



Research Article

Energy-Efficient Resource Scheduling and Computation Offloading Strategy for Solar-Powered Agriculture WSN

Juan Gao ¹, Runze Wu ¹, Jianhong Hao,¹ Chen Xu,² Haobo Guo,¹ and Haonan Wang¹

¹Electrical and Electronic Engineering Department, North China Electric Power University, Beijing 102206, China

²School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China

Correspondence should be addressed to Runze Wu; wurz@ncepu.edu.cn

Received 25 March 2022; Revised 23 September 2022; Accepted 26 September 2022; Published 11 April 2023

Academic Editor: Zhenxing Zhang

Copyright © 2023 Juan Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

IoT-based smart agriculture plays a significant role in building a high-yield, sustainable, and intelligent modern agriculture. However, limited battery capacity and low-power processors of sensors cannot accommodate the exponential expansion of data from smart agriculture sensing terminals. To overcome the challenges, we introduced solar harvesting and multiaccess edge computing (MEC) to investigate sustainable monitoring of smart agriculture in solar-powered MEC-enabled WSNs. Considering the cyclical and day-night fluctuations of solar energy, we formulate a joint optimization problem for resource scheduling and computation offloading strategy to maximize the minimum weighted computation capacity across the time slots under solar energy constraints. To solve the mixed-integer nonlinear program (MINLP), we propose a multiply-iterated decoupling optimization algorithm by jointly optimizing a computation offloading strategy, energy provision of the solar-powered hybrid access point (HAP), and local CPU frequency as well as time scheduling. Simulation results show that the proposed algorithm can efficiently use solar energy to balance network calculations, improve network energy efficiency, and realize unmanned and sustainable agricultural WSN.

1. Introduction

1.1. Background. The flourishing development of IoT has brought new opportunities and challenges to modern agriculture, especially in precision agriculture and smart irrigation applications [1]. Wireless sensor networks (WSNs), as the sensing layer of IoT, can achieve comprehensive sensing and timely response to environmental status with their highly scalable and ubiquitous architecture, which can effectively assist the smart agriculture systems in maximizing yields and minimizing wastage [2]. However, the characteristics of sensors, including the low-power processor and energy-constrained battery, make it difficult to process complex tasks sustainably. It seriously affects the upgrade application of agriculture WSNs, especially in precision agriculture, where WSN not only needs to consume more energy to continuously monitor feature parameters, such as soil moisture, soil nutrients, crop growth, and pests, but also requires to pay more computation to process data and transmit commands with low latency [3]. Therefore, how to

tackle the energy constraints and computation limitations of sensors simultaneously is a critical problem in the development of sustainable agriculture WSN.

Conventional sensors are powered by limited capacity batteries. Networks that prolong the network lifetime by regularly replacing batteries can result in significant maintenance costs and serious environmental pollution. To satisfy the energy requirements for comprehensive monitoring, existing research focuses on either open-source or cost-saving approaches [4]. Specific energy-efficient approaches for battery-constrained WSNs cover clustering-based schemes [5], node deployment strategies [6], node scheduling algorithms [7], energy-efficient routing schemes [8], and energy-efficient joint designs [9], all of which prolong the network lifetime at the expense of network performance and fail to inherently provide a sustainable energy supply to the network. Moreover, the deployment of batteries in soil and water quality monitoring applications is not allowed since it is necessary to prevent environmental damage caused by battery leakage or damage. Based on RF, wireless

power transfer (WPT) technology can provide a continuous and controlled energy supply for sensors and effectively solve the energy constraint problem [10]. However, the existing wireless energy mainly derives from high-capacity batteries or the nearby power grid, which is problematic in agricultural applications due to the difficulty of introduction and high maintenance costs. Renewable environmental energy sources, such as solar energy, wind energy, thermal energy, and vibration, can provide a continuous, convenient, and clean energy supply for remote agriculture by virtue of their wide distribution and accessibility [11]. Solar energy, as the most popular ambient microenergy source, enjoys a higher power density (15 mW/cm^2), greater geographic flexibility, easier installation, and periodicity available, making it particularly appropriate for the perpetual energy supply of nodes in agricultural WSNs [12].

To make up for the limited computing capacity of sensors, the majority of agricultural systems use cloud computing to process, analyse, and store large amounts of heterogeneous data through multiple network layers, which imposes a significant burden on the information and communication infrastructure, causing enormous energy costs and more significant information return latency [13]. It obviously fails to match the low latency requirements of precision agriculture with exponentially increasing monitoring tasks. The multiaccess edge computing (MEC) allows the sensors to offload intensive computations to nearby servers located at the edge of the radio access network, thus facilitating better performance than the cloud computing paradigm in terms of latency, effective bandwidth, energy consumption, and load balancing, which is considered to be a promising solution to enhance the computing capability of WSN [14]. Existing MEC research has mainly focused on networks that rely on limited batteries [15] or stable and controllable RF signals [16] for energy supply, achieving maximum computation rate [17], minimum energy consumption [18], and maximum energy efficiency [19] by jointly optimizing offloading decisions and resource allocation, which makes it difficult to apply in remote agricultural IoT with the requirement of unmanned, low-latency, comprehensive state monitoring. It is significant to build a solar-powered MEC green agricultural IoT to satisfy the requirements of smart agriculture for comprehensive sensing and real-time processing. However, due to the randomness and volatility of solar energy, research on energy-efficient offloading decisions and resource scheduling of MEC based on solar energy supply faces significant challenges.

1.2. Related Works. In MEC systems, it is essential to design efficient offloading decision and resource scheduling for improving network performance in terms of energy efficiency, spectral efficiency, latency, and lifetime [14]. Compared to single-user offloading [20], the multiuser offloading decision is more suitable for the smart agricultural IoT as it requires a combination of communication resources and collaboration between multiple users [18].

From the perspective of task features, there are two basic computation offloading decisions for MEC, such as binary

and partial offloading [14]. Partial offloading allows a task to be partitioned into two parts, with one executed locally and the other offloaded for edge computing, which is mostly used for high-volume and complex tasks such as environmental monitoring and healthcare applications. Binary offloading requires a task to be executed as a whole packet either locally or offloaded to a remote MEC server, which is more suitable for applications with high relevance, such as anomaly detection. Research on binary and partial offloading is also abundant [16, 21, 22]. In [21], Zhou et al. investigated the optimal resource allocation for maximizing the network computation efficiency under the partial and binary offloading mode. Mao et al. [22] designed the online partial offloading and resource allocation algorithm to trade-off between energy efficiency and delay. In [16], Bi et al. proposed two efficient solution algorithms (such as the coordinate descent method and the ADMM-based method) to tackle the binary combinatorial computing mode selection. Partial offloading is a relaxed form of binary offloading from a mathematical point of view, which is simpler to compute since it eliminates the discrete random variables. However, in many practical scenarios, especially those suffering from timing characteristics, computing tasks may not be arbitrarily divisible; thus, the binary offloading will be considered in this work.

From the energy supply perspective, traditional MEC are powered by limited capacity batteries. Researchers have focused on proposing excellent offload decisions and resource allocation strategies that can reduce node energy consumption and prolong network lifetime, but this is limited. In [15], Xu et al. investigated the energy minimization task offloading and resource allocation for MEC in NOMA-HetNets under the constraints of QoS requirements. Literature [23] proposes a cooperative offloading technique based on the Lagrangian Suboptimal Convergent Computation Offloading Algorithm (LSCCOA) to minimize weighted sum of transmit power consumption. Considering the limitations and complexity of the integrated batteries in nodes, more and more scholars are focusing on introducing WPT to provide continuous and controlled energy supply for sensors sensing, forwarding, etc., thus improving network energy efficiency and prolonging network lifetime [24]. In a wireless powered multiuser MEC system, Literature [16] jointly optimized the binary offloading mode and the transmission time allocation to maximize the sum computation rate of all the devices. Literature [18] proposed a power minimized resource scheduling strategy by jointly optimizing downlink energy beamforming, uplink computation offloading, and local task execution at users. To enhance computation efficiency and prolong the network lifetime, Literature [17] jointly optimizes the allocation of the communication, computing, and energy resources with the aid of WPT in Full-Duplex (FD) mode. In [25], Li et al. proposed a joint user association and dynamic offloading scheme to enhance the computing capability and achieve sustainable device operation. Different from the works in [18] where channel state information (CSI) is assumed to be fixed, the authors in [26] consider a practical scenario with casual task state

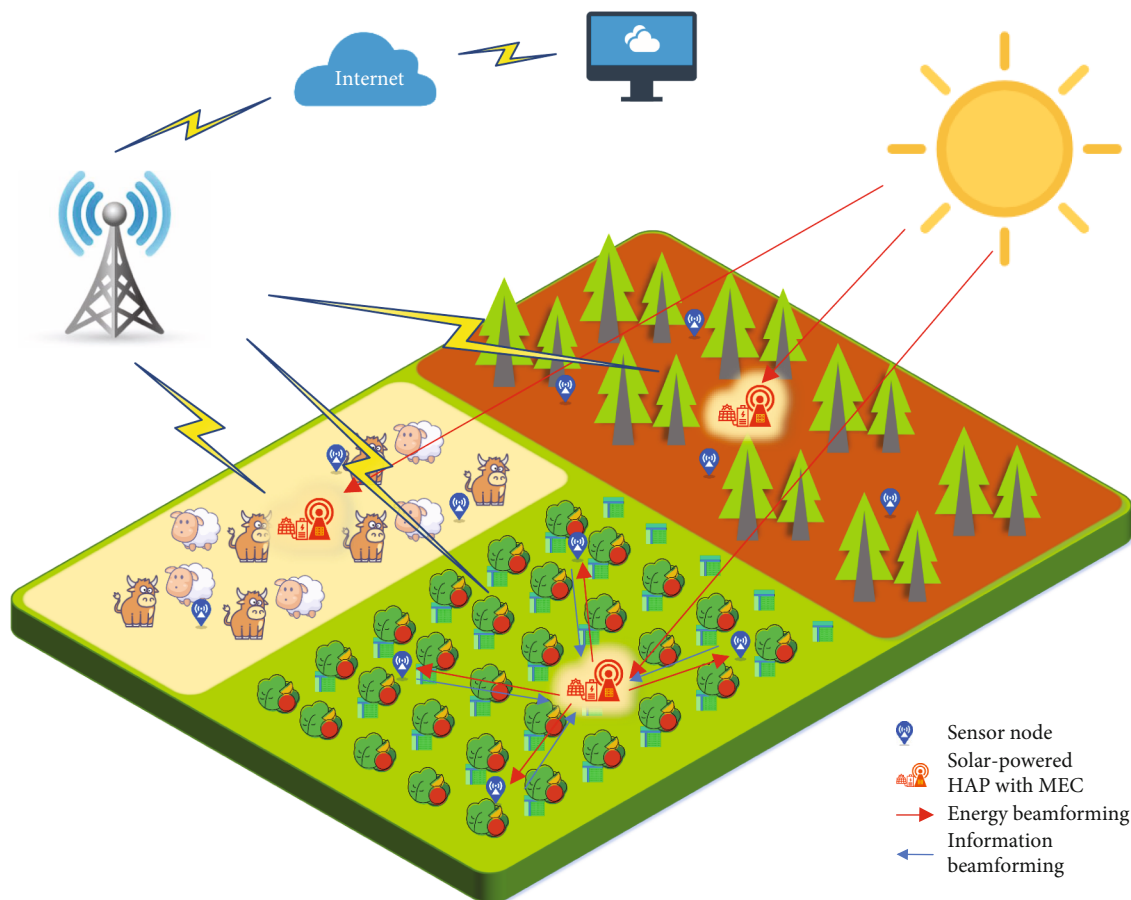


FIGURE 1: The overview of solar-powered MEC-enabled agriculture WSN. The HAP (embedded MEC) consistently harvests and stores solar energy for charging a set of battery-free sensor nodes and processing tasks offloaded by sensor nodes that makes offloading decisions.

information (TSI) and CSI and propose a real-time resource allocation strategy to minimize the total system energy consumption. However, these studies are based on the continuous and stable energy transmitting power provided by high-capacity batteries or the nearby power grid, which is not suitable for the agricultural environment far from cities and complex terrain, so we need to find a more suitable energy source for agricultural WSNs. Given the stochastic and volatile nature of environmental energy, few existing studies on renewable energy-powered MEC are available. In [27], the multiuser energy consumption and computing resource minimization problem for hybrid renewable energy and grid supply is studied under energy harvesting and QoS constraints. Literature [28] investigated the decentralized partially observable offloading problem in the multiuser EH-enabled system, in which multiple IoT devices cooperate to maximize the network performance while meeting QoE requirements. Although MEC-based optimal offloading decisions and resource scheduling problems have been studied under conventional battery supply, WPT supply, and renewable energy access supply, all of these have failed to fully exploit both the periodic cleanliness of solar energy and the continuous controllability advantages of WPT through technological complementarity in MEC networks to meet the operational requirements of precision agricultural IoT.

1.3. Contributions. In this paper, we consider a solar-powered multinode MEC-enabled WSN system, as shown in Figure 1, where a solar-powered hybrid access point (HAP) integrated with a MEC server jointly recharges all sensors and computes the offloading task. Each sensor uses up the charged energy to make the optimal offloading decision given the system objective. In particular, we are interested in maximizing the minimum weighted system computation capacity as indicated by throughput across the time slots subject to the periodicity and day-night variability of solar energy. To our knowledge, this is the first paper that studies the optimal design in a multinode solar-WPT-powered MEC network using binary offloading strategy for agricultural WSN. Our contributions are detailed below.

- (1) We constructed a solar-powered multinode MEC-enabled agricultural WSN framework to achieve balanced system computation capacity, which is necessary for building efficient, clean, unmanned sustainable smart agriculture
- (2) We formulate a maximize the minimum weighted system throughput problem for Solar-Powered MEC-enabled WSN to determine the computation offloading strategy of each node in each time slot

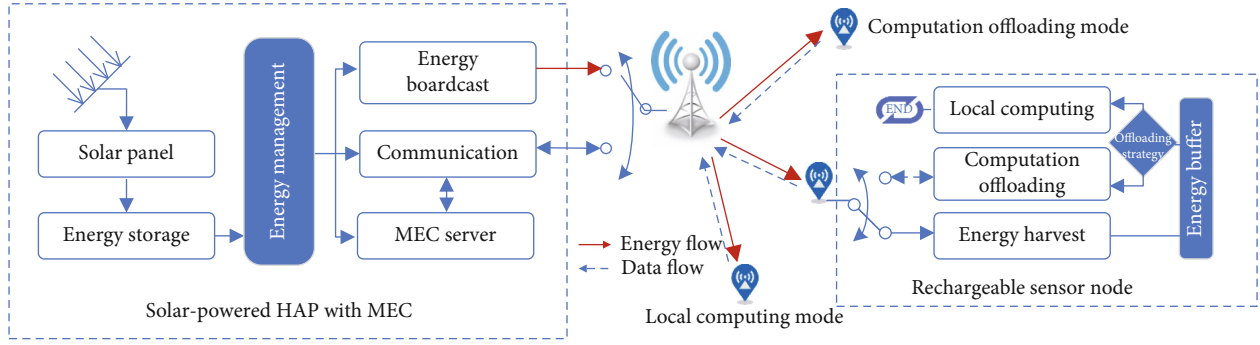


FIGURE 2: A block diagram of the energy broadcasting and task computing. The HAP harvest-store-broadcast energy for member nodes to perform local computing or computation offloading.

and the time trade-off between energy broadcast and transmission, which jointly optimize the energy provision of solar-powered HAP, multinode computation offloading strategy, and local CPU frequency as well as computing offloading time schedule

- (3) To solve the mixed-integer nonlinear program (MINLP), we propose a multiply-iterated decoupling optimization method. First, we initialize a computation offloading strategy to obtain a semiclosed-form solution for the optimal local CPU frequency and the optimal edge computation communication time allocation subject to a specific HAP energy provision. We then propose a low-complexity algorithm to compute the optimal solution for HAP energy provision and time allocation by using the dual and successive convex approximation (SCA) methods. Subsequently, we propose a coordinate descent (CD) method to update the previous computation offloading strategy and finally get the optimal solution of the model by several iterations of the loop

The rest of the paper is organized as follows. Section 2 introduces the system model and assumption. Section 3 formulates the maximize the minimum weighted system throughput problem for solar-powered MEC-enabled battery-free WSN. Section 4 presents the details of the multiply-iterated decoupling optimization method for the nonconvex problem mentioned above. Section 5 evaluates the simulation results, and Section 6 concludes the paper.

2. System Model and Assumption

We consider a solar-powered multinode MEC-enabled WSN system with perfect CSI as shown in Figure 2, where a solar-powered HAP integrated with a MEC server consistently harvests and stores solar energy with battery capacity B_j^{\max} , then employs WPT to charge a set $N \triangleq \{1, 2, \dots, n\}$ of battery-free sensor nodes and compute offloading tasks. It is assumed that all channels are reciprocity and follow a quasistatic block fading model [29], in which the channel gain of a channel remains constant over a slot but varies between slots. Each node makes full use of recharged energy to perform environmental monitoring and task computing, which

follows a binary offloading strategy, i.e., operating in either local computing or computation offloading at each slot. Local computing is where the sensor node performs the generated computing tasks by the on-chip microprocessor, which has low computing capability due to energy and size constraints. Computation offloading means that the sensor node offloads the entire task to the MEC server embedded in HAP with much more processing power, which provides cloud-like computation capability for WSN to realize the requirements of computationally intensive applications in precision agriculture. To achieve real-time state monitoring for smart agriculture, we need to guarantee that the network sensor nodes are in a full-time sensing state, which imposes greater requirements for better management of the HAP's energy provision and offloading decisions of the network. This is illustrated in Figure 2.

2.1. Time Scheduling Model. Considering the periodicity and day-night variability of solar harvesting, we propose a time division computing offloading scheduling (TDCOS) to improve the system computation capacity and solar energy utilisation of each time slot. To avoid interference, we assume that HAP and its member nodes run on different frequencies. Each node adopts a half-duplex time mode to communication with a HAP and uses time division multiple access (TDMA) to achieve computation offloading to avoid cochannel interference. Consider a finite time horizon T with a whole day as shown in Figure 3, which is divided into K slots with duration $\tau = T/K$. Let $T \triangleq \{1, \dots, k, \dots, K\}$ denote the set of the time slots k . Each time slot k is divided into the following two intervals: (1) energy broadcast time, where HAP charges all devices for a time τ_{bro} simultaneously; (2) computing offloading time, where each node chooses to operate at either local computing or computing offloading based on the system throughput within a time slot; then, each offload node takes turns uploading tasks in their allocated time slots τ_i . In this case, task offloading can occupy the rest of the time slot after energy broadcasting of HAP, as shown in the following equation.

$$\sum_{i=1}^N \tau_i + \tau_{bro} \leq \tau. \quad (1)$$

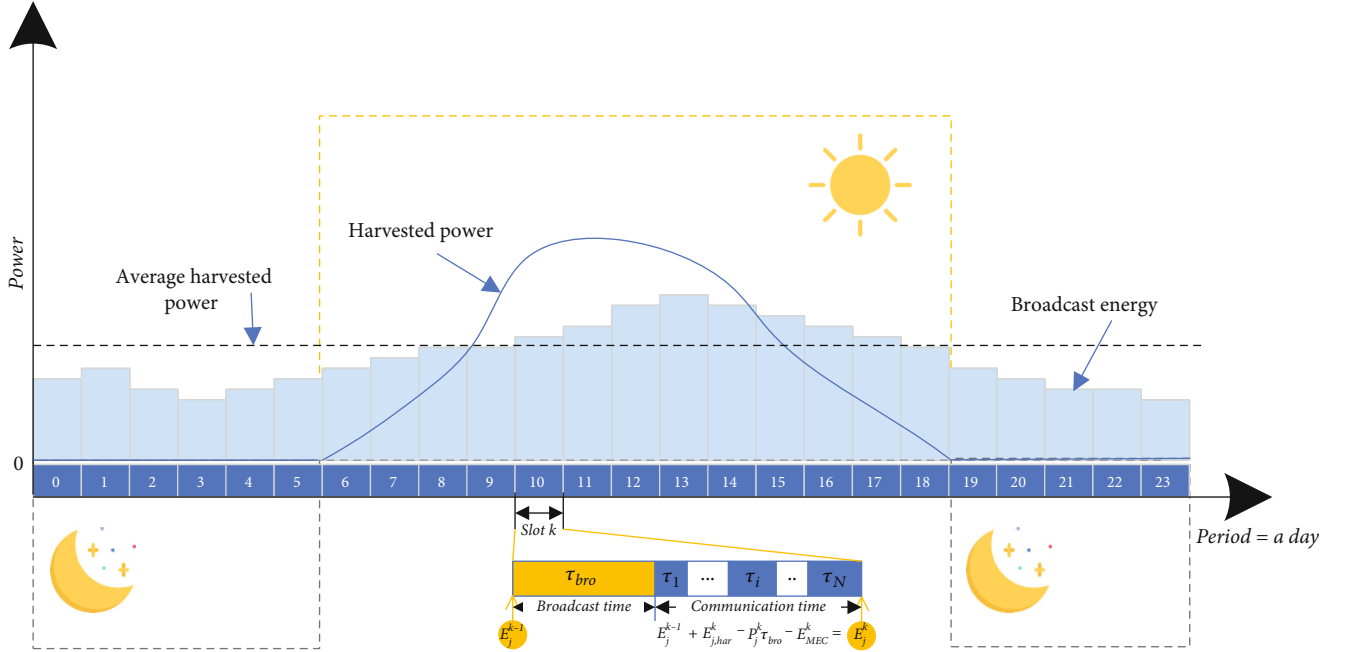


FIGURE 3: Time slot allocation during a solar-harvesting period. A period (a day) is divided equally with time interval 1 h, and each slot is composed of the broadcast time and task offloading time. The harvested energy of HAP is scheduled to reduce the impact from day-night variability of solar energy.

2.2. Energy Harvest Model. We consider a solar-powered HAP equipped with a solar panel of size W_p and has a rechargeable battery with capacity B_j^{\max} , as shown in Figure 2. We assume that solar energy arriving at HAP follows the m -state Markov chain model [30], where each state $m \in \{1, \dots, M\}$ represents an EH profile expressed as a probability distribution with a given mean μ_m and variance ρ_m . Specifically, letting ψ_j^k represent the solar intensity on the HAP at time slot k , the solar-harvested energy of HAP at slot k can be expression by the following equation.

$$E_{j,har}^k = \eta_j \psi_j^k W_p \tau, \quad \forall k \in T, \quad (2)$$

where $\eta_j \in (0, 1)$ denotes the solar energy conversion rate of HAP. Within each time slot k of period T , the HAP converts the available energy E_j^{k-1} at the start of time slot k to charge all member nodes with wireless power transfer (WPT) and process tasks offloaded by some nodes on the MEC server, followed by storing the residual solar energy in the battery for the next slot. Let P_j^k denote the transmitting power of HAP, which characterizes the antenna performance of HAP and is satisfied with $0 \leq P_j^k \leq P_j^{\max}$. Here, E_{MEC}^k denotes the energy consumption of HAP processing the tasks offloaded by nodes. Consequently, the available energy of HAP at the end of slot k can be defined as the following equation.

$$E_j^k = E_j^{k-1} + E_{j,har}^k - P_j^k \tau_{bro} - E_{MEC}^k, \quad \forall k \in T, \quad (3)$$

where the broadcast energy $P_j^k \tau_{bro}$ of HAP is constrained by $P_j^k \tau_{bro} \leq E_j^{k-1}$ as a result of the causal energy limitations. Also, to ensure proper monitoring at night within the constraints of day-night variability of solar-harvesting, we need to further constrain the lower limit of available energy for the HAP at any time slot as shown in the following equation.

$$P_j^k \tau_{bro} + E_{MEC}^k \leq E_j^{k-1}, \quad \forall k \in T. \quad (4)$$

For avoiding energy overflow, E_j^k is limited by the capacity of the HAP battery B_j^{\max} , as shown in the following equation.

$$0 \leq E_j^k \leq B_j^{\max}, \quad \forall k \in T. \quad (5)$$

During the energy broadcasting phase, each member node harvests the wireless energy for task monitoring and offloading during the time slot. Let h_{ij}^k denote the channel gain from the HAP to node i based on the Rayleigh fading model [31]. Ignoring interference from noise, we can get the charged energy of each node at slot k , which can be expressed as the following equation.

$$E_i^k = P_j^k h_{ij}^k \eta_i \tau_{bro}, \quad \forall k \in T, i \in N, \quad (6)$$

where $\eta_i \in (0, 1)$ denotes the energy-harvesting efficiency, which can be considered as a fixed value since the transmitting power is usually small in WSNs. In order to establish a sustainable agricultural WSN, we need the network to

achieve energy neutral operation (ENO), which means the energy harvesting and consumption of HAP is balanced in one period. The mathematical expression is as in the following equation.

$$\left| E_j^N - E_j^0 \right| \leq \varepsilon, \quad (7)$$

where ε denotes the acceptable error range.

2.3. Computing Model. For each node, task computing can be performed locally or alternatively by offloading to the MEC server embedded in HAP. Here, we introduce the set of binary offloading variable $S^k \triangleq \{s_1^k, \dots, s_i^k, \dots, s_N^k\}$, where $s_i = 0$ denotes that the node i decides to offload its computation task to the MEC server. Otherwise, $s_i = 1$. Based on the given computing offloading strategy, the detailed operation of each node is illustrated as follows.

2.3.1. Local Computing. When node i makes the decision of local computing, it would perform the task computation independently of the others with the charged energy. Let δ_i indicate the number of CPU cycles required to complete a task with unit data size and f_i^{loc} denote the CPU frequency, which characterizes the computational power of the node and satisfies with $0 \leq f_{i,loc}^k \leq f_{i,max}^{loc}$. It is assumed that $\Phi_{i,loc}^k$ is the total throughput on local computing at the slot k ; the corresponding energy consumption and computation latency can be calculated as shown in the Equations (8) and (9), respectively.

$$E_{i,loc}^k = \kappa_{mob,i} \left(f_{i,loc}^k \right)^3 \delta_i \Phi_{i,loc}^k, \quad \forall k \in T, i \in N, \quad (8)$$

$$t_{i,loc}^k = \frac{\Phi_{i,loc}^k \delta_i}{f_{i,loc}^k}, \quad \forall k \in T, i \in N, \quad (9)$$

where $\kappa_{mob,i}$ denotes energy consumption parameters, which depend on the chip architecture [32]. Notice that local computing can last for the entire time slot τ , since the energy charging and local computing can be done simultaneously based on the circuit as shown in Figure 2. Accordingly, the total local computing throughput $\Phi_{i,loc}^k$ are constrained by computation latency and energy consumption in the Equations (10) and (11), respectively.

$$0 \leq t_{i,loc}^k \leq \tau, \quad \forall k \in T, i \in N, \quad (10)$$

$$0 \leq E_{i,loc}^k \leq E_i^k, \quad \forall k \in T, i \in N. \quad (11)$$

2.3.2. Computing Offloading. Apart from local computing, each node needs to take turns using up the charged energy for offloading the entire task to the MEC server embedded in HAP, and downloading the computation results after MEC server has completed the task calculation when offloading decision $s_i = 0$. According to the Shannon theory [33], the available offload computing rate of node i , denoted by $r_{i,off}^k$, can be calculated by the following equation.

$$r_{i,off}^k = W \log_2 \left(1 + \frac{P_{i,off}^k H_{ij}^k}{N_0} \right), \quad \forall k \in T, i \in N, \quad (12)$$

where W denotes the communication bandwidth and N_0 denotes the receiver noise power. Let $P_{i,off}^k = E_i^k / \tau_i^k$ denote the offloading transmitting power of node i at time slot k . The total throughput $\Phi_{i,off}^k$ of each offload node in its allocated time τ_i can be calculated by the following equation.

$$\Phi_{i,off}^k = \tau_i^k r_{i,off}^k = W \tau_i^k \log_2 \left(1 + \frac{P_{i,off}^k H_{ij}^k}{N_0} \right), \quad \forall k \in T, i \in N. \quad (13)$$

After receiving the raw data of all the member nodes, the HAP computes and sends back the output result back to the corresponding node. Given that the calculation results at MEC are smaller and the downlink transmission rate is higher, the downlink energy consumption and latency can be ignored [34]. Thus, the energy consumption of computation offloading at the MEC server, denoted by E_{MEC}^k , can be expressed as the following equation.

$$E_{MEC}^k = \kappa_{mob,j} (f_{MEC})^3 \delta_j \Phi_{ij,off}^k, \quad \forall k \in T, i \in N, \quad (14)$$

where $\kappa_{mob,j}$ is the capacitance coefficient specified by the MEC server's CPU architecture and f_{MEC} is the CPU frequency of the HAP during each slot, which is fixed during the calculation cycle of the task by the DVFS technique [35].

3. Problem Formulation

In this subsection, we formulate a maximize the minimum weighted system computation problem for the solar-powered MEC-enabled battery-free WSN to determine the computation offloading strategy of each node in each time slot and the time trade-off between energy broadcast and transmission, which jointly optimize the multinode computation offloading strategy $\{S^k\}$, the amount of energy provision by solar-powered HAP $\{P_j^k, \tau_{bro}\}$, and local CPU frequency $\{f_i^{loc}\}$, as well as computing offloading time scheduling $\{\tau\}$. Here, we qualify the volume of network computation in terms of throughput. Accordingly, the weighted total throughput maximization problem is formulated as the following equation.

$$\mathbf{P0} \max_{\{P, \tau, s, f\}} \min_k \sum_{i=1}^N w_i \left[s_i \Phi_{i,loc}^k + (1 - s_i) \Phi_{i,off}^k \right], \quad (15)$$

Subject to

$$C1 : 0 \leq P_j^k \leq P_j^{\max}, \quad \forall k \in T, \quad (16)$$

$$C2 : 0 \leq E_j^k \leq B_j^{\max}, \quad \forall k \in T, \quad (17)$$

$$C3 : P_j^k \tau_{bro}^k + E_{MEC}^k \leq E_j^{k-1}, \quad \forall k \in T, \quad (18)$$

$$C4 : \left| E_j^N - E_j^0 \right| \leq \varepsilon, \quad (19)$$

$$C5 : \sum_{i=1}^N \tau_i^k + \tau_{bro}^k \leq \tau, \quad \forall i \in N, k \in T, \quad (20)$$

$$C6 : s_i^k \in \{0, 1\}, \quad \forall i \in N, k \in T, \quad (21)$$

$$C7 : f_{\min} \leq f_{i,loc}^k \leq f_{\max}, \quad \forall i \in N, k \in T, \quad (22)$$

$$C8 : 0 \leq t_{i,loc}^k \leq \tau, \quad \forall i \in N, \quad (23)$$

where $w_i \in [0, 1]$ is a weight factor that accounts for the priorities of different nodes and is specified by the application layer. C1 is the transmission power constraint which ensures that the transmission power of the HAP is nonnegative and bounded by the maximum transmission power P_j^{\max} depending on the system architecture. Battery capacity constraint C2 ensures that the battery energy level of HAP is upper bounded by the battery capacity B_j^{\max} . Energy consumption constraint C3 specifies that the energy consumption for broadcasting and computation offloading does not exceed the available energy of HAP at the start of the time slot, which can effectively guarantee the energy requirements for night-time network monitoring. C4 is an energy neutral operation (ENO) constraint wherein the energy harvesting and consumption of HAP is balanced in one period. Time slot constraint C5 specifies that all nodes need to accomplish energy charging and computational offloading within one time slot. C6 indicates that the variables of offloading decision variables are 0-1 binary decision variables. The frequency constraint C7 denotes that the local CPU frequency should be limited within a range of $[f_{\min}, f_{\max}]$. C8 indicates that the local computing time of the node cannot exceed the duration of a time slot. Due to the combinatorial nature of the multinode computing offloading strategy and its strong coupling with resource scheduling, problem (P0) is hard to tackle.

4. Joint Optimization Method

In this section, we introduce a joint optimization between resource scheduling and computation offloading strategy to maximize the minimum weighted system throughput for solar-powered MEC-enabled battery-free WSN. In response to this mixed-integer nonlinear program (MINLP), we propose a multiply-iterated decoupling optimization method, where we first initialize a computation offloading strategy $S^k(0)$ to obtain a semiclosed-form solution for the optimal local CPU frequency $f_{i,loc}^{k*}$ and the optimal edge computation communication time allocation subject to a certain HAP energy provision. We then propose a low-complexity algorithm to compute the optimal solution Γ^{k*} for HAP energy provision and time allocation. Subsequently, we propose a coordinate descent (CD) method to update the previous computation offloading strategy and finally get the optimal solution of the model. Next, we propose a corresponding solution for each iteration.

4.1. Optimal Local Computing. In this subsection, we investigate the optimization of the local CPU computation frequency given the computation offload strategy. From problem (P0), it can be seen that independently optimizing the local CPU frequency f_i^{loc} does not affect the performance of other nodes when the $\Gamma^k = P_j^k \tau_{bro}^k$ is fixed. Combining Equations (8)–(11) and constraint C7, we can conclude $f_{i,loc}^k$ by the following equation.

$$f_{i,loc}^k \in \left[\left(\frac{E_i^k}{\kappa_{mob,i} \tau} \right)^{1/4}, f_{\max} \right], \quad \forall i \in N, k \in T. \quad (24)$$

Note that the function $\Phi_i^{loc}(f_i^{loc})$ is monotonically decreasing on the domain. Hence, we can obtain the optimal semiclosed solution for the local CPU frequency for each node, which is shown in the following equation.

$$f_{i,loc}^{k*} = \min \left[\left(\frac{E_i^k}{\kappa_{mob,i} \tau} \right)^{1/4}, f_{\min} \right], \quad \forall i \in N, k \in T. \quad (25)$$

To ensure that the network works properly, $E_{i,loc}^k \geq \kappa_{mob,i} f_{i,loc}^{k*} \tau$ needs to be satisfied. By substituting $f_{i,loc}^{k*}$ into Equation (8), we can obtain the maximum local computing data volume by the following equation.

$$\Phi_{i,loc}^{k*} = \frac{E_i^k}{\kappa_{mob,i} (f_{i,loc}^k)^3 \delta_i} = \eta_1 \left(\Gamma^k h_{ij}^k \right)^{1/4}, \quad \forall i \in N, k \in T, \quad (26)$$

where $\eta_1 \triangleq (\eta \tau^3 / \kappa_{mob,i} \delta_i^4)^{1/4}$ is a fixed parameter.

4.2. Joint Iterative Optimization of Broadcast Power and Transmission Time. In this subsection, we continue to study the optimal computation offloading resource scheduling with the assumption that S^k is known and Γ^k is fixed. Here, we first transform problem (P0) into problem (P1) by removing the variables $f_{i,loc}^{k*}$ and making formal transformations to the objective function, which can be formulated as the following equation.

$$\begin{aligned} \mathbf{P1} \quad & \max_{\{P, \tau, s\}} \min_k \sum_{i=1}^N w_i s_i \eta_1 \left(\Gamma^k h_{ij}^k \right)^{1/4} + \sum_{i=1}^N w_i (1 - s_i^k) W_p \tau_i^k \log_2 \left(1 + \frac{\Gamma^k (h_{ij}^k)^2}{\tau_i^k N_0} \right), \\ & \text{s.t. } C1 \sim C6. \end{aligned} \quad (27)$$

To solve the problem (P1) for nodes with offloading decision, we need to adopt the iteration optimization algorithm for offloading time allocation and energy broadcast scheduling. Let $\tau^k(l)$ and $\Gamma^k(l)$ denote the set of variables related to offloading time and energy charging allocation for nodes in the l -th iteration, respectively. Specifically, each iteration can be divided into two steps. Firstly, the optimal

Input: given a computation offloading strategy $S^k = \{s_1^k, s_2^k, \dots, s_n^k\}$.
Output: the optimal $\{P_j^{k*}, \tau_{bro}^{k*}, \tau^{k*}\}$ to Problem (P2) given S^{k*} .

- 1 Initialize $\Gamma^k(0)$ and $l = 1, \alpha(0), \beta(0)$
- 2 Repeat
- 3 Calculate local CPU frequency according to Equation (25);
- 4 Repeat
- 5 Calculate edge offloading time allocation according to Equation (32);
- 6 Update $\lambda_1, \lambda_3, \lambda_4$ according to Equations (33)–(35);
- 7 Until $\lambda_1, \lambda_3, \lambda_4$ converges;
- 8 Let $\tau^{k*}(l) \leftarrow \max_{i=1, \dots, n} \Phi_i^k(l)$;
- 9 Repeat
- 10 Repeat
- 11 Calculate energy provision according to Equation (43);
- 12 Update μ_3, μ_4, μ_5 , according to Equations (44)–(46);
- 13 Until μ_3, μ_4, μ_5 converges;
- 14 Repeat
- 15 Calculate $\alpha_i^k(l_p), \beta_i^k(l_p)$ according to Equations (39) and (40);
- 16 Update number of iterations $l = l + 1$;
- 17 Until Γ^k converges;
- 18 Update outer loop iterations $l = l + 1$ and tolerance ξ^l
- 19 Until $\xi^l \leq \sigma$
- 20 Let $\Gamma^{k*}(l) \leftarrow \max_{i=1, \dots, n} \Phi_i^k(l)$
- 21 Calculate the optimal energy broadcast power P_j^{k*} and corresponding time τ_{bro}^{k*} according to Equations (48) and (49)
- 22 Return the optimal solution by f^*, \mathbf{P}, τ

ALGORITHM 1: Iterative algorithm with combination of SCA and Lagrangian dual theory for optimal resource scheduling.

offloading time allocation of iteration l is derived from the given value of $\Gamma^k(l-1)$. Then, with the fixed value of $\tau^k(l)$, the optimal energy broadcast scheduling $\Gamma^k(l)$ is obtained. The iterative algorithm is repeated until it converges to stopping criterion $\xi^l \leq \sigma$; then, we can obtain a closed-form solution for optimal offloading time allocation and energy broadcast scheduling for problem (P1). The criterion is specified by the total throughput variation value ξ^l , which can be expressed by the following equation.

$$\xi^l = \left\| \begin{array}{c} \sum_{i=1}^N w_i \left[s_i^k \Phi_{i,loc}^k(l) + (1 - s_i^k) \Phi_{i,off}^k(l) \right] \\ - \sum_{i=1}^N w_i \left[s_i^k \Phi_{i,loc}^k(l-1) + (1 - s_i^k) \Phi_{i,off}^k(l-1) \right] \end{array} \right\|, \quad \forall i \in N, k \in T. \quad (28)$$

As shown in Algorithm 1, we need to solve problem (P2) and problem (P3) to obtain an optimal solution for offloading time allocation and energy broadcast scheduling, respectively. The problem P2 is shown as the following equation.

$$\text{P2} \max_{\tau^k} \sum_{i \in \{s_i^k=1\}} w_i W_p \tau_i^k \log_2 \left(1 + \frac{\Gamma^k(l-1) (h_{ij}^k)^2}{\tau_i^k N_0} \right), \text{s.t.} \quad C2 \sim C6. \quad (29)$$

Obviously, problem (P2) is a convex problem, so we can introduce Lagrangian multipliers $\lambda = \{\lambda_i | \lambda_i \geq 0\}$ with constraints C1~C5 to form a partial Lagrangian and achieve the optimal objective value $\{\tau^*\}$ by the strong duality, as is shown in the following equation.

$$\begin{aligned} L(\tau, \lambda) = & \sum_{i \in \{s_i=1\}} w_i W_p \tau_i^k \log_2 \left(1 + \frac{\Gamma^k(l-1) (h_{ij}^k)^2}{\tau_i^k N_0} \right) \\ & - \lambda_1 \left(E_j^k - B_j^{\max} \right) - \lambda_2 \left(\Gamma^k - E_j^{k-1} \right) \\ & - \lambda_3 \left(\sqrt{(E_j^N - E_j^0)^2} - \varepsilon \right) \\ & - \lambda_4 \left(\sum_{i=1}^N \tau_i^k + \tau_{bro}^k - \tau \right). \end{aligned} \quad (30)$$

The corresponding dual function can be calculated by the following equation.

$$d(\lambda) = \max_{\tau} \{L(\tau, \lambda), \tau_i \geq 0\}. \quad (31)$$

According to KKT condition $\partial L / \partial \tau_i^k(l) = 0$, we can obtain the optimal $\{\tau^*\}$ which satisfies the following equation.

$$\frac{\tau_i^{k*}(l)}{\Gamma^k(l-1)} = \frac{\eta_2 \left(h_{ij}^k\right)^2}{(-W(x))^{-1} - 1}, \quad \forall k \in T, i \in \{i | s_i = 0\}, \quad (32)$$

where $W(x)$ denotes the Lambert-W function and is satisfied with $x = -(1/\exp(1 + \lambda_4/(w_i W_p + (\lambda_1 + \lambda_3)\eta_3)))$.

Then, the solution of the dual problem $\min_{\lambda} d(\lambda)$ can be determined by a subgradient method as shown in Equations (33) and (34).

$$\lambda_1^{l+1} = \left[\lambda_1^l + \sigma_{\lambda_1} \left(E_j^k - B_j^{\max} \right) \right]^+, \quad (33)$$

$$\lambda_3^{l+1} = \left[\lambda_3^l + \sigma_{\lambda_3} \left(\sqrt{(E_j^N - E_j^0)^2} - \varepsilon \right) \right]^+, \quad (34)$$

$$\lambda_4^{l+1} = \left[\lambda_4^l + \sigma_{\lambda_4} \left(\sum_{i=1}^N \tau_i + \tau_{bro} - \tau \right) \right]^+. \quad (35)$$

Here, $[x]^+ = \max(x, 0)$. With the fixed value τ_i^{k*} found above, Problem (P1) can be transformed into Problem (P3), which can be shown as Equation (35).

$$\begin{aligned} \text{P3} \max_{\{P_{r_0}\}} & \sum_{i=1}^N w_i s_i \eta_1 \left(\Gamma^k(l) h_{ij}^k \right)^{1/4} + \sum_{i=1}^N w_i (1 - s_i) W \tau_i^{k*}(l) \log_2 \left(1 + \frac{\Gamma^k(l) \left(h_{ij}^k \right)^2}{\tau_i^{k*}(l) N_0} \right), \\ & \text{s.t.} \quad C2 \sim C6. \end{aligned} \quad (36)$$

Obviously, Problem (P3) is a nonconvex problem because the second term of the objective function is a convex function. In order to solve this problem, we use the SCA approach [36] to propose the energy broadcast scheduling algorithm with logarithmic approximation. Instead of directly dealing with the highly nonconcave rate function, we apply the logarithmic approximation method to convert the throughput function into $\widehat{\Phi}_i^{off}(P_j^k \tau_{bro}(l_p))$, which is shown in Equation (36).

$$\widehat{\Phi}_{i,off}^k \left(\Gamma^k(l_p) \right) = W_p \tau_i^{k*} \alpha_i^k(l_p) + W_p \tau_i^{k*} \beta_i^k(l_p) \log_2 \left(\frac{\Gamma^k(l_p) \left(h_{ij}^k \right)^2}{\tau_i^{k*} N_0} \right), \quad (37)$$

where $\alpha(l_p)$ and $\beta(l_p)$ are the parameter variable in the SCA process, which can be calculated as seen in Equations (37) and (38), respectively. l_p is the iteration number in the SCA process. $\widehat{\Phi}_{i,off}^k(\Gamma^k(l_p))$ denotes the low bound of the original throughput function $\widehat{\Phi}_{i,off}^k(\Gamma^k(l))$, which is satisfied with $\widehat{\Phi}_{i,off}^k(\Gamma^k(l)) \geq \widehat{\Phi}_{i,off}^k(\Gamma^k(l_p))$.

$$\begin{aligned} \alpha_i^k(l_p) &= W_p \tau_i^{k*}(l) \log_2 \left(1 + \frac{\Gamma^k(l_p - 1) \left(h_{ij}^k \right)^2}{\tau_i^{k*}(l) N_0} \right) \\ &\quad - W_p \tau_i^{k*} \beta(l_p) \log_2 \left(\frac{\Gamma^k(l_p - 1) \left(h_{ij}^k \right)^2}{\tau_i^{k*}(l) N_0} \right), \end{aligned} \quad (38)$$

$$\beta_i^k(l_p) = \frac{\Gamma^k(l_p - 1) \left(h_{ij}^k \right)^2 / \tau_i^{k*}(l) N_0}{1 + \Gamma^k(l_p - 1) \left(h_{ij}^k \right)^2 / \tau_i^{k*}(l) N_0}. \quad (39)$$

Unfortunately, $\widehat{\Phi}_{i,off}^k(\Gamma^k(l_p))$ is still a nonconcave function owing to the existence of logarithmic function $\log_2(\Gamma^k(l_p)(h_{ij}^k)^2 / \tau_i^{k*} N_0)$. Let $\widehat{\Gamma}^k = \log_2(\Gamma^k)$; then, the logarithmic function can be converted into a log-sum-exp function and the problem (P3) can be reformulated into a convex problem (P4), which is shown as Equation (39).

$$\begin{aligned} \text{P4} \max_{\{P_{r_0}\}} & \sum_{i=1}^N w_i s_i \eta_1 \left(e^{\widehat{\Gamma}^k(l_p)} h_{ij}^k \right)^{1/4} + \sum_{i=1}^N w_i (1 - s_i) W_p \tau_i^k(l) \\ & \times \left[\alpha(l_p) + \beta(l_p) \log_2 \left(\frac{e^{\widehat{\Gamma}^k(l_p)} \left(h_{ij}^k \right)^2}{\tau_i^{k*}(l) N_0} \right) \right], \\ & \text{s.t.} \quad C1, C4, C5, C6, \end{aligned}$$

$$C2 : 0 \leq E_j^k(l_p) \leq B_j^{\max},$$

$$C3 : e^{\widehat{\Gamma}^k(l_p)} \leq E_j^{k-1}, \quad \forall k \in T, \quad (40)$$

where $\alpha(l_p) = \{\alpha_i^k(l_p) | i \in N\}$ and $\beta(l_p) = \{\beta_i^k(l_p) | i \in N\}$. Utilizing Lagrangian duality, we propose the SCA-based power allocation algorithm based on the logarithmic approximation method. The Lagrangian duality of the problem is defined as Equation (40).

$$\begin{aligned} L(\widehat{\Gamma}^k(l_p), \mu) &= \sum_{i=1}^N w_i s_i \eta_1 \left(e^{\widehat{\Gamma}^k(l_p)} h_{ij}^k \right)^{1/4} \\ &\quad - \sum_{i=1}^N w_i (1 - s_i) W \tau_i^k(l) \left[\alpha(l_p) + \beta(l_p) \log_2 \left(\frac{e^{\widehat{\Gamma}^k(l_p)} \left(h_{ij}^k \right)^2}{\tau_i^{k*}(l) N_0} \right) \right] \\ &\quad - \mu_1 (P_j^k - P_j^{\max}) - \mu_2 (E_j^k - B_j^{\max}) - \mu_3 \left(e^{\widehat{\Gamma}^k(l_p)} - E_j^{k-1} \right) \\ &\quad - \mu_4 \left(\sqrt{(E_j^N - E_j^0)^2} - \varepsilon \right) - \mu_5 \left(\sum_{i=1}^N \tau_i + \tau_{bro} - \tau \right). \end{aligned} \quad (41)$$

```

Input: initial computation offloading strategy  $S^k(0)$ .
Output: an approximate solution  $\{P_j^{k*}, \tau_{bro}^*, f_{local}^*, \tau^*, S^*\}$ .
1 Initialization:  $l \leftarrow 0$ 
2 Repeat
3  $l \leftarrow l + 1$ 
4 For each node  $i$  do
5 Calculate  $R_i^k(l)$  in Equation (20) using Algorithm 1
6 End
7 Let  $V^{k*}(l) \leftarrow \max_{i=1, \dots, n} R_i^k(l)$  and  $i^{k*}(l) \leftarrow \arg \max_{i=1, \dots, n} R_i^k(l)$ ;
8 Update  $S^k(l) \leftarrow S_{i^{k*}}^k(l-1)$  using Equation (51);
9 Until  $V_i^* \leq 0$  :
10 Let  $S^* = S^k(l)$  and calculate the optimal resource scheduling  $\{P_j^{k*}, \tau_{bro}^*, f_{local}^*, \tau^*\}$ 
11 Return the optimal computation offloading strategy  $\{P_j^{k*}, \tau_{bro}^*, f_{local}^*, \tau^*, S^*\}$  to P1

```

ALGORITHM 2: Coordinate descent algorithm for computation offloading strategy optimization.

The corresponding dual function can be shown as Equation (41).

$$D(\mu) = \underset{\left\{ \widehat{\Gamma}^k(l_p) \right\}}{\text{maximize}} \left\{ L\left(\widehat{\Gamma}^k(l_p), \mu \right) \right\}. \quad (42)$$

According to KKT condition $\partial L / \partial \Gamma^k(l_p) = 0$, we can obtain the optimal $\{\Gamma^k(l_p)^*\}$ which satisfies Equation (42).

$$\Gamma^k(l_p)^* = \sqrt{\frac{(\mu_4 - \mu_3 - \mu_5) \left(h_{ij}^k \right)^4 w_i s_i^k \eta}{[w_i(1-s_i)W_p + (\mu_3 + \mu_5)\eta_3] \tau_i^{k*}(l)\beta(l_p)}}, \quad \forall i \in N, k \in T. \quad (43)$$

The solution of the dual problem $\min_{\mu} D(\mu)$ can be determined by a subgradient method as shown in Equations (42) and (43).

$$\mu_3^{l_p+1} = \left[\mu_3^l + \varepsilon_{\mu_3} \left(E_j^k - B_j^{\max} \right) \right]^+, \quad (44)$$

$$\mu_4^{l_p+1} = \left[\mu_4^l + \varepsilon_{\mu_4} \left(e^{\widehat{\Gamma}^k(l_p)} - E_j^{k-1} \right) \right]^+, \quad (45)$$

$$\mu_5^{l_p+1} = \left[\mu_5^l + \varepsilon_{\mu_5} \left(\sqrt{\left(E_j^N - E_j^0 \right)^2 - \varepsilon} \right) \right]^+. \quad (46)$$

By substituting energy broadcast scheduling $\Gamma^k(l)^* = \Gamma^k(l_p)^*$ and computation offloading time scheduling $\{\tau^{k*}\}$ into C5, we can get the optimal solution of broadcast power P_j^{k*} and time τ_{bro}^* by Equations (44) and (45), respectively.

$$\tau_{bro}^{k*} = \tau - \sum_{i=1}^N \tau_i^{k*}, \quad \forall i \in N, k \in T, \quad (47)$$

$$P_j^{k*} = \frac{\Gamma^{k*}}{\tau_{bro}^{k*}}, \quad \forall k \in T. \quad (48)$$

Algorithm 1 shows the pseudocode of iterative optimization with combination of SCA and Lagrangian dual theory for optimal resource scheduling.

4.3. Coordinate Descent for Computing Mode Optimization.

In this subsection, we introduce the coordinate descent (CD) method [37] to jointly optimize the N binary decision variable $\mathbf{S}^k = \{s_1^k, s_2^k, \dots, s_n^k\}$, which can significantly lower the computational complexity and faster convergence by along the direction of only one variable s_i at a time; see the pseudocode of Algorithm 2 for details. First, we initialize a binary decision variable $\mathbf{S}^k(0)$ in slot k and denote $\mathbf{S}^k(l-1)$ as the computation offloading strategy at $(l-1)$ th iteration, where the optimal system throughput $V^{\mathbf{S}^k(l-1)}$ can be solved by Algorithm 1. In order to find the optimal decision in each iteration, we switch the decision variables for each node i in turn and calculate its system throughput gain $R_i^k(l)$ in the (l) th iteration, which can be calculated by Equation (50).

$$R_i^k(l) = V^k\left(\mathbf{S}_i^k(l-1)\right) - V^k\left(\mathbf{S}^k(l-1)\right), \quad (49)$$

where $\mathbf{S}_i^k(l-1)$ denotes the switched computation offloading strategy with node i and can be calculated by Equation (51).

$$\mathbf{S}_i^k(l-1) = \left[S_1^k(l-1), S_2^k(l-1), \dots, S_i^k(l-1) \oplus 1, \dots, S_n^k(l-1) \right], \quad (50)$$

where \oplus denotes the modulo-2 summation operator. Obviously, $R_i^k(l) > 0$ denotes a better decision for node i in $(l-1)$ th iterations, and we need $\mathbf{S}^k(l) = \mathbf{S}_i^k(l-1)$ and vice versa. Therefore, the optimal decision under this iteration is obtained by comparing the selection of the best

$i_i^* = \arg \max_{i=1, \dots, n} R_i^k(l)$. Meanwhile, R^k increases monotonically with the number of iterations and the optimal value is bounded, so the method can converge to the optimal value.

Next, we consider the complexity of the proposed algorithm. Here, the complexity of the proposed algorithm is calculated based on the two stages. During the computation offloading strategy definition phase, the number of iterative updates for each strategy increases linearly with n , and the complexity is $O(n)$, where n is the number of nodes. In iteration optimization phases of $\Gamma^k(l)$ and τ^k , the number of messages exchanged in each iteration which is equal to the total number of nodes with computational complexity is $O(n^2)$. Therefore, the overall complexity of the proposed algorithm is $O(n^3)$.

5. Results and Discussion

In this section, we evaluate the proposed algorithm for solar-powered agriculture WSN in terms of the system average throughput, energy efficiency, and relevant parameters against the existing *All Local Computing*, *All Offloading*, *Algorithm* [16], and *Optimal*. *All Local Computing* means the sensor uses the charged energy to process their tasks locally. *All Offloading* denotes that the sensor offloads all computing tasks to the MEC for processing. *Algorithm* [16] is presented to address maximizing the computation rate of all users for WPT-powered WSN by jointly optimizing the individual computing mode selection and the system transmission time allocation where it is assumed that HAP is well powered and the transmitting power is fixed. *Optimal* refers to finding the optimal mode and allocating energy employing exhaustive enumeration. All except *Algorithm* [16] use the same energy provision constraint as the *proposed*. The following firstly illustrates the simulation environment and then presents the simulation results.

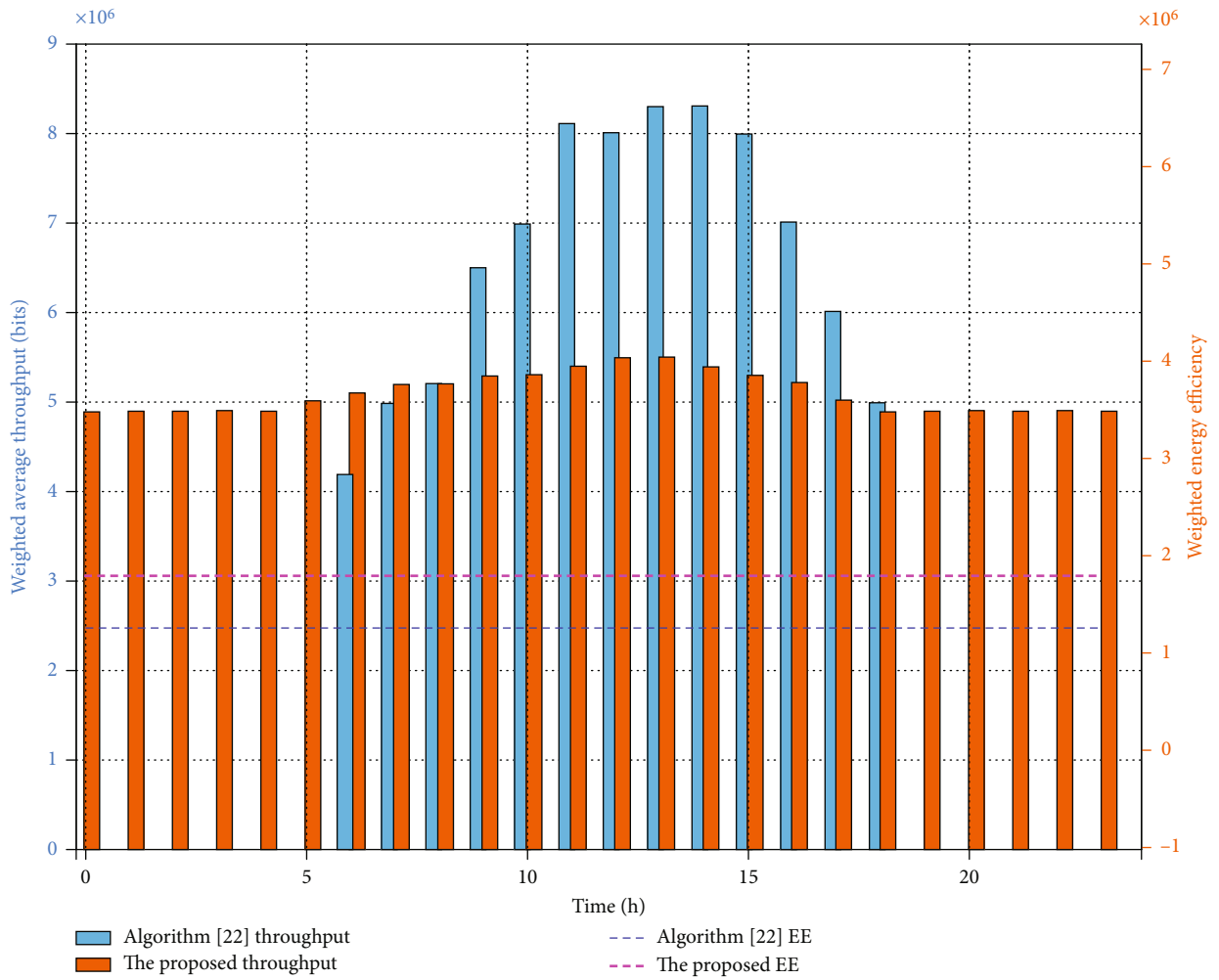
5.1. Simulation Setup. We consider a coverage area of $100 \text{ m} \times 100 \text{ m}$ with total node number [5, 10, 15, 21, 25] for solar-powered multinode MEC-enabled agriculture WSN, which includes a HAP-integrated MEC and several sensor nodes randomly distributed. HAP are equipped with a $1 \text{ m} \times 1 \text{ m}$ solar panel, which has an energy-harvesting efficiency of 20%. Solar energy arrivals are calculated in a similar way to [30], which follows the hidden Markov model with four states, including *excellent*, *good*, *fair*, and *poor*. The distribution of each state has a mean of 94.6, 76.0, 45.6, and 17.9, and variance of 0.31, 1.55, 1.48, and 0.71, respectively. The HAP has a battery with a capacity of 10 kJ, and the maximum transmitting power is set to 5 W. All nodes recharge themselves with energy efficiency 0.51, and the parameter of computing efficiency is set 109. The local CPU frequency of each node is set ranging from 0.2 GHz to 2 GHz, and the CPU frequency of each MEC server is set as 4 GHz. The channel gain from the sensor node to HAP follows the Rayleigh fading channel model, where the path loss factor is set varying from 2.0 Hz to 3.5 Hz. The system bandwidth is set to 200 kHz, and the noise spectral density is set to -95 dB/

Hz. The numerical experiments are performed by the Monte Carlo method as follows. All nodes are randomly distributed in the sensing area for 100 times, and the corresponding maximum throughput is calculated, which are probabilistically distributed by interval, and the number with the highest throughput value from the highest probability interval is selected as the optimal throughput of this network, and the corresponding offloading calculation decision and resource scheduling scheme is the best. All the simulations are performed on a desktop computer with an Intel Core i7-8700U 3.2 GHz CPU and 24 GB memory.

5.2. Performance for Network. To achieve sustainable monitoring for solar-powered agriculture WSNs, we propose a joint optimization method for resource scheduling and computation offloading strategy to maximize the minimum system throughput of each time slot during one period under the solar energy periodicity and volatility constraints. Figure 4 shows in detail the variation of system throughput over time and the system period energy efficiency under two different algorithms. Here, the period energy efficiency is defined as the ratio of network throughput to network energy consumption during one period, where the energy consumption mainly comes from the energy broadcast and the energy consumption of the MEC in HAP and can be measured as the solar-harvested of HAP, which can be expressed as Equation (50).

$$EE = \frac{\sum_{k=1}^T \sum_{i=1}^N w_i \left[s_i^k \Phi_{i,loc}^k + (1 - s_i^k) \Phi_{i,off}^k \right]}{\sum_{k=1}^T E_{j,har}^k}, \quad \forall i \in N, k \in T. \quad (51)$$

Figure 4 illustrates the variation of the system throughput over time under different weather conditions. It is obvious to observe that the system throughput of *Algorithm* [16] fluctuates significantly at different time slots within one period compared to the proposed, which can reach the maximum during the day but zero at night. This is because *Algorithm* [16] is designed to maximize the system computation rate with the assumption that the HAP is always energetic and that the transmit power is constant. This is clearly not applicable to highly volatile solar energy. Conversely, the proposed has a balanced throughput across time slots by maximizing the minimum one slot throughput to balance the energy broadcast by the HAP across time slots, which can achieve a rational energy allocation and storage during the day to ensure normal monitoring at night. Also, it is evident from Figure 4 that the proposed can achieve a higher system period energy efficiency than the *Algorithm* [16] by managing the broadcast power of harvested solar energy. Compared to rainy days, the system has higher throughput and energy efficiency on sunny days because it has more solar energy and its value varies less over time. More notably, the algorithm proposed in this paper has better stability in terms of throughput either over diurnal or climate change processes, which shows that it is more adaptable to real-time energy changes in the network.



(a)

FIGURE 4: Continued.

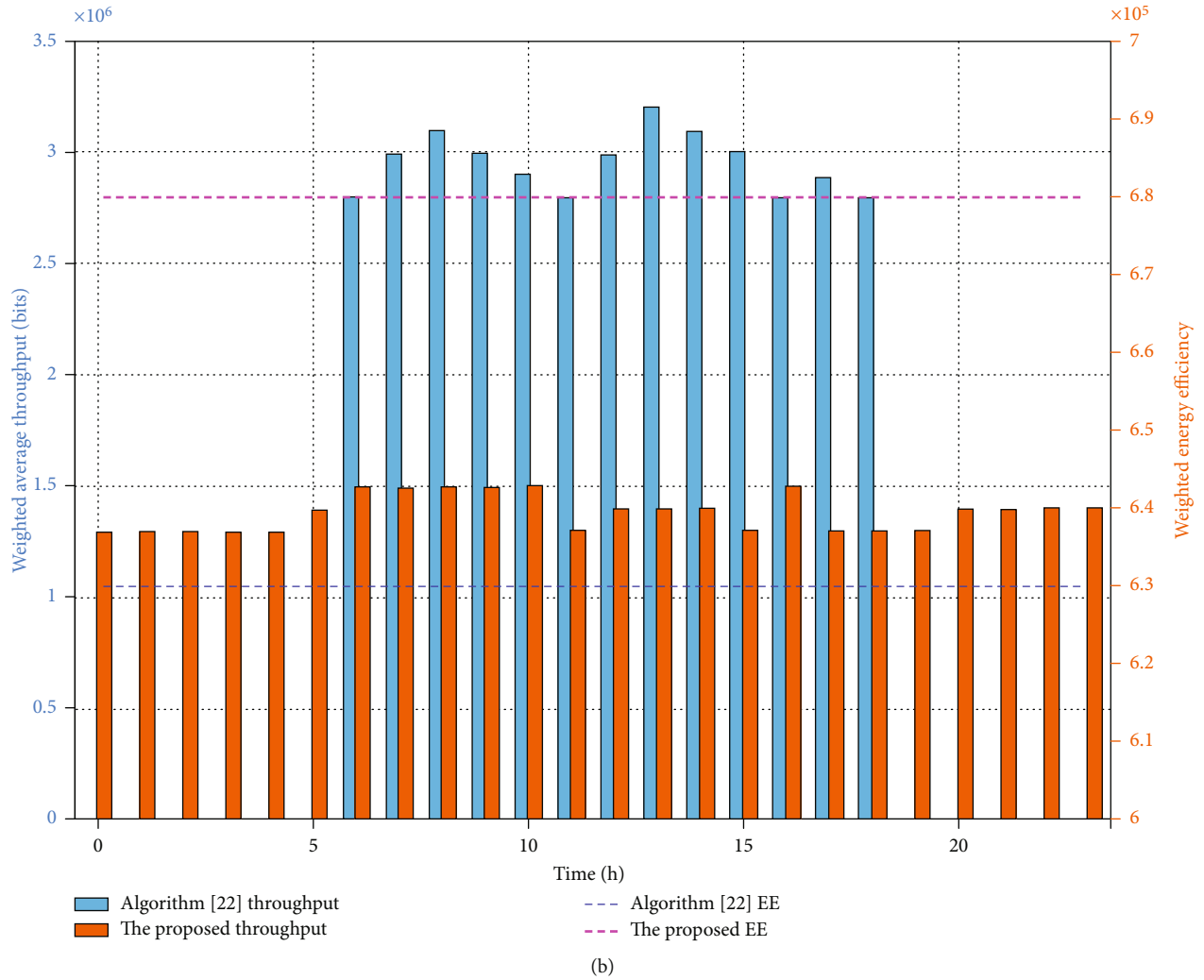


FIGURE 4: Plot of the weighted sum throughput of each slot and weighted energy efficiency versus the time with respect to (a) sunny and (b) rainy.

Figure 5 shows the effect of different algorithms on the system average throughput during one period versus total node number for path loss factor $\alpha = 2.4$. It is obvious that the *proposed* can achieve near-optimal performance and is superior to the other algorithms as the number of nodes increases. As more nodes are added gradually, the system average throughput of all the above algorithms increases, especially in the offload mode. But when the number of nodes is particularly large, the throughput in the offload mode drops abruptly. The reason for this is that when the number of nodes exceeds a threshold, wireless channel interference will increase, and this will lead to a reduction in offload link capacity. However, the local computing mode grows smoothly with the number of nodes due to the high independence of each node.

Figures 6(a) and 6(b) compare the effects of the path loss factor α and the average distance \bar{d} between HAP and nodes on the system average throughput under different algorithms, respectively, where both α and \bar{d} are important factors that affect the channel state. It can be seen from

Figure 6(a) that, for a given \bar{d} , the average throughput of the above algorithms all suffer varying degrees of decline as the channel state deteriorates with α increasing from 2 to 3.5. *All Offloading* is close to optimal performance when α is small and drops significantly as α increases, which is because task offloading suffers from both the energy broadcast downlink state and the data transmission uplink state. In contrast, local computation has a more moderate degradation due to its dependence on the downlink state only. Note that the *Proposed* is perfectly aligned with the optimal algorithm because it jointly optimizes energy scheduling and offloading strategy, which drives the network to choose more local computing to reduce losses as the channel state progressively deteriorates. In Figure 6(b), we compare the average throughput variation of systems with different algorithms when the average distance \bar{d} of HAP-to-nodes varies from 10 m to 50 m when $\alpha = 2.4$. We observe that the throughput of the above methods decreases as the average distance increases, because the distance between HAP and nodes affects the channel gain h_{ij} , which affects the

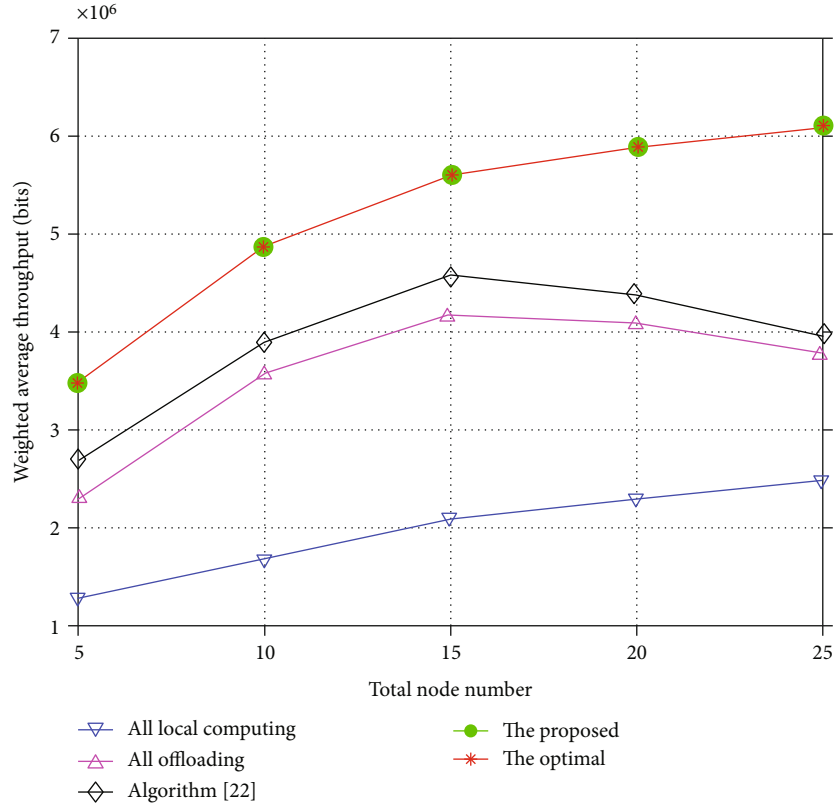
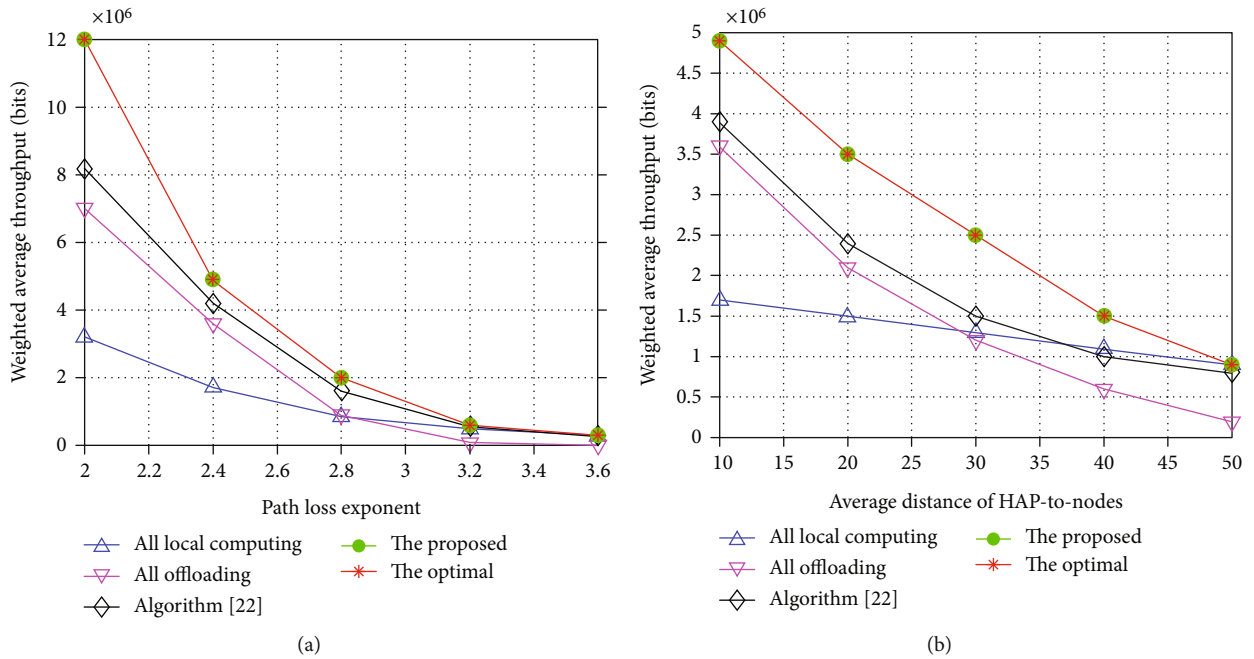


FIGURE 5: Plot of weighted average throughput versus total node number.

FIGURE 6: Plot of weighted average throughput with respect to (a) path loss exponent α when $\bar{d} = 20$ and (b) average distance \bar{d} of HAP-to-nodes when $\alpha = 2.4$.

harvested energy E_i of node i and subsequently affects local throughput $\Phi_{i,loc}$ and computation offloading throughput $\Phi_{i,off}$, as in α . Comparing Figures 6(a) and 6(b), we can

observe that α and \bar{d} have a similar effect on system throughput, as both α and \bar{d} influence network throughput by regulating the channel state.

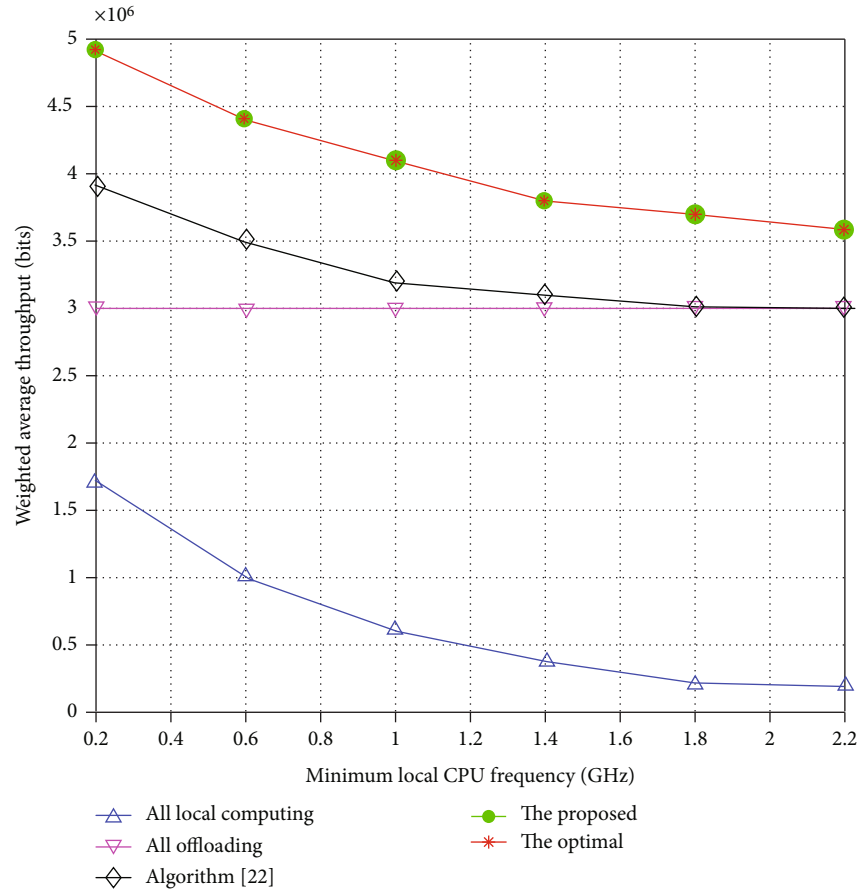


FIGURE 7: Plot of weighted average throughput versus minimum local CPU frequency.

Finally, we investigated the effect of the local minimum local CPU frequency on the network average throughput under different algorithms in Figure 7, from which we can observe that as the minimum local CPU frequency gradually increases, the average throughput under *all local computing* is affected the most with a significant downward trend, with some impact under the *proposed*, but almost no impact under *all offloading*. This is because the increase of the minimum local CPU frequency can drive more nodes to choose offloading. It is interesting to note that when the minimum local CPU frequency gradually increases to a certain value, the throughput under proposed tends to under offloading. This is because all nodes of the network would choose the computation offloading when the minimum local CPU frequency is greater than a certain threshold.

6. Conclusions

This paper studied a real-time environmental monitoring system for solar-powered MEC-enabled smart agriculture. To alleviate the cyclical and day-night fluctuation influence of solar energy on network performance, we formulated a joint resource scheduling and computation offloading strategy optimization problem, which maximizes the minimum weighted computation capacity across the time slots by

jointly optimizing the computation offloading strategy, energy provision of the solar-powered hybrid access point (HAP), local CPU frequency as well as time scheduling. Based on the coordinate descent method and successive convex approximation theory, we developed a multiply-iterated decoupling optimization algorithm to solve the strong coupling between binary offloading strategy, broadcast power, and time allocation. The simulation results demonstrate that the proposed algorithm outperforms in terms of energy efficiency and computational volume time slot balance and can better adapt to solar energy variations to meet the real-time status monitoring requirements of the smart agriculture IoT.

In order to further improve green energy utilization, future research can be extended as follows. First, the assumption of the solar harvesting rate in this paper is linear. In practice, it varies nonlinearly with the transmitting power, so it will be more relevant to examine the optimal offloading decision under the nonlinear harvesting conditions. Next, HAP broadcast power can be extended by using MIMO technology to reduce interference and improve energy utilization. In addition, NOMA-based protocols can be used instead of TDMA-based multinode task offloading to improve the network's spectral efficiency. Finally, the long-term optimization model of network energy can be constructed by considering the time-varying nature of

network energy state information, channel state information, and task state information.

Data Availability

The data are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (NSFC) under Grant 62071179.

References

- [1] O. Friha, M. A. Ferrag, L. Shu, L. Maglaras, and X. Wang, "Internet of things for the future of smart agriculture: a comprehensive survey of emerging technologies," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 718–752, 2021.
- [2] D. Popescu, F. Stoican, G. Stamatescu, L. Ichim, and C. Dragana, "Advanced UAV-WSN system for intelligent monitoring in precision agriculture," *Sensors*, vol. 20, no. 3, p. 817, 2020.
- [3] M. Ayaz, M. Ammad-Uddin, Z. Sharif, A. Mansour, and E. H. M. Aggoune, "Internet-of-Things (IoT)-based smart agriculture: toward making the fields talk," *IEEE Access*, vol. 7, pp. 129551–129583, 2019.
- [4] A. Boukerche, Q. Wu, and P. Sun, "Efficient green protocols for sustainable wireless sensor networks," *IEEE Transactions on Sustainable Computing*, vol. 5, no. 1, pp. 61–80, 2020.
- [5] C. Iwendi, P. K. R. Maddikunta, T. R. Gadekallu, K. Lakshmana, A. K. Bashir, and M. J. Piran, "A metaheuristic optimization approach for energy efficiency in the IoT networks," *Software: Practice and Experience*, vol. 51, no. 12, pp. 2558–2571, 2021.
- [6] E. A. Khalil and S. Ozdemir, "Reliable and energy efficient topology control in probabilistic wireless sensor networks via multi-objective optimization," *The Journal of Supercomputing*, vol. 73, no. 6, pp. 2632–2656, 2017.
- [7] R. Wan, N. Xiong, and N. T. Loc, "An energy-efficient sleep scheduling mechanism with similarity measure for wireless sensor networks," *Human-centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1–22, 2018.
- [8] T. D. Nguyen, J. Y. Khan, and D. T. Ngo, "A distributed energy-harvesting-aware routing algorithm for heterogeneous IoT networks," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 4, pp. 1115–1127, 2018.
- [9] C. Buratti and R. Verdone, "Joint scheduling and routing with power control for centralized wireless sensor networks," *Wireless Networks*, vol. 24, no. 5, pp. 1699–1714, 2018.
- [10] X. Gu, L. Grauwin, D. Dousset, S. Hemour, and K. Wu, "Dynamic ambient RF energy density measurements of Montreal for battery-free IoT sensor network planning," *IEEE Internet of Things Journal*, vol. 8, no. 17, pp. 13209–13221, 2021.
- [11] A. Riaz, M. R. Sarker, M. H. M. Saad, and R. Mohamed, "Review on comparison of different energy storage technologies used in micro-energy harvesting, WSNs, low-cost micro-electronic devices: challenges and recommendations," *Sensors*, vol. 21, no. 15, p. 5041, 2021.
- [12] H. Sharma, A. Haque, and Z. A. Jaffery, "Solar energy harvesting wireless sensor network nodes: a survey," *Journal of Renewable and Sustainable Energy*, vol. 10, no. 2, article 023704, 2018.
- [13] P. Corcoran and S. K. Datta, "Mobile-edge computing and the internet of things for consumers: extending cloud computing and services to the edge of the network," *IEEE Consumer Electronics Magazine*, vol. 5, no. 4, pp. 73–74, 2016.
- [14] B. Ali, M. A. Gregory, and S. Li, "Multi-access edge computing architecture, data security and privacy: a review," *IEEE Access*, vol. 9, pp. 18706–18721, 2021.
- [15] C. Xu, G. Zheng, and X. Zhao, "Energy-minimization task offloading and resource allocation for mobile edge computing in NOMA heterogeneous networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 16001–16016, 2020.
- [16] S. Bi and Y. J. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Transactions on Wireless Communications*, vol. 17, no. 6, pp. 4177–4190, 2018.
- [17] 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 *GLOBECOM 2017-2017 IEEE Global Communications Conference*, Singapore, 2017.
- [18] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 17, no. 3, pp. 1784–1797, 2018.
- [19] L. Shi, Y. Ye, X. Chu, and G. Lu, "Computation energy efficiency maximization for a NOMA-based WPT-MEC network," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10731–10744, 2021.
- [20] F. Wang, X. Jie, and S. Cui, "Optimal energy allocation and task offloading policy for wireless powered mobile edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 19, no. 4, pp. 2443–2459, 2020.
- [21] F. Zhou and H. Rose Qingyang, "Computation efficiency maximization in wireless-powered mobile edge computing networks," *IEEE Transactions on Wireless Communications*, vol. 19, no. 5, pp. 3170–3184, 2020.
- [22] 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 Transactions on Wireless Communications*, vol. 19, no. 3, pp. 1855–1867, 2019.
- [23] J. H. Anajemba, T. Yue, C. Iwendi, M. Alenezi, and M. Mittal, "Optimal cooperative offloading scheme for energy efficient multi-access edge computation," *Access*, vol. 8, pp. 53931–53941, 2020.
- [24] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "Mobile edge computing: Survey and research outlook," 2017, <https://arxiv.org/abs/1701.01090v3>.
- [25] B. Li, Z. Fei, J. Shen, X. Jiang, and X. Zhong, "Dynamic offloading for energy harvesting mobile edge computing: architecture, case studies, and future directions," *Access*, vol. 7, pp. 79877–79886, 2019.
- [26] F. Wang, H. Xing, and X. Jie, "Real-time resource allocation for wireless powered multiuser mobile edge computing with

- energy and task causality,” *IEEE Transactions on Communications*, vol. 68, no. 11, pp. 7140–7155, 2020.
- [27] 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 Transactions on Network and Service Management*, vol. 18, no. 2, pp. 2154–2165, 2021.
- [28] Q. Tang, R. Xie, F. R. Yu, T. Huang, and Y. Liu, “Decentralized computation offloading in IoT fog computing system with energy harvesting: a dec-POMDP approach,” *IEEE Internet of Things Journal*, vol. 7, no. 6, pp. 4898–4911, 2020.
- [29] W. Yang, G. Durisi, T. Koch, and Y. Polyanskiy, “Quasi-static multiple-antenna fading channels at finite blocklength,” *IEEE Transactions on Information Theory*, vol. 60, no. 7, pp. 4232–4265, 2014.
- [30] M.-L. Ku, Y. Chen, and K. J. R. Liu, “Data-driven stochastic models and policies for energy harvesting sensor communications,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 8, pp. 1505–1520, 2015.
- [31] B. Sklar, “Rayleigh fading channels in mobile digital communication systems.I. Characterization,” *IEEE Communications magazine*, vol. 35, no. 7, pp. 90–100, 1997.
- [32] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, “Energy-optimal mobile cloud computing under stochastic wireless channel,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, 2013.
- [33] C. E. Shannon, “A mathematical theory of communication,” *ACM SIGMOBILE mobile computing and communications review*, vol. 5, no. 1, pp. 3–55, 2001.
- [34] X. Chen, L. Jiao, W. Li, and X. Fu, “Efficient multi-user computation offloading for mobile-edge cloud computing,” *IEEE/ACM transactions on networking*, vol. 24, no. 5, pp. 2795–2808, 2016.
- [35] C. Jiang, T. Fan, H. Gao et al., “Energy aware edge computing: a survey,” *Computer Communications*, vol. 151, pp. 556–580, 2020.
- [36] B. R. Marks and G. P. Wright, “Technical note—a general inner approximation algorithm for nonconvex mathematical programs,” *Operations Research*, vol. 26, no. 4, pp. 681–683, 1978.
- [37] P. Breheny and J. Huang, “Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection,” *The annals of applied statistics*, vol. 5, no. 1, pp. 232–253, 2011.