{Q}(\boldsymbol{h})$ versus learning rate when using TDMA.

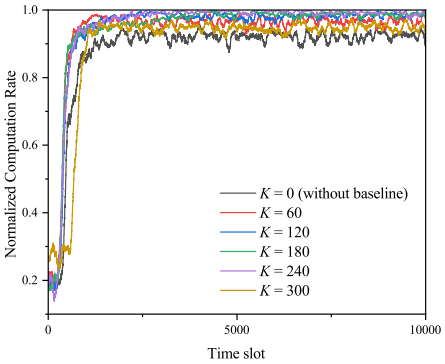

Fig. 10 $\overline{Q}(\boldsymbol{h})$ versus baseline when using TDMA.

approximately 96% of the maximum computation rate.

Fig. 10 examines the algorithm's performance under different K. When $K = 0$, indicating the absence of a baseline, $\overline{Q}(\boldsymbol{h})$ is approximately 0.90. Following this figure, let $K = 120$.

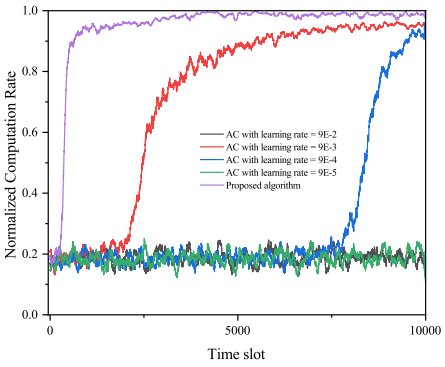

Fig. 11 Our algorithm versus AC algorithm when using TDMA.

Fig. 11 compares our algorithm with AC. As depicted in Fig. 11, despite setting different learning rates for the critic network in AC, its performance consistently lags behind that of the proposed algorithm. Furthermore, our proposed algorithm exhibits faster convergence compared to AC.

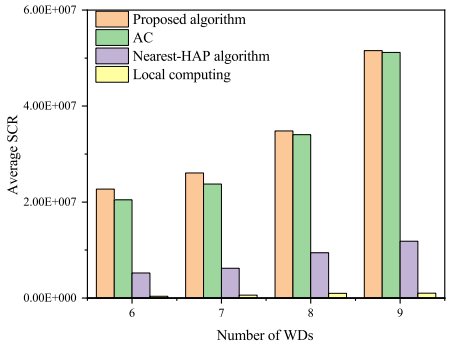

Fig. 12 Comparison of different algorithms when using TDMA.

Fig. 12 evaluates various offloading algorithms. If all WNs perform the task locally, the SCR is smallest. On the contrary, if all WNs offload data to their nearest H-APs, the achieved average SCR is also significantly lower than that of our proposed algorithm. Furthermore, there is still a performance gap between our algorithm and AC.

Table 1 Running time of different algorithms.

WNs number	ES algorithm	Our algorithm	AC algorithm
5	0.1813s	0.0055s	0.0076
6	0.8179s	0.0062s	0.0089s
7	3.2327s	0.0071s	0.0099s
8	12.5052s	0.0094s	0.0109s
9	50.8865s	0.0108s	0.0122s

Table I shows the average running time of our algorithm and ES. ES's execution time shows significant growth as N increases, as the execution time exponentially increases as N grows. In contrast, our proposed algorithm exhibits a roughly linear computational complexity in relation to N. Additionally, AC's average CPU running time is approximately 1.4 times longer than our proposed algorithm. This disparity arises from the fact that the AC algorithm requires the maintenance of two networks, and computing state-value by critic network takes more time compared to our algorithm, which only maintains a single policy network.

6. Conclusions

This paper tackles the SCR maximization in a WP-MEC system having multiple H-APs. The study focuses on binary offloading and proposes DRL-based algorithms for TDMA and FDMA, respectively. Our algorithms demonstrate rapid convergence and achieve approximately the maximum SCR while maintaining low complexity. Furthermore, comparing to the commonly used AC algorithm, our algorithms surpass the

AC algorithm in terms of computation rate, convergence speed, and running time.

Some possible future works are as follows. Firstly, it presupposes the availability of perfect CSI, whereas the scenario with imperfect CSI is more practical. Secondly, we have focused on the single-slot determinant optimization, whereas optimizing the long-term SCR, which poses greater challenges, merits further attention and endeavors.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (62272414), and the Researchers Supporting Project Number (RSP2024R102), King Saud University, Riyadh, Saudi Arabia.

References

[1] D. C. Nguyen, M. Ding, P. N. Pathirana, et al. "6G Internet of Things: A comprehensive survey," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 359-383, 2021.

[2] X. Wang, Y. Han, V. C. M. Leung, D. Niyato, X. Yan and X. Chen, "Convergence of Edge Computing and Deep Learning: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 869-904, 2020.

[3] X. Wang, J. Li, Z. Ning, et al. "Wireless powered mobile edge computing networks: A survey," *ACM Computing Surveys*, 2023.

[4] K. Chi, Y. Zhu, Y. Li, L. Huang, and M. Xia. "Minimization of transmission completion time in wireless powered communication networks," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp.1671-1683, 2017.

[5] B. Zhu, K. Chi, J. Liu, K. Yu, and S. Mumtaz, "Efficient offloading for minimizing task computation delay of NOMA-based multiaccess edge computing," *IEEE Transactions on Communications*, vol. 70, no. 5, pp. 3186-3203, 2022.

[6] H. Guo, J. Liu, J. Zhang, W. Sun, N. Kato, "Mobile-edge computation offloading for ultradense IoT networks," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 4977-4988, 2018.

[7] J. Wang, L. Zhao, J. Liu, N. Kato, Smart resource allocation for mobile edge computing: A deep reinforcement learning approach. *IEEE Transactions on emerging topics in computing*, vol. 9, no. 3, pp. 1529-1541, 2021.

[8] X. Zhu, F. Ma, F. Ding, Z. Guo, J. Yang and K. Yu, "A Low-latency Edge Computation Offloading Scheme for Trust Evaluation in Finance-Level Artificial Intelligence of Things," *IEEE Internet of Things Journal*, doi: 10.1109/JIOT.2023.3297834.

[9] P. Zhang, N. Chen, G. Xu, N. Kumar, A. Barnawi, M. Guizani, Y. Duan and K. Yu, "Multi-Target-Aware Dynamic Resource Scheduling for Cloud-Fog-Edge Multi-Tier Computing Network," *IEEE Transactions on Intelligent Transportation Systems*, doi: 10.1109/TITS.2023.3330419.

[10] J. -H. Syu, J. C. -W. Lin, G. Srivastava and K. Yu, "A Comprehensive Survey on Artificial Intelligence Empowered Edge Computing on Consumer Electronics," *IEEE Transactions on Consumer Electronics*, doi: 10.1109/TCE.2023.3318150.

[11] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4177-4190, 2018.

[12] L. Huang, S. Bi and Y. -J. A. Zhang, "Deep Reinforcement Learning for Online Computation Offloading in Wireless Powered Mobile-Edge Computing Networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 11, pp. 2581-2593, 2020.

[13] M. Zeng, R. Du, V. Fodor, C. Fischione, "Computation rate maximization for wireless powered mobile edge computing with NOMA," *2019 IEEE WoWMoM*, 2019: 1-9.

[14] M. Poposka, Z. Hadzi-Velkov, "Edge Computing: System Overview and Fusion with Wireless Power Transfer," 2021 International Balkan Conference on Communications and Networking (BalkanCom), pp. 51-55, 2021.

[15] S. Zhang, S. Bao, K. Chi, K. Yu, and S. Mumtaz, "DRL-Based Computation Rate Maximization for Wireless Powered Multi-AP Edge Computing," *IEEE Transactions on Communications*, vol. 72, no. 2, pp. 1105-1118, 2024.

[16] S. Zhang, H. Gu, K. Chi, L. Huang, K. Yu and S. Mumtaz, "DRL-Based Partial Offloading for Maximizing Sum Computation Rate of Wireless Powered Mobile Edge Computing Network," *IEEE Transactions on Wireless Communications*, vol. 21, no. 12, pp. 10934-10948, 2022.

[17] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 9, pp. 1927-1941, 2018.

[18] X. Zhou, L. Huang, T. Ye, and W. Sun, "Computation bits maximization in UAV-assisted MEC networks with fairness constraint," *IEEE Internet of Things Journal*, vol. 9, no. 21, pp. 20997-21009, 2022.

[19] S. Mao, N. Zhang, L. Liu, J. Wu, M. Dong, K. Ota, T. Liu, D. Wu, "Computation Rate Maximization for Intelligent Reflecting Surface Enhanced Wireless Powered Mobile Edge Computing Networks," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 10, pp. 10820-10831, 2021.

[20] P. Chen, B. Lyu, Y. Liu, H. Guo and Z. Yang, "Multi-IRS Assisted Wireless-Powered Mobile Edge Computing for Internet of Things," *IEEE Transactions on Green Communications and Networking*, vol. 7, no. 1, pp. 130-144, 2023.

[21] L. Shi, Y. Ye, X. Chu, and G. Lu, "Computation energy efficiency maximization for a NOMA-based WPT-MEC network," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10731-10744, 2021.

[22] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784-1797, 2018.

[23] F. Wang, H. Xing, and J. Xu, "Real-time resource allocation for wireless powered multiuser mobile edge computing with energy and task causality," *IEEE Transactions on Communications*, vol. 68, no. 11, pp. 7140-7155, 2020.

[24] R. Malik and M. Vu, "On-request wireless charging and partial computation offloading in multi-access edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 20, no. 10, pp. 6665-6679, 2021.

[25] C. Zhao, S. Xu and J. Ren, "AoI-Aware Wireless Resource Allocation of Energy-Harvesting-Powered MEC Systems," *IEEE Internet of Things Journal*, vol. 10, no. 9, pp. 7835-7849, 2023.

[26] G. Xu, H. Liu, Z. Dai and L. Li, "Joint Offloading and Energy Harvesting Design in Multiple Time Blocks for FDMA Based Wireless Powered MEC," 2021 International Conference on Intelligent Computing, Automation and Systems (ICICAS), Chongqing, China, pp. 487-493, 2021.

[27] P. X. Nguyen, D.-H. Tran, O. Onireti, P. T. Tin, S. Q. Nguyen, S. Chatzinotas, and H. Vincent Poor,

"Backscatter-assisted data offloading in OFDMA-based wireless-powered mobile edge computing for IoT networks," *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9233-9243, 2021.

[28] F. Zhou and R. Q. Hu, "Computation efficiency maximization in wireless-powered mobile edge computing networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3170-3184, 2020.

[29] K. Zheng, G. Jiang, X. Liu, K. Chi, X. Yao, J. Liu, "DRL-based offloading for computation delay minimization in wireless-powered multi-access edge computing," *IEEE Transactions on Communications*, vol. 71, no. 3, pp. 1755-1770, 2023.

[30] J. Liu, K. Xiong, D. Ng, P. Fan, Z. Zhong, "Optimal design of wireless-powered hierarchical fog-cloud computing networks," *2019 IEEE GLOBECOM*, 2019: 1-6.

[31] J. Liu, K. Xiong, D. Ng, P. Fan, Z. Zhong, K. Letaief, "Max-min energy balance in wireless-powered hierarchical fog-cloud computing networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 11, pp. 7064-7080, 2020.

[32] M. Wu, W. Qi, J. Park, P. Lin, L. Guo and I. Lee, "Residual Energy Maximization for Wireless Powered Mobile Edge Computing Systems With Mixed-Offloading," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 4, pp. 4523-4528, 2022.

[33] D. Samardzija, N. Mandayam, "Unquantized and uncoded channel state information feedback in multiple-antenna multiuser systems," *IEEE Transactions on Communications*, vol. 54, no. 7, pp. 1335-1345, 2006.

[34] T. Marzetta, B. Hochwald, "Fast transfer of channel state information in wireless systems," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1268-1278, 2006.

[35] X. Chen, C. Yuen, and Z. Zhang, "Wireless Energy and Information Transfer Tradeoff for Limited Feedback Multi-antenna Systems with Energy Beamforming," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 1, pp. 407-412, 2014.

[36] M. Pei, A. L. Swindlehurst, D. Ma, J. Wei, "Adaptive limited feedback for MISO wiretap channels with cooperative jamming," *IEEE Transactions on Signal Processing*, vol. 62, no. 4, pp. 993-1004, 2014.

[37] Stephen Boyd and Lieven Vandenberghe, Convex optimization. Cambridge University Press, 2004.

Guanqun Shen is with the School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His current research focuses on wireless-powered edge computing, wireless networks, etc.

Kaikai Chi received the B.S. and M.S. degrees from Xidian University, Xi'an, China, in 2002 and 2005, respectively, and the Ph.D. degree from Tohoku University, Sendai, Japan, in 2009. He is currently a professor in the School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, China. His current research focuses on wireless cellular network, wireless ad hoc network and wireless sensor network. He was the recipient of the Best Paper Award at the IEEE Wireless Communications and Networking Conference in 2008. He has published more than 50 referred technical papers in proceedings and journals like IEEE Transactions on Wireless Communications, IEEE Transactions on Vehicular Technology, IEEE Transactions on Parallel and Distributed Systems, etc.

Osama Alfarraj received the masters and Ph.D. degrees in information and communication technology from Griffith University, in 2008 and 2013, respectively. He is currently a Professor of computer sciences with King Saudi University, Riyadh, Saudi Arabia. His current research interests include eSystems (eGov, eHealth, and ecommerce), cloud computing, and big data. For two years, he has served as a Consultant and a member for the Saudi National Team for Measuring E-Government, Saudi Arabia.

Amr Tolba received the M.Sc. and Ph.D. degrees from the Mathematics and Computer Science Department, Faculty of Science, Menoufia University, Egypt, in 2002 and 2006, respectively. He is currently a Full Professor in computer science with King Saud University (KSU), Saudi Arabia. He has authored/coauthored over 180 scientific articles in top-ranked (ISI) international journals, such as IEEE INTERNET OF THINGS JOURNAL, ACM TOIT, IEEE SYSTEMS JOURNAL, etc. He served as a TPC Member at several conferences, such as DSIT 2022, CICA2022, EAI MobiHealth 2021, DSS 2021, etc. He has been included in the list of the top 2% of influential researchers globally (prepared by scientists from Stanford University, USA) during the calendar years 2020, 2021, 2022, and 2023, respectively. His main research interests include artificial intelligence (AI), the Internet of Things (IoT), data science, and cloud computing.