ITC 3/55
Information Technology and Control
Vol. 55 / No. 3/ 2026
pp. 997-1019
DOI 10.5755/j01.itc.55.3.43745

Communication Optimization Method for Power Internet of Things Based on Multi-agent Deep Reinforcement Learning

Received 2025/12/02 | Accepted after revision 2026/05/11

HOW TO CITE: Hu, Y., Zhu, Y., He, X., Zhu, Y. (2026) Communication Optimization Method for Power Internet of Things Based on Multi-agent Deep Reinforcement Learning. *Information Technology and Control*, 55(3), 996-1018. https://doi.org/10.5755/j01.itc.55.3.43745

Communication Optimization Method for Power Internet of Things Based on Multi-agent Deep Reinforcement Learning

Yueshen Hu*, Xinggu He, Youwei Zhu
China Southern Power Grid EHV Transmission Company Dali Bureau, Dali, 671000, China

Yifeng Zhu
China Southern Power Grid EHV Transmission Company, Guangzhou, 510000, China

Corresponding author: ylf18866@126.com

To solve the bottleneck of power Internet of Things terminals in remote areas in terms of communication interruption, energy depletion and insufficient computing power, a Space-Air-Ground Integrated Power Internet of Things communication and computing collaborative optimization framework is constructed, and a communication optimization method based on Multi-Agent Distributed Deep Deterministic Policy Gradient (MAD3PG) is proposed. Firstly, the energy replenishment and task offloading problem is modeled as a multi-agent Markov decision-making process, and the Actor-Critic structure of centralized training and distributed execution is used to jointly optimize cross-agent trajectories, power allocation, user association and other strategies. Secondly, a hybrid offloading strategy and a shared result cache pool are introduced to reduce repeated calculations, and wireless information and energy simultaneous transmission technology are combined to collaboratively execute task transmission and energy replenishment. The results showed that when the MAD3PG was deployed in a 10km^2 area, the average communication throughput reached 28.65Mbps, the minimum user rate remained at 4.12Mbps, the total system energy consumption was only 4.82kJ, the task offloading success rate reached 99.1%, and the edge computing resource utilization rate was 85.2%. The research method can provide a set of high-efficiency, low-energy consumption and adaptive communication optimization solutions for the communication-empowered power Internet of Things, which is of great value in ensuring the long-term stable operation of new power systems with new energy as the main body.

KEYWORDS: Reinforcement Learning, Agent Model, Markov Strategy, Power Internet of Things, Communication, Task Offloading

1. Introduction

With the advancement of the "double carbon" goal, the new power system with new energy as the main body is accelerating. Distributed energy stations such as wind energy and solar energy are often deployed in remote areas. Their inherent intermittency and volatility have brought serious challenges to the safe and stable operation [13]. To achieve comprehensive and accurate perception and intelligent control of all aspects of power system generation, transmission, distribution and utilization, the deployment of large-scale Power Internet of Things (PIoT) has become key [21]. The computing power and battery capacity of PIoT devices are extremely limited, which greatly restricts high-computing tasks and threatens the long-term stable operation [11, 23]. Strengthening PIoT communication optimization and providing solutions for its computing offloading and energy supplement have become the key to the current development of the PIoT. The Space-Air-Ground Integrated PIoT (SAG-PIoT) uses Low Earth Orbit (LEO) satellites and Unmanned Aerial Vehicles (UAVs) to provide ubiquitous communication coverage and computing resources, which can effectively make up for the shortcomings of the ground network [18, 25].

Therefore, to solve the communication interruption and resource waste faced by PIoT equipment in remote areas, the research first builds a communication and computing collaborative optimization framework based on the SAG-PIoT architecture. Secondly, the research uses multi-agent Deep Reinforcement Learning (DRL) to achieve power IoT communication optimization. That is, the Multi-Agent Deep Deterministic Policy Gradient (MADDPG) is used to solve the UAV decision variable, and the observed action information is used to dynamically balance system energy consumption and task delay. The improved Multi-Agent Distributed Deep Deterministic Policy Gradient (MAD3PG) is taken to realize hybrid task offloading and decision-making selection in the shared mode to better improve the utilization efficiency of edge computing resources in the SAG-PIoT architecture. Although the unified modeling of hybrid action spaces and the integration of shared cache pools are not entirely novel paradigms in reinforcement learning, the contribution of the research lies in constructing a unified multi-agent decision-making model for joint energy-delay optimization in PIoT, encompassing discrete communication mode selection, continuous power control, and continuous UAV trajectory planning. By adaptively extending and modifying existing MARL frameworks, the study effectively addresses this highly coupled hybrid decision-making problem. Specifically, the research proposes a unified hybrid action space mapping mechanism that integrates multiple heterogeneous decision variables into an end-to-end MARL framework. Additionally, it introduces task-oriented shared cache pools as part of the state in MARL, demonstrating their effectiveness in reducing redundant computations during multi-agent collaboration. The superiority of research algorithms is rooted in their core framework of "centralized training and distributed execution". During the training phase, the centralized Critic can obtain the global state and actions of all agents, which not only effectively overcomes the problem of environmental non stationarity caused by policy interaction in multi-agent systems, but more importantly, it achieves accurate credit allocation. This enables intelligent agents to learn complex collaborative behaviors, rather than the short-sighted decisions of single agent DDQN or greedy strategies, thereby driving the system to converge towards Pareto optimality. The research aims to provide a set of efficient, reliable and adaptive communication optimization solutions for the future PIoT empowered by mobile communication technology to ensure the safe and stable operation.

2. Related Work

In response to high energy consumption and low routing efficiency in IoT wireless sensor networks, Han et al. [7] used an improved ant colony algorithm to optimize routing selection and find the optimal path by optimizing transition probability and pheromone concentration. The results showed that this method significantly improved performance on energy consumption, delay and network throughput [7]. In view of the resource constraints of the air-ground integrated heterogeneous PIoT, Qin et al. explored the task offloading, and decomposed the original problem through Lyapunov optimization, effectively reducing the energy consumption of PIoT equipment [15]. Faced with energy consumption caused by node failures and

route changes in wireless sensor networks supporting the Internet of Things (IoT), Vazhuthi et al. designed a hybrid adaptive neuro-fuzzy inference system reptile optimization algorithm to identify optimal routes. This method reduced energy consumption to 0.01J at a thousand-level node scale [22]. To solve the power supply of large-scale IoT nodes, Portilla et al. explored the potential of using large-area electronic technology to develop wireless power supply solutions, providing a technical path for realizing environmentally friendly and battery-free IoT devices [14].

To optimize the efficiency of federated learning in the wireless IoT, Chen et al. built a framework to optimize communication efficiency and resources, which improved model training efficiency while protecting data privacy through client scheduling and power bandwidth allocation [3]. To analyze the energy efficiency of device-to-device communication, Su et al. optimized UAV positioning and resource allocation, and achieved significant improvement in system energy efficiency under the Simultaneous Wireless Information and Power Transfer (SWIPT) mechanism [19]. To solve the energy consumption, Al-Abiad et al. proposed a resource allocation scheme that combined IoT, power allocation and computing frequency, which greatly reduced the overall energy consumption of the federated learning process [1]. Liu et al. coordinated the scheduling and trajectory of multiple UAVs in IoT services through fair energy efficiency resource optimization, ensuring the balance of energy efficiency distribution [9]. To solve the spectrum scarcity and energy constraints, Guo et al. combined cognitive IoT with radio frequency energy harvesting to propose a transmission algorithm based on DRL to maximize long-term uplink throughput [6]. Na et al. analyzed the physical layer security of UAV relay communication, and effectively improved the minimum average confidentiality rate and communication fairness by optimizing UAV trajectories and resource allocation [12].

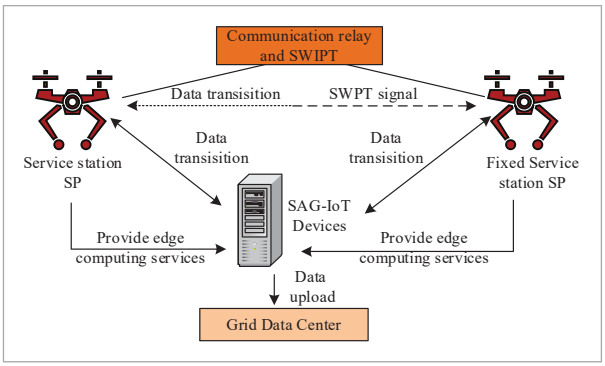
In recent years, research on IoT communication optimization has mainly focused on routing, task offloading, and energy management. For example, physical layer energy efficiency models and wireless information energy simultaneous transmission technology, multi-agent reinforcement learning, and intelligent algorithms such as Harris Eagle and Gray Wolf are used to achieve task offloading and power communications. Most work focuses more on improving energy consumption efficiency and mission success rate in static topology and stable channel environments, without taking into account real-time decision-making under "highly dynamic topology, random channel, and strong energy-computing coupling". There are challenges such as insufficient strategy coordination under cross-layer communication coupling, weak training stability, and low robustness in dynamic environments. Therefore, to take into account communication fairness, energy efficiency and task delay, a communication optimization method for the PIoT based on MADDPG is proposed, and energy supply and task offloading are discussed and analyzed, aiming to provide a better reference for the development of the PIoT.

3. Design of Communication Optimization Method for PIoT

3.1. Design of Multi-agent Collaborative Framework for SAG-PIoT

The energy supplement of PIoT equipment is a key bottleneck that restricts its long-term stable operation in remote areas. The traditional method is susceptible to weather changes, and its stability and sustainability are difficult to guarantee. Therefore, depending on the flexible and easy-to-deploy characteristics of UAV, equipped with SWIPT technology, it provides reliable wireless power supply and communication relay services for PIoT devices in remote areas. Figure 1 presents the PIoT device.

Figure 1
PIoT device architecture.



From Figure 1, the framework includes a fixed Service Point (SP), a group of UAVs and a group of ground PIoT devices. SP provides SWIPT services (charging) and edge computing services for UAV, while UAV provides energy supplement and data relay for PIoT devices.

The study assumes that the system contains M UAVs and N PIoT devices, whose sets are expressed as $V=\{1,2,...,M\}$ and $U=\{1,2,...,N\}$, respectively. Each PIoT device $u \in U$ is fixed on the power facility, its position is $L_u=(x_u, y_u, z_u)$, and the position of the SP is $L_s=(x_s, y_s, z_s)$. All UAV equipment flies at a fixed altitude H. The total task cycle T_{total} is divided into T discrete and equal-length gaps, the set is $T=\{1,2,...,T\}$, and the length of each time slot is $\delta_t=T_{total}/T$. Since δ_t is small enough, the position of the UAV within a single time slot remains unchanged. Therefore, under the time slot $t \in T$, the 3D spatial coordinates of the UAV is presented in Equation (1) [24].

$$L_v(t)=(x_v(t), y_v(t), H). \tag{1}$$

In Equation (1), $(x_v(t),y_v(t))$ is the horizontal coordinate of UAV v. The distances between the UAV v and the PIoT device u and the SP are $D_{u,v}(t)=\sqrt{\|L_u-L_v(t)\|^2}$ and $D_{s,v}(t)=\sqrt{\|L_s-L_v(t)\|^2}$, respectively. To avoid PIoT communication being limited by terrain, the study considers Line-of-Sight (LS) and Non-Line-of-Sight (NLS) transmission, and adopts the Rician fading channel model. The channel gain $h_{u,v}(t)$ between UAV and PIoT devices is shown in Equation (2) [20].

$$h_{u,v}(t)=\frac{\beta_0}{D_{u,v}(t)^{\alpha}}\left(\sqrt{\frac{K}{K+1}}h_{LoS}+\sqrt{\frac{1}{K+1}}h_{nLoS}\right). \tag{2}$$

In Equation (2), β_0 signifies the channel power gain at a reference distance of 1m. α signifies the path loss index. K signifies the Rice factor. h_{LoS} and h_{nLoS} are the channel responses of LS and NLS components. In time slot t, if the UAV transmits signals to the PIoT device, the signal-to-interference-to-noise ratio at the receiving end is presented in Equation (3).

$$\lambda_{u,v}(t)=\frac{P_v(t)h_{u,v}(t)}{\sum_{k\in V,k\neq v}P_k(t)h_{u,k}(t)+\sigma^2}. \tag{3}$$

In Equation (3), $P_v(t)$ is the transmit power of the UAV. $P_k(t)$ is the transmitting power of the transmitter. σ^2 signifies the additive. According to Shannon's formula, the outgoing transmission rate of UAVs can be expressed as Equation (4) [8, 2, 16].

$$C_{u,v}(t)=W\log_2(1+\lambda_{u,v}(t)). \tag{4}$$

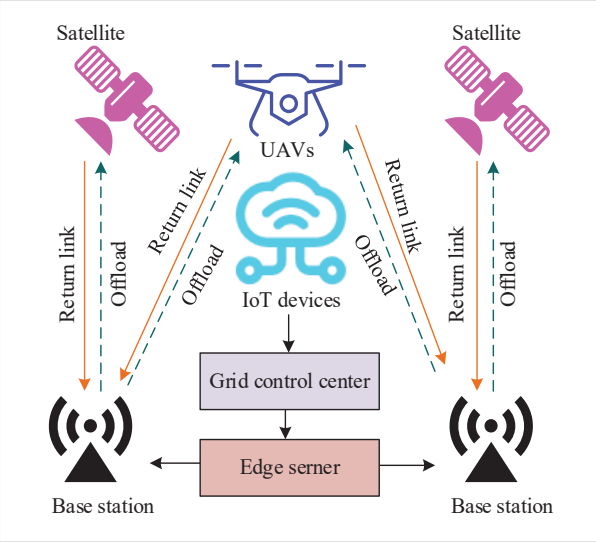
In Equation (4), W is the channel bandwidth. PIoT devices generate computing tasks within time slots, which will generate corresponding transmission energy consumption during local calculation and offloading. Therefore, under the premise of satisfying various constraints, the study jointly optimizes the user association strategy $A=\{a_{u,v}(t)\}$, UAV transmission power $P=\{P_v(t)\}$ and power allocation strategy $R=\{\rho_v(t)\}$ to maximize the weighted sum of the system's lowest average user rate and global energy efficiency. The optimization problem is expressed as Equation (5).

$$\max_{A,P,R}\omega_1(\min\frac{1}{T}\sum_{t=1}^{T}C_{us}^{eff}(t))+\omega_2(\frac{1}{T}\sum_{t=1}^{T}GEE(t)). \tag{5}$$

The objective function shown in Equation (5) balances communication fairness and system energy efficiency goals. $C_{us}^{eff}(t)$ is the effective communication rate of user us in the time slot. $GEE(t)$ signifies the global energy efficiency of the system under the time slot. T signifies the quantity of time slots in the total task cycle. ω_1 and ω_2 are weight factors. PIoT devices face limited network coverage, insufficient computing power, and difficulty in energy acquisition. Therefore, the research uses LEO satellites and UAVs as aerial computing and communication platforms to design a SAG-PIoT device architecture [10]. Figure 2 presents the SAG-PIoT architecture.

In Figure 2, the architecture consists of LEO satellites, multiple UAVs, ground base stations and massive PIoT devices. To avoid UAV collision, it is necessary to ensure that its flight trajectory meets basic constraints such as collision avoidance, speed limit and periodic return. Considering that PIoT devices are mostly deployed in open areas, and the channel between UAV and PIoT devices is mainly LS, the study uses the free space path loss model to calculate the transmission energy, and uses orthogonal frequency division multiple access to eliminate PI-

Figure 2
Schematic diagram of the SAG-PIoT architecture.



oT-UAV link interference [17]. Equation (6) is the transmission rate $C_{d,u}(t)$ from PIoT device d to UAV u in the PIoT-UAV link.

$$C_{d,u}(t) = W_{UAV} \log_2\left(1 + \frac{P_d^{tx} g_{d,u}(t)}{\sigma_0^2}\right). \quad (6)$$

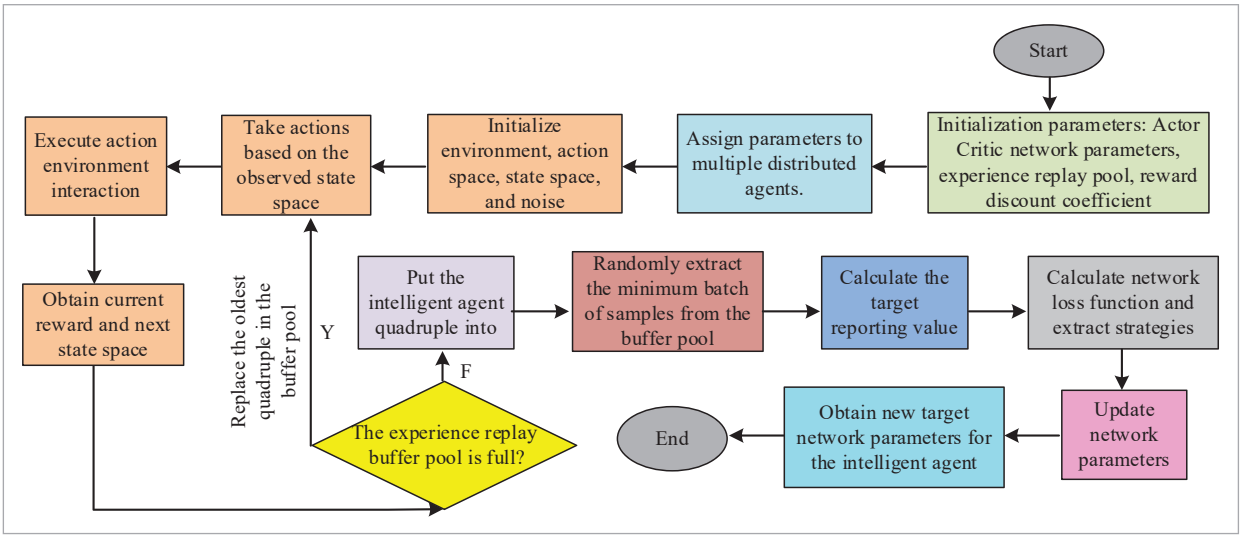
In Equation (6), W_{UAV} signifies the channel bandwidth of UAV. P_d^{tx} signifies the transmit power. σ_0^2 is the noise power. $g_{d,u}(t)$ is the path loss index under LS.

The core of a cache pool is the ability to uniquely identify and quickly retrieve tasks and their results. $TaskID=(Device_{ID},Task_{Type},Input_{Hash})$ vague task description can lead to cache pollution and incorrect hits.

3.2. Communication Optimization for PIoT Based on Multi-agent Deep Learning

Considering that the decision variables faced in the UAV collaborative process are relatively complex, to better handle the objective function, the research attempts to model the problem as a Markov Decision Process (MDP) and solve it based on the MADDPG. Each UAV exerts as an agent, interacts with the environment and learns the optimal strategy. The MADDPG employs a "centralized training, distributed execution" framework. Each agent v includes an Actor and a Critic. During the training phase, the experience tuples (s,a,r,s') of all agents (where s, a and s' are the joint states, actions and next states of all agents, and r is the immediate reward) are stored in a shared experience pool. Data are sampled from the pool to update the network. After training is completed, during the execution phase, each UAV agent only needs to use its own Actor to make independent decisions based on local observations to achieve efficient distributed control. Su et al. proposed attention loss adjustment priority experience replay and direct attention loss adjustment priority experience replay, using error compensation and incentive mechanisms

Figure 3
The process of MADDPG algorithm.



to improve training performance [4, 5]. Figure 3 is a schematic flow of the MADDPG algorithm.

From Figure 3, the MADDPG algorithm first initializes the Actor-Critic and the experience pool. The agent selects noisy actions based on observations, interacts with the environment and stores experience. When the experience pool is full, the oldest experience will be replaced with a fixed strategy. Subsequently, a mini-batch of samples is obtained from the pool, the critic loss and the actor policy gradient are calculated, and the network parameters are updated to stabilize the training process.

To collaboratively optimize complex energy replenishment and complete task offloading decisions, this study further improves the MADDPG algorithm and proposes to use the MAD3PG algorithm to model and solve the mixed task offloading. Specifically, a shared offloading strategy is designed to ensure that each PIoT device can share the result pool data. A structured task ID generation rule was designed in the study, where when a PIoT device generates a task, its task ID is defined as a triplet<DeviceID, TaskType, DataHash>. Among them, DeviceID is the unique identifier of the device that generates the task; TaskType is the task type identifier, and DataHash is a fixed length string obtained by hashing the input data of the task. Only when the device source, task type, and input data of two tasks are exactly the same, are they considered to be the same task. After the task is successfully unloaded, the sequence number of the task unit will be matched with the cache pool and stored in the sequence pool based on the matching result. The data and calculation results of the power Internet of Things have strong timeliness, so the study adopts an adaptive Time To Live (TTL) setting mechanism based on task types, which achieves better data matching and cleaning by setting different adaptive TTL for different tasks. The study uses a hybrid offloading strategy to analyze different situations and establish a unified cost model for each decision considering delay and energy consumption. If the task is processed locally, the execution delay $T_{d,k}^{loc}$ and energy consumption $E_{d,k}^{loc}$ can be expressed as Equation (7).

$$T_{d,k}^{loc} = \frac{S_k w_k}{F_d}, E_{d,k}^{loc} = \kappa_d (F_d)^{co} T_{d,k}^{loc} . \tag{7}$$

In Equation (7), F_d is the CPU frequency of the PIoT device. κ_d is the effective capacitance coefficient of the chip. co signifies a constant. k is the task generated by the PIoT device. S_k and w_k are the data volume and computing density of PIoT device tasks. The comprehensive cost of local calculation is $\mathrm{Cos}t_{d,k}^{loc} = \lambda_t T_{d,k}^{loc} + \lambda_e E_{d,k}^{loc}$, where λ_t and λ_e are the weighting factors of time and energy, respectively. When tasks are offloaded, there are two modes. One is binary offloading. The entire task k is completely offloaded to UAV or LEO. The total delay $T_{d,u,k}^{bin}$, total energy consumption $E_{d,u,k}^{bin}$ and total cost $\mathrm{Cos}t_{d,k}^{bin}$ of offloading to UAV can be expressed as Equation (8).

$$\begin{cases} T_{d,u,k}^{bin} = T_{d,u,k}^{tx} + T_{u,k}^{comp} \\ E_{d,u,k}^{bin} = E_{d,u,k}^{tx} + E_{u,k}^{comp} \\ \mathrm{Cos}t_{d,k}^{bin} = \lambda_t T_{d,u,k}^{bin} + \lambda_e E_{d,u,k}^{bin} \end{cases} \tag{8}$$

In Equation (8), $T_{d,u,k}^{tx}$ and $E_{d,u,k}^{tx}$ are the transmission delay and energy consumption. $T_{u,k}^{comp}$ and $E_{u,k}^{comp}$ are edge computing delay and computing energy consumption. When there is partial offloading, task k is divided, a proportion of data of β_k is offloaded, and the remaining part $(1-\beta_k)$ is calculated locally in parallel. Then, its total delay and total energy consumption are $T_{d,u,k}^{part} = \max\{T_{d,k}^{loc}(1-\beta_k), T_{d,u,k}^{tx}(\beta_k) + T_{u,k}^{comp}(\beta_k)\}$ and $E_{d,u,k}^{part} = E_{d,k}^{loc}(1-\beta_k) + E_{d,u,k}^{tx}(\beta_k) + T_{u,k}^{comp}(\beta_k)\}$. The cost of partial offloading can be obtained by weighting the total delay and total energy consumption.

To optimize the offloading problem, the goal is to minimize the total cost of all PIoT, which is the sum of the task processing cost and the wireless power transmission service cost. The decision variables include task offloading mode, offloading target, charging request, etc. Therefore, the optimization problem is displayed in Equation (9).

$$\max_{\Phi} \sum_{t \in T} \sum_{d \in D} (\cos t_d^{task}(t) + \mathrm{Cos}t_d^{wpt}(t)) . \tag{9}$$

In Equation (9), Φ is the set of all decision variables. $cost_d^{task}(t)$ and $Cost_d^{wpt}(t)$ are the task processing cost and the wireless power transmission service cost. This problem is a complex mixed integer non-convex optimization problem, which is difficult to solve by traditional optimization methods. The joint optimization problem is transformed into a multi-agent

MDP model (S, A, P, R, γ), where the state space S signifies the local observation state of the agent PIoT device (including residual energy and channel gains for all available UAVs) under the time slot. The action space A includes the charging decision *ChargeReq*, the unloading decision *OffloadTarget* and the unloading mode *OffloadMode*, and its action $a_d(t)$ is presented in Equation (10).

$$a_d(t) = \{ChargeReq, OffloadTarget, OffloadMode\}. \quad (10)$$

The goal of all agents is to minimize the total cost, so the reward function is taken to be the negative value of the total cost, at which time the reward is displayed in Equation (11).

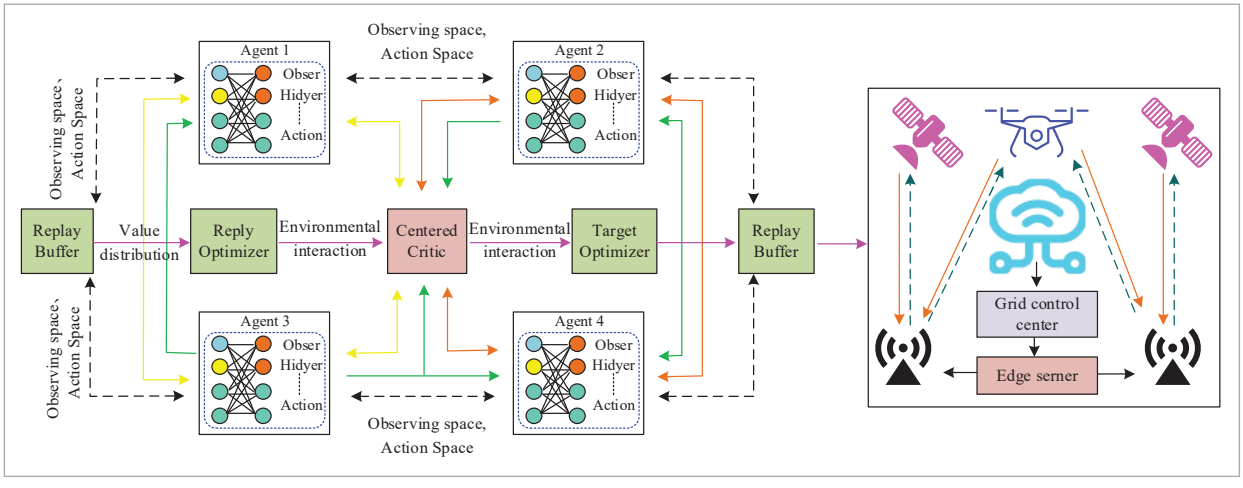
$$r(t) = -(\sum_{d \in D} \cos t_d^{task}(t) + \mathrm{Cos} t_d^{wpt}(t)) + Bonuses / Penalties. \quad (11)$$

In Equation (11), *Bonuses* is the reward term. *Penalties* is the penalty item. Mixed action space: The study formalizes the mixed action space A_i of UAV agent i as a Cartesian product: $A_i = A_i^d + A_i^c$. The discrete action space A_i^d contains task offloading decisions, while the continuous action space A_i^c contains communication power and flight speed/direction. Considering that the action space defined by the MAD3PG algorithm is a mixed space containing continuous variables (transmission power) and discrete variables (computing offloading decisions, channel selection), the mapping mechanism is studied to clarify that for each agent, its Actor network outputs a continuous action vector. The first element of the vector is directly activated by the Sigmoid function and multiplied by the maximum transmission power to obtain the final continuous transmission power. Afterwards, discrete action processing was performed, dividing the latter half of the vector into channel selection and two parts. The argmax function was applied to each of these two sub vectors to achieve the mapping from continuous output to discrete selection. The Actor network is designed to have two parallel output heads: one uses the Gumbel Softmax technique to output the probability distribution of discrete actions, and the other directly outputs continuous action values (scaled to a valid range using the tanh function). This design unifies heterogeneous actions under the same policy gradient update framework. Figure 4 is a schematic diagram of the MAD3PG architecture.

In Figure 4, the MAD3PG algorithm uses the Actor-Critic framework of centralized training and distributed execution, and outputs a value distribution in the Critic network, which can more accurately model the uncertainty of the environment. Multiple parallel Actors interact with the environment to produce experience data, which is uniformly stored in a global experience pool to implement network updates. To overcome the inherent training instability of the MADDPG algorithm, a small coefficient τ=0.01 is used to slowly update the parameters of the target network, i.e. $\theta' \leftarrow \tau\theta + (1-\tau)\theta'$ to suppress overestima-

Figure 4

Schematic diagram of MAD3PG architecture.



tion of the target Q value and reduce the oscillation of policy updates. At the same time, a delay strategy is adopted to ensure that the update frequency of the Actor network and the target Actor network is lower than that of the Critic network. Table 1 shows the pseudocode content of the research algorithm.

Table 1
Pseudo code content of MAD3PG algorithm.

MAD3PG algorithm
Initialization: For each intelligent agent i=1,..., N:
Initialize Actor network μ_i and its parameter θ_i
Initialize Critic network Q_i and its parameter ϕ_i
Initialize the target Actor network μ'_i and its parameters $\theta'_i \leftarrow \theta_i$.
Initialize the target Critic network Q'_i and its parameters $\phi'_i \leftarrow \phi_i$.
Initialize the replay buffer D with a capacity of M.
for episode=1 to $\max_{episodes}$ **do**
Initialize the environment and obtain the initial state s=(o′1,..., o′N)
Initialize random noise N for exploration
for t=1 to $\max_{episodes}$steps **do**
// **Environmental interaction and data collection**
For each intelligent agent i, select the action $a'_i=\mu_i(o'_i)$ based on the current observation o'_i
Execute the joint action a=(a_1,..., a_N) of all agents in the environment.
Obtain joint rewards $r=(r_1,..., r_N)$ and the next state $s=(o'_1,..., o'_N)$.
Store the experience quadruple (s, a, r, s′) into the experience replay pool D
Update current status: s←s'.
// **Centralized training network**
if the sample size in D is sufficient **then**
Randomly sample K samples (s^j, a^j, r^j, s^j) from a mini batch of D.
for agent i=1 to N **do**
// **Update Critic Network**
Calculate the next action predicted by the target netwodrk, $a'^k=\mu'^k(o'^j_k)$.
Calculate the target Q value: $y^j=r^{ij}+\gamma*Q'^i(s^j, a'^1,..., a'^N)$
Update Critic Network by Minimizing Loss Function
// **Update Actor Network**
// Update Actor Network with Sampling Strategy Gradient
end for
// **Update all target networks**
for agent i = 1 to N **do**
$\phi'_i \leftarrow \tau*\phi_i+(1-\tau)*\phi'_i$。
$\Theta'_i \leftarrow \tau*\theta_i+(1-\tau)*\theta'_i$。
end for
end if
end for
end for

4. Testing and Application on Communication Optimization Results of PIoT

4.1. Experimental Environment Parameter Settings and Indicator Selection

To deal with the core problems such as communication interruption, energy depletion and computing power bottlenecks faced by PIoT in remote areas, the performance of the proposed MAD3PG power IoT communication optimization algorithm is evaluated. Table 2 presents the experimental parameters.

The communication physical layer modeling uses MATLAB R2023b and NS-3 joint simulation framework to build a simulation platform. The experimental scenario is a power IoT deployment area in a remote mountainous area of 10km × 10km, including 6 UAVs, 120 ground PIoT devices, one SP and one LEO. PIoT device task requests follow Poisson arrival distribution (average λ=1.5/s). The MAD3PG is compared with Lyapunov Optimization (LO), Robust Energy-Efficiency Maximization Resource Allocation Algorithm based on Convex Optimization (ROBEEM), Cluster Cooperative Jamming Strategy based on Improved Genetic Algorithm (CCJS-IGA), Improved Genetic Algorithm (IGA), IGWO algorithm, Independent DDPG (IDDPG) and Multi-Agent PPO (MAPPO) algorithms. Heuristic algorithm classes are used as a reference for theoretical performance upper bounds, assuming that all global and future information can be obtained for offline optimization. They cannot ensure the consistency of hyperparameters with DRL benchmark algorithms, but can ensure that the basic parameter settings of similar algorithms are consistent, such as learning rate, iteration times, etc. The performance of the algorithm in terms of energy efficiency, task offloading success rate, latency fairness, UAV energy recovery balance and network throughput rate is evaluated. The processing power or reward mechanism of the system is positively correlated with the task load, and the average reward value can be obtained at higher task throughput. To verify the robust-

Table 2
Experimental related parameters.

Main server configuration	**CPU: Intel Xeon Gold 6430 (32 cores, 2.1GHz) Memory: 128GB**	**Energy harvesting efficiency**	**0.75**
Node configuration	Each node is equipped with 2 x NVIDIA RTX A6000 GPUs (48GB of video memory)	Task size	Uniform distribution [0.3, 2.5] MB
operating system	Ubuntu 22.04 LTS	Initial energy state	Beta (3, 2)
programming language	Python 3.10	Rice factor	6 dB
dependency library	TensorFlow 2.15, PyTorch 2.3	Path loss index	2.2
UAV altitude	150-250 m	Task generation probability	0.7
Maximum speed of UAV	20 m/s	Single task data volume	[0.5, 5] MB
Time gap interval	1 second	Calculate the number of cycles	[500, 1,500] Mcycle/bit
Total time gap length	1,000	Initial energy of PIoT (J)	[5, 20] J
Number of neurons per layer	256	Learning rate	0.001
activation function	ReLU	Discount factor	0.95
Optimizer	Adam	Target network update cycle	100
Experience pool capacity	1,000,000	Batch processing size	512

Table 3
Parameter sensitivity analysis.

Hyperparameter	Test value	Average system energy efficiency (bits/Joule)	Hyperparameter	Test value	Average system energy efficiency (bits/Joule)
Learning rate	0.01	1.65×10^7	Discount factor	0.90	1.69×10^7
	0.001	1.85×10^7		0.95	1.79×10^7
	0.0001	1.78×10^7		0.99	1.85×10^7
Batch Size	64	1.80×10^7	Number of hidden layer neurons	128	1.75×10^7
	128	1.83×10^7		256	1.85×10^7
	256	1.85×10^7		512	1.81×10^7

ness of the algorithm and the impact of key parameters on system performance, sensitivity analysis was conducted on the learning rate and task arrival rate of the Actor network. The results are shown in Table 3.

In Table 3, the model performing best when the value is 0.001. As the discount factor increases from 0.90 to 0.99, the system's energy efficiency steadily improves. The batch size of 256 achieves the best balance between the accuracy of gradient estimation and convergence efficiency, with an appropriate model capacity of 256 neurons.

4.2. SAG-PIoT Equipment System Communication and Transmission Performance Analysis

Core communication performance is the primary indicator to measure the effectiveness of the SAG-PIoT device system. The study first evaluated the system average communication throughput rate and the minimum user transmission rate, as presented in Table 4. To ensure the scientific validity of the conclusions, all performance indicators were based on the mean of multiple independent experiments, and p-values were calculated using a two sample t-test compared to the research method (MAD3PG) to quantify the statistical significance of its performance advantages.

From Table 4, the average communication throughput rate of the MAD3PG method reached 28.65Mbps, which was 10.7% higher than that of the sub-optimal MAPPO algorithm and 63.3% higher than that of the traditional LO method. More importantly, the data in the p-value column shows that MAD3PG has extremely high statistical significance (p<0.01 or

Table 4
Comparative analysis of core communication performance.

Algorithm	Average communication throughput rate (Mbps)	Minimum user transmission rate (Mbps)	Rate fairness index (Jain's)	Uplink utilization rate (%)	Downlink utilization rate (%)
MAD3PG	28.65 ± 0.41	4.12 ± 0.25	0.95 ± 0.01	89.5 ± 1.5	92.3 ± 1.1
MAPPO	25.88 ± 0.62 (< 0.001)	3.51 ± 0.34 (< 0.01)	0.91 ± 0.02 (< 0.001)	84.1 ± 2.1	86.7 ± 1.9
IDDPG	21.43 ± 1.15 (< 0.001)	2.15 ± 0.51 (< 0.001)	0.78 ± 0.04 (< 0.001)	75.3 ± 3.5	78.4 ± 3.2
CCJS-IGA	23.17 ± 0.89 (< 0.001)	2.98 ± 0.42 (< 0.001)	0.85 ± 0.03 (< 0.001)	80.2 ± 2.8	81.5 ± 2.6
IGA	22.56 ± 0.95 (< 0.001)	2.81 ± 0.45 (< 0.001)	0.83 ± 0.03 (< 0.001)	78.6 ± 3.0	80.1 ± 2.9
IGWO	22.95 ± 0.91 (< 0.001)	2.92 ± 0.44 (< 0.001)	0.84 ± 0.03 (< 0.001)	79.4 ± 2.9	80.9 ± 2.8
ROBEEM	18.91 ± 1.42 (< 0.001)	1.88 ± 0.65 (< 0.001)	0.71 ± 0.05 (< 0.001)	68.7 ± 4.2	70.2 ± 4.0
LO	17.54 ± 1.55 (< 0.001)	1.65 ± 0.71 (< 0.001)	0.67 ± 0.06 (< 0.001)	65.1 ± 4.8	67.9 ± 4.5

p<0.001) in terms of average throughput, minimum user rate, and fairness index compared to all other comparison algorithms. Table 5 shows the average total task delay and task offloading success rate, where the total task delay includes transmission delay, queuing delay and calculation delay.

In Table 5, the MAD3PG algorithm compresses the average total task latency to 85.6 ms, with latency standard deviations of 6.8 ms and 14.2 ms for the MAPPO and LO methods. The MAD3PG algorithm has a task offload success rate of 99.1%. The p-values in the table (all p<0.001) provide strong statistical evidence that MAD3PG's leading advantage in these two core task processing indicators is extremely significant. The study evaluated the single step average decision delay and model parameter size of each algorithm, and the results are shown in Table 6.

In Table 6, the MAD3PG method achieved an extremely low average decision delay of 18.5 ms, and the p-value test (p<0.001) showed that this delay advantage is extremely significant. Traditional methods such as LO (Local Offloading) have the lowest decision latency (5.2 ms), but this is almost at the cost of sacrificing all communication and task processing performance. To better evaluate the performance of the research method, it was compared with

Table 5
Comparative analysis of task processing performance.

Algorithm	Average total task delay (ms)	Task offloading success rate (%)	Task timeout rate (%)	Proportion of local computing tasks (%)	Edge computing task proportion (%)
MAD3PG	85.6 ± 3.5	99.1 ± 0.2	0.9 ± 0.2	21.5± 0.2	78.5 ± 2.1
MAPPO	102.4 ± 6.8 (< 0.001)	97.8 ± 0.4 (< 0.001)	2.2 ± 0.4	26.8± 0.4	73.2 ± 3.5
IDDPG	145.2 ± 10.1 (< 0.001)	92.3 ± 0.9 (< 0.001)	7.7 ± 0.9	38.1± 0.9	61.9 ± 5.1
CCJS-IGA	121.7 ± 8.5 (< 0.001)	95.6 ± 0.6 (< 0.001)	4.4 ± 0.6	31.5± 0.6	68.5 ± 4.2
IGA	128.9 ± 9.2 (< 0.001)	94.9 ± 0.7 (< 0.001)	5.1 ± 0.7	33.2± 0.7	66.8 ± 4.6
IGWO	124.5 ± 8.8 (< 0.001)	95.2 ± 0.6 (< 0.001)	4.8 ± 0.6	32.4± 0.6	67.6 ± 4.4
ROBEEM	168.3 ± 12.5 (< 0.001)	88.7 ± 1.2 (< 0.001)	11.3 ± 1.2	45.3± 1.2	54.7 ± 6.2
LO	179.6 ± 14.2 (< 0.001)	86.5 ± 1.5 (< 0.001)	13.5 ± 1.5	49.6± 1.5	50.4 ± 6.8

Table 6
Analysis of online computational complexity and decision latency.

Algorithm	Average Decision Delay (ms)	p value (vs. MAD3PG)	Model parameter quantity (in millions)
MAD3PG	18.5 ± 1.2	-	1.85
MAPPO	21.2 ± 1.8	< 0.05	2.10
IDDPG	15.8 ± 1.1	< 0.05	1.25
CCJS-IGA	35.7 ± 2.5	< 0.001	1.73
IGA	41.3 ± 3.1	< 0.001	1.56
IGWO	38.9 ± 2.9	< 0.001	1.89
ROBEEM	55.4 ± 4.5	< 0.001	1.64
LO	5.2 ± 0.5	< 0.001	1.31

Table 7
Comparative performance of different communication optimization algorithms.

Indicators	MAD3PG	DDPG	D3QN	CERL-SSA	AS-MPCA	MOIBBO-CNN-LSTM	KF-ADR
Avg. System Throughput (Mbps)	28.65	26.50	25.15	21.50	20.88	19.40	16.25
Avg. Task Processing Delay (ms)	85.6	98.5	110.2	135.8	142.1	155.4	180.5
Total System Energy Consumption(kJ)	4.82	5.58	6.21	5.90	6.85	7.52	8.10
Task Offloading Success Rate (%)	99.1	98.2	97.4	94.5	95.1	92.0	88.6

Dueling Double Deep Q-Network (D3QN), Kalman Filter based Adaptive Data Rate (KF-ADR), Deep Deterministic Policy Gradient (DDPG), Adaptive Scheduling Algorithm based on Multi Population Co evolution (AS-MPCA), Hybrid CNN-LSTM with MOIBO optimization (MOIBBO-CNN-LSTM), and Chaos and Elite Reverse Learning Enhanced Sparrow Search Algorithm (CERL-SSA) was compared, and the comparison results are shown in Table 7.

In Table 7, the MAD3PG method achieved the highest average system throughput (28.65 Mbps) and task success rate (99.1%), while minimizing delay (85.6 ms) and energy consumption (4.82 kJ). The DDPG method exhibited a 15% increase in latency and a 16% increase in energy consumption, highlighting the performance penalties associated with the lack of direct collaboration. The D3QN method with discrete action space performed poorly in continuous control tasks, with a latency of 29% higher than the MAD3PG method. To evaluate the fairness of different algorithms in resource allocation more deeply and intuitively, the study further calculated the distribution of cumulative service volume obtained by all PIoT devices throughout the entire simulation cycle. The research algorithm was compared with three benchmark algorithms: Random, Greedy, and Double Deep Q-Network (DDQN). The results are shown in the Table 8.

In Table 8, the MAD3PG algorithm performs the best in overall service level and balance, with the highest average service volume (85.2) and the lowest standard deviation (5.8) and coefficient of variation (0.068) among all algorithms, indicating its high and extremely uniform resource allocation efficiency. To more intuitively evaluate the fairness of different algorithms in resource allocation, the cumulative service distribution obtained by each PIoT device throughout the simulation cycle was studied and statistically analyzed, and Table 9 was obtained.

Table 8
Distribution Statistics of Cumulative Service Volume for PIoT Devices Under Different Algorithms.

Statistical Metric	MAD3PG	DDQN	Greedy	Random
Mean Service Volume	85.2	76.5	68.3	45.1
Median Service Volume	84.5	75.0	62.1	44.8
Standard Deviation	5.8	12.3	18.9	15.5
Coefficient of Variation	0.068	0.161	0.277	0.344
(Min Service Volume	72.1	55.4	30.5	10.2
Max Service Volume	96.3	98.1	105.7	80.3
Jain's Fairness Index	0.991	0.953	0.902	0.856

Table 9
Distribution statistics of cumulative service volume of PIoT devices under different algorithm.

Non statistical indicators	MAD3PG	DDQN	Greedy	Random
Average service volume	85.2	76.5	68.3	45.1
standard deviation	5.8	12.3	18.9	15.5
coefficient of variation	0.068	0.161	0.277	0.344
Jain's Fairness Index	0.991	0.953	0.902	0.856

In Table 9, the service standard deviation and coefficient of variation of MAD3PG algorithm are the lowest among all algorithms, and Jain's fairness index is the highest, which proves that through collaborative strategy, it achieves optimal resource allocation fairness while pursuing overall system performance, effectively avoiding the "Matthew effect".

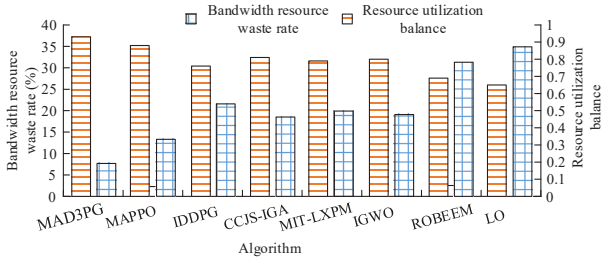
4.3. System Resource Utilization and Application Robustness Analysis of SAG-PIoT Equipment

Efficient resource utilization is the key to reducing system operating costs and improving service capabilities. The study evaluated the utilization performance of the comparison algorithm, as presented in Figure 5.

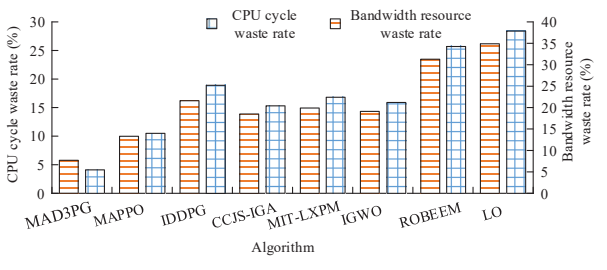
In Figure 5(a), the MAD3PG algorithm achieves an edge computing resource waste rate of 85.2% with a utilization balance of 0.93, indicating that the computational load is efficiently distributed to each edge node. In Figure 5(b), the CPU cycle waste rate (4.1%) and bandwidth resource waste rate (7.7%) shown by the MAD3PGalgorithm were much smaller than those of other algorithms. Afterwards, the energy consumption and transmission rate of the above algorithms in the agent are compared and analyzed, as shown in Figure 6.

In Figure 6(a), the MAD3PG algorithm consistently maintains the highest value, peaking at about 1.92×10^5%, significantly outperforming MAPPO (1.75×10^5%) and CCJS-IGA (1.68×10^5%). The energy efficiency of the ROBEEM algorithm and LO does not exceed 1.30×10^5%. In Figure 6(b), MAD3PG has the highest average transfer rate of about 1.85×10^3 bit/s, followed by MITLXPM (1.72×10^5bit/s) and MAPPO (1.68×10^5bit/s), while the IGWO, ROBEEM, and LO algorithms all have an average rate of no more than 1.5×10^3bit/s. The fundamental reason why the success rate of energy replenishment cannot reach 100% is the endogenous conflict between physical and service constraints, that is, the flight time and charging time of drones are rigid. In some extreme cases, when multiple remote devices simultaneously request charging, the drone cannot arrive before all devices have depleted their power.

Figure 5
Comparison results of utilization performance of different algorithms.



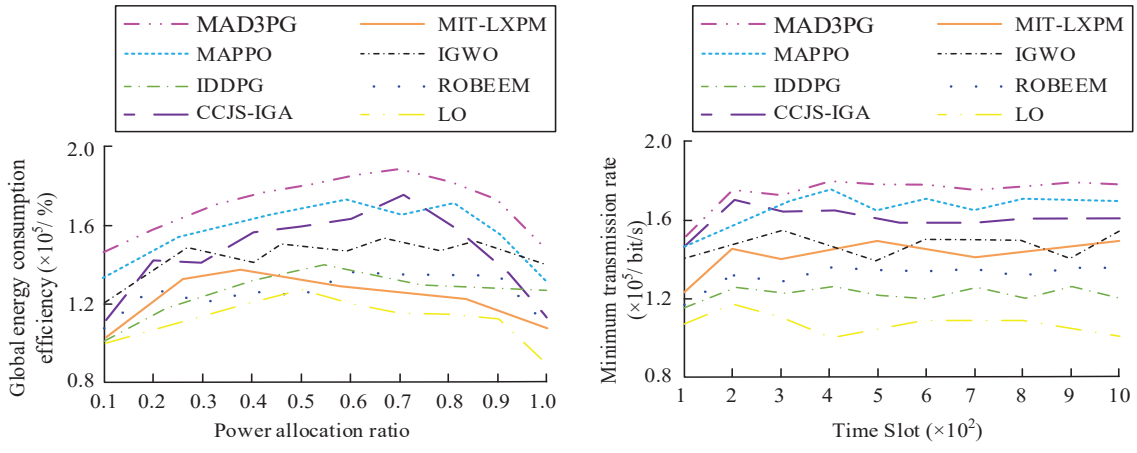
(a) Edge computing resource utilization ratio and resource utilization balance



(b) CPU cycle and bandwidth resource waste rate

Figure 6

Energy consumption efficiency and transmission rate of different algorithms.

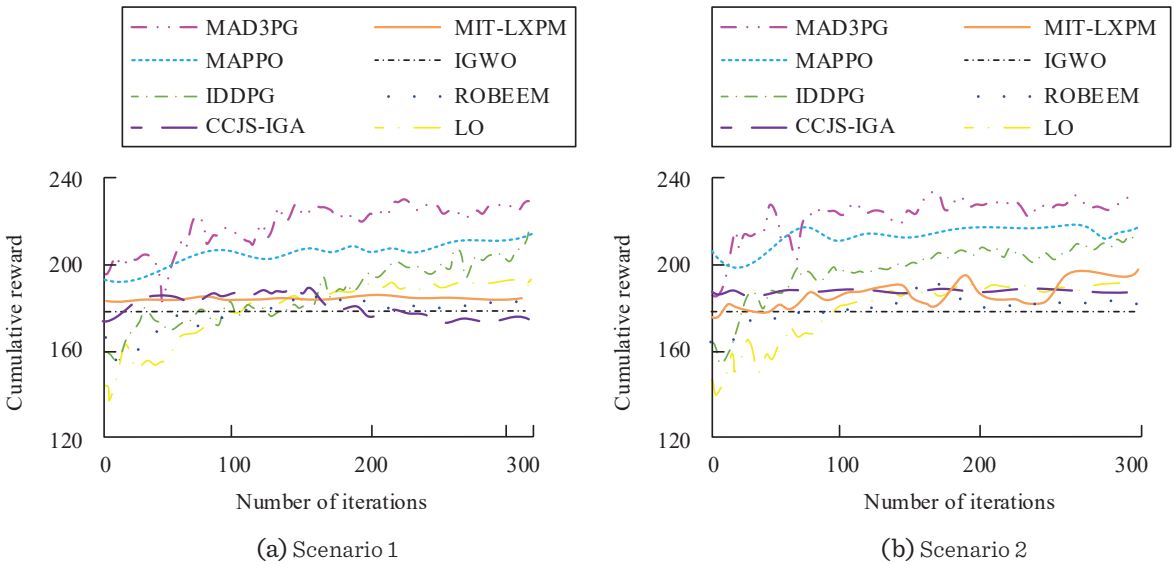


(a) Energy consumption efficiency under different power divisions

(b) Minimum transmission rate results of different algorithms

Figure 7

Convergence performance of different algorithms in power grid transmission environment.



(a) Scenario 1

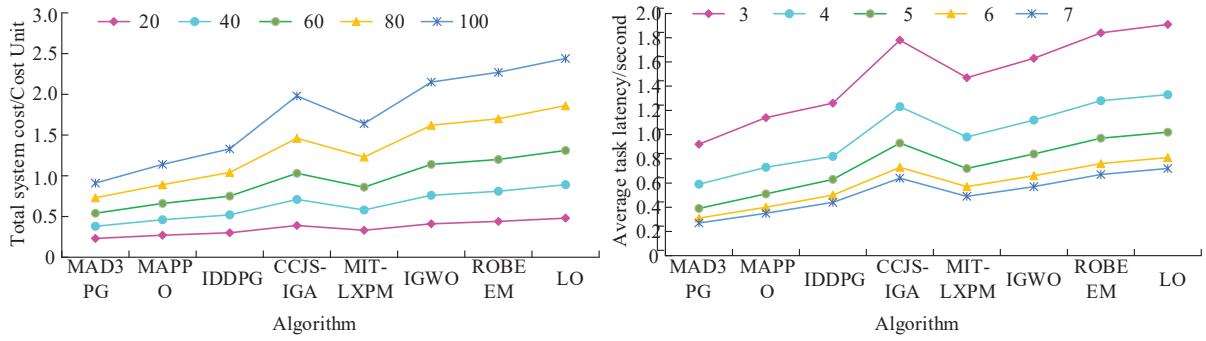
(b) Scenario 2

Then, the convergence performance of different algorithms in the power grid transmission environment was compared in two scenarios (scenario 1: early morning, scenario 2: noon). The higher the cumulative reward, the better the iterative selection decision. The results are shown in Figure 7.

In Figure 7(a), the MAD3PG algorithm had the highest reward value, while the IGWO, ROBEEM and LO algorithms converged slowly, and the MAPPO algorithm tended to converge after reaching 200 times. In Figure 7(b), the maximum reward value of the MAD4PG MO in scenario 2 exceeded

Figure 8

Algorithm robustness under different numbers of PIoT devices and UAVs.



(a) The impact of different numbers of PIoT devices on the total system cost

(b) The impact of different UAV numbers on average task latency

200, which was faster than that of the MAPPO algorithm. The CCJS-IGA and MIT-LXPM algorithm had slower convergence speeds and lower reward values. Later, to verify the robustness of the MAD-3PG when the network scale was expanded, the algorithm performance under different numbers of PIoT devices and UAVs was compared. The results are shown in Figure 8.

In the results of Figure 8(a), the growth curve of the MAD3PG algorithm is the most stable as the devices increase. In a high-load scenario, with 100 PoT devices, the system cost is approximately 0.9 cost units, which is significantly lower than other comparison methods. The initial 'upward' stage is due to the agent gradually learning how to balance communication power consumption and computational benefits through exploration, effectively offloading tasks to edge nodes with good channel conditions, thereby improving energy efficiency. Entering the 'stable' stage indicates that the agent's strategy has converged and achieved Pareto optimality in energy efficiency under the current system load and resources. In the results of Figure 8(b), the delay of MAD3PG is about 0.9 seconds when resources are limited and there are only 3 drones, while the average task delay of the LO algorithm is 1.8 seconds or more. Table 10 presents the performance scalability under changes in the number of PIoT devices and UAVs.

In the results in Table 10, when the PIoT device is boosted to 100, the average latency of the MAD-3PG algorithm remains at 115.8 ms, and the uninstall success rate remains at 98.3%, which is better than other comparison algorithms. As the number of drones increased from 3 to 7, the MAD3PG increased its average throughput from 19.8Mbps to 30.1Mbps, improving energy efficiency by nearly 50%. Average latency dropped from 132.5ms to 78.3ms, and the uninstall success rate remained at 99.5%. Afterwards, in order to better test the performance of the algorithm, the complexity, stability, and security of the algorithm in conventional environments were studied, and a probabilistic obstacle model was introduced in the simulation. The communication link had a certain probability of instantaneous obstruction, and the channel switched from Rayleigh channel to Rayleigh channel to simulate the dynamic changes in channel quality caused by small displacements in real environments, that is, the robustness results of the algorithm in degraded environments. Table 11 was obtained.

As shown in Table 11, MAD3PG shows extremely high system stability and operational safety, with a load volatility of only 0.15, which is much lower than other comparison methods, and its UAV collision avoidance safety rate is 99.98%, indicating that it can effectively learn safe cooperative flight strategies and integrate safety distance constraints into the reward function or state space. In a deteriorating channel environment, the MAD3PG method proposed in the study has the smallest performance degradation, with an average throughput decrease

Table 10
Scalability of performance under changes in the number of PIoT devices and UAVs.

Changes in PIoT Devices quantity				
PIoT Devices quantity	**Algorithm**	**Average delay (ms)**	**Offloading success rate (%)**	**Minimum speed (Mbps)**
20	MAD3PG	65.1	99.8	5.81
	MAPPO	78.3	99.2	4.95
	IGWO	95.2	98.1	4.12
40	MAD3PG	85.6	99.1	4.12
	MAPPO	102.4	97.8	3.51
	IGWO	124.5	95.2	2.92
100	MAD3PG	115.8	98.3	3.05
	MAPPO	148.7	95.5	2.24
	IGWO	183.1	90.7	1.83
Changes in the number of UAV				
UAV quantity	**Algorithm**	**Average throughput rate (Mbps)**	**Offloading success rate (%)**	**Energy efficiency (Mbit/J)**
3	MAD3PG	19.80	97.5	4.15
	MAPPO	17.20	94.8	3.21
	LO	11.50	82.3	1.58
4	MAD3PG	23.55	98.2	4.91
	MAPPO	20.81	96.1	3.90
	LO	14.12	84.0	1.85
5	MAD3PG	26.78	98.8	5.52
	MAPPO	23.95	97.2	4.45
	LO	16.33	85.6	2.06
6	MAD3PG	28.65	99.1	5.94
	MAPPO	25.88	97.8	4.70
	LO	17.54	86.5	2.18
7	MAD3PG	30.15	99.4	6.25
	MAPPO	27.32	98.3	5.03
	LO	18.61	87.2	2.31

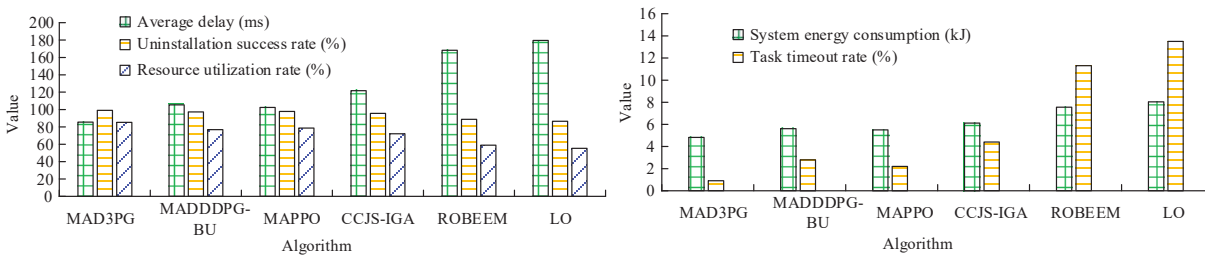
Table 11

Performance results of algorithms in different environments.

Algorithm	Stability and Safety in Conventional Environments						Robustness in degraded environments	
	System stability index (load fluctuation rate)	UAV collision avoidance safety rate (%)	Average task queue length	Network congestion rate (%)	PIoT device dropout rate (%)	UAV return charging on-time rate (%)	Average throughput (Mbps)	Drone collision safety rate (%)
MAD3PG	0.15	99.98	2.1	1.2	0.3	99.5	24.11	99.98
MAPPO	0.23	99.91	3.5	2.8	0.9	98.7	20.87	99.91
IDDPG	0.41	98.50	7.8	8.5	3.1	92.1	18.34	98.50
CCJS-IGA	0.31	99.65	5.2	4.9	1.8	96.3	19.12	99.65
IGA	0.34	99.58	5.9	5.7	2.2	95.8	19.05	99.58
IGWO	0.32	99.61	5.5	5.2	2.0	96.1	20.04	99.61
ROBEEM	0.52	99.12	10.3	12.4	4.8	90.5	20.11	99.12
LO	0.58	99.01	11.8	14.1	5.6	88.9	13.98	99.01

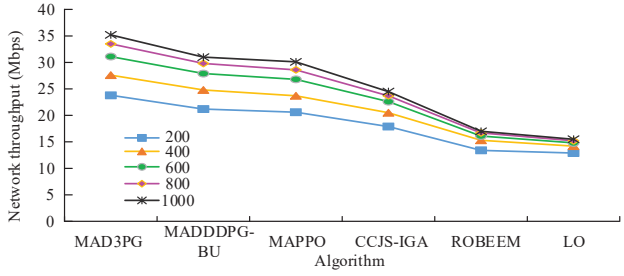
Figure 9

Benefit results of mixed unloading.



(a) Average delay, uninstallation success rate and resource utilization rate

(b) System energy consumption and task timeout rate



(c) Network throughput under different time slots

of only 15.8%, and the unloading success rate still remains at a high level of 97.5%, indicating that the collaborative adaptive learning mechanism of MAD-3PG can effectively dynamically compensate for unpredictable environmental challenges and has higher stability.

To quantify the difference between the hybrid offloading strategy (allowing partial offloading) and the binary offloading strategy (only local or complete offloading), the study limited the offloading mode of MAD3PG to binary offloading (Multi-Agent Distributed Deep Deterministic Policy Gradient-Binary offloading, MADDDPG-BU). Figure 9 shows the benefit results of hybrid unloading.

In Figure 9(a), the MAD3PG algorithm performed best, with an average delay of only about 85.6ms, which was far lower than that of the LO algorithm (about 179.3ms). In Figure 9(b), the task timeout rate of MAD3PG was controlled at around 1%, showing extremely high reliability. The timeout rates of ROBEEM and LO soared to about 11% and 13%, respectively, reflecting their inefficiency and instability when handling complex tasks. In Figure 9(c), the maximum throughput of MAD3PG was 35Mbps. To further analyze the scalability of the algorithm, the performance was compared under changes in the number of PIoT devices and drones, and the results are shown in Table 12.

In Table 12, as the system scale expands, the performance of all algorithms decreases due to intensified resource competition, with the MAD3PG algorithm studied exhibiting the best scalability. When the scale increased from (20,4) to (60,8), the energy efficiency of MAD3PG algorithm only decreased by 21.9%, far better than the 46.4% reduction of heuristic algorithm CCJS-IGA, and the task delay increase was significantly lower than that of the comparison algorithm. Quantitative analysis is conducted through ablation experiments to examine the independent contribution of each core component. The experimental results are shown in Table 13.

Table 12
Scalability of Performance under Changes in the Number of PIoT Devices and Drones.

System scale (number of PIoT devices, number of drones)	algorithm	Average system energy efficiency (×10^7 bits/Joule)	Average task completion delay (ms)
(20, 4)	MAD3PG	2.15	75.3
	MAPPO	1.98	88.1
	CCJS-IGA	1.81	95.6
(40, 6)	MAD3PG	1.85	85.6
	MAPPO	1.62	102.4
	CCJS-IGA	1.35	121.7
(60, 8)	MAD3PG	1.68	98.2
	MAPPO	1.39	125.5
	CCJS-IGA	0.97	158.3

Table 13
Core component ablation experiment and statistical significance test.

Algorithm configuration	Average energy efficiency (bits/Joule)	*p*-value (vs. MAD3PG)[1]	Average task completion delay (s)	*p*-value (vs. MAD3PG)[1]
MAD3PG	1.85×10^7	1.25	4.82	85.2
MAD3PG w/o Cache Pool	1.62×10^7 (p=0.008)[1]	1.58 (p=0.011)[1]	5.71	73.1
MAD3PG w/o Mixed Action	1.41×10^7(p=0.002)[1]	1.89 (p=0.003)[1]	5.35	79.5
MADDPG	1.29×107 (p<0.001)[1]	2.15 (p<0.001)[1]	—	—

Note: The "p^1" value represents statistical significance compared to the complete MAD3PG model, calculated based on a two sample t-test, and $p<0.05$ is considered significant.

In Table 13, compared with the core benchmark MADDPG, the proposed MAD3PG model achieved statistically significant performance improvements in energy efficiency and latency ($p<0.001$). After removing the 'shared result cache pool', the average energy efficiency of the model significantly decreased to 1.62×10^7, the average latency significantly increased by 0.33s, the total energy consumption increased by 18.5%, and the edge resource utilization rate decreased by 14.2%. This indicates that the caching mechanism plays a key role in reducing system energy consumption and latency, and improving resource efficiency by avoiding duplicate calculations. After removing the 'mixed action mapping mechanism', there was also a statistically significant deterioration in performance metrics. The average energy efficiency decreased to 1.41×10^7, the latency increased by 0.64s, and the total energy consumption and resource utilization also performed poorly. This confirms that the hybrid action space endows intelligent agents with more refined control capabilities, which is a necessary condition for achieving high energy efficiency and low latency decision-making.

4. Conclusion

This study constructed a SAG-PIoT multi-agent collaboration framework and proposed the MAD3PG algorithm to optimize PIoT communication. The average communication throughput of MAD3PG reaches 28.65Mbps, and the minimum user rate is guaranteed to exceed 4.12Mbps. The average speed of MAD-3PG is the highest, about 1.85×10^3 bit/s, followed by MITLXPM (1.72 × 105 bit/s) and MAPPO (1.68 × 105 bit/s). In high load scenarios with 100 PIoT devices, the system cost of MAD3PG algorithm is approximately 0.9 cost units, significantly lower than other comparison methods. The hybrid offload and shared cache mechanism makes the resource utilization rate of edge computing reach 85.2%, which is higher than the comparison algorithm. Although there is still a gap in latency for some extreme power protection and control tasks that require sub millisecond response. However, the average single step decision delay of MAD3PG algorithm is 18.5ms, which is much lower than the second level iterative optimization time required by traditional heuristic algorithms, and fully meets the real-time requirements of most task scheduling and network resource management scenarios in the power Internet of Things (usually in the hundreds of milliseconds). The research method transforms complex optimization problems into lightweight online reasoning through offline training, ensuring high performance and practical deployment value. In terms of research methods, the definition of state space (such as PIoT device energy, specific channel models), the construction of action space (such as task offloading mode for power data), and the weights of each component in the reward function are all tailored to solve SAG-PIoT communication optimization problems. The multi-agent DRL method (MAD3PG) used for collaborative optimization of trajectories, resource allocation, and discrete service selection under centralized training and distributed execution paradigms, especially its idea of handling mixed action spaces and learning collaborative mechanisms through shared resource models, is universal. It can be adapted to other multi robot or distributed resource management scenarios, such as smart logistics, smart agriculture, or public safety networks. The fundamental reason for the superior performance of research algorithms lies in the deep application of multi-agent reinforcement learning theory to overcome the key challenges of collaborative decision-making. The "centralized training, decentralized execution" framework allows each agent to evaluate joint actions using global information during training, effectively solving the problem of environmental non stationarity caused by policy changes of other agents. This enables intelligent agents to escape from local optima and learn true global collaborative strategies, thus achieving breakthroughs in throughput and fairness. In addition, the DDPG mechanism enables precise control of continuous variables such as drone position and power, endowing the algorithm with high robustness in dynamic environments and ensuring its superior performance and scalability.

Although the study has verified the effectiveness of the algorithm, there are still some limitations and areas for improvement: firstly, the simulation environment is relatively idealized and fails to fully consider the complex electromagnetic interference, weather

changes, multipath fading and other channel effects in the real physical world, and lacks a collaborative obstacle avoidance mechanism between unmanned aerial vehicles. Secondly, the system design ignored the data synchronization latency and overhead in real networks, and did not delve into data security and privacy protection issues in depth. Again, the scalability and deployment strategy of the algorithm in larger or heterogeneous drone clusters need to be validated. Therefore, future research will focus on: 1) constructing channel and energy consumption models based on measured data, or building semi physical simulation platforms for verification; 2) Introducing lightweight messaging mechanisms such as MQTT to achieve efficient cache consistency maintenance; 3) Explore more scalable multi-agent algorithms and design collaboration and reward mechanisms for heterogeneous clusters to address more complex real-world deployment challenges.

Meanwhile, integrating richer dynamic information into the state space is expected to significantly enhance the decision-making foresight and strategy robustness of intelligent agents. For example, introducing historical channel information will transform intelligent agents from simple "reactive" decisions to "predictive" decisions; Perceiving the state of neighboring intelligent agents can promote more efficient collaboration; Accurate energy consumption prediction helps intelligent agents plan long-term tasks. However, the expansion of this state space also brings serious challenges, with the most direct impact being a sharp increase in the state dimension. Therefore, how to design an information rich and learning efficient state representation that strikes a balance between improving decision quality and controlling training costs is a core challenge and important direction for future research.

Appendix A

Table of Formula Symbols.

Formula symbol	Meaning
$L_v(t)$	The three-dimensional spatial coordinates of UAV
H	Height
t	Time slot
$(x_v(t), y_v(t))$	Horizontal coordinate of UAV
$h_{u,v}(t)$	Channel gain between UAV and PIoT devices
β_0	Channel power gain at a reference distance of 1 meter
α	Path loss index
K	Rice factor
h_{LoS}	Channel response of line of sight component
h_{nLoS}	Channel response of non line of sight components
$\lambda_{u,v}(t)$	Signal to interference and noise ratio of the receiving end of UAV transmitting signals to PIoT devices
$P_v(t)$	Launch power of UAV
$P_k(t)$	Transmitting power of the transmitter
σ^2	Additive Gaussian White Noise Variance at the PIoT Device End
$C_{u,v}(t)$	Transmission rate
W	Channel bandwidth
$C_{us}^{eff}(t)$	Effective communication rate of users in time slots
$GEE(t)$	Global energy efficiency of the system in time slots

Formula symbol	Meaning
T	The number of time slots in the total task cycle
ω_1, ω_2	weight factor
$C_{d,u}(t)$	The transmission rate of PIoT devices to UAVs
W_{UAV}	Channel bandwidth of UAV
P_d^{tx}	Transmission power of PIoT devices
σ_0^2	noise power
$g_{d,u}(t)$	Path loss index under line of sight
$L(\theta_v^Q)$	The loss function of Critic network
$T_{d,k}^{loc}$	Delay in task execution
$E_{d,k}^{loc}$	Energy consumption of tasks
F_d	CPU frequency of PIoT devices
κ_d	Effective capacitance coefficient of chip
co	Constant
k	Tasks generated by PIoT devices
S_k	The data volume of PIoT device tasks
w_k	The computational density of PIoT device tasks
$\mathrm{Cost}_{d,k}^{loc}$	The comprehensive cost of local computing is
λ_t	Time weighting factor
λ_e	Energy weight factor
$T_{d,u,k}^{bin}$	Total delay of task offloading to UAV
$E_{d,u,k}^{bin}$	Total energy consumption of task offloading to UAV
$T_{d,u,k}^{tx}$	Transmission delay
$E_{d,u,k}^{tx}$	Transmission energy consumption
$T_{u,k}^{comp}$	Edge computing delay
$E_{u,k}^{comp}$	Edge computing energy consumption
Φ	The set of all decision variables
$\mathrm{cost}_d^{task}(t)$	The cost of task processing expenses
$\mathrm{Cost}_d^{wpt}(t)$	Cost of wireless power transmission service
ChargeReq	Charging Decision
OffloadTarget	Unloading Decision
OffloadMode	Uninstall mode
$a_d(t)$	The actions of intelligent PIoT devices
Bonuses	Reward items
Penalties	Penalty term

References

1. Al-Abiad, M. S., Hassan, M. Z., Hossain, M. J. Energy-Efficient Resource Allocation for Federated Learning in NOMA-Enabled and Relay-Assisted Internet of Things Networks. IEEE Internet of Things Journal, 2022, 9, 24736-24753. https://doi.org/10.1109/JIOT.2022.3194546
2. Ari, I. A Low Carbon Pathway for the Turkish Electricity Generation Sector. Global Landscape and Climate Economy, 2023, 1, 147-153. https://doi.org/10.47852/bonviewGLCE3202552
3. Chen, H., Huang, S., Zhang, D., Mao, M., Skoglund, M., Poor, H. V. Federated Learning over Wireless IoT Networks with Optimized Communication and Resources. IEEE Internet of Things Journal, 2022, 9, 16592-16605. https://doi.org/10.1109/JIOT.2022.3151193
4. Chen, Z., Li, H., Wang, Z. Directly Attention Loss Adjusted Prioritized Experience Replay. Complex & Intelligent Systems, 2025, 11, 267. https://doi.org/10.1007/s40747-025-01852-6
5. Chen, Z., Li, H., Yan, B. Efficient Training Framework for Multi-USV System Based on Off-Policy Deep Reinforcement Learning. Information Technology and Control, 2025, 54, 1159-1177. https://doi.org/10.5755/j01.itc.54.4.40496
6. Guo, S., Zhao, X. Deep Reinforcement Learning Optimal Transmission Algorithm for Cognitive Internet of Things with RF Energy Harvesting. IEEE Transactions on Cognitive Communications and Networking, 2022, 8, 1216-1227. https://doi.org/10.1109/TCCN.2022.3142727
7. Han, H., Tang, J., Jing, Z. Wireless Sensor Network Routing Optimization Based on Improved Ant Colony Algorithm in the Internet of Things. Heliyon, 2024, 10, e23577. https://doi.org/10.1016/j.heliyon.2023.e23577
8. Li, Y., Qu, W., Zhang, Z. Intelligent Algorithm Operation and Data Management of Electromechanical Engineering Power Communication Network Based on the Internet of Things. Scalable Computing: Practice and Experience, 2024, 25, 3330-3341. https://doi.org/10.12694/scpe.v25i5.3087
9. Liu, X., Liu, Z., Lai, B., Peng, B., Durrani, T. S. Fair Energy-Efficient Resource Optimization for Multi-UAV Enabled Internet of Things. IEEE Transactions on Vehicular Technology, 2022, 72, 3962-3972. https://doi.org/10.1109/TVT.2022.3219613
10. Liu, Y., Jiang, L., Qi, Q., Xie, S. Energy-Efficient Space-Air-Ground Integrated Edge Computing for Internet of Remote Things: A Federated DRL Approach. IEEE Internet of Things Journal, 2022, 10, 4845-4856. https://doi.org/10.1109/JIOT.2022.3220677
11. Mamaghani, M. T., Hong, Y. Aerial Intelligent Reflecting Surface-Enabled Terahertz Covert Communications in Beyond-5G Internet of Things. IEEE Internet of Things Journal, 2022, 9, 19012-19033. https://doi.org/10.1109/JIOT.2022.3163396
12. Na, Z., Ji, C., Lin, B., Zhang, N. Joint Optimization of Trajectory and Resource Allocation in Secure UAV Relaying Communications for Internet of Things. IEEE Internet of Things Journal, 2022, 9, 16284-16296. https://doi.org/10.1109/JIOT.2022.3151105
13. Nagajayanthi, B. Decades of Internet of Things Towards Twenty-First Century: A Research-Based Introspective. Wireless Personal Communications, 2022, 123, 3661-3697. https://doi.org/10.1007/s11277-021-09308-z
14. Portilla, L., Loganathan, K., Faber, H., Eid, A., Hester, J. G. D., Tentzeris, M. M., Fattori, M., Cantatore, E. Wirelessly Powered Large-Area Electronics for the Internet of Things. Nature Electronics, 2023, 6, 10-17. https://doi.org/10.1038/s41928-023-00927-x
15. Qin, P., Fu, Y., Zhao, X., Wu, K., Liu, J., Wang, M. Optimal Task Offloading and Resource Allocation for C-NOMA Heterogeneous Air-Ground Integrated Power Internet of Things Networks. IEEE Transactions on Wireless Communications, 2022, 21, 9276-9292. https://doi.org/10.1109/TWC.2022.3175472
16. Qin, P., Zhao, H., Fu, Y., Geng, S., Chen, Z., Zhou, H., Zhao, X. Energy-Efficient Resource Allocation for Space-Air-Ground Integrated Industrial Power Internet of Things Network. IEEE Transactions on Industrial Informatics, 2023, 20, 5274-5284. https://doi.org/10.1109/TII.2023.3331127
17. Qin, Z., Yao, H., Mai, T., Wu, D., Zhang, N., Guo, S. Multi-Agent Reinforcement Learning Aided Computation Offloading in Aerial Computing for the Internet-of-Things. IEEE Transactions on Services Computing, 2022, 16, 1976-1986. https://doi.org/10.1109/TSC.2022.3190562
18. Senthil, G. A., Raaza, A., Kumar, N. Internet of Things Energy-Efficient Cluster-Based Routing Using Hybrid

Particle Swarm Optimization for Wireless Sensor Network. Wireless Personal Communications, 2022, 122, 2603-2619. https://doi.org/10.1007/s11277-021-09015-9

19. Su, Z., Feng, W., Tang, J., Chen, Z., Fu, Y., Zhao, N., Wong, K. K. Energy-Efficiency Optimization for D2D Communications Underlaying UAV-Assisted Industrial IoT Networks with SWIPT. IEEE Internet of Things Journal, 2022, 10, 1990-2002. https://doi.org/10.1109/JIOT.2022.3142026
20. Sun, Y., An, K., Luo, J., Zhu, Y., Zheng, G., Chatzinotas, S. Outage Constrained Robust Beamforming Optimization for Multiuser IRS-Assisted Anti-Jamming Communications with Incomplete Information. IEEE Internet of Things Journal, 2022, 9, 13298-13314. https://doi.org/10.1109/JIOT.2022.3140752
21. Van Hoang, T. Impact of Integrated Artificial Intelligence and Internet of Things Technologies on Smart City Transformation. Journal of Technical Education Science, 2024, 19, 64-73. https://doi.org/10.54644/jte.2024.1532
22. Vazhuthi, P. P. I., Prasanth, A., Manikandan, S. P., Sowndarya, K. K. D. A Hybrid ANFIS Reptile Optimization Algorithm for Energy-Efficient Inter-Cluster Routing in Internet of Things-Enabled Wireless Sensor Networks. Peer-to-Peer Networking and Applications, 2023, 16, 1049-1068. https://doi.org/10.1007/s12083-023-01458-0
23. Wu, X., Liu, C., Wang, L., Bilal, M. Internet of Things-Enabled Real-Time Health Monitoring System Using Deep Learning. Neural Computing and Applications, 2023, 35, 14565-14576. https://doi.org/10.1007/s00521-021-06440-6
24. Zhang, H., Liu, W., Zhang, Q., Liu, B. Joint Customer Assignment, Power Allocation, and Subchannel Allocation in a UAV-Based Joint Radar and Communication Network. IEEE Internet of Things Journal, 2024, 11, 29643-29660. https://doi.org/10.1109/JIOT.2024.3397029
25. Zhou, Y., Liu, X., Zhai, X., Zhu, Q., Durrani, T. S. UAV-Enabled Integrated Sensing, Computing, and Communication for Internet of Things: Joint Resource Allocation and Trajectory Design. IEEE Internet of Things Journal, 2023, 11, 12717-12727. https://doi.org/10.1109/JIOT.2023.3335937

This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).