Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.083306

ARTICLE

Multi-UAV Collaborative Energy Charging for Battery-Free SWIPT-Enabled Sensor Networks Based on MADDPG

Xiangyi Le[1], Deyu Lin[1,2,*], Yufei Zhao[2], Wang Miao[3] and Yong Liang Guan[2]

[1]School of Software, Nanchang University, Nanchang, China
[2]School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, Singapore
[3]Computer Science Department, University of Exeter, Exeter, UK

*Corresponding Author: Deyu Lin. Email: Dashing_lin@126.com

Received: 01 April 2026; Accepted: 10 June 2026; Published: 23 July 2026

ABSTRACT: The emergence of Unmanned Aerial Vehicle (UAV)-enabled Wireless Energy Transfer (WET) and Simultaneous Wireless Information and Power Transfer (SWIPT) technology provide a promising solution to overcome the energy sustainability limitations of traditional harvesting-reliant sensor networks. However, in large-scale Battery-free SWIPT-enabled Sensor Networks (BSSN) characterized by sparse node distribution and heterogeneous energy consumption and harvesting rates, employing a single UAV for energy replenishment often suffers from insufficient operation continuity and low charging efficiency. To overcome these challenges, a Multi-UAV Collaborative Energy Charging for BSSN Based on Multi-Agent Deep Deterministic Policy Gradient (MCEC-MADDPG) is proposed in this paper. Specifically, we construct a collaborative one-to-one precision energy supply model where UAVs hover directly above specific nodes to achieve power transmission without complex beamforming requirements. To achieve collaborative scheduling among multiple UAVs in wide-area dynamic environments, the energy replenishment problem is first formulated as a Partially Observable Markov Decision Process (POMDP). Subsequently, the Centralized Training with Decentralized Execution (CTDE) architecture of the MADDPG algorithm is leveraged to solve this POMDP, which effectively tackles the non-stationarity challenge inherent in multi-agent environments. Simulation results demonstrate that MCEC-MADDPG exhibits superior performance in terms of convergence speed and stability. It enables the adaptive emergence of spatial-division collaborative strategies, significantly enhances the average residual energy of the network, and elevates the node survival rate to nearly 90%. Compared with Deep Deterministic Policy Gradient (DDPG), the traditional static Partition-Greedy method, the heuristic K-Means algorithm and the dynamic Two-Layer task allocation strategy, the proposed approach demonstrates substantial advantages.

KEYWORDS: Battery-free SWIPT-enabled sensor networks; multi-agent deep deterministic policy gradient; multi-unmanned aerial vehicle; collaborative energy charging; partially observable Markov decision process; centralized training with decentralized execution

1 Introduction

Battery-free SWIPT-enabled Sensor Networks (BSSN) play a vital role in fields such as structural health monitoring, implantable medical devices and smart cities [1,2]. However, sensor nodes are typically powered by small energy storage capacitors that rely on intermittent ambient energy harvesting, and the depletion of the energy leads to network topology fragmentation and data loss [3]. Recent advancements in spatial-temporal topological decoupling and dynamic adaptive energy sustainability have provided new perspectives for maintaining robust operations [2,4]. To further extend network lifetime, utilizing Unmanned Aerial

Vehicles (UAVs) equipped with wireless energy transfer equipment to replenish energy for sensor nodes—thereby constructing a UAV-assisted Energy-Harvesting Sensor Network (WEHSN)—has become a research hotspot in recent years [5–7].

Although employing a single UAV for energy charging has achieved significant results in small-scale networks [8], the single-UAV scheme often faces bottlenecks such as insufficient endurance and low charging efficiency in large-scale areas where sensor nodes are sparsely distributed [9]. To overcome this limitation, introducing multi-UAV coordination to perform wireless energy transfer tasks has become an inevitable trend [10]. However, compared to single-UAV scenarios, multi-UAV collaboration introduces a series of new challenges, including resource allocation conflicts between UAVs, flight collision risks, communication limitations, and high environmental dynamism and uncertainty [11–13].

To address these challenges, Multi-Agent Deep Deterministic Policy Gradient (MADDPG) has proven its efficacy in tackling dynamic environmental challenges within complex tasks [14]. In automated warehouse logistics, swarm robotics coordination, and urban traffic management, it is widely recognized for enabling multi-agent systems to reach a stable Nash Equilibrium [15,16]. Integrating MADDPG to multi-UAV energy charging tasks offers a potent framework for solving challenges like resource allocation conflicts, collision risks, and environmental uncertainties [17]. By employing a Centralized Training, Decentralized Execution (CTDE) architecture, the algorithm allows each UAV to access global state information during training, effectively overcoming the non-stationarity issues typical of multi-agent environments [18]. During the execution phase, UAVs perform real-time autonomous decision-making based solely on local observations. This paradigm encourages the emergence of precise obstacle avoidance and task-partitioning strategies, ensuring high efficiency and system robustness in highly dynamic settings [19].

In this paper, we propose a Multi-UAV Collaborative Energy Charging framework for Battery-free SWIPT-enabled Sensor Networks based on Multi-Agent Deep Deterministic Policy Gradient, termed MCEC-MADDPG. Different from existing MADDPG-based UAV scheduling studies that mainly focus on trajectory optimization, communication coverage, or generic task allocation, this work specifically targets the energy sustainability problem of capacitor-powered BSSN with heterogeneous energy consumption and sparse charging demands. The main contributions are summarized as follows.

(1) We formulate the multi-UAV charging problem by taking the maximization of the network-wide average residual energy as the primary optimization objective, rather than merely minimizing flight distance or task completion time. This objective is particularly suitable for battery-free SWIPT-enabled sensor networks, where node energy depletion may directly cause topology fragmentation and data loss. By incorporating node survival, UAV collision avoidance, task uniqueness, and motion constraints, the proposed formulation better reflects the long-term operational requirements of BSSN.

(2) To capture the partial observability and dynamic uncertainty of multi-UAV energy replenishment, the charging scheduling process is formulated as a Partially Observable Markov Decision Process POMDP. The state space jointly characterizes sensor-node residual energy, UAV positions, and UAV residual battery levels, while each UAV makes decisions based on local observations. The action space is defined as continuous velocity-vector control, allowing smooth UAV trajectory planning and hovering-based charging. The reward function integrates average residual energy improvement, node-death penalty, and collision penalty, thereby explicitly guiding the agents to balance energy sustainability, charging fairness, and flight safety.

(3) We adapt the MADDPG framework to the proposed multi-UAV BSSN charging scenario. During centralized training, each Critic evaluates joint observations and joint actions of all UAVs, enabling the agents to learn cooperative task allocation and reduce competition for the same low-energy nodes. During decentralized execution, each UAV relies only on its local observation for real-time

decision-making, which improves deployment feasibility and reduces the requirement for global communication. This design directly addresses the non-stationarity problem caused by simultaneous policy updates in multi-UAV systems.

(4) Simulation results show that MCEC-MADDPG achieves faster and more stable convergence than independent DDPG and outperforms traditional Partition-Greedy, K-Means, and Two-Layer task allocation strategies. The proposed method enables the adaptive emergence of spatial-division cooperative behaviors, improves the average residual energy of the network, reduces dead nodes, and increases the network survival rate to nearly 90%, demonstrating its effectiveness for large-scale BSSN energy sustainability.

2 Related Works

2.1 Traditional Heuristic and Static Planning for Energy Replenishment

Early research primarily focused on utilizing geometric spatial partitioning and classical combinatorial optimization theory to solve path planning problems. The K-Means clustering algorithm, as applied in wireless sensor networks by Sasikumar and Khara [20], is frequently used to partition distributed nodes into task clusters for preliminary load balancing. Similar clustering-based methods have also been investigated for sensor-node organization in wireless sensor networks [21]. Building upon clustering-based task organization, two-layer scheduling strategies generally combine upper-layer task assignment with lower-layer path planning. In the lower layer, the Traveling Salesperson Problem (TSP) and its heuristic variants, such as the Lin-Kernighan algorithm, are commonly used to plan efficient replenishment or visiting sequences [22–24]. For localized dynamic demands, partition-based greedy strategies are often adopted due to their low computational complexity within fixed sub-regions. However, these methods essentially rely on predefined static rules and struggle to address the energy harvesting and consumption heterogeneity in sensor networks, as highlighted by Xie et al. [25]. In dynamic environments where node energy consumption rates fluctuate drastically, static planning may lead to inefficient energy replenishment and network energy imbalance [26].

2.2 UAV Collaborative Scheduling Based Deep Reinforcement Learning

To enhance system adaptability in dynamic environments, Deep Reinforcement Learning (DRL) has been integrated into the field of path planning. Following the foundational work on Deep Q-Network (DQN) by Mnih et al. [27], Lillicrap et al. [28] proposed the DDPG algorithm for continuous action control, enabling smooth UAV flight control via an Actor-Critic architecture. Later actor-critic studies, such as TD3 proposed by Fujimoto et al. [29] and soft actor-critic proposed by Haarnoja et al. [30], further improved continuous-control reinforcement learning from the perspectives of value estimation stability and exploration. Nevertheless, directly extending single-agent actor-critic algorithms to multi-UAV cooperative scenarios in an independent-learning manner still faces severe challenges. Because multiple agents update their policies simultaneously, the environment becomes non-stationary from the perspective of each individual agent. This violates the stationarity assumption of conventional Markov Decision Processes (MDPs), leading to ineffective experience replay, unstable gradient updates, and difficulties in convergence or task coordination.

2.3 Distributed Collaborative Strategy Based MADDPG

To overcome the hurdle of non-stationarity, the MADDPG algorithm proposed by Lowe et al. [31] has emerged as an advanced solution for multi-agent continuous control. MADDPG employs a CTDE architecture. During the training phase, an augmented Critic network receives the joint observations and actions of all agents to grasp global information; in the execution phase, each UAV performs real-time

decision-making based solely on local observations. Recent studies by Sunehag et al. [32] have demonstrated that this framework enables UAV swarms to autonomously develop sophisticated spatial-division and obstacle avoidance strategies. Consequently, introducing the MADDPG algorithm into multi-UAV collaborative energy replenishment tasks will provide a novel pathway for resolving resource allocation conflicts in dynamic environments.

Motivated by the potential of multi-agent reinforcement learning in dynamic environments, we propose a Multi-UAV Collaborative Energy Charging for Wireless Sensor Networks Based on Multi-Agent Deep Deterministic Policy Gradient (MCEC-MADDPG) framework to address the constraints of energy autonomy and coordination in large-scale BSSN.

3 System Model and Problem Formulation

This section establishes the system model for multi-UAV-assisted wireless charging in BSSN, including network, energy transmission, UAV kinematics, and node energy dynamics models, as well as an analysis of charging efficiency. Building on these models, we then formulate the optimization problem for maximizing the expectation of the network's average residual energy, subject to constraints on flight safety, task uniqueness, and node survival.

3.1 Multi-UAV One-to-One Wireless Charging Model

One-to-many charging is generally suitable for networks with densely distributed charging demands. However, in non-deterministic scheduling scenarios, nodes requesting energy are typically sparse, which significantly degrades the efficiency of this mode. Furthermore, one-to-many charging complicates the prediction of scheduling feasibility. The system must verify whether a UAV's residual energy can sustain both the flight and the total transferred energy along the planned route. If a UAV opportunistically charges non-requesting nodes within its coverage, it introduces unpredictable energy consumption. This uncertainty exacerbates the difficulty of predicting endurance and severely undermines the reliability of scheduling decisions.

Therefore, the energy replenishment mechanism proposed in this paper adopts a one-to-one charging mode, as is shown in Fig. 1. Specifically, multiple UAVs equipped with wireless charging modules operate collaboratively under a predefined scheduling strategy, enabling dynamic task allocation. Although the service is physically conducted in a one-to-one manner, each UAV can sequentially serve multiple target nodes within a single flight mission through optimized path planning. After arriving above a target node, the UAV hovers to complete the charging process and then immediately proceeds to the next node in the sequence, continuing this process until the charging cycle is completed.

The network consists of N sensor nodes and M UAVs. Sensor nodes are denoted as $s_i \in S = \{s_1, s_2, \ldots, s_N\}$ with coordinates $s_i = (x_{s_i}, y_{s_i}, 0)$, energy consumption P_n, and storage capacitor capacity E_{cap}. UAVs are denoted as $u_j \in U = \{u_1, u_2, \ldots, u_M\}$. At a fixed altitude H, the coordinates of UAV u_j at time t are $q_{u_j}(t) = \left(x_{u_j}(t), y_{u_j}(t), z_{u_j}(t)\right)$. According to the free-space path loss model, the received power $P_r(d)$ from UAV u to sensor node s is distance-dependent. Assuming that the UAV transmits with a constant power P_t, the received power can be expressed as follows,

$$P_r(d) = \frac{\eta_0 P_t}{(d+\Delta)^{\alpha}} \tag{1}$$

where η_0 denotes the channel gain at a reference distance, $d = \left\| q_{u_j}(t) - s_i \right\|$ represents the Euclidean distance between UAV u_j and node s_i, $\alpha \geq 2$ is the path loss exponent, and Δ is a small constant introduced to avoid

singularity when $d = 0$. To complete the charging task, each node s_i is required to receive a total energy of E_{req,s_i}.

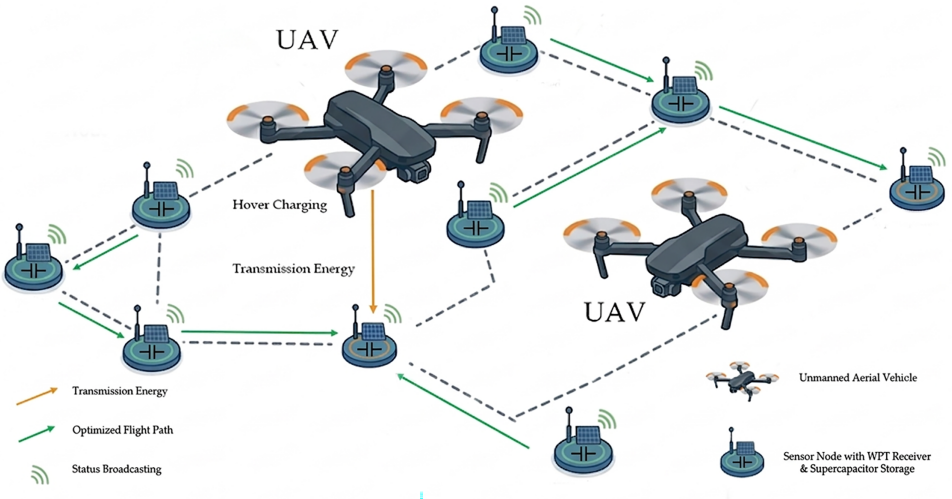

Figure 1: Multi-UAV-assisted one-to-one charging model.

3.2 Optimization Objective

Different from conventional battery-powered sensor networks, sensor nodes in BSSN rely on capacitors for short-term energy storage, making their available energy highly dynamic and vulnerable to depletion. Continuous energy degradation may cause node failure, local topology fragmentation, data forwarding interruption, and reduced network service quality. Therefore, the network-wide average residual energy is adopted to characterize the overall energy state and evaluate the effectiveness of the UAV charging policy. This metric also provides a continuous and stable optimization signal for deep reinforcement learning, whereas indicators such as node death and network lifetime are usually sparse or delayed. Nevertheless, average residual energy may conceal local energy imbalance and cannot fully represent all aspects of network performance. Hence, dead nodes, network survival rate, fairness, and throughput are regarded as complementary metrics, with dead nodes and network survival rate further evaluated in the simulation section. Let T denote the system scheduling period. The objective is to determine an optimal multi-UAV cooperative scheduling policy π that maximizes the expected average residual energy of all nodes over the period T. The objective function is defined as follows,

$$\max_{\pi} J = \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T}\frac{1}{N}\sum_{i=1}^{N}E_{s_i}^{res}\right]. \tag{2}$$

To prevent node failure, the residual energy of any sensor node s_i, denoted as $E_{s_i}^{res}(t)$, must remain positive at any time before task completion. It is defined as follows.

$$E_{s_i}^{res}(t) > 0, \quad \forall t \in [0, T], \forall s_i \in S. \tag{3}$$

The collision avoidance constraint requires that the distance between any two UAVs must always be greater than a predefined safety distance D_{safe}, which is expressed as follows.

$$\left\| q_{u_j}(t) - q_{u_k}(t) \right\| > D_{safe}, \quad \forall u_j \neq u_k, \forall t. \tag{4}$$

The kinematic constraint ensures that the velocity of each UAV does not exceed the maximum allowable speed V_{max}, which is constrained as follows.

$$\left\| v_{u_j}(t) \right\| \leq V_{max}, \forall u_j \in U. \tag{5}$$

The UAV position is updated according to the formula as below,

$$q_{u_j}(t+1) = q_{u_j}(t) + v_{u_j}(t) \cdot \delta_t \tag{6}$$

where δ_t denotes the duration of a single time slot.

4 MCEC-MADDPG

In this section, a MADDPG-based cooperative energy charging mechanism is presented, which leverages DRL to jointly optimize UAV flight trajectories and charging schedules. First, considering the limited sensing range of individual UAVs, the sequential decision-making problem is formulated as a POMDP. Subsequently, to overcome the non-stationarity dilemma caused by concurrent multi-agent exploration, the MADDPG algorithm is introduced. By leveraging the CTDE architecture, the algorithm enables UAVs to navigate complex resource competition and collision avoidance while converging toward a global optimal synergy.

4.1 POMDP Model

A POMDP can be regarded as a generalization of the classical MDP. Unlike an MDP, which assumes that the environmental state is fully observable, a POMDP allows an agent to obtain only partial or noisy observations of the environment's true state. In other words, if the observation set of a POMDP is identical to the state set and each observation corresponds deterministically to a state, the POMDP degenerates into a standard MDP. Consequently, the POMDP framework exhibits superior versatility, enabling the simulation of various real-world sequential decision-making processes. Since its inception, the concept of POMDP has garnered extensive attention and has been widely applied in fields such as robotic control [33,34], UAV formation decision-making [35,36], and reinforcement learning [37,38].

In multi-UAV cooperative energy replenishment tasks, UAVs are required to make decisions based on local observational information. Since the energy status fluctuations of sensor nodes, the positions of other UAVs, and the overall environmental dynamics cannot be fully perceived by an individual UAV, the problem naturally aligns with the characteristics of a POMDP. Accordingly, we define this process as a quadruple.

$$POMDP = \langle X, O, A, R \rangle. \tag{7}$$

State space

The global state space X of the system encompasses the operational states of all UAVs and the energy levels of the ground sensor nodes, which primarily consists of the following three components,

$$E_S^{res}(t) = \left(E_{s_1}^{res}(t), E_{s_2}^{res}(t), \ldots, E_{s_i}^{res}(t), \ldots, E_{s_N}^{res}(t) \right), \mathrm{t} \in T, \tag{8}$$

$$q_U(t) = \left(q_{u_1}(t), q_{u_2}(t), \ldots, q_{u_j}(t), \ldots, q_{u_M}(t) \right), \mathrm{t} \in T, \tag{9}$$

$$E_U(t) = \left(E_{u_1}^{res}(t), E_{u_2}^{res}(t), \ldots, E_{u_j}^{res}(t), \ldots, E_{u_M}^{res}(t) \right), \mathrm{t} \in T \tag{10}$$

where $E_S^{res}(t)$ denotes the residual energy of all sensor nodes at time slot t, and $E_{s_i}^{res}(t)$ represents the residual energy of the i-th node. The term $q_U(t)$ denotes the ground-projected positions of all UAVs at time slot t, while $E_U(t)$ represents the residual battery energy of all UAVs at time slot t.

Observation space

Since each UAV cannot access the global state information, it can only observe local information related to itself. The observation space of the j-th UAV at time slot t, denoted as o_{u_j}, is defined as below,

$$o_{u_j} = \left\{ \hat{E}_{nodes}^{u_j}(t), q_{u_j}(t), E_{u_j}^{res}(t) \right\} \tag{11}$$

where $\hat{E}_{nodes}^{u_j}(t)$ denotes the energy states of the ground sensor nodes observed by the j-th UAV within its charging range. For nodes outside the coverage area, the UAV may only retain the latest observation or treat their states as unknown. The terms $q_{u_j}(t)$ and $E_{u_j}^{res}(t)$ represent the current position and residual energy of the UAV itself, respectively.

Action space

To adapt to the continuous action space capability of the MADDPG algorithm and to achieve smooth UAV flight control, the action space is defined as continuous velocity-vector control rather than discrete movement commands. The action of each UAV at time slot t is defined as below,

$$a_u^t = \left\{ v_x, v_y, v_z \right\} \tag{12}$$

where v_x, v_y and v_z are normalized velocity components in the range $[-1, 1]$. The actual flight velocity is as follows.

$$V_{actual} = a_u^t \cdot V_{max}. \tag{13}$$

When the UAV moves above a target node and its velocity approaches zero, the hovering-based charging operation is automatically triggered. This continuous action-space formulation is consistent with the actor-critic structure of DDPG and enables more precise flight actions.

Reward function

The reward function is designed to guide multiple UAVs to collaboratively complete the charging task, improve the overall residual energy level of sensor nodes, and avoid unsafe flight behaviors. The reward function r_u^t of the UAV at time slot t is defined as below.

$$r_u^t = r_{avg}^t + r_{dead}^t + r_{collision}^t. \tag{14}$$

The global average energy reward r_{avg}^t measures the average residual energy level of all sensor nodes in the network at time slot t, which is defined as (14a),

$$r_{avg}^t = \frac{1}{N} \sum_{i=1}^{N} \frac{E_{s_i}^{res}(t)}{E_{cap}} \tag{14a}$$

where $E_{s_i}^{res}(t)$ represents the residual energy of sensor node s_i at time slot t, and E_{cap} is the capacitor capacity limit of each node. Since $E_{s_i}^{res}(t) \le E_{cap}$, the value of r_{avg}^t is normalized into $[0, 1]$. This term encourages the UAV swarm to maintain the overall energy baseline of the network rather than focusing only on a single low-energy node. r_{dead}^t is the node-death penalty. When a node within the sensing area is depleted of energy, a large penalty is imposed on the system to ensure charging fairness and node survival. It is defined as below,

$$r_{dead}^{t} = \begin{cases} -\lambda, & E_{s_i}^{res}(t) = 0 \\ 0, & otherwise \end{cases} \tag{14b}$$

where λ is the node-death penalty coefficient. Since the normalized average energy reward is bounded by 1 at each time slot, λ is set to be larger than the maximum possible one-step positive reward, so that node failure cannot be offset by a temporary increase in average residual energy. In the experiments, λ was selected through preliminary sensitivity tests from a candidate set, and the value that achieved stable convergence with fewer dead nodes was adopted. $r_{collision}^{t}$ is the collision penalty term, which is triggered when the distance between UAV u_j and another UAV is smaller than the safety threshold D_{safe}, thereby ensuring flight safety. It is defined as below,

$$r_{collision}^{t} = \begin{cases} -C, & if \left\| q_{u_j}(t) - q_{u_k}(t) \right\| \le D_{safe} \\ 0, & otherwise \end{cases} \tag{14c}$$

where C is the collision penalty coefficient. Since collision avoidance is a hard safety requirement in multi-UAV coordination, C is assigned a value no smaller than λ. This setting ensures that unsafe actions are strongly suppressed during policy learning. In the experiments, the final value of C was selected by balancing training stability, collision avoidance performance, and charging efficiency. $q_{u_j}(t)$ and $q_{u_k}(t)$ denote the position vectors of UAV j and UAV k at time slot t, respectively. D_{safe} is the predefined safety distance threshold.

It should be noted that the collision penalty provides a soft safety constraint rather than a strict safety guarantee. During the exploration stage of reinforcement learning, UAVs may still generate unsafe actions before the policy converges, even when a large collision penalty is imposed. Therefore, the current reward-based collision avoidance mechanism can reduce collision risks during learning, but it cannot completely eliminate temporary constraint violations. Safer mechanisms, such as constrained reinforcement learning, action shielding, control barrier functions, and model-predictive safety filters, will be considered in future work.

4.2 Energy Charging

For the multi-UAV cooperative energy supply task, the system must simultaneously satisfy continuous flight control, partial observability, collision avoidance, and global energy optimality. Since UAV motion control is a continuous-action decision problem rather than a discrete state transition process, conventional DQN methods cannot be directly applied. To address this issue, we first adopt the DDPG framework, which is based on the Actor-Critic architecture [39], as the basic form for single-agent continuous control, and then further extends it to MADDPG to accommodate the cooperative decision-making scenario involving multiple UAVs.

The DDPG framework consists of four neural networks. The Actor network $\mu(o_u|\theta^{\mu})$, outputs a deterministic action a_u according to the input UAV local observation o_u, while the Critic network $Q\left(o_u, a_u|\theta^{Q}\right)$ evaluates the value of that action under the current observation. The Target Actor network $\mu'\left(o_u|\theta^{\mu'}\right)$ and the Target Critic network $Q'\left(o_u, a_u|\theta^{Q'}\right)$ are copies of the current networks and are used to compute the target Q-value, thereby stabilizing the training process.

To improve training stability, DDPG introduces experience replay and target networks. The replay buffer D stores transition samples $\left(o_u^t, a_u^t, r^t, o_u^{t+1}\right)$ generated through interactions between the agent and the environment. During training, mini-batches are randomly sampled from the buffer for parameter

updates, thereby weakening temporal correlations among samples. The target network is updated slowly via a soft-update mechanism, as follows,

$$\theta^{\mu'} \leftarrow \tau\theta^{\mu} + (1-\tau)\,\theta^{\mu'}, \theta^{Q'} \leftarrow \tau\theta^{Q} + (1-\tau)\,\theta^{Q'} \tag{15}$$

where $\tau \ll 1$ is the soft-update coefficient. To enhance exploration in unknown regions and sparse states, exploration noise ξ is added to the action output as below.

$$a_u = \mu\left(o_u|\theta^{\mu}\right) + \xi. \tag{16}$$

The Critic network is updated by minimizing the mean-squared error between the predicted Q-value and the target Q-value, which is expressed as follows,

$$L\left(\theta^{Q}\right) = \frac{1}{W}\sum_{n=1}^{W}\left(z_n - Q\left(o_u^n, a_u^n|\theta^{Q}\right)\right)^2 \tag{17}$$

where W denotes the mini-batch size, representing the number of transitions randomly sampled from the experience replay buffer D for each training step, and n represents the index of a sampled transition from the replay buffer. The target value z_n serves as the Temporal Difference (TD) target, providing a stable learning baseline by aggregating the immediate reward with the discounted future expected return estimated by the target networks. It is expressed as below.

$$z_n = r_u^n + \gamma Q'\left(o_u^{n+1}, \mu'\left(o_u^{n+1}|\theta^{\mu'}\right)|\theta^{Q'}\right). \tag{18}$$

The Actor network is updated by maximizing the expected cumulative return through policy gradients, and its update rule is expressed as follows.

$$\nabla_{\theta^{\mu}} J \approx \frac{1}{W}\sum_{n=1}^{W}\nabla_a Q\left(o_u, a_u|\theta^{Q}\right)|_{o_u=o_u^n, a_u=\mu(o_u^n)}\nabla_{\theta^{\mu}}\mu\left(o_u|\theta^{\mu}\right)|_{o_u=o_u^n}. \tag{19}$$

Although DDPG provides continuous-action control capability, the synchronous update of strategies among multiple UAVs makes the environment highly non-stationary from the perspective of any single agent. This violates the stationarity assumption of conventional MDPs and makes independent training difficult to converge. To address this issue, a MADDPG-based multi-agent cooperative solution framework is further constructed in our study to achieve joint path planning and charging scheduling for multiple UAVs under partial observability constraints.

MADDPG follows the CTDE paradigm, which is suitable for the problem considered in our study. During training, the Critic network of each UAV not only uses its own local observation and action, but also jointly takes the observations and actions of all UAVs as input, which is expressed as below.

$$O = \left\{o_{u_1}, \ldots, o_{u_M}\right\}, A = \left\{a_{u_1}, \ldots, a_{u_M}\right\}. \tag{20}$$

In this way, the Critic can perceive the global cooperative state during training, thereby evaluating the value of joint actions more accurately and alleviating the non-stationarity problem in multi-agent environments. During execution, the Actor network of j-th UAV generates actions only based on its own local observation, without requiring high-bandwidth global communication, which is more consistent with practical online deployment scenarios of multi-UAV systems. For an agent j, the Critic network is trained by

minimizing the mean-squared error between the predicted Q-value and the target Q-value over a mini-batch of transitions sampled from the replay buffer. The loss function is defined as follows,

$$L\left(\theta_j^Q\right) = \frac{1}{W}\sum_{n=1}^{W}\left[\left(z_j^n - Q_j\left(O^n, A^n|\theta_j^Q\right)\right)^2\right] \tag{21}$$

where W denotes the mini-batch size and n represents the index of a sampled transition from the replay buffer. The symbols O^n and A^n denote the joint observation and joint action contained in the n-th sampled transition, respectively. The target value z_j^n is defined as TD target, which incorporates both the immediate reward and the estimated future return obtained from the target networks. It is computed as below.

$$z_j^n = r_{u_j}^n + \gamma Q_j^{'}\left(O^{n+1}, a_{u_1}^{'}, \ldots, a_{u_M}^{'}|\theta_j^{Q'}\right) \tag{22}$$

where $r_{u_j}^n$ denotes the reward received by the j-th UAV in the n-th sampled transition, γ is the discount factor, and $Q_j^{'}$ represents the target Critic network. The target actions $\mu_j^{'}$ using the next-state observations $o_{u_j}^{n+1}$ from the sampled mini-batch as follows,

$$a_{u_j}^{'} = \mu_j^{'}\left(o_{u_j}^{n+1}\middle|\theta_j^{\mu'}\right), j = 1, \ldots, \mathrm{M}. \tag{23}$$

The Actor network of each agent is updated by maximizing the expected cumulative return through the policy gradient method. Using the joint observations O^n and actions A^n from the replay buffer, the parameter update gradient for the Actor network of agent j is approximated as follows,

$$\nabla_{\theta_j^\mu} J \approx \frac{1}{W}\sum_{n=1}^{W}\nabla_{a_{u_j}} Q_j(O^n, A^n|\theta_j^Q)|_{a_{u_j}=\mu_j(o_{u_j}^n)}\nabla_{\theta_j^\mu}\mu_j(o_{u_j}|\theta_j^\mu)|_{o_{u_j}=o_{u_j}^n}. \tag{24}$$

In the multi-UAV cooperative energy replenishment problem studied in this paper, the advantages of MADDPG are mainly reflected in two aspects. On the one hand, the centralized Critic explicitly models the influence of other UAVs' behaviors, enabling agents to gradually learn cooperative task allocation strategies during training, thereby avoiding multiple UAVs competing for the same low-energy node. On the other hand, the decentralized execution mechanism allows each UAV to make real-time decisions based solely on its local observations, which enhances the scalability and robustness of the system. Therefore, MADDPG is adopted as the core solution framework for multi-UAV path planning and charging scheduling, aiming to maximize the global average residual energy while reducing collision risks. The detailed training and execution procedure of the proposed MCEC-MADDPG algorithm is summarized in Algorithm 1.

Algorithm 1: MCEC-MADDPG Procedure

Input: Number of UAVs M, maximum training episodes K, maximum steps per episode T, Actor network learning rate α, Critic network learning rate β, discount factor γ, soft update coefficient τ, replay buffer capacity D, mini-batch size W

Output: Trained multi-UAV policy networks $\mu = \{\mu_1, \mu_2, \ldots, \mu_M\}$

Initialize experience replay buffer D, an exploration noise process ξ;

for each agent $j = 1$ to M **do**

  Initialize actor network $\mu_j\left(o_{u_j}|\theta_j^\mu\right)$ and critic network $Q_j\left(O, a_{u_1}, \ldots, a_{u_M} \mid \theta_j^Q\right)$ with random weights;

  Initialize target actor network $\mu_j^{'}$ and target critic network $Q_j^{'}$;

(Continued)

Algorithm 1 (continued)

Set $\theta_j^{\mu'} \leftarrow \theta_j^{\mu}, \theta_j^{Q'} \leftarrow \theta_j^{Q}$;
end for;
for $episode = 1$ to K **do**
  Initialize the environment and obtain the initial joint local observations $O = \{o_{u_1}, \ldots, o_{u_M}\}$ based on the energy and positions of sensor nodes;
  for ***step t* = 1** to ***T*** **do**
    for each agent $j = 1$ to M **do**
      Each UAV selects action $a_{u_j}^t = \mu_j\left(o_{u_j}^t|\theta_j^{\mu}\right) + \xi_j^t$;
    end for;
    Execute actions $A = \{a_{u_1}, \ldots, a_{u_M}\}$ and observe reward r_u^t and next state observations o_u^{t+1};
    Store transition $\left(O^t, A^t, r^t, O^{t+1}\right)$ in D;
    Update $o_u^t \leftarrow o_u^{t+1}, O^t \leftarrow O^{t+1}$;
    if ***size*** $(\boldsymbol{D}) \geq \boldsymbol{W}$ **then**
      for each agent $j = 1$ to M **do**
      Randomly sample W transitions $\left(O^n, A^n, r^n, O^{n+1}\right)$ from D;
        for each sampled transition $n = 1$ to W **do**
          Compute target actions $a_{u_j}^{'} = \mu_j^{'}\left(o_{u_j}^{n+1}\left|\theta_j^{\mu'}\right.\right), j = 1, \ldots, M$;
          Compute target value $z_j^n = r_{u_j}^n + \gamma Q_j^{'}\left(O^{n+1}, a_{u_1}^{'}, \ldots, a_{u_M}^{'}|\theta_j^{Q'}\right)$;
        end for
          Update Critic network by minimizing $L\left(\theta_j^Q\right) = \frac{1}{W}\sum_{n=1}^{W}\left[\left(z_j^n - Q_j\left(O^n, A^n|\theta_j^Q\right)\right)^2\right]$;
        Update Actor network using the policy gradient
        $\nabla_{\theta_j^{\mu}} J \approx \frac{1}{W}\sum_{n=1}^{W}\nabla_{a_{u_j}} Q_j\left(O^n, A^n|\theta_j^Q\right)|_{a_{u_j}=\mu_j\left(o_{u_j}^n\right)} \nabla_{\theta_j^{\mu}}\mu_j\left(o_{u_j}|\theta_j^{\mu}\right)|_{o_{u_j}=o_{u_j}^n}$;
      end for
      for each agent $j = 1$ to M **do**
        Update target networks $\theta_j^{\mu'} \leftarrow \tau\theta_j^{\mu} + (1-\tau)\,\theta_j^{\mu'}, \theta_j^{Q'} \leftarrow \tau\theta_j^{Q} + (1-\tau)\,\theta_j^{Q'}$;
      end for;
    end if;
  end for;
end for;
Return trained multi-UAV policy networks $\mu = \mu_1, \mu_2, \ldots, \mu_M$;

In terms of time complexity, the algorithm requires action selection, environment interaction, and network parameter updates at each iteration. Its time complexity can be expressed as below,

$$O\left(K \cdot T \cdot W \cdot M^2\right) \tag{25}$$

where W denotes the batch size sampled from the experience replay buffer during each training update. The space complexity is mainly constrained by the storage capacity of the replay buffer D, and can be expressed as follows.

$$O\left(D \cdot M \cdot (|O| + |A|)\right) \tag{26}$$

where $|O|$ denotes the dimension of the local observation state space of a single UAV, and $|A|$ denotes the dimension of the action space of a single UAV. Since the algorithm uses the Actor-Critic architecture

and the Target Network mechanism to address non-stationarity and training instability in the multi-agent environment, it additionally requires storage space for $4M$ neural network parameter sets.

5 Performance Evaluation and Discussion

5.1 Simulation Parameters

Considering a square monitoring region of 1000 m × 1000 m, forty sensor nodes are randomly distributed in the area. To emulate the real operating state of the network, the initial energy of each node is randomly generated within the range of $[100, 600]$ J, and the energy depletion rate of each node dynamically varies within $[0.05, 0.15]$ J/s, thereby creating a highly stressed test environment characterized by energy heterogeneity and urgent task demands. The system is equipped with $M = 3$ rotary-wing UAVs, which start from the center of the region. The UAV flight speed is fixed at 15 m/s, and the effective charging distance is 10 m. To reflect the actual efficiency of wireless charging, the effective received charging power is set to 15 W, which is much higher than the node energy consumption rate, ensuring the feasibility of emergency energy replenishment. The detailed simulation parameters are listed in Table 1. All simulation experiments are conducted on a laptop equipped with an AMD 7800X3D CPU and an NVIDIA GeForce RTX 4070 GPU.

Table 1: Simulation parameters.

Parameter	Value
Network size	1000 m × 1000 m
Number of UAVs	3
Sensor node capacitor capacity	800 J
Initial energy range	$[100, 600]$ J
Sensor energy consumption	$[0.05, 0.15]$ J/s
UAV flight speed	15 m/s
Effective charging power	15 W
Effective charging radius	10 m
Actor network learning rate α	0.0001
Critic network learning rate β	0.001
Discount factor γ	0.95
Soft update coefficient τ	0.01
Replay buffer capacity D	100,000
Batch size W	256
Maximum training episodes K	5000
Maximum steps per episode T	500

It should be noted that the current simulation setting with 40 static sensor nodes and 3 UAVs is designed as a representative benchmark scenario to validate the feasibility and comparative effectiveness of the proposed MCEC-MADDPG framework. This setting allows us to clearly examine the convergence behavior, cooperative trajectory formation, residual energy improvement, throughput performance, dead-node reduction, and network survival rate under heterogeneous energy consumption. However, this setting does not fully cover all possible large-scale deployment conditions. In particular, the influence of different numbers of sensor nodes and UAVs, mobile sensor nodes, variable UAV speeds, and adaptive charging radii is not exhaustively evaluated in the current simulation. These factors will be further investigated in future scalability experiments.

To comprehensively evaluate the performance of the proposed MCEC-MADDPG, four representative algorithms are selected as comparison baselines. In the IDDPG algorithm, each UAV is treated as an independent agent and trains its own Actor and Critic networks solely based on its local observations, without any information sharing among agents [28,29]. This algorithm is used to verify the necessity of the proposed mechanism in addressing environment non-stationarity. In the Partition-Greedy strategy, the monitoring region is divided into three fixed subregions, with each UAV responsible for one subregion. Within each region, the UAV follows a greedy strategy and always prioritizes the nearest node whose energy falls below a preset threshold [22]. This algorithm represents a traditional static planning method. In the K-Means algorithm based on the classical heuristic divide-and-conquer strategy, sensor nodes are clustered according to their geographic coordinates using K-Means, and the entire network is divided into three clusters [20]. This method is used to examine the performance of static task partitioning based only on spatial distribution when facing heterogeneous node energy consumption. The Two-Layer task allocation strategy uses an upper-layer clustering mechanism to dynamically or periodically update task assignments, while the lower layer guides UAV charging operations for each assigned cluster through a TSP [21] approximate solution or heuristic path planning [19]. This algorithm represents a commonly used hierarchical divide-and-conquer optimization approach in wireless charging scheduling.

5.2 Performance Evaluation

In multi-agent deep reinforcement learning, whether an algorithm can converge stably in a complex dynamic environment is a core indicator of its feasibility. Fig. 2 shows the trend of cumulative reward for each algorithm over 5000 training episodes. It can be clearly observed that MCEC-MADDPG explores rapidly at the beginning of training and achieves fast convergence within relatively few episodes. By contrast, IDDPG, which treats each UAV as an independent agent, exhibits significant fluctuations: its cumulative reward oscillates for a long time within a wide range of [−150, 50] and fails to converge easily. This confirms that, in the absence of global information exchange, the continuous policy updates of other UAVs make the learning problem faced by a single UAV a non-stationary Markov decision process, leading to ineffective experience replay and chaotic gradient updates.

Figure 2: Convergence comparison.

To further analyze the behavioral characteristics of multiple UAVs in a continuous action space, Fig. 3 presents the projected flight trajectories of the three UAVs in the two-dimensional plane after convergence. As shown in Fig. 3, although no partition rule is predefined in the algorithm, the three UAVs spontaneously exhibit a clear spatial division of labor. This self-organized partitioning behavior is a manifestation of a multi-agent game reaching a Nash equilibrium. Through centralized training, the agents learn that entering a teammate's responsible area will create resource competition and thus reduce the overall reward. As a result, they autonomously form non-interfering patrol regions, achieving spatial load balancing and greatly improving charging coverage efficiency.

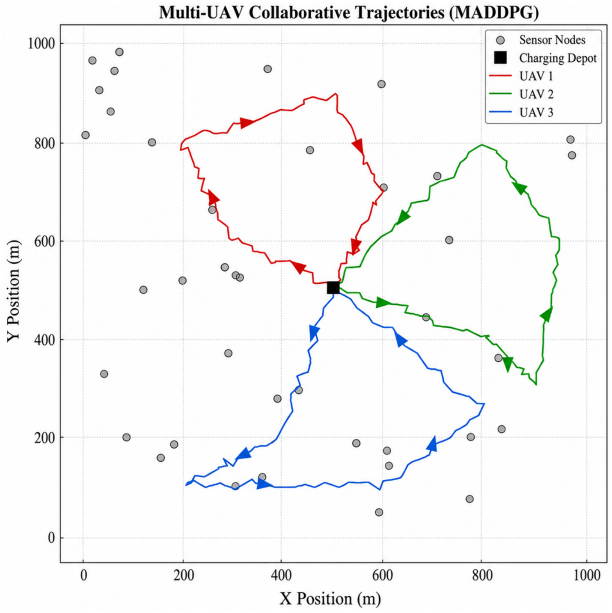

Figure 3: Cooperative flight trajectories of multiple UAVs based on MADDPG.

The core objective of MCEC-MADDPG is to maximize the average residual energy of the entire sensor network within a fixed scheduling period, thereby enhancing the network's resilience to risk. Fig. 4 illustrates the dynamic variation of the average residual energy during system operation. The experimental results show that MCEC-MADDPG achieves the best performance. Although IDDPG also maintains a relatively high energy level, its curve shows more noticeable fluctuations, indicating weaker continuity and global coordination in charging scheduling. Since the energy consumption rate of sensor nodes changes dynamically, relying solely on distance priority, static clustering, or a fixed two-layer divide-and-conquer strategy often causes UAVs to spend too much flight time in local regions or fail to respond in time to sudden high-energy-demand nodes across different regions, ultimately leading to a decline in the overall network energy level. Therefore, the traditional Partition-Greedy algorithm, the K-Means-based divide-and-conquer algorithm, and the Two-Layer algorithm perform poorly in maintaining network energy.

In highly dynamic environments, ensuring continuous and efficient data transmission is key to evaluating the quality of survival in large-scale sensor networks. Fig. 5 illustrates the network throughput comparison across different algorithms during the training process. The results demonstrate that the proposed MCEC-MADDPG algorithm consistently outperforms all baselines, achieving the highest stable throughput of approximately 75 kbps. While IDDPG also exhibits an upward learning trend, its throughput remains at a suboptimal level. This further validates that the CTDE architecture effectively mitigates multi-agent non-stationarity, allowing UAVs to coordinate efficiently and ensure nodes receive timely

charging to maintain active data transmission. Conversely, traditional static methods such as K-Means and Partition-Greedy achieve significantly lower throughput due to their lack of adaptability to dynamic network conditions.

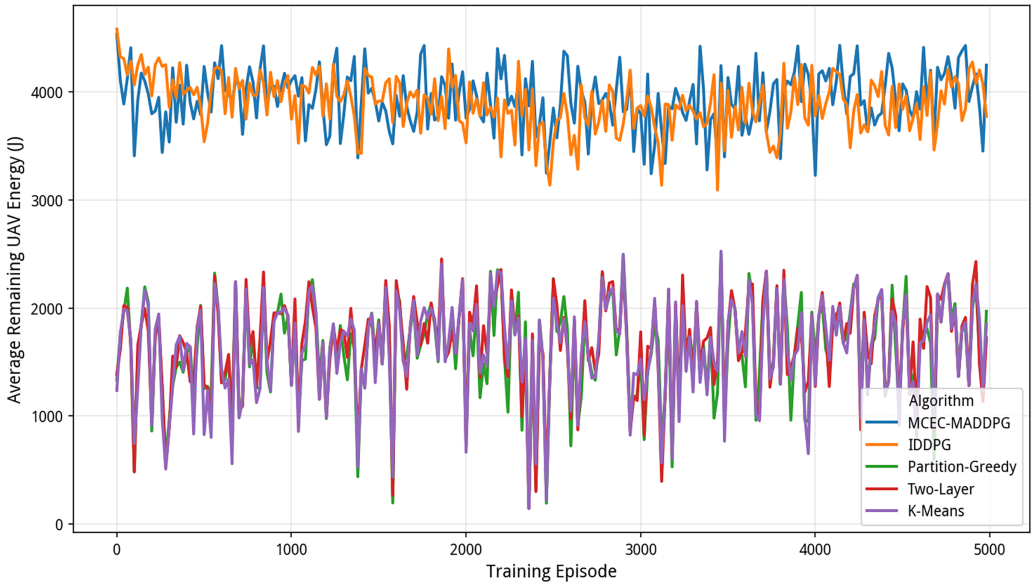

Figure 4: Comparison of average residual energy.

Figure 5: Comparison of network throughput across different algorithms.

Preventing nodes from dying due to energy depletion is the foundation of energy replenishment in wireless sensor networks. Fig. 6 shows the performance of each algorithm in terms of the number of dead nodes and network survival rate as training episodes increase. Fig. 6a shows a comparison of dead node numbers, all algorithms start with relatively high numbers of dead nodes, around 15–20, because the agents' policies are not yet mature at the early stage. As training progresses, MCEC-MADDPG demonstrates strong dynamic adaptability: the number of dead nodes decreases sharply and eventually stabilizes at around 5,

significantly outperforming the other algorithms. Fig. 6b shows a comparison of the network survival rate, the advantage of MCEC-MADDPG becomes even more evident. Its survival-rate curve rises steadily and eventually approaches 90%. In contrast, IDDPG, which cannot explicitly learn the behavioral patterns of other UAVs, is prone to resource competition and only achieves a survival rate of about 80%, while Partition-Greedy and Two-Layer remain in the range of 60%–70%. The detailed quantitative results are further summarized in Table 2.

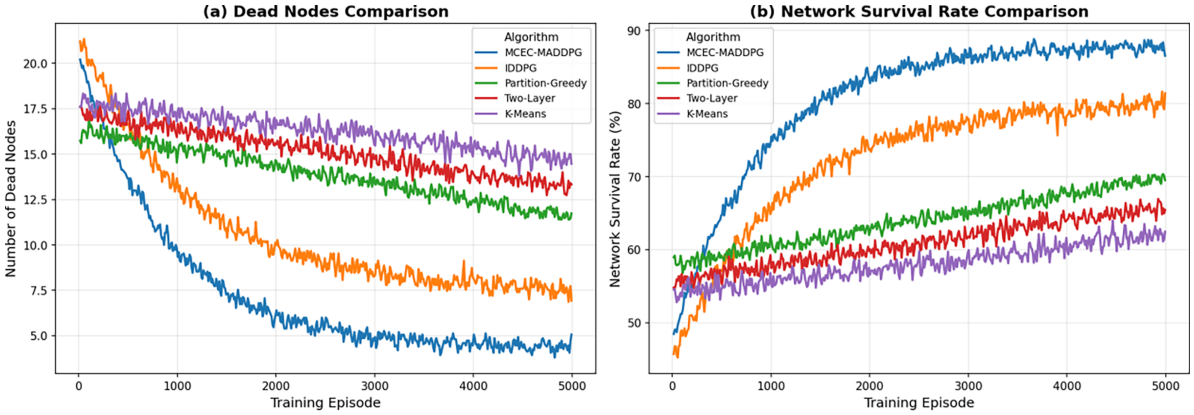

Figure 6: Comparison of dead nodes and network survival rate.

Table 2: Quantitative performance comparison of different algorithms.

	Cumulative Reward	Average Remaining Energy (J)	Throughput	Survival Rate (%)
MCEC-MADDPG	90 (±7.021)	3875 (±575.342)	732 (±12.846)	87.5 (±1.849)
IDDPG	−39 (±70.531)	3750 (±600.092)	652 (±12.506)	79.5 (±1.543)
Partition-Greedy	88 (±9.223)	1300 (±1000.245)	530 (±10.443)	69 (±1.007)
Two-Layer	77 (±15.662)	1400 (±1000.553)	495 (±10.247)	64.5 (±1.763)
K-Means	−114 (±136.457)	1350 (±1150.432)	460 (±10.398)	61 (±1.946)

6 Conclusion and Future Research Directions

To address the challenges of limited UAV endurance, insufficient charging efficiency, and heterogeneous energy demands in large-scale Battery-free SWIPT-enabled Sensor Networks, this paper proposes the MCEC-MADDPG algorithm. The proposed mechanism constructs a cooperative one-to-one wireless charging model, in which UAVs hover above target nodes to deliver energy precisely. This design avoids the need for complex beamforming and improves both the stability and efficiency of the energy replenishment process. In terms of the optimization objective, instead of minimizing flight time alone, this paper takes the maximization of the average residual energy of all nodes in the network as the core criterion, with the aim of enhancing the overall robustness and long-term operability of the network. Meanwhile, a CTDE framework is adopted to effectively alleviate the non-stationarity caused by continuously changing policies in multi-agent environments, and continuous action control together with collision avoidance mechanisms are incorporated to improve the safety and smoothness of UAV coordination.

Simulation results demonstrate that MCEC-MADDPG outperforms the comparison baselines, including IDDPG, Partition-Greedy, and Two-Layer task allocation strategy, K-Means, in terms of convergence

speed, training stability, and overall scheduling performance. The proposed method is able to maintain the average residual energy of the entire network more effectively under dynamic energy consumption conditions, significantly reduce the number of dead nodes, and improve the network survival rate to a high level, thereby validating its effectiveness and superiority in multi-UAV cooperative charging scenarios.

Nevertheless, this work still has several limitations. First, the current charging model mainly relies on a simplified path-loss-based formulation, which provides a clear and controllable basis for algorithm design but cannot fully capture all physical factors in practical UAV-enabled wireless power transfer. In real environments, the received energy may be affected by antenna gain, beamforming design, nonlinear energy harvesting characteristics, small-scale fading, UAV hovering instability, and LoS/NLoS channel conditions. Second, the current problem formulation mainly takes the maximization of network-wide average residual energy as the optimization objective. Although this metric can effectively reflect the overall energy state of BSSN and provide a stable continuous feedback signal for reinforcement learning, it cannot fully replace other important metrics, such as the number of dead nodes, network lifetime, energy fairness, and data throughput. In particular, when node energy consumption is highly heterogeneous or node distribution is uneven, simply increasing the average residual energy may not sufficiently guarantee the long-term survival of critical nodes. Third, collision avoidance in the current framework is mainly handled through reward penalties and safety-distance constraints. Although this design can discourage unsafe behaviors during policy learning, it still belongs to a soft-constraint mechanism. During the exploration stage, RL agents may still generate actions that temporarily violate safety constraints before the policy converges. Therefore, the current method cannot provide strict safety guarantees for collision-free multi-UAV coordination.

Future work will be extended in the following directions. First, a more realistic UAV-enabled WPT model will be developed by incorporating antenna gain, directional beamforming, nonlinear energy harvesting, fading effects, hovering instability, and probabilistic LoS/NLoS channel conditions, so as to improve the physical fidelity of the charging process. Second, a multi-objective optimization framework will be developed by jointly considering average residual energy, minimum residual energy, the number of dead nodes, network lifetime, fairness, and throughput, so as to construct a more comprehensive multi-UAV cooperative charging strategy. Third, systematic scalability evaluations will be conducted under different numbers of sensor nodes and UAVs. Specifically, the proposed framework will be tested under larger node densities, different UAV-team sizes, mobile-node scenarios, variable UAV speeds, and adaptive charging radii, thereby more comprehensively verifying its scalability, robustness, and applicability in large-scale dynamic BSSN scenarios. Fourth, safer multi-UAV reinforcement learning mechanisms will be investigated. Constrained reinforcement learning, control barrier functions, action shielding, safe exploration, and model-predictive safety filters can be introduced to prevent unsafe actions before execution, rather than relying only on post-action reward penalties. Fifth, UAV hovering locations can be jointly optimized with energy replenishment, task scheduling, and path planning to further improve charging accuracy and energy efficiency. Finally, hardware-in-the-loop simulations and real UAV experiments can be conducted to evaluate the actual performance of the proposed algorithm under realistic communication, computation, mobility, safety, and energy constraints, providing further support for future engineering deployment.

Acknowledgement: Not applicable.

Funding Statement: This work was supported by the National Natural Science Foundation of China (Grant No. 62461041), the Natural Science Foundation of Jiangxi Province (Grant No. 20224BAB212016), and the China Scholarship Council (Grant No. 202106825021).

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Xiangyi Le, Deyu Lin; data collection: Xiangyi Le; methodology and algorithm design: Xiangyi Le, Deyu Lin, Yufei Zhao;

simulation implementation: Xiangyi Le, Yufei Zhao; analysis and interpretation of results: Xiangyi Le, Deyu Lin, Yufei Zhao, Wang Miao; draft manuscript preparation: Xiangyi Le, Deyu Lin; manuscript review and editing: Deyu Lin, Yufei Zhao, Wang Miao, Yong Liang Guan; supervision: Deyu Lin, Yong Liang Guan. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data used to support the findings of this study are available from the corresponding author upon request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lin D, Wan J, Wang J, Kong L, Zhao Y, Guan YL. A novel topology-scale-adaptive and energy-efficient clustering scheme for energy sustainable large-scale SWIPT-enabled WSNs. IEEE Trans Mob Comput. 2024;23(12):13172–88. doi:10.1109/TMC.2024.3429235.
2. Yu Y, Lin D, Zhao Y, Su C, Wei J, Guan YL. Spatial-temporal fuzzy logic driven topological decoupling mechanism with balanced supply and demand in data and energy for BSSN. IEEE J Sel Areas Commun. 2026;44:5105–21. doi:10.1109/JSAC.2026.3699610.
3. Yetgin H, Cheung KTK, El-Hajjar M, Hanzo L. A survey of network lifetime maximization techniques in wireless sensor networks. IEEE Commun Surv Tutor. 2017;19(2):828–54. doi:10.1109/COMST.2017.2650979.
4. Yu Y, Lin D, Zhang Y, Zhao Y, Miao W, Guan YL. A queuing theory-based dynamic adaptive energy sustainability algorithm in battery-free sensor networks. In: Proceedings of the 2026 IEEE International Conference on Communications (ICC); 2026 May 24–28; Glasgow, UK. p. 1–7.
5. Xie L, Cao X, Xu J, Zhang R. UAV-enabled wireless power transfer: a tutorial overview. IEEE Trans Green Commun Netw. 2021;5(4):2042–64. doi:10.1109/TGCN.2021.3093718.
6. Ojha T, Raptis TP, Passarella A, Conti M. Wireless power transfer with unmanned aerial vehicles: state of the art and open challenges. Pervasive Mob Comput. 2023;93(9):101820. doi:10.1016/j.pmcj.2023.101820.
7. Liu Y, Dai HN, Wang Q, Shukla MK, Imran M. Unmanned aerial vehicle for Internet of everything: opportunities and challenges. Comput Commun. 2020;155(3):66–83. doi:10.1016/j.comcom.2020.03.017.
8. Xu J, Zeng Y, Zhang R. UAV-enabled wireless power transfer: trajectory design and energy optimization. IEEE Trans Wirel Commun. 2018;17(8):5092–106. doi:10.1109/TWC.2018.2838134.
9. Shi J, Cong P, Zhao L, Wang X, Wan S, Guizani M. A two-stage strategy for UAV-enabled wireless power transfer in unknown environments. IEEE Trans Mob Comput. 2024;23(2):1785–802. doi:10.1109/TMC.2023.3240763.
10. Shan T, Wang Y, Zhao C, Li Y, Zhang G, Zhu Q. Multi-UAV WRSN charging path planning based on improved heed and IA-DRL. Comput Commun. 2023;203(1):77–88. doi:10.1016/j.comcom.2023.02.021.
11. Mu J, Sun Z. Trajectory design for multi-UAV-aided wireless power transfer toward future wireless systems. Sensors. 2022;22(18):6859. doi:10.3390/s22186859.
12. Wang X, Wu P, Hu Y, Cai X, Song Q, Chen H. Joint trajectories and resource allocation design for multi-UAV-assisted wireless power transfer with nonlinear energy harvesting. Drones. 2023;7(6):354. doi:10.3390/drones7060354.
13. Zhao N, Cheng Y, Pei Y, Liang YC, Niyato D. Deep reinforcement learning for trajectory design and power allocation in UAV networks. In: Proceedings of the ICC 2020—2020 IEEE International Conference on Communications (ICC); 2020 Jun 7–11; Dublin, Ireland. p. 1–6.
14. Nguyen TT, Nguyen ND, Nahavandi S. Deep reinforcement learning for multiagent systems: a review of challenges, solutions, and applications. IEEE Trans Cybern. 2020;50(9):3826–39. doi:10.1109/TCYB.2020.2977374.
15. Oroojlooy A, Hajinezhad D. A review of cooperative multi-agent deep reinforcement learning. Appl Intell. 2023;53(11):13677–722. doi:10.1007/s10489-022-04105-y.
16. Ning Z, Xie L. A survey on multi-agent reinforcement learning and its application. J Autom Intell. 2024;3(2):73–91. doi:10.1016/j.jai.2024.02.003.

17. Zhang K, Yang Z, Başar T. Multi-agent reinforcement learning: a selective overview of theories and algorithms. In: Handbook of reinforcement learning and control. Cham, Switzerland: Springer; 2021. p. 321–84.
18. Foerster J, Nardelli N, Farquhar G, Afouras T, Torr PHS, Kohli P, et al. Stabilising experience replay for deep multi-agent reinforcement learning. In: Proceedings of the 34th International Conference on Machine Learning; 2017 Aug 6–11; Sydney, Australia.
19. Rashid T, Samvelyan M, de Witt CS, Farquhar G, Foerster J, Whiteson S. QMIX: monotonic value function factorisation for deep multi-agent reinforcement learning. In: Proceedings of the 35th International Conference on Machine Learning; 2018 Jul 10–15; Stockholm, Sweden. p. 4295–304.
20. Sasikumar P, Khara S. K-means clustering in wireless sensor networks. In: Proceedings of the 2012 Fourth International Conference on Computational Intelligence and Communication Networks; 2012 Nov 3–5; Mathura, India. p. 140–4.
21. Tran TN, Le LB, Nguyen D, Tran NH. Sensor clustering using a K-means algorithm in wireless sensor networks. Sensors. 2023;23(4):2345. doi:10.3390/s23042345.
22. Helsgaun K. An effective implementation of the Lin-Kernighan traveling salesman heuristic. Eur J Oper Res. 2000;126(1):106–30. doi:10.1016/S0377-2217(99)00284-2.
23. Lin S, Kernighan BW. An effective heuristic algorithm for the traveling-salesman problem. Oper Res. 1973;21(2):498–516. doi:10.1287/opre.21.2.498.
24. Applegate DL, Bixby RE, Chvátal V, Cook WJ. The traveling salesman problem: a computational study. Princeton, NJ, USA: Princeton University Press; 2006.
25. Xie L, Shi Y, Hou YT, Lou A. Wireless power transfer and applications to sensor networks. IEEE Wirel Commun. 2013;20(4):140–5. doi:10.1109/MWC.2013.6590061.
26. Shi Y, Xie L, Hou YT, Sherali HD. On renewable sensor networks with wireless energy transfer. In: Proceedings of IEEE INFOCOM 2011; 2011 Apr 10–15; Shanghai, China. p. 1350–8. doi:10.1109/INFCOM.2011.5934919.
27. Mnih V, Kavukcuoglu K, Silver D, Rusu AA, Veness J, Bellemare MG, et al. Human-level control through deep reinforcement learning. Nature. 2015;518(7540):529–33. doi:10.1038/nature14236.
28. Lillicrap TP, Hunt JJ, Pritzel A, Heess N, Erez T, Tassa Y, et al. Continuous control with deep reinforcement learning. In: Proceedings of the 4th International Conference on Learning Representations; 2016 May 2–4; San Juan, Puerto Rico.
29. Fujimoto S, van Hoof H, Meger D. Addressing function approximation error in actor-critic methods. In: Proceedings of the 35th International Conference on Machine Learning; 2018 Jul 10–15; Stockholm, Sweden.
30. Haarnoja T, Zhou A, Abbeel P, Levine S. Soft actor-critic: off-policy maximum entropy deep reinforcement learning with a stochastic actor. In: Proceedings of the 35th International Conference on Machine Learning; 2018 Jul 10–15; Stockholm, Sweden. p. 1861–70.
31. Lowe R, Wu Y, Tamar A, Harb J, Abbeel P, Mordatch I. Multi-agent actor-critic for mixed cooperative-competitive environments. In: Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017); 2017 Dec 4–9; Long Beach, CA, USA.
32. Sunehag P, Lever G, Gruslys A, Czarnecki WM, Zambaldi V, Jaderberg M, et al. Value-decomposition networks for cooperative multi-agent learning. In: Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems (AAMAS); 2018 Jul 10–15; Stockholm, Sweden. p. 2085–7.
33. Kaelbling LP, Littman ML, Cassandra AR. Planning and acting in partially observable stochastic domains. Artif Intell. 1998;101(1–2):99–134. doi:10.1016/S0004-3702(98)00023-X.
34. Lauri M, Hsu D, Pajarinen J. Partially observable Markov decision processes in robotics: a survey. IEEE Trans Robot. 2023;39(1):21–40. doi:10.1109/TRO.2022.3200138.
35. Ding Z, Schober R, Poor HV. Partially observable Markov decision process for UAV formation decision-making. IEEE J Sel Areas Commun. 2020;38(12):2857–70. doi:10.1109/JSAC.2020.3000827.
36. Sun J, Tang J, Lao S. Collision avoidance for cooperative UAVs with optimized artificial potential field algorithm. IEEE Access. 2017;5:18382–90. doi:10.1109/ACCESS.2017.2746752.
37. Hausknecht M, Stone P. Deep recurrent q-learning for partially observable MDPs. AAAI Fall Symp. 2015;45:141. doi:10.1016/j.ins.2022.11.065.

38. Zhu K, Zhang T. Deep reinforcement learning based mobile robot navigation: a review. Tsinghua Sci Technol. 2021;26(5):674–91. doi:10.26599/TST.2021.9010012.
39. Grondman I, Busoniu L, Lopes GAD, Babuska R. A survey of actor-critic reinforcement learning: standard and natural policy gradients. IEEE Trans Syst Man Cybern Part C Appl Rev. 2012;42(6):1291–307. doi:10.1109/TSMCC.2012.2218595.