Computation Energy Efficiency Maximization for a NOMA-Based WPT-MEC Network

Liqin Shi, Yinghui Ye, *Member, IEEE*, Xiaoli Chu, *Senior Member, IEEE*, and Guangyue Lu

***Abstract*—Emerging smart Internet-of-Things (IoT) applications are increasingly relying on mobile-edge computing (MEC) networks, where the energy efficiency (EE) of computation is one of the most pertaining issues. In this article, considering the limited computation capacity at the MEC server and a practical nonlinear energy harvesting (EH) model for IoT devices, we propose a scheme to maximize the system computation EE (CEE) of a wireless power transfer (WPT) enabled nonorthogonal multiple access (NOMA)-based MEC network by jointly optimizing the computing frequencies and execution time of the MEC server and the IoT devices, the offloading time, the EH time and the transmit power of each IoT device, as well as the transmit power of the power beacon (PB). We formulate the joint optimization into a nonlinear fractional programming problem and devise a Dinkelbach-based iterative algorithm to solve it. By means of convex theory, we derive closed-form expressions for parts of the optimal solutions, which reveal several instrumental insights into the maximization of the system CEE. In particular, the system CEE increases as the optimal computing frequencies of both the IoT devices and the MEC server decrease, and the system CEE is maximized when the MEC server and the IoT devices use the maximum allowed time to complete their computing tasks. Simulation results demonstrate the superiority of the proposed scheme over benchmark schemes in terms of system CEE.**

***Index Terms*—Computation energy efficiency (CEE), mobile-edge computing (MEC), nonorthogonal multiple access (NOMA), wireless power transfer (WPT).**

I. Introduction

IN THE era of Internet of Things (IoT), there will be massive smart devices irregularly deployed in various communication systems, e.g., intelligent agriculture and smart home automation, to monitor, generate data, and process the data timely for intelligent services [1], [2]. However, owing to the stringent device size constraint and production cost consideration, the IoT devices are usually energy-constrained and computation limited [3]–[6], and thus how to efficiently solve the above two major limitations is of importance for the application of IoT.

Wireless power transfer (WPT) [3], [4], [7] and mobile-edge computing (MEC) [5], [6] are two promising technologies to prolong the device lifetime and enhance the device computation capacity, respectively. The key idea of WPT is to let the energy source, e.g., power beacon (PB), base station, to charge the IoT devices on demand via microwave irradiation. For example, Zhou *et al.* [7] considered an energy harvesting (EH)-based cognitive Machine-to-Machine (M2M) communication system underlaying a single-cell cellular network, where multiple M2M transmitters harvest energy from ambient radio frequency (RF) signals and investigated the energy efficiency (EE) maximization problem. While in MEC, the IoT devices are able to offload their partial tasks to nearby MEC servers with more computation capabilities so that the tasks can be successfully processed within the delay budge. However, only using WPT or MEC cannot address the energy-constrained and computation-limited problems simultaneously in IoT systems, and this motivated us to combine the above two advanced technologies together.

To date, there are a considerable number of studies on the combination of WPT and MEC [8]–[22]. You *et al.* [8] maximized the successful computation probability for a WPT-MEC network with a single edge user (EU) by proposing a binary computation offloading scheme, where each task is either computed locally or completely offloaded as a whole. Note that in this work, the IoT device and the EU are used interchangeably. Then the work in [8] was extended into a multiple EUs scenario where the weighted sum computation bits were maximized by using convex theory [9] and deep learning [10]. Recently, partial offloading schemes, where a task can be divided into independent parts for offloading or local computing, were proposed [11]–[17]. In [11], the partial offloading decisions, computation resource allocation, and the trajectory of the unmanned aerial vehicle (UAV) were jointly designed to maximize the weighted sum computation bits in a UAV assisted wireless powered MEC network. The energy consumption of the MEC server was minimized subject to the energy-causality constraint and the maximum computation latency constraint in the EU noncooperation scenario [12] and in the EU cooperation scenario [13].

To evaluate the tradeoff between the computation bits and the energy consumption, the authors in [14]–[17] introduced a new performance metric, called the computation EE (CEE), into wireless powered MEC networks and defined it as the ratio of the computation bits to the energy consumption for

Manuscript received August 3, 2020; revised October 24, 2020 and November 19, 2020; accepted December 30, 2020. Date of publication January 4, 2021; date of current version June 23, 2021. This work was supported in part by the Science and Technology Innovation Team of Shaanxi Province for Broadband Wireless and Application under Grant 2017KCT-30-02, and in part by the Natural Science Foundation of Shaanxi Province under Grant 2021JQ-713. *(Corresponding author: Yinghui Ye.)*

Liqin Shi, Yinghui Ye, and Guangyue Lu are with the Shaanxi Key Laboratory of Information Communication Network and Security, Xi'an University of Posts and Telecommunications, Xi'an 710121, China (e-mail: liqinshi@hotmail.com; connectyyh@126.com; tonylugy@163.com).

Xiaoli Chu is with the Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield S10 2TN, U.K. (e-mail: x.chu@sheffield.ac.uk).

Digital Object Identifier 10.1109/JIOT.2020.3048937

communication and computation. Mao *et al.* [14] proposed a joint computation offloading and resource allocation algorithm to maximize the minimum CEE among EUs in wireless powered full-duplex MEC systems. Lim and Hwang [15] maximized the CEE of all the EUs for a wireless powered MEC network. The minimum CEE among EUs was maximized in a two-EU WPT-MEC network [16] and in a wireless powered MEC network [17]. In [14]–[17], EUs adopted orthogonal multiple access (OMA) to offload tasks to the MEC server.

To better support computation offloading in WPT-MEC networks, the spectral efficiency and EE of transmission links need to be enhanced. Since nonorthogonal multiple access (NOMA) can offer a spectral efficiency gain over OMA, NOMA has been recently considered for task offloading in MEC and WPT-MEC networks [18]–[22]. For a NOMA-based WPT-MEC network, the max-min CEE problem for the EUs was investigated under a partial offloading scheme in [21] and under both partial and binary offloading schemes in [22].

However, there are limitations in the above existing works on CEE maximization [14]–[17], [21], [22], which are listed as follows.

1) The CEE maximization-based resource allocation scheme can be designed from the EU's perspective or the system's perspective [23]. We note that most existing studies [14]–[17], [21], [22] focused on the CEE maximization from the user's perspective, i.e., maximizing the CEE of all the EUs [15], or solving the max-min CEE problem for improving fairness among EUs [14], [16], [17], [21], [22]. There has not been any work reported on designing the CEE maximization-based resource allocation scheme for a WPT-MEC network from the system's perspective.
2) All the existing works [14]–[17], [21], [22] assumed that the computing capacity of the MEC server is unlimited and the execution time at the MEC server is negligible. However, in practice, although the MEC server has a more powerful computing capacity than that of the EUs, it may still take nonignorable time to execute the received tasks [18].

In this article, we consider a limited computing capacity and nonnegligible execution time at the MEC server, and study the CEE problem of a WPT-MEC network from the system's perspective. Following [21] and [22], we employ uplink NOMA for task offloading at each EU.

Our contributions are summarized as follows.

1) We study the CEE maximization for a NOMA-based WPT-MEC network under the partial offloading scheme from the system's perspective while considering the computation resource allocation of the MEC server and a practical nonlinear EH model for the EUs. More specifically, we propose to maximize the system CEE by jointly optimizing the computing frequencies and execution time of the MEC server and the EUs, and the transmit power, offloading time and EH time of EUs., as well as the transmit power of the PB. This joint optimization is formulated into a nonconvex fractional programming problem.

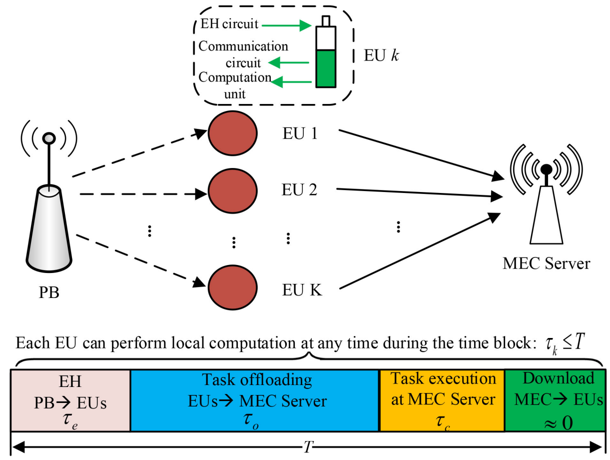


Fig. 1. Frame structure of the considered network.

2) To solve the formulated nonconvex fractional programming problem, we develop a Dinkelbach-based iterative algorithm to obtain the optimal resource allocation scheme. Besides, we derive closed-form expressions for parts of the optimal solutions by means of convex theory. Based on the derived results, we obtain several key insights into the maximization of the system CEE as follows. First, the system CEE increases as the optimal computing frequencies of both the EUs and the MEC server decrease. Second, the system CEE is maximized when the total task bits offloaded by all the EUs equal the maximum computation bits for the MEC server during the task execution phase, and the MEC server and the EUs use the maximum allowed time to complete their computing tasks, e.g., each EU performs local computing throughout each time block.

The remainder of this article is organized as follows. The system model is presented in Section II. Sections III presents a system CEE maximization problem by jointly optimizing the computing frequencies and execution time of the MEC server and the EUs, and the transmit power, offloading time and EH time of EUs, etc. and provides a Dinkelbach-based iterative algorithm to obtain the optimal solutions, as well as shows several instrumental insights into the maximization of the system CEE. Simulation results are provided in Section IV. This article is concluded in Section V.

II. System Model

As shown in Fig. 1, we consider a NOMA-based WPT-MEC network that consists of one MEC server, one PB and K EUs, each equipped with a rechargeable battery. Following [9], [11], and [20]–[22], we assume that each device is equipped with a single antenna. Following the "harvest-then-transmit" protocol, in each transmission block, the K EUs first harvest energy from the RF signals transmitted by the PB, and then use the harvested energy to process and offload their tasks, thus avoiding consuming the energy in their batteries and prolonging the operation time of each EU.[1] Accordingly, we assume that in each transmission block, the energy consumed at each EU for processing and offloading their tasks is less than the energy harvested from the RF signals transmitted by the PB [8]–[11], [17], [22]. Assuming the EU transceivers working in the half-duplex mode [20], [22], task offloading can only start after the EH has finished. We assume that the data bits of each task are bitwise independent [11]–[14], [17]–[19], [22] and the partial offloading scheme can be used for efficient computation within a given time block T. Let g_k ($k \in \{1, 2, \ldots, K\}$) and h_k denote the channel power gains of the PB-to-the kth EU link and the MEC server-to-the kth EU, respectively. All the channels are modeled as quasi-static fading, i.e., remain static within T but may change between adjacent time blocks. Assuming perfect channel state information available at the MEC server, the MEC server determines the optimal resource allocation scheme. Following [8]–[15] and [20]–[22], we assume that all the devices in the considered system are time synchronized.

The entire time block T is divided into four phases. In the first phase of duration τ_e, PB broadcasts energy signals and the K EUs work in the EH mode. The second phase of duration τ_o is used for task offloading, where the K EUs offload parts of their tasks to the MEC server via uplink NOMA. The third phase of duration τ_c is the task execution phase, in which the K EUs stop offloading tasks and the MEC server executes all the received computation tasks. In the fourth phase, the MEC server sends the computation results to the EUs, where we assume that the downlink transmission time is negligible as the size of the computation results is much smaller than that of the task data [11]–[14], [17]–[20], [22]. Accordingly, the fourth phase of each time block will be ignored hereafter. Note that during the second to the fourth phase, the PB keeps silent. Following [9], [11], and [22], we assume that each EU can perform local computation at any time during the time block as each EU can have separate circuits for the computation unit and the transmission unit.

A. Energy Harvesting Phase

In this phase, the PB transmits energy signals to the K EUs with the transmit power P_t, and each EU works in the EH mode. The existing works, e.g., [24]–[26], have shown that resource allocation schemes designed under the linear EH model will lead to a significant performance loss in practice owing to the mismatching between the linear EH model and the nonlinear behavior of EH circuits. This motivates us to consider a nonlinear EH model, i.e., the piecewise linear EH model with $N+1$ segments [24], to characterize the energy harvester at each EU. Note that different EH models are designed based on different functions and have different accuracies. Compared with the nonlinear EH models in [25] and [26], the piecewise linear EH model with $N \geq 3$ is more accurate.

[1] In this work, we focus on the computation-intensive scenarios where the EUs are not able to compute all their computation bits locally within a given delay budget and have to offload partial data to the MEC server [21], [22]. It is worth noting that in scenarios where some EUs can process all their computation bits locally within a given delay budget, allowing such EUs to perform completely local computing may further improve the system CEE, which is outside the scope of this work.

Besides, each segment of the piecewise linear EH model is given by a simple linear function, which facilitates analytical tractability in the design of the optimal resource allocation scheme. Thus, we employ the piecewise linear EH model in this work.

Based on [24], the harvested power at the kth EU is given by

$$P_h^k = \begin{cases} 0, & P_{\mathrm{RF}}^k \in \left[P_{\mathrm{th}}^0, P_{\mathrm{th}}^1\right] \\ a_{j_k} P_{\mathrm{RF}}^k + b_{j_k}, & P_{\mathrm{RF}}^k \in \left[P_{\mathrm{th}}^{j_k}, P_{\mathrm{th}}^{j_k+1}\right], j_k = 1, \ldots, N-1 \\ P_m^k, & P_{\mathrm{RF}}^k \in \left[P_{\mathrm{th}}^N, P_{\mathrm{th}}^{N+1}\right] \end{cases} \tag{1}$$

where $P_{\mathrm{RF}}^k = P_t g_k$ is the received RF power at the kth EU; $P_{\mathrm{th}} = \{P_{\mathrm{th}}^{j_k} | 0 \leq j_k \leq N+1\}$ with $P_{\mathrm{th}}^0 = 0$ and $P_{\mathrm{th}}^{N+1} = +\infty$ denotes thresholds on P_{RF}^k for the $N+1$ linear segments; a_{j_k} and b_{j_k} denote the slope and the intercept of the linear function in the j_kth ($j_k \in \{1, \ldots, N-1\}$) segment at the kth EU, respectively, and P_m^k is the maximum harvestable power at the kth EU when the EH circuit is saturated. Note that P_{th}^1 also denotes the circuit sensitivity of the EH circuit (i.e., the minimum required received power). For convenience, we let a_0 (a_N) and b_0 (b_N) be the slope and the intercept for the 0th (Nth) segment. Since in the 0th (Nth) segment, the harvested power is 0 (P_m^k) for any $P_{\mathrm{RF}}^k \in [P_{\mathrm{th}}^0, P_{\mathrm{th}}^1]$ ($P_{\mathrm{RF}}^k \in [P_{\mathrm{th}}^N, P_{\mathrm{th}}^{N+1}]$), we have $a_0 = b_0 = 0$ ($a_N = 0$ and $b_N = P_m^k$). Based on (1), the total harvested energy at the kth EU can be computed as $E_h^k = \tau_e P_h^k$.

B. Task Offloading Phase

In this phase, K EUs offload parts of their tasks to the MEC server simultaneously via uplink NOMA. The MEC server performs successive interference cancelation (SIC) to obtain each EU's task. Based on the principle of uplink NOMA, the MEC server decodes the message from the EU with the best channel condition first, subtracts the decoded message from the received composite signal, and then continues to decode the message from the EU with the next best channel condition [18]. Accordingly, we assume that $\{h_k\}_{k=1}^K$ is ranked in the descending order, i.e., $h_1 \geq h_2 \geq \cdots \geq h_K$. For the kth EU, the offloaded task is denoted by c_k, $k \in \{1, \ldots, K\}$. After the MEC server has decoded c_k and subtracted it from the received composite signal, it continues to decode c_{k+1}, until all K received tasks are decoded. Such a decoding order allows decoding the weakest EU's message without interference, thus maximizing the sum uplink transmission throughput. Note that the system CEE under different decoding orders may be different and including the decoding order in the joint optimization may further improve the system CEE, while this is outside the scope of this work and will be studied in our future work. Denote the achievable throughput for the kth EU by R_o^k, which can be calculated as

$$R_o^k = \tau_o B \log_2\left(1 + \frac{p_k h_k}{\sum_{i=k+1}^K p_i h_i + \sigma^2}\right) \tag{2}$$

where B is the bandwidth of uplink NOMA, p_k denotes the transmit power of the kth EU, and σ^2 is the noise power. Based on (2), we can first compute $R_o^K + R_o^{K-1}$ as

$$\begin{aligned} R_o^K + R_o^{K-1} &= \tau_o B\log_2\left(1 + \frac{p_K h_K}{\sigma^2}\right) \\ &\quad + \tau_o B\log_2\left(1 + \frac{p_{K-1}h_{K-1}}{p_K h_K + \sigma^2}\right) \\ &= \tau_o B\log_2\left(1 + \frac{p_{K-1}h_{K-1}}{p_K h_K + \sigma^2} + \frac{p_K h_K}{\sigma^2} + \frac{p_K h_K p_{K-1}h_{K-1}}{(p_K h_K + \sigma^2)\sigma^2}\right) \\ &= \tau_o B\log_2 \\ &\quad \times \left(1 + \frac{p_{K-1}h_{K-1}(\sigma^2 + p_K h_K) + (p_K h_K + \sigma^2)p_K h_K}{(p_K h_K + \sigma^2)\sigma^2}\right) \\ &= \tau_o B\log_2\left(1 + \frac{p_{K-1}h_{K-1} + p_K h_K}{\sigma^2}\right). \end{aligned} \tag{3}$$

Based on (3) and the expression of R_o^{K-2}, we can obtain $R_o^K + R_o^{K-1} + R_o^{K-2} = \tau_o B\log_2$ $(1+ [(p_{K-1}h_{K-1}+ p_K h_K+ p_{K-1}h_{K-1} + p_{K-2}h_{K-2})/\sigma^2])$. Continuing with such calculations, the total achievable throughput of the K EUs can be computed as

$$\begin{aligned} R_{\text{total}}^o &= \sum_{k=1}^{K} R_o^k = \tau_o B \sum_{k=1}^{K} \log_2\left(1 + \frac{p_k h_k}{\sum_{i=k+1}^{K} p_i h_i + \sigma^2}\right) \\ &= \tau_o B\log_2\left(1 + \sum_{k=1}^{K} \frac{p_k h_k}{\sigma^2}\right). \end{aligned} \tag{4}$$

C. Task Execution Phase

After successfully decoding the received tasks, the MEC server starts to execute the received tasks. Let f_m denote the central processing unit (CPU) frequency at the MEC server. Then the maximum bits computed by the MEC server during the task execution phase are given by

$$R_m = \frac{\tau_c f_m}{C_{\text{cpu}}^m} \tag{5}$$

where C_{cpu}^m is the number of CPU cycles required for computing 1 bit at the MEC server.

Let R_m^e denote the number of effective computation bits at the MEC server and R_m^e is determined by not only the total achievable throughput of all the EUs, but also the maximum computation bits at the MEC server. That is, when the computing time and frequency of the MEC server are large enough, i.e., $R_m > R_{\text{total}}^o$, R_m^e is determined by R_{total}^o. Otherwise, the MEC server cannot compute all the received tasks within the given time and R_m^e is equal to R_m. Accordingly, R_m^e is given by

$$R_m^e = \min\{R_m, R_{\text{total}}^o\} \tag{6}$$

where R_{total}^o is the total achievable throughput given in (4).

Let ε_m be the energy consumption coefficient (ECC) of the processor's chip at the MEC server. Then the energy consumption of the MEC server in this phase is given by [27]

$$E_m^e = \varepsilon_m f_m^3 \tau_c. \tag{7}$$

Note that each EU can perform local computation at any time during the time block. Let τ_k ($0 \le \tau_k \le T$) and f_k be the local computation time and the CPU frequency for the kth EU, respectively. Then the effective computation bits at the kth EU are calculated as

$$R_k^e = \frac{\tau_k f_k}{C_{\text{cpu}}^k} \tag{8}$$

where C_{cpu}^k is the number of CPU cycles required for computing 1 bit at the kth EU. Accordingly, the energy consumption for local computation at the kth EU is given by

$$E_k^e = \varepsilon_k f_k^3 \tau_k \tag{9}$$

where ε_k is the ECC of the processor's chip at the kth EU.

III. Computation Energy Efficiency Maximization

A. Problem Formulation

We define the system CEE of the considered network as the ratio of the total achievable computation bits of the whole network to the total energy consumption of the system. The total computation bits in a time block consist of the local computation bits completed by the K EUs and the bits computed at the MEC server, which can be given by $R_m^e + \sum_{k=1}^{K} R_k^e$. According to [8] and [11], the total energy consumption of the system in a time block consists of three parts, which are the energy consumed for the EH, the local computing and task offloading of the K EUs, and the information decoding and task computing of the MEC server. Thus, the system energy consumption in a time block can be computed as $[(P_t + P_{sc})\tau_e - \sum_{k=1}^{K} P_h^k \tau_e] + [P_{rc}\tau_o + \varepsilon_m f_m^3 \tau_c] + [\sum_{k=1}^{K}(p_k + p_{c,k})\tau_o + \sum_{k=1}^{K} \varepsilon_k f_k^3 \tau_k]$, where P_{sc} and P_{rc} denote the constant circuit power consumption of the PB during the EH phase and that of information decoding at the MEC server, respectively, and $p_{c,k}$ denotes the constant circuit power consumption of the kth EU during the task offloading phase.

Accordingly, the system CEE of the considered network is given by

$$\begin{aligned} &q_s\left(\tau_e, \tau_o, \tau_c, \{\tau_k\}_{k=1}^K, P_t, \{p_k\}_{k=1}^K, f_m, \{f_k\}_{k=1}^K\right) \\ &= \frac{\min\left\{\frac{\tau_c f_m}{C_{\text{cpu}}^m}, \tau_o B\log_2\left(1 + \sum_{k=1}^{K} \frac{p_k h_k}{\sigma^2}\right)\right\} + \sum_{k=1}^{K} \frac{\tau_k f_k}{C_{\text{cpu}}^k}}{(P_t + P_{sc})\tau_e - \sum_{k=1}^{K} E_h^k + P_{rc}\tau_o + \varepsilon_m f_m^3 \tau_c + \sum_{k=1}^{K}(p_k + p_{c,k})\tau_o + \sum_{k=1}^{K} \varepsilon_k f_k^3 \tau_k}. \end{aligned} \tag{10}$$

On this basis, we propose to maximize the system CEE of the NOMA-based WPT-MEC network under the nonlinear EH model, by jointly optimizing the transmit power of the PB and the EUs, the CPU frequencies and execution time of the MEC server and the EUs, the offloading time and the EH time. Accordingly, we formulate the system CEE maximization problem for the considered network as

$$\begin{aligned} \mathbf{P_0} :\ & \max_{\{p_k\}, \{f_k\}, \{\tau_k\}, \tau_e, \tau_o, \tau_c, f_m, P_t} q_s \\ \text{s.t.}\ & \text{C1} : R_m^e + \sum_{k=1}^{K} R_k^e \ge L_{\min} \\ & \text{C2} : (p_k + p_{c,k})\tau_o + \varepsilon_k f_k^3 \tau_k \le E_h^k \quad \forall k \\ & \text{C3} : \frac{p_k h_k}{\sum_{i=k+1}^{K} p_i h_i + \sigma^2} \ge \gamma_{\text{th}}^k\ \forall k \\ & \text{C4} : \tau_e + \tau_o + \tau_c \le T \end{aligned}$$

$$\text{C5}: 0 \le f_m \le f_{\max}, 0 \le f_k \le f_k^{\max} \quad \forall k$$
$$\text{C6}: 0 \le P_t \le P_{\max}, p_k \ge 0 \quad \forall k$$
$$\text{C7}: \tau_e, \tau_o, \tau_c \ge 0$$
$$\text{C8}: 0 \le \tau_k \le T \quad \forall k$$

where $L_{\min}$ denotes the minimum required computation bits of all the EUs; $f_k^{\max}$ and $f_{\max}$ are the maximum CPU frequencies for the kth EU and the MEC server, respectively; γ_{th}^k denotes the minimum required signal to interference and noise ratio (SINR) for the kth EU; $P_{\max}$ is the maximum transmit power for the PB.

In $\mathbf{P_0}$, constraint C1 guarantees the minimum required computation task bits of the whole system, where $L_{\min}$ can be adjusted to obtain a desirable tradeoff between the CEE and the total computation bits. C2 constrains that the total energy consumption at the kth EU should not exceed its total harvested energy over each EH phase. Note that it is not definite that each EU will use up all the harvested energy when the maximum system CEE is achieved and any residual harvested energy in the current time slot can be saved into its battery for future use. C3 is the minimum required SINR constraint for the kth EU. C4 constrains that all the offloaded computation task bits should be executed within T. C5 constrains the maximum CPU frequencies of each EU and the MEC server, while C6 is the constraint on the transmit power of the PB and each EU. C8 states that the local computation task bits at each EU should be executed within T.

It is worth noting that $\mathbf{P_0}$ is a typical nonconvex fractional optimization problem, where the coupling relationships between different optimization variables (i.e., P_t and τ_e, f_k and τ_k, etc.) exist in both the objective function and the constraints, making them nonconvex. In the next section, we design an efficient iterative algorithm to obtain the optimal solution to $\mathbf{P_0}$.

B. Solution and Iterative Algorithm

In order to deal with the coupling relationship between variables P_t and τ_e, we first divide both the numerator and the denominator of (10) by τ_e and then let $t_e = (1/\tau_e)$, $t_o = (\tau_o/\tau_e)$, $t_c = (\tau_c/\tau_e)$, and $t_k = (\tau_k/\tau_e)$. Correspondingly, the optimization problem $\mathbf{P_0}$ is reformulated as

$$\begin{aligned}
\mathbf{P}_1: \quad & \max_{\{p_k\},\{f_k\},\{t_k\},t_e,t_o,t_c,f_m,P_t} q_s^{(1)} \\
\text{s.t.} \quad & \text{C1}-1: \min \\
& \times \left\{\frac{t_c f_m}{C_{\text{cpu}}^m}, t_o B\log_2\left(1+\sum_{k=1}^{K}\frac{p_k h_k}{\sigma^2}\right)\right\} \\
& + \sum_{k=1}^{K}\frac{t_k f_k}{C_{\text{cpu}}^k} \ge L_{\min} t_e \\
& \text{C2}-1: (p_k + p_{c,k})t_o + \varepsilon_k f_k^3 t_k \le P_h^k \ \forall k \\
& \text{C3, C5, C6} \\
& \text{C4}-1: 1 + t_o + t_c \le T t_e \\
& \text{C7}-1: t_e, t_o, t_c \ge 0 \\
& \text{C8}-1: 0 \le t_k \le T t_e \quad \forall k
\end{aligned}$$

where

$$q_s^{(1)} = \frac{\min\left\{\frac{t_c f_m}{C_{\text{cpu}}^m}, t_o B\log_2\left(1+\sum_{k=1}^{K}\frac{p_k h_k}{\sigma^2}\right)\right\} + \sum_{k=1}^{K}\frac{t_k f_k}{C_{\text{cpu}}^k}}{P_t + P_{sc} - \sum_{k=1}^{K} P_h^k + P_{rc} t_o + \varepsilon_m f_m^3 t_c + \sum_{k=1}^{K}(p_k + p_{c,k}) t_o + \sum_{k=1}^{K}\varepsilon_k f_k^3 t_k}.$$

To further simplify the optimization problem $\mathbf{P}_1$, we introduce a slack variable λ ($\lambda \ge 0$), where $\lambda = \min\{[(t_c f_m)/(C_{\text{cpu}}^m)], t_o B\log_2(1+\sum_{k=1}^{K}[(p_k h_k)/(\sigma^2)])\}$, to remove the min function in the objective function and C1 − 1. Then, $\mathbf{P_1}$ is equivalently transformed into

$$\begin{aligned}
\mathbf{P}_2: \quad & \max_{\{p_k\},\{f_k\},\{t_k\},t_e,t_o,t_c,f_m,P_t,\lambda} q_s^{(2)} \\
\text{s.t.} \quad & \text{C1}-2: \lambda + \sum_{k=1}^{K}\frac{t_k f_k}{C_{\text{cpu}}^k} \ge L_{\min} t_e \\
& \text{C2}-1, \text{C3}, \text{C4}-1, \text{C5}, \text{C6}, \text{C8}-1 \\
& \text{C7}-2: t_e, t_o, t_c, \lambda \ge 0 \\
& \text{C9}: \frac{t_c f_m}{C_{\text{cpu}}^m} \ge \lambda \\
& \text{C10}: t_o B\log_2\left(1+\sum_{k=1}^{K}\frac{p_k h_k}{\sigma^2}\right) \ge \lambda
\end{aligned}$$

where

$$q_s^{(2)} = \frac{\lambda + \sum_{k=1}^{K}\frac{t_k f_k}{C_{\text{cpu}}^k}}{P_t + P_{sc} - \sum_{k=1}^{K} P_h^k + P_{rc} t_o + \varepsilon_m f_m^3 t_c + \sum_{k=1}^{K}(p_k + p_{c,k}) t_o + \sum_{k=1}^{K}\varepsilon_k f_k^3 t_k}.$$

Since the optimization problem $\mathbf{P_2}$ is still a nonconvex fractional optimization problem, based on Dinkelbach's method [28], we introduce Proposition 1 so that we can transform $\mathbf{P_2}$ into a more tractable optimization problem in the subtractive form.

Proposition 1: Let $\{p_k^*\}_{k=1}^K$, $\{f_k^*\}_{k=1}^K$, $\{t_k^*\}_{k=1}^K$, t_e^*, t_o^*, t_c^*, f_m^*, P_t^*, and λ^* denote the optimal solution to $\mathbf{P}_2$ and q^* be the corresponding maximized CEE of the considered network. Then the optimal solution can be obtained if and only if the following equation holds:

$$\begin{aligned}
& \max_{\{p_k\},\{f_k\},\{t_k\},t_e,t_o,t_c,f_m,P_t,\lambda} \lambda + \sum_{k=1}^{K}\frac{t_k f_k}{C_{\text{cpu}}^k} \\
& \quad - q^* E_{\text{total}}\left(\{p_k\}_{k=1}^K, \{f_k\}_{k=1}^K, \{t_k\}_{k=1}^K, t_e, t_o, t_c, f_m, P_t, \lambda\right) \\
& = \lambda^* + \sum_{k=1}^{K}\frac{t_k^* f_k^*}{C_{\text{cpu}}^k} - q^* E_{\text{total}}\Big(\{p_k^*\}_{k=1}^K, \{f_k^*\}_{k=1}^K, \{t_k^*\}_{k=1}^K \\
& \qquad t_e^*, t_o^*, t_c^*, f_m^*, P_t^*, \lambda^*\Big) = 0
\end{aligned} \tag{11}$$

where $E_{\text{total}}(\{p_k\}_{k=1}^K, \{f_k\}_{k=1}^K, \{t_k\}_{k=1}^K, t_e, t_o, t_c, f_m, P_t, \lambda) = P_t + P_{sc} - \sum_{k=1}^{K} P_h^k + P_{rc} t_o + \varepsilon_m f_m^3 t_c + \sum_{k=1}^{K}(p_k + p_{c,k}) t_o + \sum_{k=1}^{K}\varepsilon_k f_k^3 t_k$.

Proof: Proposition 1 can be proven based on the generalized fractional programming theory following a method similar to [28]. The detailed proof is omitted here for brevity. ■

According to Proposition 1, we develop a Dinkelbach-based iterative algorithm to obtain the optimal solution to $\mathbf{P_2}$, which

Algorithm 1 Dinkelbach-Based Iterative Algorithm for $\mathbf{P_2}$

```
Set the maximum error tolerance ε;
Set the iteration index l = 1 and the maximum system CEE q = 0;
repeat
    Solve P3 with a given q, and obtain the optimal solution, denoted by
    {{p_k^+}_{k=1}^K, {f_k^+}_{k=1}^K, {t_k^+}_{k=1}^K, t_e^+, t_o^+, t_c^+, f_m^+, P_t^+, λ^+};
    Compute the CEE of the system as
    q^+ = (λ^+ + Σ_{k=1}^K t_k^+ f_k^+ / C_cpu^k) / E_total({p_k^+}_{k=1}^K, {f_k^+}_{k=1}^K, {t_k^+}_{k=1}^K, t_e^+, t_o^+, t_c^+, f_m^+, P_t^+, λ^+);
    if |q^+ − q| ≤ ε then
        The obtained solution is the optimal solution to P2 and set Flag = 1;
    else
        Set q = q^+, Flag = 0 and l = l + 1;
    end if
until Flag = 1.
```

is summarized in Algorithm 1. As shown in Algorithm 1, in each iteration, the optimization problem $\mathbf{P}_3$ (defined below) is solved for a given q, returning the corresponding solution $\{\{p_k^+\}_{k=1}^K, \{f_k^+\}_{k=1}^K, \{t_k^+\}_{k=1}^K, t_e^+, t_o^+, t_c^+, f_m^+, P_t^+, \lambda^+\}$. Then based on the obtained solution, the CEE of the system is computed as

$$q^+ = \frac{\lambda^+ + \sum_{k=1}^K \frac{t_k^+ f_k^+}{C_{\text{cpu}}^k}}{E_{\text{total}}\left(\{p_k^+\}_{k=1}^K, \{f_k^+\}_{k=1}^K, \{t_k^+\}_{k=1}^K, t_e^+, t_o^+, t_c^+, f_m^+, P_t^+, \lambda^+\right)}.$$

Given an error tolerance ϵ, if $|q^+ - q| \le \epsilon$ is satisfied, then the obtained solution is the optimal solution to $\mathbf{P_2}$. Otherwise, we should update q as q^+ and repeat the above steps

$$\begin{aligned}
\mathbf{P}_3: \max_{\substack{\{p_k\},\{f_k\},\{t_k\},t_e,\\ t_o,t_c,f_m,P_t,\lambda}} \ & \lambda + \sum_{k=1}^K \frac{t_k f_k}{C_{\text{cpu}}^k} \\
& - q\Bigg(P_t + P_{sc} - \sum_{k=1}^K P_h^k + \sum_{k=1}^K \left(p_k + p_{c,k}\right)t_o \\
& \quad + P_{rc}t_o + \varepsilon_m f_m^3 t_c + \sum_{k=1}^K \varepsilon_k f_k^3 t_k\Bigg) \\
\text{s.t.}\ & \text{C1}-2, \text{C2}-1, \text{C3}, \text{C4}-1, \text{C5} \\
& \text{C6}, \ \text{C7}-2, \ \text{C8}-1, \ \text{C9}, \ \text{C10}
\end{aligned}$$

where q is a given parameter in each iteration and will be updated iteration by iteration.

To solve the nonconvex problem $\mathbf{P}_3$, which includes coupling relationships between multiple variables, i.e., t_k and f_k, t_o and p_k, etc., we introduce the following auxiliary variables: $x_k = t_k f_k$, $y_k = t_k f_k^3$, $x_m = t_c f_m$, $y_m = t_c f_m^3$, and $P_k = p_k t_o$. Accordingly, we have $t_k = \sqrt{(x_k^3/y_k)}$, $f_k = \sqrt{(y_k/x_k)}$, $t_c = \sqrt{(x_m^3/y_m)}$, $f_m = \sqrt{(y_m/x_m)}$, and $p_k = (P_k/t_o)$. Then $\mathbf{P}_3$ can be transformed as

$$\begin{aligned}
\mathbf{P}_4: \max_{\substack{\{P_k\},\{y_k\},\{x_k\},t_e,\\ t_o,x_m,y_m,P_t,\lambda}} \ & \lambda + \sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k} \\
& - q\Bigg(P_t + P_{sc} - \sum_{k=1}^K P_h^k + \sum_{k=1}^K \left(P_k + p_{c,k}t_o\right) \\
& \quad + P_{rc}t_o + \varepsilon_m y_m + \sum_{k=1}^K \varepsilon_k y_k\Bigg) \\
\text{s.t.}\ & \text{C1}-3: \lambda + \sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k} \ge L_{\min} t_e \\
& \text{C2}-2: P_k + p_{c,k}t_o + \varepsilon_k y_k \le P_h^k \quad \forall k \\
& \text{C3}-1: P_k h_k \ge \gamma_{\text{th}}^k \left(\sum_{i=k+1}^K P_i h_i + t_o\sigma^2\right) \forall k \\
& \text{C4}-2: 1 + t_o + \sqrt{\frac{x_m^3}{y_m}} \le T t_e \\
& \text{C5}-1: 0 \le y_m \le x_m f_{\max}^2 \\
& \qquad\quad\ 0 \le y_k \le x_k \left(f_k^{\max}\right)^2 \quad \forall k \\
& \text{C6}-1: 0 \le P_t \le P_{\max}, P_k \ge 0 \quad \forall k \\
& \text{C7}-3: t_e, t_o, \lambda, x_m \ge 0 \\
& \text{C8}-2: \sqrt{\frac{x_k^3}{y_k}} \le T t_e \\
& \text{C9}-1: x_m \ge \lambda C_{\text{cpu}}^m \\
& \text{C10}-1: t_o B \log_2\left(1 + \sum_{k=1}^K \frac{P_k h_k}{t_o\sigma^2}\right) \ge \lambda.
\end{aligned}$$

Besides, the consideration of the piecewise linear EH model also makes the optimization problem $\mathbf{P_4}$ more challenging to solve. Specifically, it is hard to determine the value of P_h^k since we do not know which segment P_{RF}^k belongs to. To address the problem brought by the used piecewise linear EH model, we propose the following three steps to obtain the optimal solution to $\mathbf{P_4}$.

Step 1: Compute the maximum number of segments that the energy harvester of each EU can operate on. Let s_k ($s_k \in \{0, 1, \ldots, N\}$) denote the maximum number of segments for the kth EU and s_k can be determined by the maximum number of s_k that satisfies $P_{\max} g_k \ge P_{\text{th}}^{s_k}$. If $\min(s_1, s_2, \ldots, s_K) = 0$, then at least one EU cannot harvest energy from the RF signals. In this case, $\mathbf{P_4}$ is infeasible. If $\min(s_1, s_2, \ldots, s_K) > 0$, go to step 2.

Step 2: Let $\{P_k^\dagger k = 1^K, \{y_k^\dagger\}_{k=1}^K, \{x_k^\dagger\}_{k=1}^K, t_e^\dagger, t_o^\dagger, x_m^\dagger, y_m^\dagger, \ \lambda^\dagger\}$ be the optimal solution to $\mathbf{P_4}$ and $q^\dagger$ be the corresponding system CEE. When the kth EU works on the j_kth segment where $1 \le j_k \le s_k$, the range of P_t is given by $P_L \le P_t \le P_U$ with $P_L = \max([P_{\text{th}}^{j_1}/g_1], [P_{\text{th}}^{j_2}/g_2], \ldots, [P_{\text{th}}^{j_K}/g_K])$ and $P_U = \min([P_{\text{th}}^{j_1+1}/g_1], \ldots, [P_{\text{th}}^{j_K+1}/g_K], P_{\max})$. If $P_L \le P_U$, go to step 3.

Step 3: Solve the optimization problem $\mathbf{P_4}$ for given $\{j_k\}_{k=1}^K$, given by

$$\begin{aligned}\mathbf{P}_5:\quad &\max_{\substack{\{P_k\},\{y_k\},\{x_k\},t_e,\\ t_o,x_m,y_m,P_t,\lambda}} \lambda+\sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k}\\ &-q\Bigg(P_t+P_{sc}-\sum_{k=1}^K\left(a_{j_k}P_t g_k+b_{j_k}\right)\\ &\qquad+P_{rc}t_o+\varepsilon_m y_m\\ &\qquad+\sum_{k=1}^K\left(P_k+p_{c,k}t_o\right)+\sum_{k=1}^K\varepsilon_k y_k\Bigg)\\ \text{s.t.}\quad &\text{C2}-3: P_k+p_{c,k}t_o+\varepsilon_k y_k\\ &\le a_{j_k}P_t g_k+b_{j_k}\quad \forall k\\ &\text{C1}-3,\text{C3}-1,\text{C4}-2,\text{C5}-1,\text{C6}-1\\ &\text{C7}-3,\text{C8}-2,\text{C9}-1,\text{C10}-1\\ &\text{C11}: P_{\text{th}}^{j_k}\le P_t g_k\le P_{\text{th}}^{j_k+1}\quad\forall k\end{aligned}$$

where constraint C11 is to ensure that the energy harvester of the kth EU works on the j_kth segment. Then the corresponding optimal solution can be obtained. On this basis, update $q^\dagger$ and $\{\{P_k^\dagger\}_{k=1}^K, \{y_k^\dagger\}_{k=1}^K, \{x_k^\dagger\}_{k=1}^K, t_e^\dagger, t_o^\dagger, x_m^\dagger, y_m^\dagger, \lambda^\dagger\}$ based on the obtained solution with the aim of obtaining a higher $q^\dagger$ until $j_k = s_k, \forall k$, is satisfied.

In order to tackle $\mathbf{P_5}$, Proposition 2 is provided.

Proposition 2: The optimization problem $\mathbf{P_5}$ is convex and can be solved by using existing convex methods (e.g., interior point method, Lagrange duality, etc.) efficiently.

Proof: See Appendix A. ■

The whole process for solving $\mathbf{P_4}$ is summarized in Algorithm 2. Combining Algorithms 1 and 2, we can obtain the optimal solution to the original optimization problem $\mathbf{P_0}$. Specifically, Algorithm 1 is used to solve $\mathbf{P_0}$, while in each iteration of Algorithm 1, Algorithm 2 is applied to obtain the optimal solution to the problem $\mathbf{P_3}$. Note that the proposed iterative algorithm in this work is the combination of Algorithms 1 and 2.

The computational complexity of the proposed iterative algorithm is analyzed as follows. If the interior point method is adopted to obtain the optimal solution to $\mathbf{P_5}$, then according to [29], the computational complexity of the proposed algorithm can be calculated as $N_u \prod_{k=1}^K s_k O(\sqrt{m_1}\log(m_1))$, where m_1 denotes the number of inequality constraints of $\mathbf{P_5}$, and N_u denotes the number of iterations required for Algorithm 1. We can see that the computational complexity is scaled up by $\prod_{k=1}^K s_k$, due to the use of the nonlinear EH model which makes the formulated optimization problem nonconvex. If without our proposed iterative algorithm, the nonconvex problem under the nonlinear EH model would need to be solved by using the exhaustive search method, which has a much higher complexity. Besides, the proposed iterative algorithm can be used to maximize the system CEE under the linear EH model after setting $N = 2$, $P_{\text{th}}^1 = 0$, $P_{\text{th}}^N = +\infty$, and $b_1 = 0$ in Algorithm 2 and its computational complexity will be reduced accordingly. This is because the conventionally considered linear EH model can be regarded as a special case of our considered nonlinear EH model in (1). How to further reduce the computational complexity in maximizing the CEE for a NOMA-based WPT-MEC network will be studied in our future work. Remark 1 is provided to summarize several purposes served by our developed iterative algorithm.

Algorithm 2 Three-Step-Based Iterative Algorithm for $\mathbf{P_4}$

Set $s_1 = s_2 = \cdots = s_K = N$;
for $k = 1$ **to** K **do**
  while $P_{\max} g_k < P_{\text{th}}^{s_k}$ **do**
    Set $s_k = s_k - 1$;
  end while
end for
if $\min(s_1, s_2, \cdots, s_K) == 0$ **then**
  $\mathbf{P_4}$ is infeasible;
else
  for $j_k = 1$ **to** s_k, $\forall k \in \{1, 2, \dots, K\}$ **do**
    Set $P_{\text{L}} = \max(\frac{P_{\text{th}}^{j_1}}{g_1}, \frac{P_{\text{th}}^{j_2}}{g_2}, \dots, \frac{P_{\text{th}}^{j_K}}{g_K})$ and $P_{\text{U}} = \min(\frac{P_{\text{th}}^{j_1+1}}{g_1}, \dots, \frac{P_{\text{th}}^{j_K+1}}{g_K}, P_{\max})$;
    if $P_{\text{L}} \le P_{\text{U}}$ **then**
      Solve $\mathbf{P_5}$ with given $\{j_k\}_{k=1}^K$ and obtain the optimal solution and the corresponding system CEE in $\mathbf{P_5}$, denoted by $\{\{P_k^\diamond\}_{k=1}^K, \{y_k^\diamond\}_{k=1}^K, \{x_k^\diamond\}_{k=1}^K, t_{\text{e}}^\diamond, t_{\text{o}}^\diamond, x_{\text{m}}^\diamond, y_{\text{m}}^\diamond, \lambda^\diamond\}$ and $q^\diamond$;
      if $j_k == 1$, $\forall k \in \{1, 2, \dots, K\}$ **then**
        Set $q^\dagger = q^\diamond$, $P_k^\dagger = P_k^\diamond$, $y_k^\dagger = y_k^\diamond$, $x_k^\dagger = x_k^\diamond$, $t_{\text{e}}^\dagger = t_{\text{e}}^\diamond$, $t_{\text{o}}^\dagger = t_{\text{o}}^\diamond$, $x_{\text{m}}^\dagger = x_{\text{m}}^\diamond$, $y_{\text{m}}^\dagger = y_{\text{m}}^\diamond$ and $\lambda^\dagger = \lambda^\diamond$;
      else
        Set $q^\dagger = \max(q^\dagger, q^\diamond)$ and update the optimal solution accordingly.
      end if
    end if
  end for
end if

Remark 1: Our developed iterative algorithm can serve the following four purposes. First, compared to the exhaustive search method, our proposed iterative algorithm provides a method with a lower complexity to obtain the optimal resource allocation that maximizes the system CEE of the considered network under the nonlinear EH model. Second, the proposed iterative algorithm can be used to maximize the system CEE of the considered network under the linear EH model by letting $N = 2$, $P_{\text{th}}^1 = 0$, $P_{\text{th}}^N = +\infty$, and $b_1 = 0$. Third, our proposed algorithm can be used to maximize the system CEE of the considered network for the nonlinear EH model under the complete offloading mode or fully local computing mode by letting $f_m = 0$ or $f_k = 0$ $\forall k$, respectively. Fourth, Algorithm 2 can be used to solve the computation bits maximization problem for the considered network under the nonlinear EH model by letting $q = 0$, $j_k = s_k$ $\forall k$ and solving $\mathbf{P_5}$ for given $\{j_k\}_{k=1}^K$.

C. Insights

In this section, by means of convex theory, we provide useful insights into the computationally energy-efficient design of the considered network, i.e., how many task bits should be offloaded by the EUs, how much time should be allocated for the offloading phase and for the MEC computing phase, etc. Toward this end, several findings are provided as follows.

Lemma 1: In order to obtain the maximum CEE of the considered network, the total task bits offloaded by all the EUs should equal the maximum bits computed by the MEC server during the task execution phase, i.e., $[(\tau_c^* f_m^*)/(C_{\text{cpu}}^m)] = \tau_o^* B \log_2(1 + \sum_{k=1}^K (p_k^* h_k/\sigma^2))$, where $*$ indicates the optimized variable corresponding to the optimal solution.

Proof: See Appendix B. ■

Remark 2: Lemma 1 reveals that all the offloaded computation task bits being computed at the MEC server during the task execution phase results in the maximum system CEE. This also indicates that for maximizing the system CEE, all the received tasks are completely computed by the MEC server in the task execution phase and the case that the MEC server cannot compute all the received tasks within the given time does not exist.

In order to obtain closed-form solutions, we use the Lagrange duality method to solve $\mathbf{P_5}$ and provide the following theorem.

Theorem 1: Given the nonnegative Lagrange multipliers, i.e., $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_6)$, $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$, $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_K)$, $\boldsymbol{\upsilon} = (\upsilon_1, \upsilon_2, \ldots, \upsilon_K)$, and $\boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_K)$, parts of the optimal solutions can be obtained as follows, i.e.,

$$f_m^* = \left[\frac{3\alpha_1}{2\left(\alpha_2 f_{\max}^2 + \alpha_5\right)}\right]^+ = \left[\sqrt[3]{\frac{\alpha_1}{2(q\varepsilon_m + \alpha_2)}}\right]^+ \tag{12}$$

$$\begin{aligned} f_k^* &= \left[\frac{3\upsilon_k C_{\text{cpu}}^k}{2\left(1 + \alpha_0 + \varphi_k \left(f_k^{\max}\right)^2 C_{\text{cpu}}^k\right)}\right]^+ \\ &= \left[\sqrt[3]{\frac{\upsilon_k}{2(q\varepsilon_k + \varphi_k + \theta_k\varepsilon_k)}}\right]^+ \end{aligned} \tag{13}$$

$$\sum_{k=1}^K p_k^* h_k = \left[G_k - \sigma^2\right]^+ \quad \forall k \tag{14}$$

$$\tau_c^* = \begin{cases} T - \tau_o^* - \tau_e^* = \frac{\tau_o^* B C_{\text{cpu}}^m \log_2\left(\frac{G_k}{\sigma^2}\right)}{f_m^*}, & \text{if } f_m^* > 0 \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

$$\tau_k^* = \begin{cases} T, & \text{if } f_k^* > 0 \\ 0, & \text{otherwise} \end{cases} \quad \forall k \tag{16}$$

where $[x]^+ = \max\{x, 0\}$ and $G_k = [(\alpha_6 B h_k)/((q + \theta_k - \mu_k h_k)\ln 2)]$.

Proof: See Appendix C. ■

Remark 3: From (15) we can see that when the EUs offload task bits to the MEC server, the MEC server will use as much time as possible to execute the received task bits in order to achieve the maximum system CEE. From (16) we can see that if there are task bits to be locally executed, each EU will use the entire time block for reducing its computing frequency and improving the system CEE. This also explains why most existing works, e.g., [9]–[12], [17], [20], and [22], assume that each EU can perform local computation in the entire time block. From (14) we can see that each EU chooses to offload task bits to the MEC server only when the channel gain between the MEC server and the EU is good. For example, for the kth EU, $h_k > [(\sigma^2(q + \theta_k)\ln 2)/(\alpha_6 B + \sigma^2 \mu_k \ln 2)$ must hold to ensure a nonzero throughput. From (12) and (13), we can also observe that the system CEE increases with the decrease of the optimal computing frequencies of both the EUs and the MEC server. This means that both the EUs and the MEC server should reduce their computing frequencies for maximizing the system CEE under the given constraints.

TABLE I
KEY SIMULATION SETTINGS

Parameters	Notation	Value
The entire time block	T	1 Second
The communication bandwidth	B	1 MHz
The PB's constant circuit power	P_{sc}	10 mW
The MEC server's constant circuit power	P_{rc}	10 mW
The k-th EU's constant circuit power	$p_{c,k}$	1 mW
The PB's maximum transmit power	$P_{\max}$	3 W
The number of EUs	K	4
The k-th EU's ECC	ε_k	10^{-26}
The MEC server's ECC	ε_{m}	10^{-28}
The k-th EU's maximum CPU frequency	$f_k^{\max}$	5×10^8 Hz
The MEC server's maximum CPU frequency	$f_{\max}$	10^{10} Hz
The minimum computation bits	$L_{\min}$	5×10^5 bits

Besides, based on $\tau_k^* = T$, we have $\sqrt{(x_k^3/y_k)} = Tt_e$, which can be used to reduce the computational complexity of solving $\mathbf{P_5}$. Specifically, $y_k = (x_k^3/T^2 t_e^2)$ should be satisfied for solving $\mathbf{P_5}$. By substituting $y_k = (x_k^3/T^2 t_e^2)$ into $\mathbf{P_5}$, we have

$$\begin{aligned} \mathbf{P}_6: \max_{\substack{\{P_k\},\{x_k\},t_e,\\ t_o,x_m,y_m,P_t,\lambda}} \quad & \lambda + \sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k} \\ & - q\Bigg(P_t + P_{sc} - \sum_{k=1}^K \left(a_{j_k} P_t g_k + b_{j_k}\right) \\ & + P_{rc} t_o + \varepsilon_m y_m \\ & + \sum_{k=1}^K \left(P_k + p_{c,k} t_o\right) + \sum_{k=1}^K \varepsilon_k \frac{x_k^3}{T^2 t_e^2}\Bigg) \\ \text{s.t.} \quad & C2-3: P_k + p_{c,k} t_o + \varepsilon_k \frac{x_k^3}{T^2 t_e^2} \\ & \leq a_{j_k} P_t g_k + b_{j_k} \quad \forall k \\ & \text{C1}-3, \text{C3}-1, \text{C4}-2, \text{C6}-1 \\ & \text{C7}-3, \text{C9}-1, \text{C10}-1, \text{C11} \\ & \text{C5}-2: 0 \leq y_m \leq x_m f_{\max}^2, x_k \leq t_e T f_k^{\max} \quad \forall k. \end{aligned}$$

Since $(x_k^3/T^2 t_e^2)$ is convex with respect to x_k and t_e, the transformed problem $\mathbf{P_6}$ is convex. By solving $\mathbf{P_6}$ instead of $\mathbf{P_5}$ in each iteration of Algorithm 2, the computational complexity for achieving the proposed resource allocation scheme can be reduced.

IV. NUMERICAL RESULTS

In this section, we verify the effectiveness and the superiority of the proposed iterative algorithm via computer simulations. Unless otherwise specified, the basic simulation parameters are given as shown in Table I [18], [22]. Similar to [22], we set $C_{\text{cpu}}^m = C_{\text{cpu}}^1 = C_{\text{cpu}}^2 = C_{\text{cpu}}^3 = C_{\text{cpu}}^4 = 1000$ Cycles/bit. We set $\gamma_{\text{th}}^1 = \gamma_{\text{th}}^2 = \gamma_{\text{th}}^3 = \gamma_{\text{th}}^4 = \gamma_{\text{th}} = 1$.

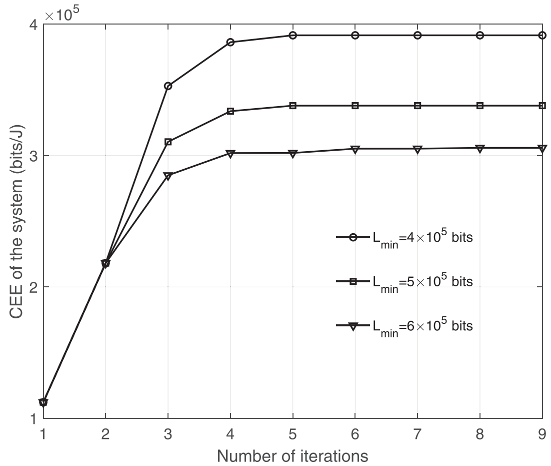


Fig. 2. Convergence of Algorithm 1.

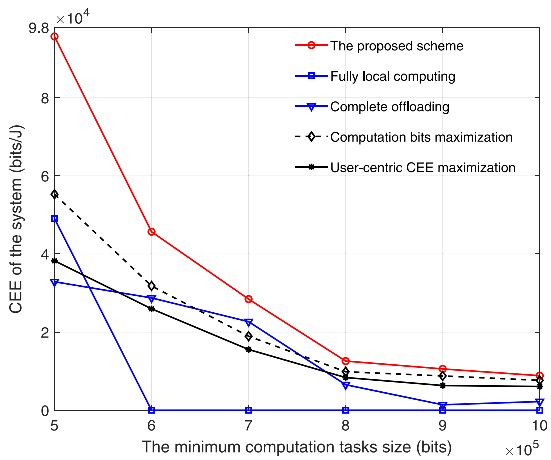


Fig. 3. System CEE under different schemes versus the minimum computation tasks size $L_{\min}$.

The channel gain between the PB and the kth EU is modeled by $g_k = g'_k d_k^{-\alpha}$ with the small-scale fading g'_k, distance d_k and path loss exponent α. Let $\alpha = 3$, $d_1 = 4.5$ m, $d_2 = 5$ m, $d_3 = 4.8$ m, and $d_4 = 4$ m. For convenience, let $(h_k/\sigma^2) = H_k h'_k$ with the small-scale fading h'_k. We set $H_1 = 110$, $H_2 = 90$, $H_2 = 70$, and $H_2 = 50$ in the following simulation. We adopt a piecewise linear EH model with $N = 3$ and the specific parameters are given as: $P_{\text{th}} = \{0, 5, 29.818, 59.51, +\infty\}$ mW, $\{a_{jk}\}_{jk=1}^{N} = \{0, 0.8260, 0.0657, 0\}$ and $\{b_{jk}\}_{jk=1}^{N} = \{0, -1.38, 21.2905, 25.2\}$ mW [24].

Fig. 2 demonstrates the convergence of Algorithm 1 under different settings of $L_{\min}$. It can be observed that less than eight iterations are required for Algorithm 1 to converge to the maximum CEE of the system, which illustrates that the proposed algorithm is computationally efficient.

For performance evaluation, we compare the proposed scheme with the following four representative benchmark schemes: 1) fully local computing scheme: all the EUs perform local computation only; 2) complete offloading scheme: all the EUs offload all their task bits to the MEC server for computation; 3) computation bits maximization scheme: this

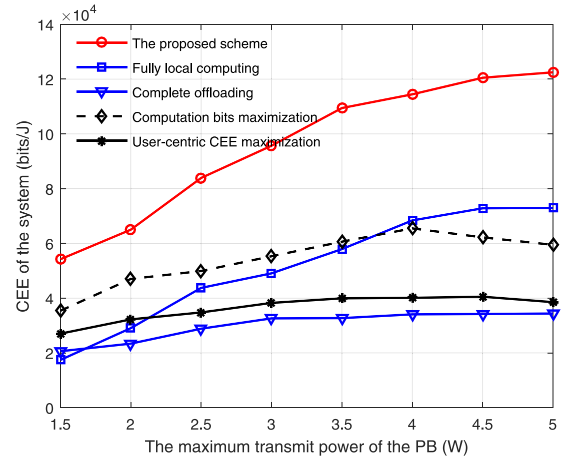


Fig. 4. System CEE under different schemes versus the maximum transmit power of the PB $P_{\max}$.

scheme maximizes the total achievable computation bits of the system under the same constraints as $\mathbf{P_0}$; and 4) user-centric CEE maximization scheme: this scheme maximizes the CEE of all the EUs under the same constraints as $\mathbf{P_0}$. Note that the fully local computing scheme, the complete offloading scheme and the computation bits maximization scheme are optimized under the same constraints as $\mathbf{P_0}$ and can be obtained based on the proposed algorithm following Remark 1. The user-centric CEE maximization scheme can be obtained based on Algorithm 1 by changing the objective function to the CEE of all the EUs and setting the optimal transmit power of the PB at $P_{\max}$.

Fig. 3 shows the system CEE under different schemes versus the minimum computation tasks size $L_{\min}$. As shown in this figure, we can see that the system CEE under all the schemes will decrease with the increasing of $L_{\min}$ since the energy consumed by computing grows faster than the growth of the computation bits. It can also be observed that the proposed scheme always outperforms the other schemes in terms of system CEE. The reasons are summarized as follows. On the one hand, the proposed scheme can utilize the available resources more efficiently for maximizing the system CEE while both the computation bits maximization scheme and the user-centric CEE maximization scheme do not aim to maximize the system CEE. On the other hand, both the fully local computing scheme and the complete offloading scheme can be regarded as special cases for the proposed scheme. By comparing the fully local computing scheme and the complete offloading scheme, we can also see that the system CEE under the fully local computing scheme is higher than that under the complete offloading scheme when $L_{\min}$ is small while for a larger $L_{\min}$, the complete offloading scheme outperforms the fully local computing scheme in terms of system CEE. This is because with a larger $L_{\min}$, EUs may not be able to compute their tasks locally due to the limitation of the harvested energy and the computation capacity, and offloading tasks to the MEC server can get the tasks computed with a less consumed energy. Moreover, we also find that the computation bits maximization scheme and the user-centric CEE maximization scheme are

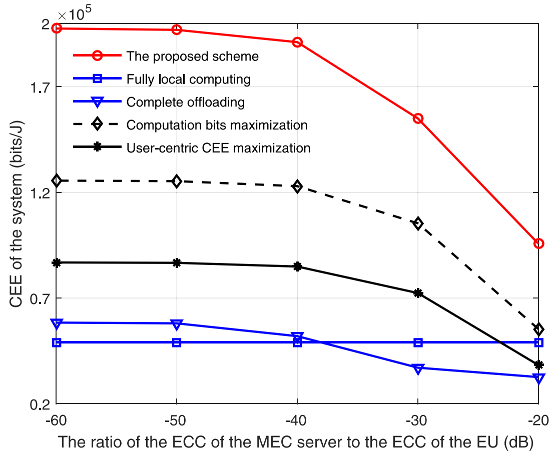


Fig. 5. System CEE under different schemes versus $(\varepsilon_m/\varepsilon_k)$.

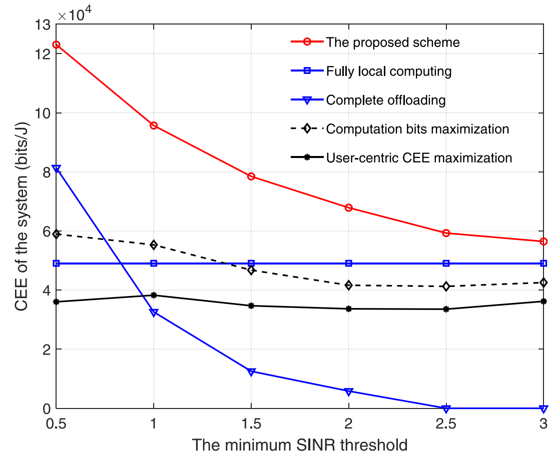


Fig. 6. System CEE under different schemes versus the SINR threshold.

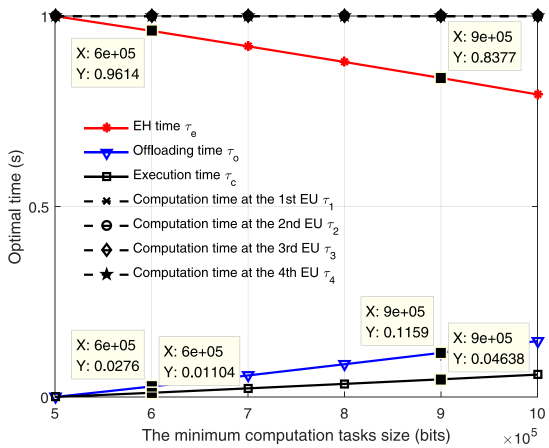
not energy efficient for the whole system, which illustrates the importance and rational of considering the system CEE maximization.

Fig. 4 shows the system CEE versus the maximum transmit power of the PB $P_{\max}$, where the above five schemes, i.e., the proposed scheme, the fully local computing scheme, the complete offloading scheme, the computation bits maximization scheme and the user-centric CEE maximization scheme, are considered. From this figure, we can observe that the system CEE under the proposed scheme, the fully local computing scheme and the complete offloading scheme will increase with the increasing of $P_{\max}$ and converge to the maximum value when $P_{\max}$ is large enough, while the system CEE under the computation bits maximization scheme and the user-centric CEE maximization scheme will increase first, reach the peak value and then decrease as $P_{\max}$ increases. The reasons are as follows. For the proposed scheme, the fully local computing scheme and the complete offloading scheme, when $P_{\max}$ is small, the optimal transmit power of the PB is constrained by $P_{\max}$ and a larger $P_{\max}$ will bring a larger system CEE, while when $P_{\max}$ is large enough, the optimal transmit power of the PB may not be influenced by $P_{\max}$, leading to an unchanged system CEE. For the computation bits maximization scheme and the user-centric CEE maximization scheme, the optimal transmit power of the PB is always $P_{\max}$, which can not bring a higher CEE for the system when $P_{\max}$ is large enough. By comparisons, we can also see that the proposed scheme can achieve the highest system CEE among these schemes.

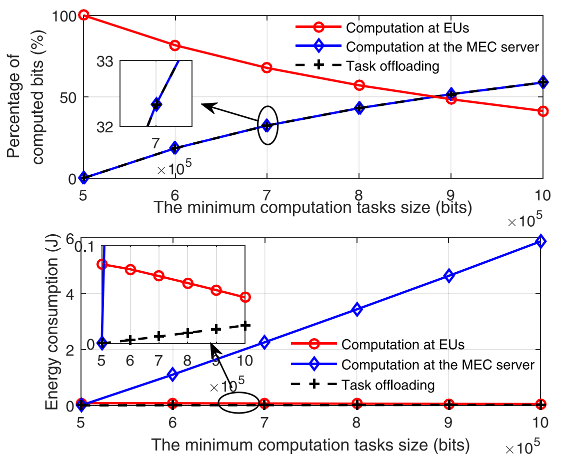
Fig. 5 shows the system CEE under the above five schemes versus $(\varepsilon_m/\varepsilon_k)$. We set $\varepsilon_k = 10^{-26}$ and the range of ε_m is set to be $[10^{-32}, 10^{-28}]$. It can be observed that with the increasing of $(\varepsilon_m/\varepsilon_k)$, the system CEE under the proposed scheme, the complete offloading scheme and the computation bits maximization scheme will decrease while the system CEE under the fully local computing scheme remains unchanged. This is because ε_m increases as $(\varepsilon_m/\varepsilon_k)$ increases and the energy consumption at the MEC server during the task execution phase also increases, leading to a decreasing CEE of the system for the proposed scheme, the complete offloading scheme and the computation bits maximization scheme, while the system CEE under the fully local computing scheme is not influenced by ε_m. By comparisons, we can still see that the proposed scheme outperforms the other schemes in terms of system CEE, which illustrates the superiority of the proposed scheme. Besides, by comparing the fully local computing scheme and the complete offloading scheme, we can also find that with a smaller ε_m, EUs tend to offload more tasks to the MEC server for computation.

Fig. 6 shows the system CEE under different schemes versus the SINR threshold γ_{th}. It can be observed that the system CEE under the proposed scheme and the complete offloading scheme decreases with the increasing of γ_{th}. When γ_{th} is large enough, the system CEE under the complete offloading scheme approaches 0, while the EUs under the proposed scheme tend to compute all the tasks locally, leading to a reduced system CEE. For all the considered SINR threshold values, the proposed scheme achieves the highest system CEE among all the schemes under comparison. The reasons are summarized as follows. On the one hand, the proposed scheme provides more flexibility for resource allocation to maximize the system CEE, while both the computation bits maximization scheme and the user-centric CEE maximization scheme do not aim to maximize the system CEE, leading to a reduced system CEE. On the other hand, the fully local computing scheme and the complete offloading scheme can not jointly utilize the computation resources at the EUs and the MEC server.

In Fig. 7, we evaluate the optimal time allocation under the proposed scheme and the tradeoff between the local computation and the MEC server's computation in terms of the computed bits and energy consumption. Fig. 7(a) plots the optimal EH time, offloading time, MEC server's execution time and EUs' computing time versus the minimum computation tasks size $L_{\min}$. It can be observed that with the increase of $L_{\min}$, our proposed scheme will allocate more time for task offloading at the EUs and for computation at the MEC server to get the required tasks computed, while the optimal EH time will decrease. Besides, we observe that the sum of the optimal EH time, offloading time and MEC server's execution time is always T and the optimal computing time of each EU is



(a)



(b)

Fig. 7. (a) Optimal time allocation versus the minimum computation tasks size $L_{\min}$. (b) Tradeoff between local computation and MEC server's computation.

always equal to T, which verifies Theorem 1. It is worth noting that for the case of $\tau_k = T$, the kth EU first uses some energy from its battery for performing local computation and then charges its battery using the harvested energy. The upper subplot of Fig. 7(b) plots the percentage of total computation bits computed by all the EUs locally and by the MEC server versus $L_{\min}$, where "computation at EUs;" denotes the percentage of total computation bits computed locally by all the EUs, "computation at the MEC server;" denotes the percentage of total computation bits computed by the MEC server, and "task offloading;" denotes the ratio of the total achievable throughput of all the EUs to the total computed task bits. It can be seen that when $L_{\min}$ grows, the proportion of local computation decreases while more task bits are offloaded to the MEC server for computation. The lower subplot of Fig. 7(b) plots the energy consumption versus $L_{\min}$, where computation at EUs; denotes the energy consumption of all the EUs for local computation, computation at the MEC server; stands for the energy consumed at the MEC server for computation, and task offloading; denotes the energy consumption of all the EUs for task offloading. We can see that with the increase of $L_{\min}$, the energy consumption at EUs for local computation decreases, while the consumed energy for task offloading and for MEC

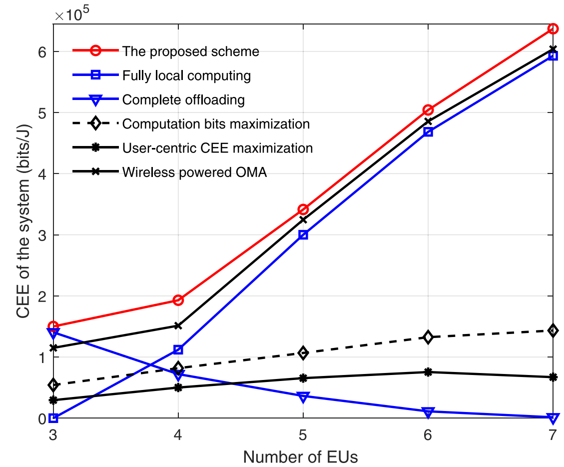


Fig. 8. System CEE under different schemes versus K.

server's computation increases. Meanwhile, we note that the offloaded task bits always equal the task bits computed by the MEC server, which verifies Lemma 1.

Fig. 8 plots the system CEE under different schemes versus the number of EUs, K. For wireless powered OMA, all the EUs take turns in time domain to offload their tasks to the MEC server during the task offloading phase while the system CEE is maximized under the same constraints as $\mathbf{P_0}$ by optimizing the transmit power and time of the PB, the offloading time, transmit power, computing frequencies and execution time of the K EUs, and the MEC server's execution time. It can be observed that the system CEE of the proposed scheme increases with the increasing K due to the fact that a larger K provides more flexibility for choosing users and allocating resources to achieve a higher system CEE. Besides, the proposed scheme achieves the highest system CEE among all the considered schemes for each given K, showing the superiority of the proposed scheme in terms of system CEE and the advantage of NOMA-based WPT-MEC over the OMA counterpart.

V. Conclusion

In this article, we have studied the CEE maximization for a NOMA-based WPT-MEC network from the system's perspective while considering a practical nonlinear EH model at the EUs and the computation resource allocation of the MEC server. By solving the system CEE maximization problem, we have proposed a Dinkelbach-based iterative algorithm to jointly optimize the transmit power and time of the PB, the EUs' computing frequencies, transmit power and offloading time, as well as the computing frequency and execution time of the MEC server. Furthermore, the closed-form expressions for parts of the optimal solutions have been derived, leading to several insights into the system CEE. Specifically, the system CEE increases as the optimal computing frequencies of both the EUs and the MEC server decrease and in order to maximize the system CEE, the total task bits offloaded by all the EUs should equal the maximum computation bits for the MEC server during the task execution phase, while the MEC server and the EUs use the maximum allowed time to complete their computing tasks. Simulations results have verified the superiority of the proposed scheme in terms of system CEE over several baseline schemes.

Based on this work, the following research directions could be explored. First, it will be interesting to include the decoding order in the joint optimization and design a scheme to maximize the system CEE by jointly optimizing the decoding order and system resources. Second, this work can be extended to the case where devices are equipped with multiple antennas. Third, the resource allocation for maximizing the system CEE will need to be carefully redesigned when considering the battery level of each EU.

APPENDIX A
PROOF OF PROPOSITION 2

As shown in $\mathbf{P_5}$, we can find that the objective function is a linear function and all the constraints except C4−2, C8−2, and C10−1 are linear constraints. As for C4−2 and C8−2, if the function $f(x, y) = \sqrt{(y^3/x)}$ is convex, then both C4−2 and C8−2 are convex constraints. By taking the second-order derivative of $f(x, y)$ with respect to x and y, the Hessian matrix is given by

$$\nabla^2 f(x, y) = \begin{bmatrix} \frac{3y^{\frac{3}{2}}}{4x^{\frac{5}{2}}} & -\frac{3\sqrt{y}}{4x^{\frac{3}{2}}} \\ -\frac{3\sqrt{y}}{4x^{\frac{3}{2}}} & \frac{3}{4\sqrt{x}\sqrt{y}} \end{bmatrix} \succeq \mathbf{0}. \tag{17}$$

Since the Hessian matrix is nonnegative definite, which indicates that $f(x, y)$ is convex, C4−2 and C8−2 are convex constraints. Besides, C10−1 can also be easily proved as a convex constraint. Thus, $\mathbf{P_5}$ is a convex optimization problem, which can be solved by using existing convex methods (i.e., interior point method, Lagrange duality, etc.) efficiently.

APPENDIX B
PROOF OF LEMMA 1

Here, we prove Lemma 1 by means of contradiction. Specifically, assume that $\{\{p_k^*\}_{k=1}^K, \{f_k^*\}_{k=1}^K, \{\tau_k^*\}_{k=1}^K, \tau_e^*, \tau_o^*, \tau_c^*, f_m^*, P_t^*\}$ is the optimal solution to $\mathbf{P_0}$, where $(\tau_c^* f_m^*/C_{\text{cpu}}^m) \neq \tau_o^* B\log_2(1 + \sum_{k=1}^K (p_k^* h_k/\sigma^2))$ always holds. That is, either $(\tau_c^* f_m^*/C_{\text{cpu}}^m) > \tau_o^* B\log_2(1 + \sum_{k=1}^K (p_k^* h_k/\sigma^2))$ or $(\tau_c^* f_m^*/C_{\text{cpu}}^m) < \tau_o^* B\log_2(1 + \sum_{k=1}^K (p_k^* h_k/\sigma^2))$ should be satisfied. Suppose that $[(\tau_c^* f_m^*)/(C_{\text{cpu}}^m)] > \tau_o^* B\log_2(1 + \sum_{k=1}^K (p_k^* h_k/\sigma^2))$ holds. Let q^* be the optimal system CEE. We can construct another solution satisfying $P_t^+ = P_t^*, p_k^+ = p_k^*, f_k^+ = f_k^*, \tau_k^+ = \tau_k^*, \tau_e^+ = \tau_e^*, \tau_o^+ = \tau_o^*, \tau_c^+ = \tau_c^*$, and $[(\tau_c^+ f_m^+)/(C_{\text{cpu}}^m)] = \tau_o^+ B\log_2(1 + \sum_{k=1}^K [(p_k^+ h_k)/\sigma^2])$. Obviously, the constructed solution $\{\{p_k^+\}_{k=1}^K, \{f_k^+\}_{k=1}^K, \{\tau_k^+\}_{k=1}^K, \tau_e^+, \tau_o^+, \tau_c^+, f_m^+, P_t^+\}$ is a feasible solution which satisfies all the constraints of $\mathbf{P_0}$. Let q^+ be the corresponding system CEE under the constructed solution. Since $[(\tau_c^+ f_m^+)/(C_{\text{cpu}}^m)] = \tau_o^+ B\log_2(1 + \sum_{k=1}^K [(p_k^+ h_k)/\sigma^2]) = \tau_o^* B\log_2(1 + \sum_{k=1}^K [(p_k^* h_k)/\sigma^2]) < [(\tau_c^* f_m^*)/(C_{\text{cpu}}^m)]$, it follows that $f_m^+ < f_m^*$. Based on (10), we can find that the constructed solution can achieve the same computation bits as the optimal one while consuming less energy. Thus, one has $q^+ > q^*$, which contradicts the assumption that $\{\{p_k^*\}_{k=1}^K, \{f_k^*\}_{k=1}^K, \{\tau_k^*\}_{k=1}^K, \tau_e^*, \tau_o^*, \tau_c^*, f_m^*, P_t^*\}$ is the optimal solution to $\mathbf{P_0}$. The same way can also be applied to the case with $[(\tau_c^* f_m^*)/(C_{\text{cpu}}^m)] < \tau_o^* B\log_2(1 + \sum_{k=1}^K [(p_k^* h_k)/\sigma^2])$ and the detailed process is omitted here for brevity. Based on the above analysis, Lemma 1 is proven.

APPENDIX C
PROOF OF THEOREM 1

Let $\boldsymbol{\alpha} = (\alpha_0, \alpha_1, \ldots, \alpha_6)$, $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_K)$, $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_K)$, $\boldsymbol{\upsilon} = (\upsilon_1, \upsilon_2, \ldots, \upsilon_K)$, and $\boldsymbol{\varphi} = (\varphi_1, \varphi_2, \ldots, \varphi_K)$ denote the nonnegative Lagrange multipliers with respect to all the constraints. Then the Lagrangian function of $\mathbf{P_5}$ is given by (18), shown at the bottom of the page, where $P_{\text{L}} = \max((P_{\text{th}}^{j_1}/g_1), (P_{\text{th}}^{j_2}/g_2), \ldots, (P_{\text{th}}^{j_K}/g_K))$ and $P_{\text{U}} = \min((P_{\text{th}}^{j_1+1}/g_1), \ldots, (P_{\text{th}}^{j_K+1}/g_K), P_{\max})$.

By taking the partial derivative of $\mathcal{L}$ with respect to each optimization variable, we have

$$\frac{\partial \mathcal{L}}{\partial y_m} = \frac{\alpha_1 \sqrt{x_m^3}}{2\sqrt{y_m^3}} - q\varepsilon_m - \alpha_2 \tag{19}$$

$$\frac{\partial \mathcal{L}}{\partial x_m} = \alpha_2 f_{\max}^2 - \frac{3\alpha_1 \sqrt{x_m}}{2\sqrt{y_m}} + \alpha_5 \tag{20}$$

$$\frac{\partial \mathcal{L}}{\partial y_k} = \frac{\upsilon_k \sqrt{x_k^3}}{2\sqrt{y_k^3}} - q\varepsilon_k - \varphi_k - \theta_k \varepsilon_k \tag{21}$$

$$\begin{aligned} \mathcal{L} = {}& \lambda + \sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k} - q\left(P_t + P_{sc} - \sum_{k=1}^K \left(a_{j_k} P_t g_k + b_{j_k}\right) + P_{rc} t_o + \varepsilon_m y_m + \sum_{k=1}^K \left(P_k + p_{c,k} t_o\right) + \sum_{k=1}^K \varepsilon_k y_k\right) + \alpha_3 (P_t - P_{\text{L}}) \\ & + \alpha_0 \left(\lambda + \sum_{k=1}^K \frac{x_k}{C_{\text{cpu}}^k} - L_{\min} t_e\right) + \sum_{k=1}^K \theta_k \left(a_{j_k} P_t h_k + b_{j_k} - P_k - p_{c,k} t_o - \varepsilon_k y_k\right) + \alpha_6 \left(t_o B\log_2\left(1 + \sum_{k=1}^K \frac{P_k h_k}{t_o \sigma^2}\right) - \lambda\right) \\ & + \alpha_1 \left(T t_e - 1 - t_o - \sqrt{\frac{x_m^3}{y_m}}\right) + \alpha_2 \left(x_m f_{\max}^2 - y_m\right) + \sum_{k=1}^K \varphi_k \left(x_k \left(f_k^{\max}\right)^2 - y_k\right) + \alpha_4 (P_U - P_t) + \sum_{k=1}^K \upsilon_k \left(T t_e - \sqrt{\frac{x_k^3}{y_k}}\right) \\ & + \alpha_5 \left(x_m - \lambda C_{\text{cpu}}^m\right) + \sum_{k=1}^K \mu_k \left(P_k h_k - \gamma_{\text{th}}^k \left(\sum_{i=k+1}^K P_i h_i + t_o \sigma^2\right)\right) \end{aligned} \tag{18}$$

$$\frac{\partial \mathcal{L}}{\partial x_k} = \frac{1+\alpha_0}{C_{\text{cpu}}^k} + \varphi_k\left(f_k^{\max}\right)^2 - \frac{3\upsilon_k\sqrt{x_k}}{2\sqrt{y_k}} \tag{22}$$

$$\frac{\partial \mathcal{L}}{\partial P_k} = \frac{\alpha_6 B h_k}{\left(\sigma^2 + \sum_{k=1}^{K}\frac{P_k h_k}{t_o}\right)\ln 2} + \mu_k h_k - q - \theta_k. \tag{23}$$

By letting $(\partial\mathcal{L}/\partial y_m) = (\partial\mathcal{L}/\partial x_m) = 0$, we can compute the optimal CPU frequency of the MEC server as

$$f_m^* = \left[\frac{3\alpha_1}{2\left(\alpha_2 f_{\max}^2 + \alpha_5\right)}\right]^+ = \left[\sqrt[3]{\frac{\alpha_1}{2(q\varepsilon_m + \alpha_2)}}\right]^+ \tag{24}$$

where $[x]^+ = \max\{x, 0\}$. Similarly, for $\forall k$, f_k^* can be expressed as

$$\begin{aligned} f_k^* &= \left[\frac{3\upsilon_k C_{\text{cpu}}^k}{2\left(1+\alpha_0+\varphi_k\left(f_k^{\max}\right)^2 C_{\text{cpu}}^k\right)}\right]^+ \\ &= \left[\sqrt[3]{\frac{\upsilon_k}{2(q\varepsilon_k + \varphi_k + \theta_k\varepsilon_k)}}\right]^+. \end{aligned} \tag{25}$$

Then by letting $(\partial\mathcal{L}/\partial P_k) = 0$ and $p_k = (P_k/t_o)$, the optimal transmit power at each EU should satisfy the following equation, i.e.,

$$\sum_{k=1}^{K} p_k^* h_k = \left[G_k - \sigma^2\right]^+ \tag{26}$$

where $G_k = [(\alpha_6 B h_k)/((q+\theta_k-\mu_k h_k)\ln 2)]$.

Based on (24), we can see that if $f_m^* > 0$ is satisfied, then $\alpha_1 > 0$ must hold. By means of the Karush–Kuhn–Tucker (KKT) conditions, we can also find that the optimal time sharing among EH phase, task offloading phase and task execution phase should satisfy the following equation: $\alpha_1(Tt_e^* - 1 - t_o^* - \sqrt{((x_m^*)^3/y_m^*)}) = 0$. Combining $\alpha_1 > 0$, we can obtain $Tt_e^* - 1 - t_o^* - \sqrt{((x_m^*)^3/y_m^*)} = Tt_e^* - 1 - t_o^* - t_c^* = 0$ in the case of $f_m^* > 0$. Through some convenient mathematical calculations, we can obtain $\tau_e^* + \tau_o^* + \tau_c^* = T$ in the case of $f_m^* > 0$. Note that in the case of $f_m^* = 0$, the MEC server can not provide computation service for the EUs. Thus, the value of τ_c^* does not influence the CEE of the considered network and we let $\tau_c^* = 0$ for convenience. Besides, we can also compute τ_c^* as $\tau_c^* = [(\tau_o^* B C_{\text{cpu}}^m \log_2(1+\sum_{k=1}^{K}[(p_k^* h_k)/\sigma^2]))/f_m^*]$ according to Lemma 1. Based on (26), τ_c^* can be rewritten as $\tau_c^* = [(\tau_o^* B C_{\text{cpu}}^m \log_2(G_k/\sigma^2))/f_m^*]$.

Likewise, $f_k^* > 0$ leads to $\upsilon_k > 0$ based on (25). Combining the complementary slackness condition $\upsilon_k(Tt_e^* - \sqrt{((x_k^*)^3/y_k^*)}) = 0$, we can obtain $Tt_e^* - \sqrt{((x_k^*)^3/y_k^*)} = Tt_e^* - t_k^* = 0$. Then we have $\tau_k^* = T$ under the case of $f_k^* > 0$. Since each EU does not perform local computation when $f_k^* = 0$, we let $\tau_k^* = 0$ for the case of $f_k^* = 0$.

References

[1] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surveys. Tuts.*, vol. 17, no. 4, pp. 2347–2376, 4th Quart., 2015.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[3] P. Ramezani and A. Jamalipour, "Toward the evolution of wireless powered communication networks for the future Internet of Things," *IEEE Netw.*, vol. 31, no. 6, pp. 62–69, Nov./Dec. 2017.

[4] K. W. Choi *et al.*, "Toward realization of long-range wireless-powered sensor networks," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 184–192, Aug. 2019.

[5] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, Oct. 2016.

[6] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[7] Z. Zhou *et al.*, "Energy-efficient resource allocation for energy harvesting-based cognitive machine-to-machine communications," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 595–607, Sep. 2019.

[8] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.

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

[10] L. Huang, S. Bi, and Y.-J. A. Zhang, "Deep reinforcement learning for online computation offloading in wireless powered mobile-edge computing networks," *IEEE Trans. Mobile Comput.*, vol. 19, no. 11, pp. 2581–2593, Nov. 2020.

[11] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation rate maximization in UAV-enabled wireless-powered mobile-edge computing systems," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 9, pp. 1927–1941, Sep. 2018.

[12] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[13] X. Hu, K.-K. Wong, and K. Yang, "Wireless powered cooperation-assisted mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2375–2388, Apr. 2018.

[14] S. Mao, S. Leng, K. Yang, X. Huang, and Q. Zhao, "Fair energy-efficient scheduling in wireless powered full-duplex mobile-edge computing systems," in *Proc. IEEE GLOBECOM*, 2017, pp. 1–6.

[15] H. Lim and T. Hwang, "Energy-efficient computing for wireless powered mobile edge computing systems," in *Proc. IEEE VTC-Fall*, 2019, pp. 1–5.

[16] L. Ji and S. Guo, "Energy-efficient cooperative resource allocation in wireless powered mobile edge computing," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4744–4754, Jun. 2019.

[17] F. Zhou, H. Sun, Z. Chu, and R. Q. Hu, "Computation efficiency maximization for wireless-powered mobile edge computing," in *Proc. IEEE GLOBECOM*, 2018, pp. 1–6.

[18] Y. Ye, R. Q. Hu, G. Lu, and L. Shi, "Enhance latency-constrained computation in MEC networks using uplink NOMA," *IEEE Trans. Commun.*, vol. 68, no. 4, pp. 2409–2425, Apr. 2020.

[19] M. Sheng, Y. Dai, J. Liu, N. Cheng, X. Shen, and Q. Yang, "Delay-aware computation offloading in NOMA MEC under differentiated uploading delay," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2813–2826, Apr. 2020.

[20] M. Zeng, R. Du, V. Fodor, and C. Fischione, "Computation rate maximization for wireless powered mobile edge computing with NOMA," in *Proc. IEEE WoWMoM*, 2019, pp. 1–9.

[21] F. Zhou, Y. Wu, R. Q. Hu, and Y. Qian, "Computation efficiency in a wireless-powered mobile edge computing network with NOMA," in *Proc. IEEE ICC*, 2019, pp. 1–7.

[22] F. Zhou and R. Q. Hu, "Computation efficiency maximization in wireless-powered mobile edge computing networks," *IEEE Trans. Wireless Commun.*, vol. 19, no. 5, pp. 3170–3184, May 2020.

[23] M. Ismail, W. Zhuang, E. Serpedin, and K. Qaraqe, "A survey on green mobile networking: From the perspectives of network operators and mobile users," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 3, pp. 1535–1556, 3rd Quart., 2015.

[24] G. Lu, L. Shi, and Y. Ye, "Maximum throughput of TS/PS scheme in an AF relaying network with non-linear energy harvester," *IEEE Access*, vol. 6, pp. 26617–26625, 2018.

[25] E. Boshkovska, D. W. K. Ng, N. Zlatanov, and R. Schober, “Practical non-linear energy harvesting model and resource allocation for SWIPT systems,” *IEEE Commun. Lett.*, vol. 19, no. 12, pp. 2082–2085, Dec. 2015.

[26] Y. Chen, K. T. Sabnis, and R. A. Abd-Alhameed, “New formula for conversion efficiency of RF EH and its wireless applications,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 11, pp. 9410–9414, Nov. 2016.

[27] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, “Mobile-edge computing: Partial computation offloading using dynamic voltage scaling,” *IEEE Trans. Commun.*, vol. 64, no. 10, pp. 4268–4282, Oct. 2016.

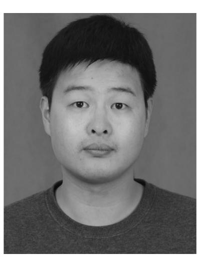
[28] W. Dinkelbach, “On nonlinear fractional programming,” *Manag. Sci.*, vol. 13, no. 7, pp. 492–498, 1967.

[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

Liqin Shi received the B.S. degree from Sichuan University, Chengdu, China, in 2015, and the Ph.D. degree from Xidian University, Xi’an, China, in 2020.

She was a Joint Ph.D. student with the Department of Electrical and Computer Engineering, Utah State University, Logan, UT, USA, from 2018 to 2019. Since 2020, she has been with the Xi’an University of Posts and Telecommunications, Xi’an, where she is currently an Associate Professor with the Department of Communication and Information Engineering. She has published more than 20 papers in IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY and IEEE ICC. Her research interests include wireless energy harvesting and mobile-edge computing.

Dr. Shi is also a reviewer of multiple international journals, including the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS and the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. She has served as a TPC member of IEEE SmartIoT in 2018 and IEEE VTC-FALL in 2019.



Yinghui Ye (Member, IEEE) received the Ph.D. degree in communication and information system from Xidian University, Xi’an, China, in 2020.

He was a joint Ph.D. student with the Department of Electrical and Computer Engineering, Utah State University, Logan, UT, USA, from 2018 to 2019. Since 2020, he has been with the Department of Communication and Information Engineering, Xi’an University of Posts and Telecommunications, Xi’an, where he is currently an Associate Professor. He has authored/coauthored more than 30 technical articles in the IEEE Transactions, journals, letters, and conferences. His research interests include cognitive radio networks, relaying networks, and wireless energy harvesting.

Dr. Ye received the Exemplary Reviewer Award from the IEEE WIRELESS COMMUNICATIONS LETTERS in 2019. He is also a reviewer of multiple international Journals, such as the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, IEEE TRANSACTIONS ON COMMUNICATIONS, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He served as a TPC member of IEEE VTC-FALL 2019 and IEEE ICCT 2019/2020.

Xiaoli Chu (Senior Member, IEEE) received the B.Eng. degree in electronic and information engineering from Xi’an Jiao Tong University, Xi’an, China, in 2001, and the Ph.D. degree in electrical and electronic engineering from Hong Kong University of Science and Technology, Hong Kong, in 2005.

She is a Professor with the Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, U.K. From 2005 to 2012, she was with the Centre for Telecommunications Research, King’s College London, London, U.K. She coauthored/co-edited the books *Fog-Enabled Intelligent IoT Systems* (Springer, 2020), *Ultra Dense Networks for 5G and Beyond* (Wiley, 2019), *Heterogeneous Cellular Networks: Theory, Simulation and Deployment* (Cambridge University Press, 2013), and *4G Femtocells: Resource Allocation and Interference Management* (Springer, 2013). She has coauthored over 150 peer-reviewed journal and conference papers.

Prof. Chu was a co-recipient of the IEEE Communications Society 2017 Young Author Best Paper Award. She received the IEEE COMMUNICATIONS LETTERS Exemplary Editor Award in 2018. She is a Senior Editor of the IEEE WIRELESS COMMUNICATIONS LETTERS, and an Editor of the IEEE COMMUNICATIONS LETTERS. She was the Co-Chair of Wireless Communications Symposium for IEEE ICC in 2015 and Workshop Co-Chair for IEEE GreenCom in 2013. She has co-organized eight workshops at IEEE ICC, GLOBECOM, WCNC, and PIMRC.

Guangyue Lu received the Ph.D. degree from Xidian University, Xi’an, China, in 1999.

From September 2004 to August 2006, he was a Guest Researcher with the Signal and Systems Group, Uppsala University, Uppsala, Sweden. Since 2005, he has been a Full Professor with the School of Communications and Information Engineering, Xi’an University of Posts and Telecommunications, Xi’an. He has been the funded by over ten projects, including the National Natural Science Foundation of China, the 863 Program, and Important National Science and Technology Specific Projects. Due to his excellent contributions in education and research. He is currently the Vice President with the Xi’an University of Posts and Telecommunications, and the Director of the Shaanxi Key Laboratory of Information Communication Network and Security. His research interests include wireless energy harvesting, cognitive radio, and cooperative spectrum sensing.

Prof. Lu was awarded by the Program for New Century Excellent Talents in University, Ministry of Education, China, in 2009.