Computation Offloading for Energy Efficiency Maximization of Sustainable Energy Supply Network in IIoT

Zhao Tong, *Member, IEEE*, Jinhui Cai, Jing Mei, Kenli Li, *Senior Member, IEEE*, and Keqin Li, *Fellow, IEEE*

***Abstract*—The efficiency of production and equipment maintenance costs in the Industrial Internet of Things (IIoT) are directly impacted by equipment lifetime, making it an important concern. Mobile edge computing (MEC) can enhance network performance, extend device lifetime, and effectively reduce carbon emissions by integrating energy harvesting (EH) technology. However, when the two are combined, the coupling effect of energy and the system's communication resource management pose a great challenge to the development of computational offloading strategies. This paper investigates the problem of maximizing the energy efficiency of computation offloading in a two-tier MEC network powered by wireless power transfer (WPT). First, the corresponding mathematical models are developed for local computing, edge server processing, communication, and EH. The proposed fractional problem is transformed into a stochastic optimization problem by Dinkelbach method. In addition, virtual power queues are introduced to eliminate energy coupling effects by maintaining the stability of the battery power queues. Next, the problem is then resolved through the utilization of both Lyapunov optimization and convex optimization method. Consequently, a wireless energy transmission-based algorithm for maximizing energy efficiency is proposed. Finally, energy efficiency, an important parameter of network performance, is used as an indicator. The excellent performance of the EEMA-WET algorithm is verified through extensive extension and comparison experiments.**

***Index Terms*—Computation offloading, energy efficiency, energy harvesting (EH), Lyapunov optimization, wireless power transfer (WPT).**

Manuscript received 28 April 2023; revised 24 August 2023; accepted 6 September 2023. Date of publication 11 September 2023; date of current version 3 April 2024. This work was supported in part by the Program of National Natural Science Foundation of China under Grants 62372172 and 62072174, in part by Distinguished Youth Science Foundation of Hunan Province, China under Grant 2023JJ10030, in part by National Natural Science Foundation of Hunan Province, China under Grant 2022JJ40278, and in part by Scientific Research Fund of Hunan Provincial Education Department, China under Grant 22A0026. Recommended for acceptance by D. Zeng. *(Corresponding author: Zhao Tong.)*

Zhao Tong, Jinhui Cai, and Jing Mei are with the College of Information Science and Engineering, Hunan Normal University, Changsha, Hunan 410081, China (e-mail: tongzhao@hunnu.edu.cn; 202020291627@hunnu.edu.cn; jingmei1988@163.com).

Kenli Li is with the College of Information Science and Engineering, Hunan University and National Supercomputing Center, Changsha, Hunan 410082, China (e-mail: lkl@hnu.edu.cn).

Keqin Li is with the College of Information Science and Engineering, Hunan University, and National Supercomputing Center, Changsha, Hunan 410082, China, and also with the Department of Computer Science, State University of New York, New Paltz, NY 12561 USA (e-mail: lik@newpaltz.edu).

Digital Object Identifier 10.1109/TSUSC.2023.3313770

I. Introduction

The rapid expansion of the Industrial Internet of Things (IIoT) has increased the interest in mobile edge computing (MEC) technology [1], [2]. MEC is an emerging computing model that moves computing resources from central data centers to edge devices to reduce data transfer latency and network congestion. This computing model can bring data processing and storage closer to the data source, thereby improving data processing speed and efficiency [3], [4], [5]. In the IIoT, MEC can facilitate real-time monitoring and control of production processes by bringing computing power and intelligent algorithms closer to the factory floor. This improves production efficiency and quality, and reduces the bandwidth and cost required for data transmission. MEC has a wide application prospect and market demand in IIoT, and it is an important means to achieve intelligence.

The rapid increase in the number of network devices also escalates the scale of carbon emissions in wireless communication networks. Energy harvesting (EH) is an emerging technology for "green communication" [6]. By using it to supply energy to the system, the carbon footprint of wireless communication network systems can be radically reduced. It is an environmentally friendly and green sustainable development. The principle of EH involves gathering green, renewable energy from the environment, which is then converted into electrical energy and either stored in the device battery or used directly. By definition, the energy sources for EH are very broad. Any energy that can be collected and utilized can be considered as a potential energy source for EH technologies [7]. Examples, wind, thermal, solar and biomass energy as well as environmental vibrations, human motion and RF signals. Energy efficiency is a significant metric for evaluating network performance. Enhancing the energy efficiency of the system through EH technology can prolong the equipment's lifespan.

A. Motivation

There is in urgent need to solve the problem of energy shortages equipment and carbon pollution to the environment. It is an emerging research area for "green energy saving" communication. By combining the collected green energy with wireless communication technology, the energy supply to the equipment can be realized anytime and anywhere. When the EH system is faced with different kinds of energy structures supplying energy, it has various ways to integrate with the communication system [8]. There are two main types from the point of view. One is to collect solar and wind energy, etc, for power supply through external devices, which needs to be applied in environments with high energy levels. The other is to convert the energy carried by the RF signal into the energy source of the system; based on this method carrying out a direct or indirect energy supply system is called a wireless energy supply communication network.

In wireless energy supply communication networks, the transmission methods induced by EH technology include two types. One is wireless power transfer (WPT), which is mainly applied to collect energy at the device side of the downlink channel in small-scale proximity communication scenarios. The other is wireless information transfer, which is mainly used for sending data from the device side to the base station (BS). The application of WPT technology addresses the energy supply issue for numerous end devices. The collected green energy can be transmitted to the device side through a wireless link using RF signals. Compared to traditional energy supply methods, it greatly reduces labor costs and carbon emissions, and increases the diversity of equipment deployment. In addition, WPT technology has outstanding advantages over the method of obtaining energy from natural resources, with high effectiveness and stability. Therefore, the application of WPT-based network communication systems is promising and can meet the ever-increasing demand for quality of service (QoS) [9]. Three typical energy harvesting architectures have been proposed in the energy harvesting system. The first is harvest-use, which directly uses the harvested energy without storage. The second is harvest-use-storage, where the harvested energy is saved and only available at the next time instance. The last is harvest-storage-use, where the harvested energy can be immediately used and the remaining is stored in the energy battery for future use. In this paper, we presume that the energy used by Internet of Things devices for offloading tasks only comes from the battery and the harvested energy in the current time slot can only be used in the next time slot, i.e., harvest-use-storage strategy.

B. Contributions

In this paper, a two-layer MEC model computation offloading strategy for WPT supply in IIoT is studied. To improve offloading energy efficiency and realize "green energy-saving" communication, WPT technology is considered as a means of system energy supply. Furthermore, the coupling of energy arrival and task transmission and the randomness of external environment are analyzed. A set of random optimization model for task offloading is constructed. By implementing the concepts of Lyapunov optimization and virtual power queues, the model can be transformed into a convex optimization problem for resolution. A iteration-based algorithm for maximizing the systems energy efficiency of WPT is proposed.

- The problem of optimizing the energy efficiency of a multi-user, two-tier MEC system with WPT supply is under consideration. We ensure the stability of the system operation by introducing task and battery power queues. Moreover, the incorporation of virtual queues and sleep-wake strategies mitigates the impact of energy coupling, enhances network energy efficiency, and reduces computational complexity.
- The problem of maximizing network energy efficiency is formulated as a stochastic optimization problem that includes energy causal constraints. The problem is decoupled into three independent subproblems. To solve the wireless channel resource allocation subproblem of offloading and EH, an offloading and energy harvesting wireless channel resource allocation algorithm (WCRAA) is proposed. It establishes the offloading priority of the device, on the basis of which the proportion of time allocated to offloading calculations and EH is rationalized. Subsequently, we designed an algorithm for maximizing the energy efficiency of wireless energy transfer (EEMA-WET) based on WCRAA. It can effectively increase the network energy efficiency with low complexity of the algorithm.
- Numerous simulation experiments show that by choosing a suitable control parameter V, the algorithm EEMA-WET based on Lyapunov optimization can infinitely converge to the optimal value. Furthermore, the experiments indicate the presence of upper bound constraints for both queues, which guarantee system stability. The efficacy of the proposed algorithm in enhancing system energy efficiency and stabilizing battery power is validated.

The paper is organized as follows. Section II provides a summary and analysis of the previous research. Section III outlines the model composition for each system component, while also providing a detailed definition and formalization of the optimization objective. In Section IV, the virtual queue is introduced to eliminate the energy coupling effect. An algorithm has been developed to optimize network energy efficiency, while also considering the trade-off with system stability. Section V conducts simulation experiments and compares them with some benchmark strategies. The superiority of the proposed EEMA-WET algoritresolution [10]. How to achieve "green energy-saving" sustainable development of mobile communication networks has received the attention and research of experts and scholars [11]. In the MEC computational offloading problem, Chen et al. [12] discovered that uncertainty characterizes both the wireless channel state and task arrival process. To solve this challenge, a dynamic energy-savinghm is verified. In conclusion, Section VI serves as the concluding section of this paper.

II. Related Work

With the arrival of 5G technology, a high quality service experience has been brought to users. However, The network also faces the issue of high energy consumption that requires offloading algorithm is proposed. It can make a computational offloading strategy with polynomial time complexity online. The experimental results demonstrate the algorithm's effectiveness in reducing system transmission energy consumption. Tong et al. [13] investigated the task queue backlog and system energy consumption balance of IoT devices in a two-tier MEC network. A Lyapunov-based online energy consumption optimization algorithm was proposed. The algorithm can significantly decrease the system's total energy consumption while satisfying the dynamic task arrival constraint. However, the aforementioned studies primarily rely on the batteries of mobile devices as their energy source, and the energy overhead of the system is reduced on this basis. Recently, some experts and scholars have started to consider the use of renewable energy as an energy supply. It can decrease energy consumption at its source.

The EH wireless transmission technology supplies energy to the system by harvesting renewable energy from nature [14]. This new energy supply technology is considered to be the best solution to achieve sustainable development of mobile communication networks. Benefiting from the excellent nature of EH wireless transmission technology, it is widely used in system optimization studies in the field of networks [15]. Gurakan et al. [16] consider a system model in which both users have EH devices and bi-directional energy collaboration. An enhanced two-dimensional directional water injection algorithm is implemented, taking into account the finite cell capacity constraint. From the two dimensions of time and users, the overall harvested energy is optimized and distributed to achieve maximum system throughput. Liu et al. [17] based on the current state of incomplete network observation. A cooperative energy transmission strategy is developed based on the Markov decision process of partial observation. The energy transfer and data transfer of the model are optimized without knowing the EH and the channel state during the current time interval. The strategy optimizes the system's long-term throughput by fully utilizing the acquired energy. The experimental results indicate that the proposed collaborative optimal algorithm outperforms other algorithms in terms of fast convergence and high throughput. It is applicable for device-to-device end communication and vehicular networking. However, EH is characterized by random and uncontrollable energy sources [18]. This can cause the system to generate energy causality constraints, which are not considered in most studies.

In the EH scenario, the long-term average objective function and the constraint with continuous variables contradict each other [19]. Lyapunov optimization is commonly employed by researchers to resolve this issue. Mao et al. [20] explored the balance between energy efficiency and delay in multi-user wireless power systems. By introducing wireless energy transfer technology into the MEC system, it enhances computing power to extend device battery life. By utilizing Lyapunov optimization theory, the network energy efficiency is optimized under the constraints of network stability, available communication resources and energy causality. Sun et al. [21] studied the resource management in end-to-end wireless powered MEC networks for devices. They consider the presence of uncertainty in the optimization problem with regards to dynamic task arrival and battery power time variation. By introducing Lyapunov optimization and virtual queues, the long-term problem is reformulated as a deterministic time slot drift-plus-penalty subproblem for resolution. The proposed scheme can be controlled by the size of the parameter V. It achieves the goal of balancing optimal system energy efficiency and a stable data queue. However, it doesn't take into account the system continue running energy demand. Therefore, an optimization framework is considered in this paper. It can attain the optimal system energy efficiency while ensuring the stability of both the device energy queue and the data queue.

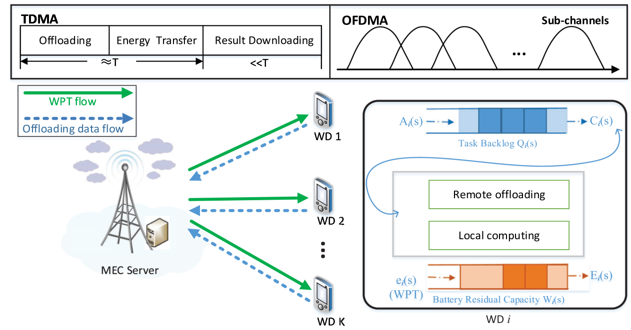

Fig. 1. Task offloading model based on wireless energy transfer.

III. System Modeling and Problem Modeling

With MEC task offloading, the system can effectively reduce energy overhead and improve service quality. However, in certain IIoT application scenarios, devices are situated in remote areas where frequent charging is not feasible [22]. Limited by their battery capacity, users expect their devices to have a long working life with fewer recharges. Therefore, in this section, WPT is considered as an energy supply scheme to be included in the study of MEC task offloading. It can effectively circumvent the problem of task processing interruptions caused by insufficient energy supply, guarantee the system's continuous optimal performance, and improve the network energy efficiency.

A. System Network Model

We consider a two-tier MEC task offloading model, which comprises of various heterogeneous IoT devices and an edge server, with the specific task offloading scenario shown in Fig. 1. Among them, each device is equipped with a power recovery device that can harness power from the surrounding environment's radio frequency signals. Let the set of n heterogeneous IoT devices be denoted as $\mathbf{N} = \{1, 2, \ldots, n\}$, indexed by variable i. In addition, the edge server in this system can be a base station or a small data center, powered by a mix of grid and renewable green energy. The edge server utilizes orthogonal frequency division multiple access (OFDMA) to communicate wirelessly with n IoT devices, while time division multiple access (TDMA) is used for energy transmission.

To assess the system's long-term stability, the operation time of the MEC system is split according to time slices. The discrete set of times is denoted as $\mathbf{S} = \{1, 2, \ldots, S\}$ and indexed by the variable s. The length of each time slot is normalized to τ. As depicted in Fig. 1, each frame is divided into two phases, the Computational Offloading and WPT phases. In the Computational Offloading phase, before the start of each cycle, each device $i \in \mathbf{N}$ generates computation tasks of size $A_i(s)$ (in bits). With ample power, some tasks are processed locally, while others are offloadinged to the edge for computation. During the

WPT phase, the device performs task offloading in the time slot $\alpha_i(s)\tau$ allocated to it, while performing local computation (without occupying the time slot). Then, supplementary power is collected via WPT for the remaining $(1-\alpha_i(s))\tau$. Since the amount of data decays sharply after the task is processed through the computing nodes, and due to the significantly higher downlink transmission rate compared to the uplink transmission rate, this paper chooses to ignore the return part of the calculation results. Therefore, the range of values of α_i should satisfy $\alpha_i(s) \in [0,1]$. To improve the efficiency of energy use, the model incorporates a sleep-wake strategy [23]. It can determine the ratio of mobile device offloading to wireless charging by weighing the remaining battery charge size. At the start of the time slot, device i determines its working or sleeping state based on the remaining battery capacity. It goes to sleep when it is low on power, spends all its time on wireless energy harvesting, and waits for the next moment of sufficient power to wake up. In this case, the value of $\alpha_i(s)$ is 0.

B. Communication Model

This model supports two computation modes for task processing, i.e., local execution and remote computation. When the remote computation mode is selected for the computation task, the data needs to be transmitted through the uplink wireless link of the MEC network; and when the IoT device is low on power it goes to sleep and needs to collect the energy emitted by the MEC network through the downlink. The assumption is made that the channel is stationary during a time slot but varies between slots. The uplink channel gain is denoted as $H_i(s)$ and the downlink channel gain is $g_i(s)$. Thus, the offloading rate that device i can achieve during time slot s is expressed as

$$R_i(s) = \omega \log_2\left(1 + \frac{P_i(s)H_i(s)}{\sigma^2 + \sum_{j\in N, j\neq i} P_j(s)H_j(s)}\right), \quad (1)$$

where ω denotes the channel bandwidth allocated by the system when the IoT device performs task offloading in time slot s, σ^2 denotes the Gaussian white noise power during data transmission over the wireless channel, and $P_i(s)$ is the transmit power of device i to transmit data in time slot s. In addition, the MEC system consists of multiple heterogeneous IoT devices, which are subject to signal interference generated by other devices when one device offloading its computing tasks. Therefore, $\sum_{j\in N, j\neq i} P_j(s)H_j(s)$ is used here to denote the interference noise from other devices on the wireless channel.

In the downlink, the base station transmits wireless energy at a constant power P_0. The energy obtained from noise and other uplink signals is negligible (because the value of P_0 is relatively large). The power collected by IoT device i during the WPT phase within a single time slot is

$$e_i(s) = \xi_i P_0 g_i(s)\left(1-\alpha_i(s)\right)\tau, \quad (2)$$

where $\xi_i \in [0,1]$ is the EH efficiency of device i.

C. Task Calculation and Energy Consumption Model

According to the sleep-wake policy, when the device is in working state will generate tasks waiting to be processed. Next, the local computation model and the offloading computation model are described in detail.

1) Local Computing Model: Local computing means that the task is handed over to the mobile device's own CPU for processing, and the required computational resources and energy consumption can be calculated by the following equations, respectively. Let φ_i be the number of CPU cycles consumed by IoT device i to process 1 b of data [24]. The size of the local computational task of device i in time slot s is

$$C_i^l(s) = \frac{f_i(s)\tau}{\varphi_i}, \quad (3)$$

where $f_i(s)$ is the CPU computation frequency of device i. Since different heterogeneous devices have different characteristics, there are some differences in their computational capabilities. By using dynamic voltage frequency scaling techniques, these devices can be controlled so that they compute at different computational frequencies at different power levels [25]. Thereby, the energy consumption in the local calculation part can be obtained as

$$E_i^l = \kappa f_i^3(s)\tau, \quad (4)$$

where κ stands for the effective switching capacitance associated with the chip architecture.

2) Offloading Calculation Model: In the offloading computing model, the task data must first be transferred to the edge. Both the channel transmission rate and the system-allocated offloading duration determine the size of the task when it is offloadinged to the edge server for processing, which is

$$C_i^M(s) = R_i(s)\alpha_i(s)\tau. \quad (5)$$

At the same time, the energy consumed to transmit the task in time slot s is

$$E_i^T(s) = P_i(s)\alpha_i(s)\tau. \quad (6)$$

In summary, when tasks are offloadinged to the edge server for processing during time slot s, the energy consumption of device i is divided into two parts. i.e., the energy consumption can be expressed as the sum of computational and transmission energy consumption, given by

$$E_i(s) = E_i^l(s) + E_i^T(s). \quad (7)$$

Correspondingly, in the time slot s the energy consumed by the MEC server computation task is

$$E^M(s) = l\sum_{i=1}^{n} C_i^M(s) + \gamma, \quad (8)$$

where γ represents the energy consumed by the MEC server in standby, independent of workload size. l is the power consumption factor required by the MEC server to calculate the unit task. Additionally, it is assumed that the MEC server has a maximum computational resource capacity of C, then there exists an offloading computational capacity constraint

$$\sum_{i=1}^{n} C_i^M(s) \leq C. \tag{9}$$

Ultimately, the energy consumption of the system for the computation task in time slot s can be expressed as

$$E^C(s) = \sum_{i=1}^{n} E_i(s) + E^M(s). \tag{10}$$

D. Queue Model

In this model, two queue models, task and power, are added and the virtual power queue is introduced to handle the power coupling. Next, the variation of the two queue backlogs is analyzed separately.

1) Task Queue Model: Each device has its own task reception and processing process. The task backlog size of each queue is $Q_i(s)$ and the task processing volume is $C_i(s)$, where $C_i(s) = C_i^l(s) + C_i^M(s)$. The task queue update formula under time slot s can be expressed as

$$Q_i(s+1) = [Q_i(s) - C_i(s)]^+ + A_i(s). \tag{11}$$

2) Power Queue Model: Every device in the system is equipped with an EH equipment, and in the previous section, we assumed that the magnitude of the energy acquired by device i from the RF signal in time slot s is $e_i(s)$. Since there is a certain loss of energy in the storage process after conversion, only a part of the energy obtained from the wireless channel in each time slot can be converted and stored as power in the battery [26]. Denote the energy successfully stored in the battery by device i as $e_i^H(s)$, which should satisfy

$$0 < e_i^H(s) \leq e_i(s). \tag{12}$$

The battery power level of IoT device i at the beginning is assumed to be $W_i(s)$. The energy consumed at the IoT device side for processing the computational task $A_i(s)$ is $E_i(s)$. The energy consumption is mainly divided into two parts: local computation and remote transmission. The energy relationship that exists in each device battery is

$$W_i(s+1) = [W_i(s) - E_i(s)]^+ + e_i^H(s). \tag{13}$$

The energy coupling effect causes the offloading decision in a time slot to impact the offloading choice in the next time slot when the device battery is deficiency. The introduction of the virtual power queue concept helps eliminate energy coupling and obtain the optimal solution. This is covered in detail in Section IV.

E. Formalization of Questions

This subsection discusses the problem of maximizing energy efficiency, a key performance index of the network, on the premise of ensuring the long-term stability of the system [27]. Its formulaic expression is

$$\begin{aligned}
&(P1) \max_{\Omega(s)} \eta_{EE} = \frac{\sum_{i=1}^{n} \bar{C}_i(s)}{\bar{E}^c(s)} \\
&\text{s.t. } C1: \ 0 \leq \alpha_i(s) \leq 1, \\
&C2: \ P_i(s) \leq P_{\max}, \\
&C3: \ f_i(s) \leq f_{\max}, \\
&C4: \ C_i^l(s) + C_i^M(s) \leq Q_i(s), \\
&C5: \ E_i(s) \leq W_i(s) \leq E_{\max}, \\
&C6: \ (9) \text{ and } (12).
\end{aligned}$$

where the set of policies defined $\Omega(s) \triangleq \{\alpha_i(s), f_i(s), e_i^H(s), i \in \mathbf{N}, s \in \mathbf{S}\}$, η_{EE} denotes the time-averaged energy efficiency of the system. $\bar{C}_i(s)$ denotes the time average system task processing volume, which is specifically sized as $\bar{C}_i(s) = \frac{1}{S}\lim_{S\to\infty}\sum_{s=1}^{S} E\{C_i(s)\}$. $\bar{E}^c(s)$ denotes the time average system energy consumption, which is specifically sized as $\bar{E}^c(s) = \frac{1}{S}\lim_{S\to\infty}\sum_{s=1}^{S} E\{E^c(s)\}$. $C1 - C6$ are the constraint conditions of the optimization problem, specifically: $C1$ is the proportion of offloading time and charging time allocated by the system. $C2$ and $C3$ represent the maximum computing frequency of the local computing CPU and the maximum transmitting power of the offloading transmission device, respectively. $C4$ indicates that the amount of tasks locally computed and offloading for time slot s should not exceed the amount of tasks present in the task queue. $C5$ constrains that the energy consumed by the mobile device during the time slot s to the current remaining battery energy and battery capacity.

As can be seen from $P1$, to solve this stochastic optimization problem and obtain an offloading strategy, it is first necessary to sense and collect information about the current network environment, including the state of the MEC server, the computing and storage capacity of the server, and other information such as the wireless channel state. The collection of network environment information provides a decision basis for the following task offloading. However, considering the abruptness of task request and the randomness of wireless channel, it is very complicated to update the prior information about system state [28], [29]. Next, the problem will be solved using Lyapunov optimization theory, so that the offloading decision can be adaptive to the network state in the system changes of each time slot.

IV. Problem Solving and Algorithm Design

In this section, an online Energy Efficiency Maximization Algorithm for Wireless Energy Transmission (EEMA-WET) is designed for solving the stochastic optimization problem established above. First, by applying the Dinkelbach optimization method, the original problem is coverted into a stochastic optimization problem. Then, an iteration-based task offloading algorithm is obtained by solving the problem using the Lyapunov optimization method. Finally, the performance of the algorithm is theoretically derived.

A. Problem Refactoring and Lyapunov Decoupling Optimization

By analyzing the form of problem P1, it is easy to see that it is a nonlinear fractional programming problem. This optimization problem is nonconvex, so the original fractional expression of the nonconvex problem needs to be transformed by the Dinkelbach optimization method [30]. Assuming that η_{EE} is the optimal system energy efficiency, then there are:

$$\eta_{EE} = \frac{\sum_{i=1}^{n} \bar{C}_i\left(\Omega^*(s)\right)}{\bar{E}^c\left(\Omega^*(s)\right)} = \max_{\Omega(s)} \frac{\sum_{i=1}^{n} \bar{C}_i\left(\Omega(s)\right)}{\bar{E}^c\left(\Omega(s)\right)}, \quad (14)$$

where $\Omega^*(s)$ represents the optimal set of solutions to the problem and $\widetilde{\Omega}(s)$ represents the set of feasible solutions. The following theorem exists for problem $P1$.

Lemma 1: The original problem obtains the same optimal solution set as the following problem when and only when the following equation obtains $f(\eta_{EE}^*) = 0$.

$$f\left(\eta_{EE}\right) = \sum_{i=1}^{n} \bar{C}_i(s) - \eta_{EE}\bar{E}^c(s). \quad (15)$$

Proof: By definition there exists

$$\eta_{EE}^* = \max_{\Omega(s)} \frac{\sum_{i=1}^{n} \bar{C}_i\left(\widetilde{\Omega}(s)\right)}{\bar{E}^c\left(\widetilde{\Omega}(s)\right)} = \frac{\sum_{i=1}^{n} \bar{C}_i\left(\Omega^*(s)\right)}{\bar{E}^c\left(\Omega^*(s)\right)} \geq \frac{\sum_{i=1}^{n} \bar{C}_i\left(\widetilde{\Omega}(s)\right)}{\bar{E}^c\left(\widetilde{\Omega}(s)\right)}. \quad (16)$$

Since the meaning of the denominator in the above equation is the time average energy consumption, it is non-negative, which can be obtained from the above equation

$$\sum_{i=1}^{n} \bar{C}_i\left(\Omega^*(s)\right) - \eta_{EE}^*\bar{E}^c\left(\Omega^*(s)\right) = 0, \quad (17)$$

$$\sum_{i=1}^{n} \bar{C}_i\left(\widetilde{\Omega}(s)\right) - \eta_{EE}^*\bar{E}^c\left(\widetilde{\Omega}(s)\right) \leq 0. \quad (18)$$

It can be seen from (17) and (18) that when $f(\eta_{EE}^*) = 0$, the original problem and (16) obtain the same optimal solution set, and the proof is complete. □

The above theorem allows for the transformation of the original problem $P1$ into the following form

$$\max_{\Omega(s)} f\left(\eta_{EE}\right) = \sum_{i=1}^{n} \bar{C}_i(s) - \eta_{EE}\bar{E}^c(s) \quad (19)$$

$$\text{s.t. } C1, C2, C3, C4, C5 \text{ and } C6.$$

Next, to eliminate the impact of energy coupling, a virtual power queue is introduced in this paper. It can effectively control the stability of the device power queue, so it can maintain long-term stable operation. Specifically, the virtual power queue of the device i in time slot s is defined as

$$\tilde{W}_i(s) = W_i(s) - \theta_i, \quad (20)$$

where θ_i is the energy perturbation parameter [31]. The initial battery power queue backlog will be maintained to fluctuate around the θ_i region provided that the virtual power queue remains steady-state. It is worth noting that the energy coupling effects between each other have been eliminated at this point, thus ensuring interference-free task offloading decisions between time slots. The values of the energy perturbation parameters satisfy the following inequality

$$\theta_i \geq \tilde{E}_i^{\max} + V \cdot \left(E_i^{\min}\right)^{-1}, \quad (21)$$

where $\tilde{E}_i^{\max} = \max\{\kappa(f_i^{\max})^3\tau, P_i^{\max}\alpha_i^{\max}\tau\}$, V is the Lyapunov control parameter, and $E_i^{\min}$ is the lowest energy level allowed for battery discharge. Therefore, by substituting (20) into (13), it is possible to express the virtual power queue as

$$\tilde{W}\left(s+1\right) = \tilde{W}(s) - E_i(s) + e_i^H(s). \quad (22)$$

Next, Lyapunov optimization theory is introduced to ensure the stability of the queue. Lyapunov function is used to represent the state of all queues in time slot s, which is defined as

$$L\left(\Theta(s)\right) = \frac{1}{2}\sum_{i=1}^{n}\left[Q_i^2(s) + \tilde{W}_i^2(s)\right]. \quad (23)$$

The vector $\Theta(s) = \{Q_1(s), \ldots, Q_n(s), H_1(s), \ldots, H_n(s)\}$ represents the backlog status of all devices in the system. At the start of the time slot, the initial value of the Lyapunov function is 0. i.e., $L(\Theta(0)) = 0$. The size of $L(\Theta(s))$ represents the relative congestion level of the system, and by reducing its value, the congestion of the queue can be effectively relieved. In this paper, $A_i(s)$ can be considered as a set of random variables. Therefore, we introduce the expectation operation. The next step involves defining the Lyapunov drift function as

$$\Delta L\left(\Theta(s)\right) = E\left\{L\left(\Theta\left(s+1\right)\right) - L\left(\Theta(s)\right)|\Theta(s)\right\}, \quad (24)$$

it reflects the variation of the system queue backlog in two adjacent time slots, and ensures the stability of the system queue by reducing its value as much as possible.

Lyapunov theory's drift-plus-penalty theory states that the addition of the penalty function gives

$$\Delta_V L\left(\Theta(s)\right) = \Delta L\left(\Theta(s)\right) - VE\left\{\sum_{i=1}^{n} C_i(s) - \eta_{EE}E^c(s)\right\}, \quad (25)$$

where V refers to the Lyapunov control parameter, which weighs the importance given to the queue length and the optimization objective.

Lemma 2: There exists $B = \frac{1}{2}\sum_{i=1}^{n}[(C_i^{\max})^2 + (A_i^{\max})^2 + (e_i^{H,\max})^2 + (E_i^{\max})^2]$, a positive real number associated with the squared term of the Lyapunov function such that an exact upper bound exists for the drift plus penalty function $\Delta_V L(\Theta(s))$ [32]

$$\begin{aligned}\Delta_V L\left(\Theta(s)\right) \leq B &+ \sum_{i=1}^{n} Q_i(s)\left[A_i(s) - C_i(s)\right] \\ &+ \sum_{i=1}^{n} \widetilde{W}_i(s)\left[e_i^H(s) - E_i(s)\right] \\ &- VE\left\{\sum_{i=1}^{n} C_i(s) - \eta_{EE}E^c(s)\right\}. \quad (26)\end{aligned}$$

Proof: Expand the two queues separately according to the inequality $(\max[Q-b,0]+a)^2 \leq Q^2+a^2+b^2-2Q(b-a)$ [33].

1) For the task queue:

$$\begin{aligned} Q_i^2(s+1)-Q_i^2(s) &\leq C_i^2(s)+A_i^2(s) \\ &\quad -2Q_i(s)\left[C_i(s)-A_i(s)\right] \\ \frac{1}{2}\sum_{i=1}^{n} Q_i^2(s) &\leq \frac{1}{2}\sum_{i=1}^{n}\left[C_i^2(s)+A_i^2(s)\right] \\ &\quad +\sum_{i=1}^{n} Q_i(s)\left[A_i(s)-C_i(s)\right]; \end{aligned} \tag{27}$$

2) For the virtual power queue:

$$\begin{aligned} \tilde{W}_i^2(s+1)-\tilde{W}_i^2(s) &\leq \left(e_i^H(s)\right)^2+E_i^2(s) \\ &\quad -2\tilde{W}_i(s)\left[E_i(s)-e_i^H(s)\right] \\ \frac{1}{2}\sum_{i=1}^{n} \tilde{W}_i^2(s) &\leq \frac{1}{2}\sum_{i=1}^{n}\left[\left(e_i^H(s)\right)^2+E_i^2(s)\right] \\ &\quad +\sum_{i=1}^{n} \tilde{W}_i(s)\left[e_i^H(s)-E_i(s)\right]. \end{aligned} \tag{28}$$

After substituting (27) and (28) into (25), the inequality is deflated to obtain the conclusion and the proof is over. □

The Lyapunov optimization technique transforms the original optimization problem into solving the upper bound on the minimization of the drift plus penalty function [34]. Thereby, the new optimization objective function is described by the following equation:

$$\begin{aligned} (P2)\ \min_{\Omega(s)} & -\sum_{i=1}^{n} Q_i(s)C_i(s)+\sum_{i=1}^{n}\tilde{W}_i(s)\left[e_i^H(s)-E_i(s)\right] \\ & -V\left[\sum_{i=1}^{n} C_i(s)-\eta_{EE}E^c(s)\right] \\ & \text{s.t.} \quad C1, C2, C3, C4, C5 \text{ and } C6. \end{aligned} \tag{29}$$

Compared with $P1$, problem $P2$ does not have energy coupling and does not require a lot of system prior information. The queue is guaranteed to be stable by controlling the size of the Lyapunov drift function, while the optimal value of the objective function is obtained by minimizing the penalty function. To achieve the global optimal solution of the problem, in the next section the problem $P2$ to be solved is solved by transforming it into three independent subproblems and solving them separately. And an energy efficiency maximization algorithm EEMA-WET based on wireless energy transfer is designed.

B. Algorithm Design

By substituting the expressions of each quantity into problem $P2$, the original problem can be decoupled into three independent subproblems by observation and integration as follows.

$$Q1: \min_{e_i^H(s)} \sum_{i=1}^{n} \tilde{W}_i(s)e_i^H(s)$$

$$\begin{aligned} Q2: \max_{\alpha_i(s)} & \sum_{i=1}^{n}\left(Q_i(s)+V\right)R_i(s)\alpha_i(s)\tau \\ & +\sum_{i=1}^{n}\tilde{W}_i(s)\left(P_i(s)+\xi_i P_0 g_i(s)\right)\alpha_i(s)\tau \\ & -\sum_{i=1}^{n} V\eta_{EE}\left(P_i(s)+lR_i(s)\right)\alpha_i(s)\tau \end{aligned}$$

$$\begin{aligned} Q3: \max_{f_i(s)} & \sum_{i=1}^{n}\left(Q_i(s)+V\right)\frac{f_i(s)\tau}{\varphi_i} \\ & +\sum_{i=1}^{n}\left(\tilde{W}_i(s)-V\eta_{EE}\right)\kappa f_i^3(s)\tau \end{aligned}$$

Among them, Q1 solves the problem of device energy harvesting, Q2 solves the problem of wireless channel resource allocation for offloading and energy harvesting, and Q3 solves the problem of local computing power decision. By solving these decoupled independent subproblems, the overall optimal solution of the problem is obtained.

1) Device Energy Harvesting:

$$Q1: \min_{e_i^H(s)} \tilde{W}_i(s)e_i^H(s) \qquad \text{s.t. } (12). \tag{30}$$

By observing the form of the problem, this subproblem is a linear programming problem, so the analytic solution of the optimization variable is

$$e_i^{H^*}(s) = \begin{cases} 0 & \tilde{W}_i(s)>0; \\ e_i(s) & \tilde{W}_i(s)\leq 0. \end{cases} \tag{31}$$

As the problem's objective function is a first-order linear function, the optimal acquisition energy depends on the virtual power queue level within the current time slot. When the coefficient of the objective function is greater than zero, the optimization variable should take the maximum value under the restriction; conversely, when the coefficient is less than zero, the optimization variable should take the minimum value of zero. This result is also consistent with the expected strategy. When the objective function coefficient, i.e., the virtual power queue level, is greater than zero, it indicates that the current device has sufficient energy for that time slot and does not need to be charged. It can perform as much task offloading as possible. Conversely, it needs to collect as much energy as possible in the current time slot to store energy for the task offloading in the next time slot. This strategy allows the virtual power queue level to be maintained near θ_i, which in turn eliminates the energy coupling effect.

Algorithm 1: Offloading and Energy Harvesting Wireless Channel Resource Allocation Algorithm (WCRAA).

Input: the maximum amount of computing resources available to the MEC server C, device battery power level $W_i(s)$, uplink radio channel gain $H_i(s)$, device wireless channel transmission rate $R_i(s)$;
Output: wireless channel resource allocation ratio $\alpha_i(s)$;
1: Initialize the radio channel resource allocation ratio $\alpha_i(s) = 0$ and the maximum amount of computing resources currently available to the MEC server $C(s) = C$;
2: Calculate the offloading priority function for each device $O_i(s)$;
3: Sort all devices in descending order based on the offloading priority function's size, obeying $O_i(s) \geq O_i(s+1)$;
4: **for all** $i \in \boldsymbol{N}$ **do**
5:   **if** $O_i(s) > 0$ **then**
6:     $M_i(s) = \min\left\{\max\left[1 - \frac{E_{\max} - W_i(s)}{\xi_i P_0 H_i(s)\tau}, 0\right], 1\right\}$;
7:     $\alpha_i(s) = \min\left\{M_i(s), \frac{C(s)}{R_i(s)\tau}\right\}$;
8:   **else**
9:     $\alpha_i(s) = 0$;
10:   **end if**
11:   $C(s) = C(s) - R_i(s)\alpha_i(s)\tau$;
12: **end for**
13: **return** $\boldsymbol{\alpha}(s) = \{\alpha_1(s), \alpha_2(s), \ldots, \alpha_n(s)\}$.

Algorithm 2: The Maximize Energy Efficiency of Wireless Energy Transmission Algorithm (EEMA-WET).

1: Initialization:
2: Energy efficiency η_{EE}, the maximum number of iterations $q_{\max}$, iteration accuracy ϵ;
3: Solve the offloading strategy $\Omega(s)$:
4: Solve for the device storage energy $e_i^H(s)$ according to (2) and (31);
5: Call Algorithm 1 to solve for the ratio of radio channel resources allocated to the device $\alpha_i(s)$;
6: According to (34), solve for the device local calculation CPU frequency $f_i(s)$;
7: Calculate energy efficiency:
8: Calculate and update each system performance index, and update energy efficiency according to (14);
9: Update iteration number $q = q + 1$;
10: Update queue information:
11: Update the backlog $Q_i(s)$ of the task queue according to (11);
12: Update the battery power level $W_i(s)$ according to (13);
13: Update the virtual power queue $\tilde{W}_i(s)$ according to (22);
14: Update energy efficiency:
15: **if** $q > q_{\max}$ or $\eta_{EE}^{q} - \eta_{EE}^{q-1} \leq \epsilon$ **then**
16:   Output optimal energy efficiency η_{EE}^{*};
17: **else**
18:   Return to Step 2 and repeat the loop.
19: **end if**

2) Offloading and Energy Harvesting Wireless Channel Resource Allocation:

$$
\begin{aligned}
Q2: \max_{\alpha_i(s)} \; & (Q_i(s) + V) R_i(s)\alpha_i(s)\tau \\
& + \tilde{W}_i(s) (P_i(s) + \xi_i P_0 g_i(s)) \alpha_i(s)\tau \\
& - V\eta_{EE} (P_i(s) + lR_i(s)) \alpha_i(s)\tau \\
& \text{s.t. } 0 \leq \alpha_i(s) \leq 1,
\end{aligned} \tag{32}
$$

where $O_i(s) = (Q_i(s) + V)R_i(s) - V\eta_{EE}(P_i(s) + lR_i(s)) + \widetilde{W}_i(s)(P_i(s) + \xi_i P_0 g_i(s))$.

Formally, the optimization problem is defined similarly to the knapsack problem, and the best solution is to select the highest value items to fill the backpack sequentially. Therefore, the algorithm sorts the IoT devices in descending order of $O_i(s)$. Devices with higher $O_i(s)$ values have higher offloading priority. The specific steps of the solution strategy are shown in Algorithm 1.

3) Local Computing Power Decisions:

$$
Q3: \max_{f_i(s)} \; (Q_i(s) + V) \frac{f_i(s)\tau}{\varphi_i} + \left(\tilde{W}_i(s) - V\eta_{EE}\right) \kappa f_i^3(s)\tau \tag{33}
$$

$$
\text{s.t. } 0 \leq f_i(s) \leq f_{\max}.
$$

The optimization objective function of subproblem $Q3$ is derivable, and the original function is convex according to the fact that the objective function's second-order derivative is negative. Therefore, one possible method is to locate the extreme value point of the objective function and compare the corresponding function values with those of the boundary points of the definition domain. In this way, the optimal solution of the objective function is determined. Assuming that the extreme value point is denoted by $f_i(s)$, the analytical solution can be expressed as

$$
f_i^*(s) = \begin{cases} \sqrt{\frac{Q_i(s) + V}{3\left(V\eta_{EE} - \tilde{W}_i(s)\right)\kappa\varphi_i}} & 0 \leq f_i(s) \leq f_{\max}; \\ f_{\max} & f_i(s) \geq f_{\max}. \end{cases} \tag{34}
$$

Based on the above analysis, aiming at the optimization problem studied, a Lyapunov optimization based task offloading method for maximizing energy efficiency of wireless energy transmission is designed in this paper. The Algorithm 2 displays the EEMA-WET pseudo-code. Algorithm 2 is an iteration-based algorithm for maximizing the energy efficiency of the system. Setting the number of iterations appropriately can effectively reduce the impact of this algorithm on the complexity of the offloading strategy.

Next, the time complexity of EEMA-WET is calculated. The feasibility of the algorithm is verified theoretically. According to Algorithm 1, WCRAA determines the offloading priority of the device. The process of traversing all devices requires $O(\log n)$ operations with the help of the fast sorting algorithm. In Algorithm 2, it need to call Algorithm 1. And the optimal energy efficiency is solved by iterations in each time slot. Finally, the time complexity of EEMA-WET is $O(n^2 \log n)$.

TABLE I
EXPERIMENTAL PARAMETER SETTINGS

Parameter	Description	Value
n	the number of devices	30
φ_i	the number of CPU cycles needed for a device to process one bit of data	U[1000,2000] cycles/bit
κ	the coefficient of capacitance switching	10^{-26}
V	the Lyapunov control parameter	2×10^{-3}
ω	the wireless channel bandwidth	1 MHz
σ^2	the Gaussian white noise power	10^{-13}
$f_{\max}$	the device maximum computing power	1.5 GHz
$P_{\max}$	the device maximum transmission power	1 W
P_0	the BS wireless energy transmission transmitting power	50 W
$A_i(s)$	the task arrival of the device	U[800,1200] bits
$H_i(s)$	the uplink wireless channel gain	$Exp(1)$

V. SIMULATION EXPERIMENTS AND RESULTS ANALYSIS

In this section, a simulation environment based on MATLAB was built to simulate a scenario consisting of multiple EH-enabled devices and an edge server, and the algorithm implementation and performance verification are carried out for maximizing energy efficiency of MEC system based on wireless energy transmission. First, the experimental parameter settings in the simulation experiments are introduced and the stability of the system is verified. Second, the influence of the main parameters in the algorithm on the performance of the algorithm is analyzed. The different task arrival rates reflect the computational load, which is usually measured by the size of the data volume of the task. The energy efficiency of the system and the robustness of the algorithm are analyzed by modifying different parameters. And finally, the algorithm's performance is verified through comparison experiments with other strategies.

In our experiments, a simulation of task computation is conducted for a scenario that includes one MEC server and 30 IoT devices. We simulate 2000 time slots and improve the reliability by running several times, where each time slot has a duration of $\tau = 1$ ms. The tasks or data involved in the experiments are partly based on extensive literature references and careful comparison with previous work. [26], [35] Some important parameter values are set in Table I.

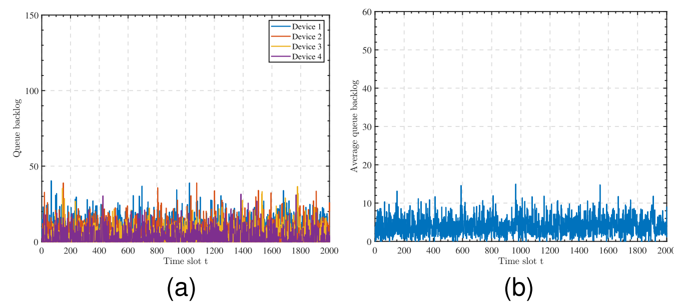

Fig. 2. (a) Devices average and (b) the system average task queue stability.

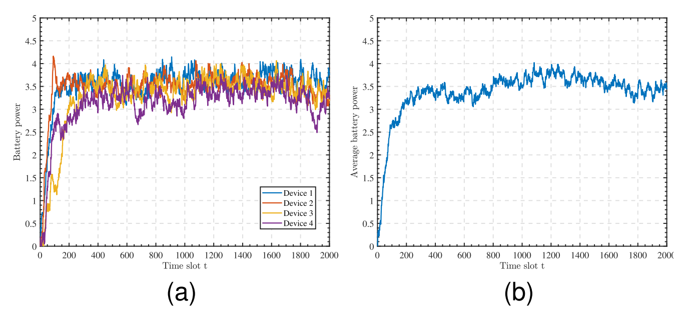

Fig. 3. (a) Devices average and (b) the system average battery power queue stability.

A. System Stability Verification

The experiment randomly selects 4 out of 30 IoT devices and observes whether their queue backlog meets the system stability prerequisites. Fig. 2(a) and (b) depict the task queue backlogs for the four selected devices and the average task queue backlog levels for all devices, respectively. In Fig. 2, as simulation time progresses, it becomes apparent that the task queue size remains at a low level for both individual devices and the system as a whole. This indicates that the computational tasks are reasonably distributed, and the MEC system is running well without congestion. Fig. 3(a) and (b) depict the power queues for the selected four devices' batteries and the average battery power queue levels for all devices, respectively. Fig. 3 shows that the size of the battery power queue grows rapidly with time and stabilizes rapidly. This is due to the low battery power at the beginning. After a period of charging, the battery power level increases rapidly and stabilizes around the energy perturbation factor, which is as expected. From the experiments, it can be found that the intra-queue backlog of the task queue is always kept at a low level. There is no significant difference in the queue backlog order of magnitude between the task queue and the energy queue. While the battery energy level of the energy queue is always maintained at a relatively high steady state. The reason for these two different steady states is that the goals of the two queues are different. The device is more capable of handling tasks, so most of the tasks have already been processed and only a small percentage of tasks are left in the queue waiting to be processed. However, the energy queue needs to be kept at a higher energy level to maintain the normal operation of the device and to improve the energy efficiency of the system. Therefore, it is experimentally verified that the system stability prerequisites are effectively guaranteed using the EEMA-WET algorithm.

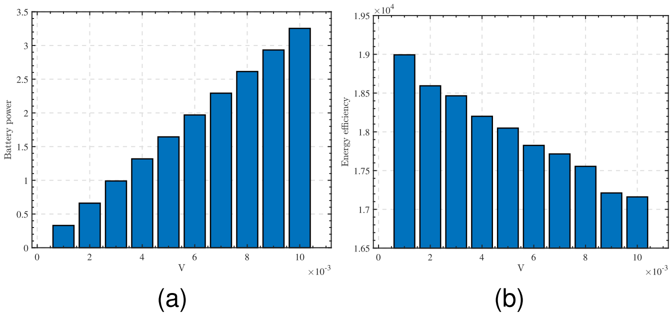

Fig. 4. Impact of V on (a) battery power queue level and (b) energy efficiency.

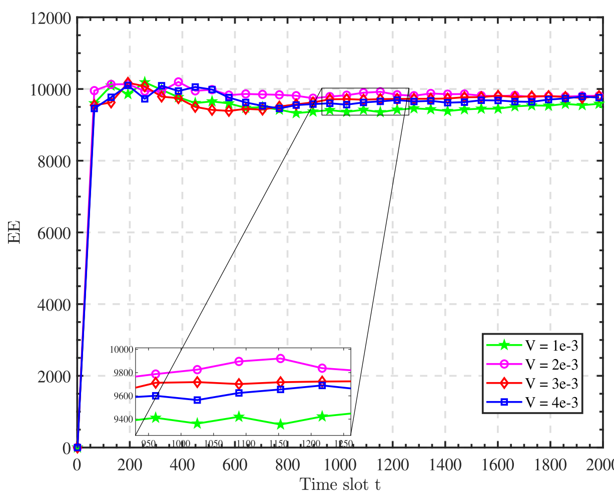

Fig. 5. Selection of control parameter V.

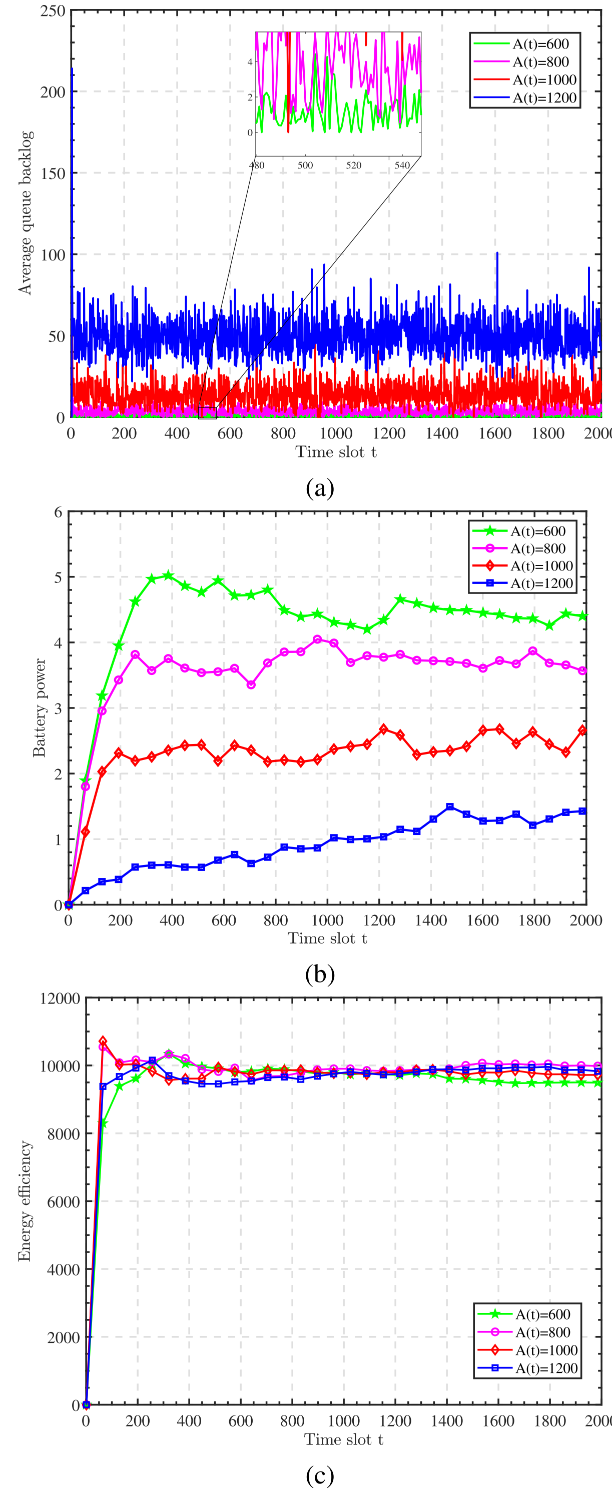

Fig. 6. Impact of $A(t)$ on (a) task queue backlog, (b) battery power queue level, and (c) energy efficiency.

B. Analysis of Algorithm Parameters

1) Impact of Control Parameter V: Fig. 4 shows the effect of the difference in the values of Lyapunov control parameters V in the EEMA-WET algorithm on the system battery power queue level and energy efficiency. As shown in Fig. 4(a), the battery power queue level is proportional to the magnitude of V. That is, as the control parameters increase, the power level of the device battery rises. This is because the algorithm places a higher emphasis on wireless energy transfer when the control parameter V is large. The device keeps its own battery power at a high level through EH.

In contrast, Fig. 4(b) shows that a decrease in the system energy efficiency occurs when the control parameter V increases. That is, the system energy efficiency is inversely proportional to the control parameter V. The experiment demonstrates that by adjusting the control parameter V's size, the proposed task offloading algorithm can strike a balance between energy efficiency and queue length. Fig. 5 shows that under the condition of satisfying the system stability, an optimal system energy efficiency can be obtained by selecting an optimal control parameter V by ablation experiments at the same order of magnitude.

2) Impact of Task Arrival $A(T)$: The impact of varying task sizes on the EEMA-WET algorithm's operation is simulated in this subsection. The initial size of the task size is set to 600 and is increased by 200 each time until it reaches 1200. From Fig. 6(a), we can see that the backlog of task queues gradually increases as the task size keeps increasing, but still remains at a low level. This indicates that the system has good processing capability in the face of tasks of different sizes. Fig. 6(b) displays

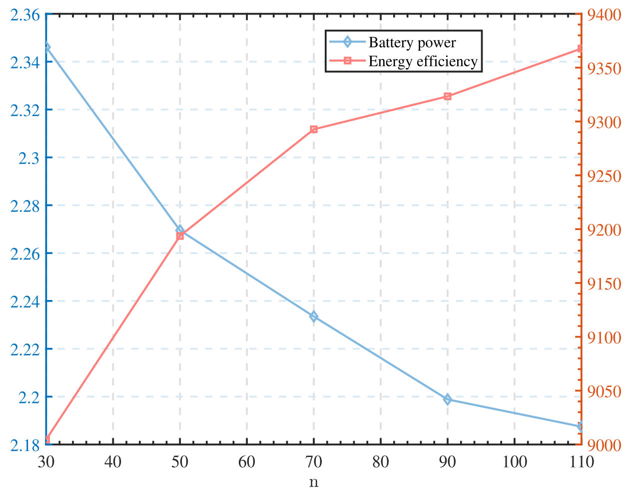

Fig. 7. Impact of control parameter V.

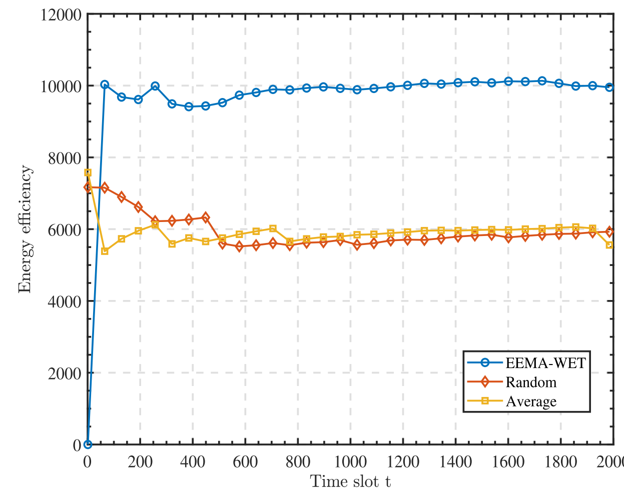

Fig. 8. Energy efficiency of the system under different algorithms.

the device power queue level's variation when dealing with tasks of different sizes. This phenomenon occurs because the resources for both computation and communication are limited during the server operation. Therefore, as the task size increases, the energy consumption required for the computational tasks increases, and the required wireless energy transfer power does not increase. Fig. 6(c) shows that the system energy efficiency can be maintained in a more stable optimal state as the task arrivals increase, reflecting the excellent performance of the algorithm.

3) Impact of the Number of Devices Accessing n: To investigate the robustness of the algorithm with increasing amount of devices connected, the battery power and energy efficiency of the system with different number of devices are depicted in Fig. 7. The battery power level decreases slowly as the amount of connected devices increases. Due to the constant wireless energy transfer power of the BS, the amount of energy collected decreases as the number of devices increases. The battery power level remains at a high level to ensure smooth operation of the MEC system. And the system energy efficiency also experiences slow growth as the amount of connected devices increases, which reflects the good robustness of the EEMA-WET algorithm.

C. Comparison Experiment

To further assess the performance of the EEMA-WET algorithm, two other benchmark strategies are selected for comparison in this section, hoping to demonstrate their performance levels more comprehensively and effectively. The first strategy is random assignment, which randomly allocates channel resources to the offloading and wireless energy transfer phases in each time slot. The other strategy is equal allocation, where the channel resources are equally allocated to offloading and wireless energy transfer in each time solt. Fig. 8 depict the system energy efficiency of three different strategies, respectively, where EEMA-WET represents the proposed algorithm in this paper, and Random and Average represent the random and average allocation strategies, respectively. It is obvious that the proposed algorithm outperforms the other two by a significant margin. This is because one of the significant advantages of EEMA-WET is its ability to quickly adapt to the continuous changes in task requests and wireless channel states caused by external reasons. And based on this adaptive characteristic, it makes balanced decisions regarding task offloading and energy allocation, maximizing the energy efficiency of the system.

VI. Conclusion

In this paper, we study the offloading strategy of task computing for a two-tier MEC system consisting of an IoT device with wireless energy transmission and an edge server, taking into account the limited battery capacity of the device. This policy enables optimal offloading proportional distribution of computing tasks on the device between local and edge servers to optimize task computing efficiency and maximize system energy efficiency. Specifically, mathematical models are constructed separately for the processes of local computing, edge server processing, communication, and energy harvesting. The balance of battery power queue and task queue is considered to ensure the smooth operation of the system. Meanwhile, the introduction of the virtual energy queue concept helps decouple the optimization problem, eliminating the energy coupling effect during long-term system operation. Finally, the problem is solved by Lyapunov optimization technique and convex optimization method, and an energy efficiency maximization algorithm EEMA-WET based on wireless energy transfer is designed.

Acknowledgment

The authors would likes to thank the editors and reviewers for their insightful comment and valuable suggestions.

References

[1] H. Duan, Y. Zheng, C. Wang, and X. Yuan, "Treasure collection on foggy islands: Building secure network archives for Internet of Things," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2637–2650, Apr. 2019.

[2] B. Pu, K. Li, S. Li, and N. Zhu, "Automatic fetal ultrasound standard plane recognition based on deep learning and IIoT," *IEEE Trans. Ind. Inform.*, vol. 17, no. 11, pp. 7771–7780, Nov. 2021.

[3] K. Gai, M. Qiu, H. Zhao, and X. Sun, "Resource management in sustainable cyber-physical systems using heterogeneous cloud computing," *IEEE Trans. Sustain. Comput.*, vol. 3, no. 2, pp. 60–72, Second Quarter 2018.

[4] G. Xie, X. Xiao, H. Peng, R. Li, and K. Li, "A survey of low-energy parallel scheduling algorithms," *IEEE Trans. Sustain. Comput.*, vol. 7, no. 1, pp. 27–46, First Quarter 2021.

[5] Y. Kang et al., "HWOA: An intelligent hybrid whale optimization algorithm for multi-objective task selection strategy in edge cloud computing system," *World Wide Web*, vol. 25, no. 5, pp. 2265–2295, 2022.

[6] P. Zhao, J. Tao, A. Rauf, F. Jia, and L. Xu, "Load balancing for energy-harvesting mobile edge computing," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. 104, no. 1, pp. 336–342, 2021.

[7] G. Piro et al., "HetNets powered by renewable energy sources: Sustainable next-generation cellular networks," *IEEE Internet Comput.*, vol. 17, no. 1, pp. 32–39, Jan./Feb. 2013.

[8] Y. Alsaba, S. K. A. Rahim, and C. Y. Leow, "Beamforming in wireless energy harvesting communications systems: A survey," *IEEE Commun. Surv. Tut.*, vol. 20, no. 2, pp. 1329–1360, Second Quarter 2018.

[9] D. Djenouri and M. Bagaa, "Energy-aware constrained relay node deployment for sustainable wireless sensor networks," *IEEE Trans. Sustain. Comput.*, vol. 2, no. 1, pp. 30–42, Jan.-Mar. 2017.

[10] J. Mei, L. Dai, Z. Tong, X. Deng, and K. Li, "Throughput-aware dynamic task offloading under resource constant for MEC with energy harvesting devices," *IEEE Trans. Netw. Service Manag.*, to be published, doi: 10.1109/TNSM.2023.3243629.

[11] S. Xia, Z. Yao, Y. Li, and S. Mao, "Online distributed offloading and computing resource management with energy harvesting for heterogeneous MEC-enabled IoT," *IEEE Trans. Wireless Commun.*, vol. 20, no. 10, pp. 6743–6757, Oct. 2021.

[12] Y. Chen, N. Zhang, Y. Zhang, X. Chen, W. Wu, and X. Shen, "Energy efficient dynamic offloading in mobile edge computing for Internet of Things," *IEEE Trans. Cloud Comput.*, vol. 9, no. 3, pp. 1050–1060, Third Quarter 2021.

[13] Z. Tong, J. Cai, J. Mei, K. Li, and K. Li, "Dynamic energy-saving offloading strategy guided by Lyapunov optimization for IoT devices," *IEEE Internet Things J.*, vol. 9, no. 20, pp. 19903–19915, Oct. 2022.

[14] M. Moradian, F. Ashtiani, and Y. J. Zhang, "Optimal relaying in energy harvesting wireless networks with wireless-powered relays," *IEEE Trans. Green Commun. Netw.*, vol. 3, no. 4, pp. 1072–1086, Dec. 2019.

[15] Y. Hu, C. Qiu, and Y. Chen, "Lyapunov-optimized two-way relay networks with stochastic energy harvesting," *IEEE Trans. Wireless Commun.*, vol. 17, no. 9, pp. 6280–6292, Sep. 2018.

[16] B. Gurakan, O. Ozel, J. Yang, and S. Ulukus, "Two-way and multiple-access energy harvesting systems with energy cooperation," in *Proc. Conf. Rec. 46th Asilomar Conf. Signals Syst. Comput.*, 2012, pp. 58–62.

[17] H. Liu, X. Zhao, H. Liang, and Z. Li, "POMDP-based energy cooperative transmission policy for multiple access model powered by energy harvesting," *IEEE Trans. Veh. Technol.*, vol. 68, no. 6, pp. 5747–5757, Jun. 2019.

[18] G. Zhang, Y. Chen, Z. Shen, and L. Wang, "Energy management for multi-user mobile-edge computing systems with energy harvesting devices and qos constraints," in *Proc. IEEE 27th Int. Conf. Comput. Commun. Netw.*, 2018, pp. 1–6.

[19] X. Ling, J. Gong, R. Li, S. Yu, Q. Ma, and X. Chen, "Dynamic age minimization with real-time information preprocessing for edge-assisted IoT devices with energy harvesting," *IEEE Trans. Netw. Sci. Eng.*, vol. 8, no. 3, pp. 2288–2300, Third Quarter 2021.

[20] S. Mao, S. Leng, S. Maharjan, and Y. Zhang, "Energy efficiency and delay tradeoff for wireless powered mobile-edge computing systems with multi-access schemes," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1855–1867, Mar. 2019.

[21] M. Sun, X. Xu, Y. Huang, Q. Wu, X. Tao, and P. Zhang, "Resource management for computation offloading in D2D-aided wireless powered mobile-edge computing networks," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8005–8020, May 2021.

[22] Y. Deng, Z. Chen, X. Yao, S. Hassan, and A. M. Ibrahim, "Parallel offloading in green and sustainable mobile edge computing for delay-constrained IoT system," *IEEE Trans. Veh. Technol.*, vol. 68, no. 12, pp. 12202–12214, Dec. 2019.

[23] G. Zhang, Y. Chen, Z. Shen, and L. Wang, "Distributed energy management for multiuser mobile-edge computing systems with energy harvesting devices and QoS constraints," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4035–4048, Jun. 2019.

[24] Y. Mao, J. Zhang, S. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5994–6009, Sep. 2017.

[25] A. Zou, K. Garimella, B. Lee, C. Gill, and X. Zhang, "F-LEMMA: Fast learning-based energy management for multi-/many-core processors," in *Proc. ACM/IEEE Workshop Mach. Learn. CAD*, 2020, pp. 43–48.

[26] M. Guo, W. Wang, X. Huang, Y. Chen, L. Zhang, and L. Chen, "Lyapunov-based partial computation offloading for multiple mobile devices enabled by harvested energy in MEC," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 9025–9035, May 2020.

[27] M. Sheng, Y. Li, X. Wang, J. Li, and Y. Shi, "Energy efficiency and delay tradeoff in device-to-device communications underlaying cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 1, pp. 92–106, Jan. 2016.

[28] C. Chen, K. Li, A. Ouyang, Z. Zeng, and K. Li, "GFlink: An in-memory computing architecture on heterogeneous CPU-GPU clusters for Big Data," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 6, pp. 1275–1288, Jun. 2018.

[29] C. Chen, K. Li, S. G. Teo, X. Zou, K. Li, and Z. Zeng, "Citywide traffic flow prediction based on multiple gated spatio-temporal convolutional neural networks," *ACM Trans. Knowl. Discov. Data*, vol. 14, no. 4, pp. 1–23, 2020.

[30] J. Mu, W. Ouyang, Z. Jing, B. Li, and F. Zhang, "Energy-efficient interference cancellation in integrated sensing and communication scenarios," *IEEE Trans. Green Commun. Netw.*, vol. 7, no. 1, pp. 370–378, Mar. 2023.

[31] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[32] M. J. Neely and L. Huang, "Dynamic product assembly and inventory control for maximum profit," in *Proc. IEEE 49th Conf. Decis. Control*, 2010, pp. 2805–2812.

[33] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synth. Lectures Commun. Netw.*, vol. 3, no. 1, pp. 1–211, 2010.

[34] Z. Chang, L. Liu, X. Guo, and Q. Sheng, "Dynamic resource allocation and computation offloading for IoT fog computing system," *IEEE Trans. Ind. Inform.*, vol. 17, no. 5, pp. 3348–3357, May 2021.

[35] F. Zhao, Y. Chen, Y. Zhang, Z. Liu, and X. Chen, "Dynamic offloading and resource scheduling for mobile-edge computing with energy harvesting devices," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 2, pp. 2154–2165, Jun. 2021.

Zhao Tong (Member, IEEE) received the PhD degree in computer science from Hunan University, Changsha, China, in 2014. He was a visiting scholar with the Georgia State University from 2017 to 2018. He is currently an associate professor with the College of Information Science and Engineering of Hunan Normal University, the young backbone teacher of Hunan Province, China. His research interests include parallel and distributed computing systems, resource management, Big Data, and machine learning algorithm. He has published more than 25 research papers in international conferences and journals, such as IEEE-TPDS, Information Sciences, FGCS, NCA, and JPDC, PDCAT, etc. He is a senior member of the China Computer Federation (CCF).

Jinhui Cai received the BE degree from the Qilu University of Technology (Shandong Academy of Sciences), Jinan, China, in 2019. He is currently working toward the MS degree degree with the College of Information Science and Engineering of Hunan Normal University, Changsha, China. His research interests include cloud edge integration, task offloading, objective optimization, and artificial intelligence.

Jing Mei received the PhD degree in computer science from Hunan University, China, in 2015. She is currently a lecturer in the College of Information Science and Engineering at Hunan Normal University. Her research interests include parallel and distributed computing, cloud computing, combinatorial optimization, etc. She has published more than 15 research articles in international conferences and journals, such as *IEEE Transactions on Computers*, IEEE-SC, IEEE-TPDS, Cluster Computing, JGC, TJSC, ICPP, etc. She is a member of China Computer Federation (CCF).

Kenli Li (Senior Member, IEEE) received the PhD degree in computer science from the Huazhong University of Science and Technology, China, in 2003. He was a visiting scholar with the University of Illinois, Urbana-Champaign from 2004 to 2005. He is currently a cheung kong professor of computer science and technology with Hunan University (HNU), the assistant president of the HNU, the dean of the College of Information Science and Engineering of HNU, and the director in the National Supercomputing Center in Changsha. His major research interests include parallel and distributed processing, high-performance computing, and Big Data management. He has published more than 350 research papers in international conferences and journals, such as *IEEE Transactions on Computers*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Cloud Computing*, HPCA, SC, MM, AAAI, DAC, ICPP, etc. He is a fellow of the China Computer Federation (CCF). He is currently serving or has served as an associate editor for IEEE-TC, IEEE-TII, and IEEE-TSUSC.

Keqin Li (Fellow, IEEE) is a SUNY distinguished professor of computer science. His current research interests include parallel computing and high-performance computing, distributed computing, energy-efficient computing and communication, heterogeneous computing systems, cloud computing, Big Data computing, CPU-GPU hybrid and cooperative computing, multicore computing, storage and file systems, wireless communication networks, sensor networks, peer-to-peer file sharing systems, mobile computing, service computing, Internet of Things, and cyber-physical systems. He has published more than 510 journal articles, book chapters, and refereed conference papers, and has received several best paper awards. He is currently serving or has served on the editorial boards of *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Computers*, *IEEE Transactions on Cloud Computing*, *IEEE Transactions on Services Computing*, and *IEEE Transactions on Sustainable Computing*.