drones

MDPI

Article

A Reward-and-Punishment-Aware Incentive Mechanism for Directed Acyclic Graph Blockchain-Based Federated Learning in Unmanned Aerial Vehicle Networks

Xiaofeng Xue, Qiong Li * and Haokun Mao

School of Cyberspace Science, Faculty of Computing, Harbin Institute of Technology, Harbin 150001, China; xfxue@hit.edu.cn (X.X.); hkmao@hit.edu.cn (H.M.)
* Correspondence: qiongli@hit.edu.cn

Highlights

What are the main findings?

- The proposed incentive mechanism integrating rewards and punishments for DAG blockchain-based federated learning in UAV networks can effectively increase the participation rate of UAVs and enhance the overall performance of federated learning.

What is the implication of the main finding?

- The proposed incentive mechanism offers an effective approach to incentivize UAV participation in federated learning under incomplete information.

Abstract

The integration of unmanned aerial vehicles (UAVs) and Federated Learning (FL) enables distributed model training while preserving data privacy. To overcome the challenges caused by centralized and synchronous model updates, we integrate Directed Acyclic Graph (DAG) blockchain-based FL into UAV networks. In this decentralized and asynchronous framework, UAVs can independently and autonomously participate in the FL process according to their own requirement. To achieve the high FL performance, it is essential for UAVs to actively contribute their computational and data resources to the FL process. However, it is challenging to ensure that UAVs consistently contribute their resources, as they may have a propensity to prioritize their own self-interest. Therefore, it is crucial to design effective incentive mechanisms that encourage UAVs to actively participate in the FL process and contribute their computational and data resources. Currently, research on effective incentive mechanisms for DAG blockchain-based FL framework in UAV networks remains limited. To address these challenges, this paper proposes a novel incentive mechanism that integrates both rewards and punishments to encourage UAVs to actively contribute to FL and to deter free riding under incomplete information. We formulate the interactions among UAVs as an evolutionary game, and the aspiration-driven rule is employed to imitate the UAV's decision-making processes. We evaluate the proposed mechanism for UAVs within a DAG blockchain-based FL framework. Experimental results show that the proposed incentive mechanism substantially increases the average UAV contribution rate from $77.04_{\pm 0.84}\%$ (without incentive mechanism) to $97.48_{\pm 1.29}\%$. Furthermore, the higher contribution rate results in an approximate 2.23% improvement in FL performance. Additionally, we evaluate the impact of different parameter configurations to analyze how they affect the performance and efficiency of the FL system.

Check for updates

Received: 26 November 2025
Revised: 8 January 2026
Accepted: 15 January 2026
Published: 21 January 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

Keywords: direct acyclic graph blockchain; federated learning; unmanned aerial vehicle; incentive mechanism; aspiration driven rule

1. Introduction

The integration of unmanned aerial vehicles (UAVs) with artificial intelligence (AI) enhances their capability to efficiently perform complex tasks [1,2]. Traditionally, UAVs collect sensor data and transmit it to a cloud server for the centralized training of the machine learning model. However, this method can incur high communication costs and introduce privacy risks [3]. In addition, the collected data cannot be processed and utilized in real time, which limits the system's responsiveness and effectiveness. Federated Learning (FL) offers a promising solution to these challenges by enabling distributed model training across multiple UAVs without sharing raw data, effectively breaking down data silos [4,5].

However, FL faces several challenges, including the need for a trusted third party (TTP) to coordinate the training process, which introduces a single point of failure [6]. Additionally, classical synchronous FL suffers from performance bottlenecks that complicate the coordination of participating clients [7]. To eliminate the reliance on a TTP, blockchain technology has been introduced into FL systems as one of the solutions [6,8]. Meanwhile, asynchronous FL, as one of the solutions, has also been introduced to alleviate synchronization bottlenecks, allowing clients to upload updated models at any time without waiting for others, thus improving flexibility and efficiency [7]. As a combination of decentralized and asynchronous FL, FL frameworks based on Directed Acyclic Graph (DAG) blockchain have been proposed [9–16]. Within these frameworks, each client can independently and autonomously participate in the FL process according to its own requirements. Therefore, the DAG blockchain-based FL frameworks are particularly well suited for UAV networks, which are characterized by high mobility.

Nevertheless, achieving high-performance FL relies heavily on UAVs actively contributing their computational and data resources. In practice, individually rational UAVs are more likely to choose to only download the aggregated model without contributing their own resources, a behavior known as free-riding. Free-riding can significantly degrade the performance of the FL system, as it reduces the diversity and quantity of data used for training, leading to suboptimal model accuracy and generalization. It is crucial to design effective incentive mechanisms that encourage UAVs to actively participate in the FL process. There are several incentive mechanisms for the traditional FL framework, such as the game-theory-based incentive mechanism, auction-theory-based incentive mechanism, and contract-theory-based incentive mechanism, that have been proposed to encourage clients to actively participate in FL [17].

However, these incentive mechanisms are typically formulated as games between the clients and the server (or model owner), where the server publishes the machine learning task and the clients participate in the training process by contributing their resources to the FL system. The server then evaluates the contributions of each client and distributes rewards accordingly. In other words, the server acts as the consumer, providing rewards to compensate clients for their training costs, while the clients act as producers. The DAG blockchain-based FL in UAV networks has three main features. First, the UAVs aim to obtain a high-quality aggregated model for their own use, which makes them consumers of the FL system. At the same time, they need to contribute their local data and computational resources to the FL system, so they are also producers. In other words, each UAV plays a dual role as both a consumer and a producer in the FL system. Second, the sizes of the local datasets on different UAVs are often heterogeneous, which leads to heterogeneous

energy consumption and training costs that should be compensated among the UAVs. In other words, the UAVs with smaller local datasets may need to compensate those with larger local datasets to incentivize cooperation. Third, all UAVs can join or leave the FL system at any time, and no UAV can prevent others from receiving the aggregated model or from free-riding. These features imply that existing incentive mechanisms designed for the traditional FL framework are not directly applicable to our proposed DAG blockchain-based FL framework in UAV networks. To the best of our knowledge, there is still a lack of incentive mechanisms specifically tailored to the unique challenges of this framework.

Based on the above analysis, we propose an incentive mechanism for DAG blockchain-based FL in UAV networks. The main contributions of this paper are summarized as follows:

1. We integrate DAG blockchain-based FL into UAV networks to establish a decentralized and asynchronous FL framework. In this framework, UAVs collaboratively train a machine learning model by exchanging and aggregating local updates through the DAG blockchain integrated into the sensing platform.
2. In light of the characteristics of the proposed framework, we design a reward- and punishment-based incentive mechanism for DAG blockchain-based FL in UAV networks. This mechanism effectively encourages UAVs to actively contribute their local data and computational resources to the FL process, while deterring and penalizing free-riding behavior.
3. We model the interactions among UAVs as an evolutionary game under incomplete information to evaluate the effectiveness of the proposed incentive mechanism. An aspiration-driven rule is adopted to emulate clients' decision-making behavior. To the best of our knowledge, this is the first work in the field of FL incentive mechanisms to employ an aspiration-driven rule for modeling client decision-making. In this approach, each participant adjusts its strategy based on the utility obtained from participating in FL in each time.
4. We conduct simulation experiments to evaluate the proposed incentive mechanism under various parameter settings. The evaluation results demonstrate that the proposed incentive mechanism can effectively encourage UAVs to submit trained models while simultaneously suppressing free-riding behavior, thereby improving the overall performance of FL.

The rest of this paper is organized as follows. Section 2 reviews related work on FL frameworks in UAV networks, DAG blockchain-based FL frameworks, and incentive mechanisms in FL. Section 3 introduces a DAG blockchain-based FL framework in UAV networks. Section 4 presents the proposed reward-and-punishment-aware incentive mechanism. In Section 5, a evolutionary game among UAVs is proposed to analyze the effectiveness of the proposed incentive mechanism. Section 6 presents the evaluation experiments conducted under various parameter settings, along with a comprehensive analysis and discussion of the results. Finally, Section 7 concludes the paper and discusses future work.

2. Related Works

2.1. Federated Learning in UAV Networks

The FL framework in UAV networks has attracted significant attention. Qu et al. proposed a decentralized FL for UAV network, where each UAV aggregates model weights from neighboring UAVs and trains the aggregated model on its local dataset [18]. Wang et al. introduced an FL framework for UAV-aided crowdsensing, where UAVs collaboratively train the machine learning model under the coordination of the server deployed in the base stations [3]. Feng et al. proposed a blockchain-enhanced decentralized FL framework

for 5G-enabled UAVs, consists of an application layer, a blockchain layer, and a network layer [19]. The UAVs realize model updates via the blockchain deployed on entities with high computing and storage capacities, such as base stations or roadside units. Li et al. proposed a UAV swarm-based FL framework for collaborative aerial-ground air quality sensing [20]. Xie et al. proposed a practical FL framework for UAV-assisted MCS [1]. In this system, UAVs train local models using sensor data. These locally trained models are transmitted to a sensing platform for aggregation. Wang et al. proposed a UAV-aided FL framework that enables collaborative model training between UAVs and the server [21].

2.2. Federated Learning with DAG Blockchain

The integration of DAG blockchain [22] with FL provides a new paradigm for achieving asynchronous and decentralized FL [23,24]. There are several DAG blockchain-based FL frameworks that have been proposed in recent years.

Lu et al. is the first to introduce DAG blockchain into FL. They proposed an FL framework for the Internet of Vehicles (IoV) with the hybrid blockchain, called PermiDAG [10]. PermiDAG includes a linear blockchain deployed on the roadside units and a DAG blockchain deployed on the Vehicles. In PermiDAG, the updated model parameters published by the vehicles are stored in the DAG blockchain to realize the sharing of the model parameters among the vehicles. Schmid et al. proposed a decentralized FL framework using the DAG blockchain [25]. In this framework, each client maintains a DAG blockchain, and each client selects the tip node (the node that has not been approved by any nodes) through the Markov-chain Monte Carlo random walk tip selection algorithm. Then, the client aggregates the model parameters stored in the selected tip nodes as the aggregated model for local model training. Beilharz et al. introduced an accuracy-biased tip selection algorithm, where clients select tip nodes based on the accuracy of model parameters stored within those nodes, as tested on local datasets [12]. Therefore, the clients can select the model updates that align with their own data distribution to realize an implicit model specialization FL. Cao et al. proposed DAG-FL, a framework that consists of three layers: the FL layer, the DAG layer, and the application layer [13]. In DAG-FL, the clients receive the multiple model parameters from the server and test the model using their local datasets. The clients select the top k highest accuracy models and aggregate them as an aggregated model. Yuan et al. proposed ChainsFL, a framework that employs a two-layer blockchain architecture [11]. The first layer consists of multiple shards, with each shard including a Raft blockchain. The second layer is a DAG blockchain designed to enable cross-shard interaction. The tip selection algorithm used in ChainsFL is similar to the one used in DAG-FL. Wu et al. proposed an asynchronous FL framework using DAG blockchain, called DAFML [26]. DAFML employs a student–teacher architecture during local training, along with a mutual distillation mechanism that enables partial model updates through the exchange of mutually transferred knowledge. A personalized weighting strategy is further incorporated to adapt the distillation process, thereby enhancing the overall model performance. Da Wu et al. proposed DTFed, which utilizes a DAG blockchain to establish an asynchronous FL framework [27]. This framework leverages the lightweight consensus mechanism to ensure the system security. In addition, the knowledge distillation method is used to reduce the communication cost. In our previous study, we introduced a clustered FL framework based on a DAG blockchain. This framework comprises a client layer, which includes multiple clients, and a server layer, which consists of several servers responsible for maintaining the DAG blockchain [15]. And a model parameter similarity-aware tip selection algorithm is proposed to select the tip nodes whose model parameters are aligned with the local data distribution of the clients.

2.3. Incentive Mechanism of Federated Learning

There are various incentive mechanisms for FL that have been proposed in recent years, including game-theory-based incentive mechanisms, auction-theory-based incentive mechanisms, contract-theory-based incentive mechanisms, and reinforcement-learning-based incentive mechanisms.

In the contract theory-based incentive mechanism, the server plays the role of the employer and the clients play the role of the employees. The server publishes the contract, and the clients select the most suitable contract to participate in the training process. For instance, Kang et al. proposed a contract theory-based incentive mechanism for FL in edge networks [28]. This mechanism uses the accuracy-related parameter of local data as the contract type to attract clients with high-quality local data to participate in the training process. The clients who have higher type data can receive more rewards. Wang et al. design a set of incentive contracts for multiple clusters in FL [29], and propose a method for finding the optimal contract that satisfies the individual rationality and incentive compatibility. Yang et al. proposed an asynchronous FL incentive mechanism based on contract theory [30]. In this mechanism, clients are divided into different levels, with each level receiving distinct contracts and rewards to encourage participation in FL.

Game theory, a theory of analyzing interaction strategies among players, has been widely used to design incentive mechanisms for FL. In the game theory-based incentive mechanism, the server is treated as a leader and the clients are treated as followers. The server publishes the task and the necessary information, and then the clients decide their strategies based on the information provided by the server to maximize their own utility. After receiving the clients' strategies, the server decides which clients are chosen to participate in FL and determines the reward for each client. Stackelberg Game is a widely used game theory for designing incentive mechanisms in FL [31–35]. In the Stackelberg Game-based FL incentive mechanism, the server is the leader and the clients are followers; both parties engage in mutual strategic interactions to achieve maximum collective benefit by designing specialized utility functions and reward mechanisms. Sarikaya and Ercetin et al. developed a Yardstick competition FL incentive mechanism to improve the utility of the clients who contribute high-quality resources [36]. In this approach, the server announces a target average delay for completing local model training. Clients who finish the training within this timeframe receive a fixed reward, which increases if they complete the task more quickly. Conversely, if the training time surpasses the target delay, the reward diminishes. This mechanism effectively incentivizes clients with high-quality resources to engage in the FL. He et al. proposed a three-stage Stackelberg Game to model the interactions among model owners, cluster heads, and UAV workers in FL to optimize the allocation of the limited resources. And a multi-agent reinforcement learning algorithm is proposed to search for the Stackelberg equilibrium [35].

Auction theory is another popular approach for designing incentive mechanisms in FL. In this framework, the server functions as the auctioneer, while clients take on the role of bidders. Clients assess the costs associated with participating in FL and submit their bids to the server accordingly. The server then selects clients to participate in the FL process and allocates rewards based on their submitted bids. Thi Le et al. employed a primal-dual greedy auction mechanism to incentivize clients to participate in the FL [37]. In this mechanism, each client submits its bid according to the energy cost of participating in the FL, and the server selects the clients to maximize the social welfare. Jiao et al. designed two auction mechanisms to improve the social welfare and effectiveness of the FL, the first mechanism is a reverse multi-dimensional auction mechanism, while another mechanism is an automated auction mechanism that integrates graph neural networks and deep reinforcement learning [38]. Zeng et al. proposed a lightweight incentive mechanism

using multi-dimensional procurement auction [39]. Tang et al. proposed a dual-reputation reverse auction mechanism [40]. In this mechanism, the first reputation of participating clients is derived from their historical performance and is used to select the participating clients. The second reputation score is based on the quality of the model parameters trained by the clients and is used to determine the reward for the clients. Pang et al. designed an FL incentive mechanism aimed at minimizing social costs [41]. This mechanism decomposes the social cost minimization problem into a winner determination problem. A greedy algorithm is used to select the winners, and their rewards are calculated based on a payment algorithm.

In addition, reinforcement learning has also been introduced in the design of FL incentive mechanisms. Wang et al. proposed a blockchain-enabled FL framework for UAV-assisted crowdsensing [42]. In this framework, the blockchain deployed on the servers is used to enable the model sharing among the UAVs, and reinforcement learning is employed to motivate the participation of client UAVs in FL systems. The Q-learning is adopted to derive the optimal quality of local model update and pricing strategy, thereby designing the optimal strategies of the UAVs and the server, respectively, when the UAVs and the server do not know each other's private parameters.

3. DAG Blockchain-Based FL Framework in UAV Networks

In UAV networks, UAVs need to collect sensor data in real time and collaboratively train machine learning models, making traditional synchronous and centralized FL unsuitable due to latency and single-point-of failure risks. To address this, we introduce the DAG blockchain-based FL framework in UAV networks that enables secure, decentralized, and privacy-preserving model training among UAVs.

As illustrated in Figure 1, we propose a new FL framework in the UAV network that integrates a two-layer DAG blockchain-based FL framework [15]. The proposed framework consists of a UAV layer and a server layer. The UAV layer consists of UAVs equipped with various sensors and computing hardware for data collection and local model training. The server layer includes multiple servers, which can be deployed in sensing platforms. Each of these servers maintains a DAG blockchain, and the servers communicate with each other through a P2P network. The UAVs collaboratively train the model by publishing their updated model parameters to the DAG-based blockchain. Each node in the DAG blockchain corresponds to an updated model parameter submitted by a UAV. Directed edges from a child node to its parent nodes indicate that the child node's model is trained based on a model aggregated from the models of its parent nodes. Nodes that have not yet been referenced by any other node are called tip nodes.

The main notations used in this paper are summarized in Table 1.

Table 1. Summary of main notations.

Notions	Meaning
N	Total number of UAVs participating in the FL process
n	Index of the UAV, $n \in \{1, \ldots, N\}$
$\mathcal{D}_n$	Local dataset of UAV n
ω	Model parameters
$L(\cdot)$	Loss function
η	Learning rate

Table 1. *Cont.*

Notions	Meaning
γ	Staking for participating in the FL process
R	Base participation fee for participating in the FL process
R_n^c	Participation fee
R_n^d	Punishment fee
Par_n	The number of times UAV n publishes its trained model
R_j	Monetary reward of tip node j obtained when being referenced by a new transaction
ς_n	Average CPU cycles for training one data sample by UAV n
c_n	Effective capacitance coefficient of the CPU chipset of UAV n
f_n	CPU cycle frequency of UAV n
E_{bit}^{down}	Energy consumption of downloading one bit of data
E_{bit}^{up}	Energy consumption of uploading one bit of data
E_n^{cmp}	Computational energy cost of UAV n
E_n^{com}	Communication energy cost of UAV n
Θ_n	Model quality evaluation function
Acc_{test}	The test accuracy of the aggregated model
R_n^o	Monetary reward obtained of UAV n
C_n^C	Cost of UAV n with "C" strategy
C_n^D	Cost of UAV n with "D" strategy
V_n^C	Payoff of UAV n with "C" strategy
V_n^D	Payoff of UAV n with "D" strategy
U_n^C	Utility of UAV n with "C" strategy
U_n^D	Utility of UAV n with "D" strategy
β	Selection intensity in the aspiration driven rule
α	Aspiration level in the aspiration driven rule
T	Predefined total number of communication rounds

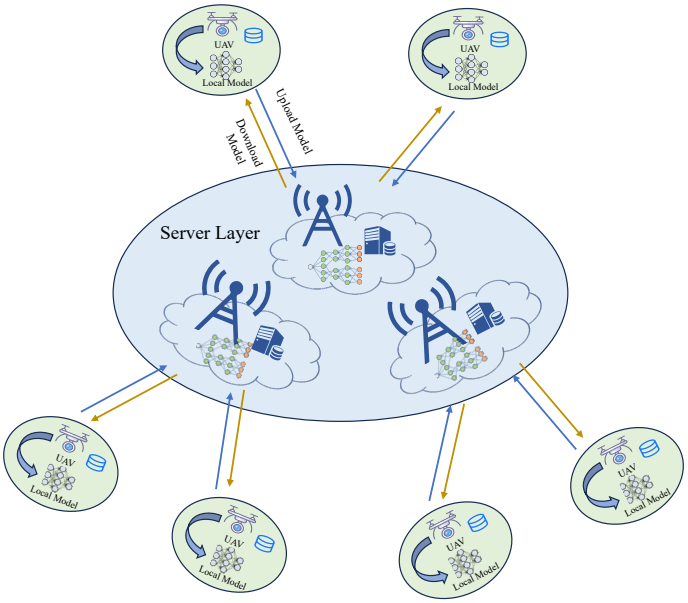


Figure 1. DAG blockchain-based FL framework in UAV networks.

The workflow of the DAG blockchain-based FL framework in UAV networks is illustrated in Figure 2.

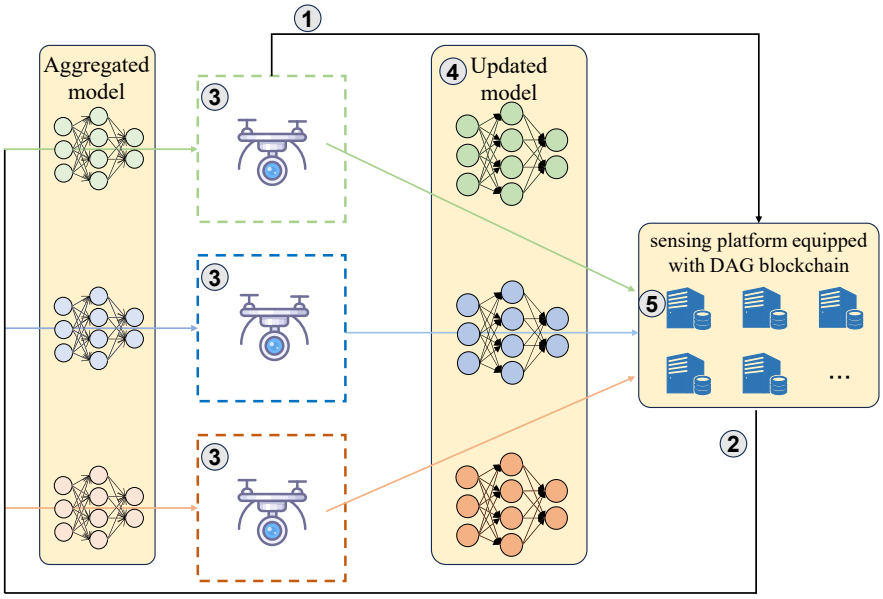


Figure 2. Workflow of DAG blockchain-based FL framework in UAV networks.

Consider an FL system with N UAVs, where each UAV n ($n = 1, 2, \ldots, N$) has a local dataset $\mathcal{D}_n = \{(\mathbf{x}_n^1, y_n^1), \ldots, (\mathbf{x}_n^{|\mathcal{D}_n|}, y_n^{|\mathcal{D}_n|})\}$ containing $|\mathcal{D}_n|$ samples. Here, $\mathbf{x}_n^i$ denotes the feature vector and y_n^i represents the label of the i-th sample. Let $\boldsymbol{\omega}_n$ denote the local model parameters of UAV n. The collaborative objective of the UAVs participating in FL is to train an optimal model $\boldsymbol{\omega}_n^*$ that minimizes the expected loss function $L(\boldsymbol{\omega}_n)$, as shown in Equation (1).

$$\begin{aligned} \omega_n^* &= \arg\min_{\omega_n} L(\omega_n) \\ &= \arg\min_{\omega_n} \frac{1}{|\mathcal{D}_n|} \sum_{i=1}^{|\mathcal{D}_n|} \ell(f(x_n^i, \omega_n), y_n^i) \end{aligned} \tag{1}$$

where $\ell(f(x_n^i, \omega_n), y_n^i)$ denotes the loss of the sample $d_i = (x_n^i, y_n^i)$ evaluated on the model ω_n.

Each UAV randomly connects to a server based on its location and network status and sends a training request to the connected server (step ① in Figure 2). The connected server then uses the predefined tip selection algorithm to select several tip nodes from its maintained DAG blockchain. In this work, we adopt the model similarity-aware adaptive tip selection algorithm proposed in our previous study [15]. First, the server obtains the UAV's latest trained model through the address of the UAV's latest transaction included in the training request. If the UAV participates in FL for the first time, the trained model, obtained by training the initial global model on the UAV's local dataset, is directly included in the training request. The server then computes the cosine similarity between the models in the tip nodes and the participating UAV's latest trained model using Equation (2) to obtain the similarity sequence S.

$$\text{similarity} = \frac{< \omega_{\text{final layer}}^{\text{UAV}}, \omega_{\text{final layer}}^{\text{tip}} >}{\|\omega_{\text{final layer}}^{\text{UAV}}\| \|\omega_{\text{final layer}}^{\text{tip}}\|} \tag{2}$$

where $\omega_{\text{final layer}}^{\text{UAV}}$ denotes the final layer's model parameter of the UAV's latest trained model, and the $\omega_{\text{final layer}}^{\text{tip}}$ denotes the final layer's model parameter of the model stored in the tip node.

Next, the similarity sequence S is sorted in descending order to obtain S'. Then, the change point detection algorithm is applied to detect the change point of S'. The tip nodes before the first change point are selected. The models in the selected tip node are then aggregated to obtain aggregated model ω_{agg}, as shown in Equation (3) (step ② in Figure 2).

$$\omega_{agg} = \sum_{j=1}^{m} p_j(\omega_{tip_j}) \tag{3}$$

where ω_{tip_j} represents the model parameters in the selected tip node j, m is the number of selected tip nodes, and p_j represents the aggregated weight.

After obtaining ω_{agg}, the UAV trains the received aggregated model on its local dataset $\mathcal{D}_n$, according to Equation (4) (step ③ in Figure 2).

$$\omega_{new} = \omega_{agg} - \eta \nabla L(\omega_{agg}, \mathcal{D}_n) \tag{4}$$

where ω_{new} denotes the trained model, η denotes the learning rate, and $\nabla L(\omega_{agg}, \mathcal{D}_n)$ represents the gradient of the local loss function L computed over the dataset $\mathcal{D}_n$ using the aggregated model parameters ω_{agg}.

Then, the UAV sends the trained model ω_{new} to the server (step ④ in Figure 2). The server packages the trained model parameters as a transaction, adds it to the DAG blockchain, returns the address of the transaction to the UAV, and broadcasts it to other servers (step ⑤ in Figure 2).

The above process is repeated until the predefined total number of communication rounds T is reached.

4. Incentive Mechanism for DAG Blockchain-Based FL in UAV Networks

In this section, we present the proposed incentive mechanism for the DAG blockchain-based FL framework in UAV networks, which aims to encourage client active participation in the FL process and prevent free-riding behavior. Existing research indicates that reward and punishment mechanisms are effective methods in promoting cooperation [43]. Therefore, we integrate the characteristics of the DAG blockchain-based FL framework with a reward and punishment scheme to design an incentive mechanism for UAVs in the FL process. The following assumptions are made in the proposed incentive mechanism:

- All participants are assumed to behave honestly during the FL training process; that is, they do not maliciously tamper with data or models, nor do they employ misleading or harmful data for model training.
- Communication between UAVs and sensing platforms, as well as among sensing platforms, is assumed to remain stable throughout the process.

The objective of the proposed incentive mechanism is to maximize the number of cooperators while minimizing the number of defectors and UAVs that leave the FL process in each FL training round. This objective can be formulated as the following optimization problem shown in Equation (5).

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n} \mathbb{I}(s_i = C) - \sum_{i=1}^{n} \mathbb{I}(s_i = D) - \sum_{i=1}^{n} \mathbb{I}(s_i = L) \\
\text{subject to} \quad & s_i = P(u_i), \quad \forall i \\
& u_i = U(R_i, \Theta_n, E_{\text{cmp}}, E_{\text{com}}) \\
& s_i \in \{C, D, L\}, \quad \forall i \\
& \sum_{i=1}^{N} \mathbb{I}(s_i \in \{C, D, L\}) = N \\
& \sum_{i=1}^{N} R_i \leq R_{\text{total}}
\end{aligned} \tag{5}
$$

In Equation (5), function $\mathbb{I}$ is a indicator function. The indicator function $\mathbb{I}(s_i = C)$ equals 1 if UAV i adopts the strategy C and 0 otherwise. $P(u_i)$ is the function that determines the strategy of UAV i based on its utility u_i, which is introduced in Section 5.2. The variable s_i denotes the strategy of UAV i, which can be cooperation "C", defection "D", or leaving "L". The constant N represents the total number of UAVs participating in the FL process. The variable R_i denotes the monetary reward that the UAV i receives, and the R_{total} denotes the total monetary reward paid by all UAVs. The function $U(R_i, \Theta_n, E_{\text{cmp}}, E_{\text{com}})$ represents the utility of UAV i, which is determined by the reward R_i it receives, the quality of the aggregated model Θ_n defined in Section 5.1.2, and the computation cost E_{cmp} and communication cost E_{com} incurred when UAV i participates in the FL process. The E_{cmp} and E_{com} are defined in Section 5.1.1.

Based on the optimal objective, we design an incentive mechanism that maximizes the number of cooperators through a reward scheme while minimizing the number of defectors through a punishment scheme. To illustrate the proposed mechanism, we present a detailed example of a UAV participating in the training process, as shown in Figure 3.

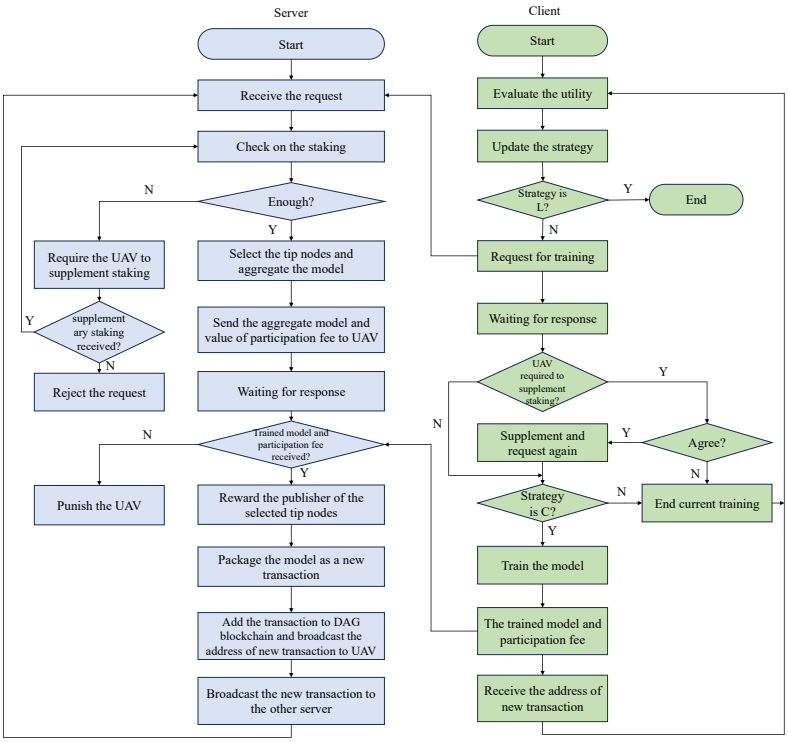


Figure 3. Workflow of the proposed incentive mechanism.

Each UAV participating in the DAG blockchain-based FL system for the first time is required to pay a staking of $\gamma_n = 100 \cdot R$ to the servers in order to qualify for participation in the training process. R is the base participation fee. The initial strategy of the UAV is "C".

In each training round, a UAV randomly connects to a server and sends the training request to the server. After receiving the request, the server will verify whether the UAV's staking is sufficient, i.e., $\gamma_n > 10 \cdot R$. If the staking is sufficient, the UAV is granted permission to participate in the FL training. Otherwise, the UAV is required to provide additional staking until its staking becomes sufficient.

After completing the check on the staking, the server executes the similarity-aware adaptive tip selection algorithm and aggregates the models in the selected tip nodes. Then, the server sends the aggregated model ω_{agg}, and the participation fee R_n^c to the UAV.

The R_n^c is defined as follows:

$$R_n^c = R \cdot (1 + \frac{max(Par) - Par_n}{T}) \tag{6}$$

where R is the base participation fee, Par_n is the number of times UAV n publishes its trained model to the DAG blockchain, $max(Par)$ represents the maximum value of Par_n among all UAVs, and T denotes the predefined total number of communication rounds. Equation (6) indicates that the fewer times a UAV participates, the higher the participation fee it must pay, which penalizes UAVs with low participation frequency. Both participation times and contribution times refer to the number of times a UAV contributes its trained model to the FL process.

The UAV that adopts the cooperation strategy trains the aggregated model using its local dataset and then sends the trained model and the participation fee to the server.

After receiving the trained model and the participation fee, the server packages the received model as a new transaction and adds it to the DAG blockchain. Then, the server allocates R_n^c to the publisher of each selected tip node. For each UAV j that publishes a selected tip node j, the monetary reward it receives is defined as follows:

$$R_j = R_n^c \cdot \{a \cdot \frac{|\mathcal{D}_j|}{\sum |\mathcal{D}|} + (1 - a) \cdot \frac{p_j}{\sum p}\} \tag{7}$$

In Equation (7), R_j is the reward allocated to the publisher of the tip node j, i.e., UAV j; $|\mathcal{D}_j|$ is the dataset size of UAV j, i.e., the number of samples held by UAV j, and p_j is the reputation of UAV j, which is the total number of times all the transactions published by the UAV j have been approved. The term $\sum |\mathcal{D}|$ denotes the total dataset size of all UAVs that published the selected tip nodes, and $\sum p$ is the total reputation of these UAVs. The parameters a is a weighting factor that adjusts the relative importance of these two components in the reward allocation.

As shown in Figure 4, the green nodes denote selected tip nodes, and the light yellow node represents the newly added node. The new node approves two tip nodes, and the participation fee R_n^c, paid by the publisher of the new transaction, is allocated to the publishers of the two selected tip nodes, according to Equation (7). In Figure 4, the solid arrows indicate the nodes have approved their parent nodes, while the dashed arrows indicate that the new node will approve the selected nodes.

If the server does not receive the updated model from the UAV within the specified time, the server regards this UAV as a defector and deducts a punishment fee $R_n^d = 10 \times R$ from its staking γ_n. The server then adds the deducted R_n^d to the final reward pool, which will be used to reward the servers or UAVs that perform well once the FL training

is completed. In addition, a UAV can leave the FL system at any time. The incentive mechanism is summarized in Algorithm 1.

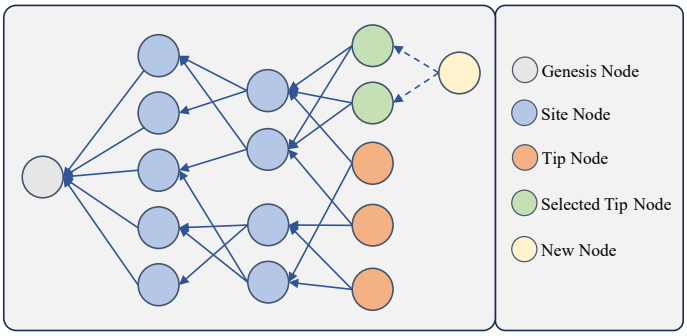


Figure 4. Schematic diagram of reward distribution.

Algorithm 1 Reward and punishment-aware incentive mechanism.

1: The server receives the training request from UAV n and checks on the staking γ_n of UAV n.
2: **if** $\gamma > 10 \cdot R$ **then**
3: The server calculates the participation fee R_n^c for UAV n using Equation (6).
4: The server sends the aggregated model to UAV n and the value of R_n^c.
5: **if** The server receive the model update from UAV n within the specified time **then**
6: The server receives ω_{agg} and the value of R_n^c from UAV n.
7: The server packages the model ω_{agg} as a transaction and adds it to the DAG blockchain.
8: The server sends the address of new transaction to the UAV.
9: The server broadcasts the new transaction to the other servers.
10: /* The server distributes R_n^c to the selected tip nodes' publishers. */
11: **for** each selected tip node tip_j **do**
12: The server calculates the reward for the publisher of tip_j using Equation (7).
13: The server distributes the reward to the publisher of tip_j.
14: **end for**
15: **else**
16: The server deducts the penalty R_n^d from the staking γ_n of UAV n.
17: **end if**
18: **else**
19: The server requires UAV to supplement the staking γ_n.
20: **if** The server receives the supplementary staking γ_n from UAV n **then**
21: Execute lines 3–17.
22: **else**
23: The server rejects the request from UAV n.
24: **end if**
25: **end if**

We analyze the effectiveness of the proposed incentive mechanism by considering a two UAVs' game. We first analyze the utilities of the UAVs without an incentive mechanism, and then analyze their utilities under the proposed incentive mechanism.

We first analyze the utilities of the UAVs without an incentive mechanism. We assume that the two UAVs have the local datasets of the same size. In each game, if both UAVs choose the cooperation strategy, they will receive an aggregated model with quality a evaluated by Θ_n defined in Section 5.1.1, but they need to pay an energy cost b. If a UAV chooses the cooperation strategy while the other UAV chooses the defection strategy or leave strategy, the UAV that cooperates receives an aggregated model with quality a', where $a' < a$, and still needs to pay energy cost b. The UAV who chooses the defection will also receive the same aggregated model with quality a'. If both UAVs choose the defection or

leave strategy, they do not receive any model, and their utilities are 0. Therefore, the utility matrix is given in Table 2. In Table 2, the pairs (x, y) represent the utilities of UAV 1 and UAV 2, respectively.

Table 2. Utility matrix of two UAVs' game without incentive mechanism.

UAV 1 \ UAV 2	C	D	L
C	$(a-b, a-b)$	$(a'-b, a')$	$(a'-b, 0)$
D	$(a', a'-b)$	$(0, 0)$	$(0, 0)$
L	$(0, a'-b)$	$(0, 0)$	$(0, 0)$

If $a' > a - b$, the strategy equilibria are (D,D) and (L,L). If $a' < a - b$, the strategy equilibria are (C,C), (D,D) and (L,L)

Next, we analyze the utilities of the UAVs with the proposed incentive mechanism. In the proposed incentive mechanism, the utilities of the UAVs are decided by the quality of the aggregated model a, the energy cost b, the participation fee c, the monetary reward d and the punishment fee c'. During the game, if both UAVs choose the cooperation strategy, they receive an aggregated model with quality a and a monetary reward d. They also pay energy cost b and the participation fee c. If one UAV chooses the cooperation strategy while the other UAV chooses the defection strategy, the UAV that cooperates receives a lower quality aggregated model with quality a' and the monetary reward d. In addition, it pays the participation fee c and the energy cost b. The UAV that chooses the defection strategy receives the aggregated model with quality a' and pays the punishment fee c'. If the both UAVs choose the defection strategy, they do not receive any aggregated model, and they should pay the punishment fee c'. For a UAV that chooses the leaving strategy, its utility is 0. Therefore, the utility is given in Table 3. In Table 3, the pairs (x, y) represent the utilities of UAV 1 and UAV 2, respectively.

Table 3. Utility matrix of two UAVs' game with incentive mechanism.

UAV 1 \ UAV 2	C	D	L
C	$(a+d-b-c, a+d-b-c)$	$(a'+d-b-c, a'-c')$	$(a'+d-b-c, 0)$
D	$(a'-c', a'+d-b-c)$	$(-c', -c')$	$(-c', 0)$
L	$(0, a'+d-b-c)$	$(0, -c')$	$(0, 0)$

The strategy pair (C,C) yields utilities of $(a+d-b-c, a+d-b-c)$. This can be a strategy equilibrium because $a+d-b-c \geq a'+d-b-c$ and $a+d-b-c \geq a'-c'$. The strategy pairs (C,D) or (D,C) yield utilities $(a'+d-b-c, a'-c')$ or $(a'-c', a'+d-b-c)$. This cannot be a strategy equilibrium because a UAV that unilaterally deviates to the cooperation strategy can obtain a higher utility when $d-b-c > -c'$. This is due to the punishment mechanism, which makes the utility of defection lower. The strategy pairs (C,L) or (L,C) yield utilities $(a'+d-b-c, 0)$ or $(0, a'+d-b-c)$. These cannot be a strategy equilibrium because a UAV that unilaterally deviates to the cooperation strategy can obtain a higher utility when $a'+d-b-c > 0$. The strategy pair (D,D) cannot be a strategy equilibrium when $a'+d-b-c > -c'$. The strategy pairs (D,L), (L,D) and (L,L) cannot be a strategy equilibrium when $a'+d-b-c > 0$. In summary, when $d-b-c > 0$ and $a+d-b-c > a'-c'$, the unique strategy equilibrium of the system is (C,C).

Therefore, if the reward can offset the cost of choosing the cooperation strategy and the punishment fee is sufficiently high, the UAVs are more inclined to adopt the cooperative strategy.

According to the above analysis, it can be concluded that, in the absence of an incentive mechanism, although both (C,C) and (D,D) are strategy equilibria, the system may naturally gravitate towards (D,D) due to the immediate and guaranteed payoff it offers in the absence of trust or external reinforcement. The proposed incentive mechanism can effectively encourage UAVs to choose the cooperation strategy and discourage them from choosing the defection strategy. In the scenario involving multiple UAVs, the analysis becomes more complex. Therefore, we employ simulations to evaluate the effectiveness of the proposed incentive mechanism.

Then, we compare and analyze the differences between the existing incentive mechanisms and the proposed incentive mechanism, as shown in Table 4.

Table 4. Comparison of different theory-based incentive mechanisms.

Incentive Mechanism	Reward Provider	Model User	Model Evaluator
Contract theory-based [28]	Server	Server	Server
Stackelberg Game theory-based [31]	Server	Server	Server
Auction theory-based [40]	Server	Server	Server
Our proposed mechanism	Each UAV	Each UAV	Each UAV

From Table 4, it can be observed that, in the existing incentive mechanisms, the server plays a central role in providing rewards, utilizing the model, and evaluating the model's performance. In contrast, the proposed incentive mechanism allows each UAV to actively participate in rewarding, utilizing, and evaluating the model. The servers in the proposed mechanism still perform model aggregation, as in the existing works. In addition, they are responsible for maintaining the DAG blockchain and selecting tip nodes using the similarity-aware adaptive tip selection algorithm.

5. Multiplayer Evolutionary Game

In this section, we construct a multiplayer evolutionary game to analyze the effectiveness of the proposed incentive mechanism. As shown in Figure 5, UAVs can choose to act as cooperators or defectors, or leave the FL system.

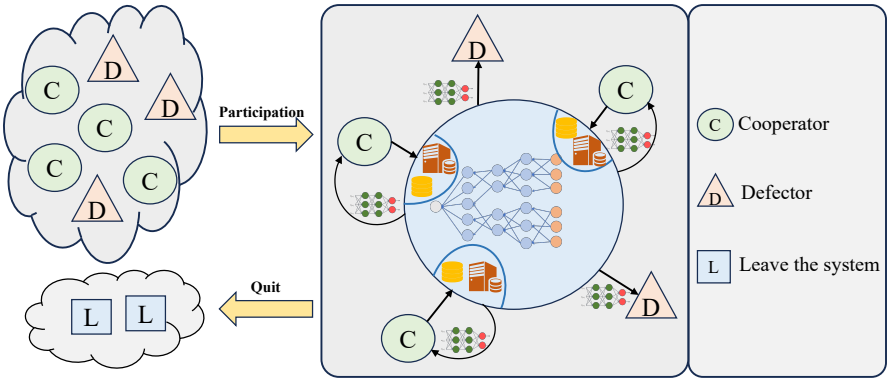


Figure 5. Evolutionary game model of the DAG blockchain-based FL.

The game model can be represented by the triplet $G = \{M, S, U\}$.

- $M = \{1, 2, \ldots, N\}$ denotes the set of UAVs in the game.
- $S = \{s_1, s_2, \ldots, s_N\}$ denotes the strategy set of each UAV in the FL framework, where $s_n \in \{C, D, L\}$. Here, "C" denotes cooperation strategy, "D" denotes defection strategy, "L" denotes the strategy that a UAV exits the FL process and no longer participates in the training.

- $U = \{U_1, U_2, \ldots, U_N\}$ denotes the utility functions of the UAVs.

5.1. Utility Analysis of the UAV

In this section, we analyze the utility of a UAV participating in a single training round. The utility function of UAV n is denoted by U_n and consists of two components: the cost function C_n and the payoff function V_n.

5.1.1. Cost Function

The cost function includes the computation energy cost E_n^{cmp}, the communication energy cost E_n^{com}, and either the participation fee R_n^c or the penalty fee R_n^d.

Computation energy cost

The average number of CPU cycles for UAV n to process one data sample is denoted by ς_n. Since all samples $\{x_n^i, y_n^i\} \in \mathcal{D}_n$ have the same size, the total number of CPU cycles required for UAV n to perform one local iteration is $\varsigma_n|\mathcal{D}_n|$. The CPU-cycle frequency of UAV n is denoted as f_n. Consequently, the CPU energy cost (i.e., computation energy cost) of UAV n for one local training process is given by Equation (8) [44,45].

$$E_n^{cmp} = c_n \varsigma_n |\mathcal{D}_n| f_n^2 \tag{8}$$

where c_n is the effective capacitance coefficient of the CPU chipset.

Communication energy cost

The communication energy cost of a UAV is determined by the model parameters. Let the model size be denoted by $|\omega_n|$, the energy cost for uploading one bit to the server is E_{bit}^{up}, and the energy cost for downloading one bit from the server is E_{bit}^{down}. For a UAV n whose current strategy is "C", the communication energy cost comprises downloading model parameters from the server and uploading the updated model parameters to the server. In contrast, for a UAV n whose current strategy is "D", the communication energy cost only includes downloading the model parameters from the server. Therefore, the communication energy cost can be expressed as in Equation (9).

$$E_n^{com} = \begin{cases} E_{bit}^{up}|\omega_n| + E_{bit}^{down}|\omega_n|, & C \\ E_{bit}^{down}|\omega_n|, & D \end{cases}. \tag{9}$$

In summary, the total cost for UAV n, whose strategy is "C", to participate in one round of FL can be expressed as in Equation (10).

$$\begin{aligned} C_n^C &= E_n^{cmp} + E_n^{com} + R_n^c \\ &= c_n \varsigma_n |\mathcal{D}_n| f_n^2 + (E_{bit}^{up} + E_{bit}^{down})|\omega_n| + R_n^c. \end{aligned} \tag{10}$$

The total cost for a UAV adopting strategy "D" to participate in one round of FL can be expressed as in Equation (11).

$$C_n^D = E_n^{com} + R_n^d = E_{bit}^{down}|\omega_n| + R_n^d. \tag{11}$$

5.1.2. Payoff Function

The payoff of UAV n consists of two parts: the quality of the received aggregated model and received monetary reward.

Model quantify evaluation

The quality of the aggregated model is evaluated based on the accuracy of the model tested on each UAV's local dataset. Variations in the size of each UAV's local dataset result in different quality evaluation results for the aggregated model. UAVs with larger local

datasets typically have higher expectations for the quality of the aggregated model, as they can train relatively high-quality models using their own data. In contrast, UAVs with smaller local datasets have lower expectations, as the models they trained locally are of lower quality. Therefore, we propose a function Θ_n to evaluate the quality of the aggregated model. The function Θ_n is defined in Equation (12).

$$\Theta_n = \mathcal{F}(|\mathcal{D}_n|) \cdot Acc_{test} = \lambda \cdot e^{(-k \cdot (|\mathcal{D}_n| - \overline{|\mathcal{D}|}))} \cdot Acc_{test}. \tag{12}$$

The function Θ_n is defined in terms of the aggregated model's test accuracy Acc_{test} on UAV n's local dataset and the size of this local dataset $|\mathcal{D}_n|$. The function $\mathcal{F}(|\mathcal{D}_n|)$ characterizes the relationship between the size of the local dataset and the function value. To illustrate the function $\mathcal{F}(|\mathcal{D}_n|)$, we provide an example in Figure 6. In this example, we set the parameter $\lambda = 0.0005$, $\overline{|\mathcal{D}|} = 500$ and $k = 0.005$.

Figure 6 indicates that a larger the local dataset leads to a higher expectation for the aggregated model and a lower value of the function $\mathcal{F}(|\mathcal{D}_n|)$, whereas UAVs with smaller local datasets yield higher value of the function $\mathcal{F}(|\mathcal{D}_n|)$.

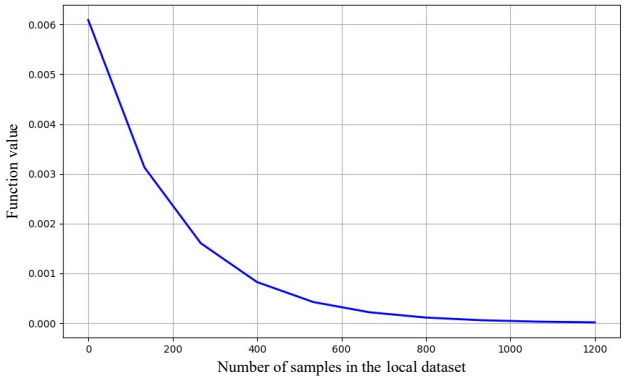


Figure 6. Function $\mathcal{F}(|\mathcal{D}_n|)$.

Monetary Reward

The monetary reward for a cooperative UAV participating in one round of training is defined as the sum of monetary reward obtained when its most recently published tip node is approved by a new transaction. The monetary reward is denoted by R_n^o. The monetary reward of UAV n, which published tip node j, can be expressed as follows.

$$R_n^o = \sum_{z}^{Z} R_z \tag{13}$$

where Z denotes the number of times that the node is approved, and R_z is the monetary reward received each time it is approved. The more frequently the node is approved, the greater the monetary reward it obtains.

In summary, the payoff of a UAV whose strategy is "C" can be expressed by Equation (14).

$$V_n^C = \Theta_n + R_n^o. \tag{14}$$

The payoff of a UAV whose strategy is "D" can be expressed by Equation (15).

$$V_n^D = \Theta_n. \tag{15}$$

5.1.3. Utility Function

According to the analysis in Section 5.1.1 and 5.1.2, the utility function of cooperative UAV n participating in one training round is given by Equation (16).

$$\begin{aligned} U_n^C &= V_n^C - C_n^C \\ &= \Theta_n + R_n^o - \nu E_n^{\text{cmp}} - \nu E_n^{\text{com}} - R_n^c \\ &= \Theta_n + \sum_z^Z R_z - \nu c_n \varsigma_n |\mathcal{D}_n| f_n^2 - \nu (E_{bit}^{down} + E_{bit}^{up}) |\omega_n| - R_n^c \end{aligned} \tag{16}$$

where ν is the electricity price.

The utility function of UAV n when its participation strategy is "D" is given by Equation (17).

$$U_n^D = V_n^D - C_n^D = \Theta_n - \nu E_{bit}^{down} |\omega_n| - R_n^d. \tag{17}$$

In summary, the utilities of UAVs under different strategies are shown in Equation (18).

$$U_n = \begin{cases} \Theta_n + \sum_z^Z R_z - \nu c_n \varsigma_n |\mathcal{D}_n| f_n^2 - \nu (E_{bit}^{down} + E_{bit}^{up}) |\omega_n| - R_n^c, & C \\ \Theta_n - \nu E_{bit}^{down} |\omega_n| - R_n^d, & D. \\ 0 & L \end{cases} \tag{18}$$

5.2. Strategy Update

In the FL process, the UAVs adjust their strategies in real time. In the evolutionary game, the strategy update rules typically include the Moran process [46], the Wright–Fisher process [47], the Fermi process [48,49], and aspiration-driven updates [50,51]. The first three strategy update rules require players to compare themselves and imitate other players to update their strategies. However, the game among the UAVs is an incomplete information game, the UAVs cannot obtain information from other UAVs. Therefore, we adopt the aspiration-driven update rule to model the strategy update of the UAVs. The core idea of the aspiration-driven update is to change each UAV's behavior based on its own utility. In the aspiration-driven rule, each UAV adjusts its strategy according to its utility. Under the aspiration-driven rule, the probability that a UAV updates its strategy from "C" to "D" can be expressed by Equation (19).

$$P_{C \to D} = \frac{1}{1 + e^{-\beta(\alpha - U_n^C)}} \tag{19}$$

where the parameter $\beta > 0$ represents the selection intensity, and α denotes the aspiration level, which reflects the UAV's expectations regarding the utility U_n^C from the game. When $P_{C \to D} = \frac{1}{2}$, the UAV has the same inclination towards two strategies; if $P_{C \to D} < \frac{1}{2}$, the UAV tends to select "C" strategy in the next round of training; whereas if $P_{C \to D} > \frac{1}{2}$, the UAV tends to select "D" strategy in the next round of training.

The probability that a UAV updates its strategy from "D" to "C" can be expressed by Equation (20).

$$P_{D \to C} = \frac{1}{1 + e^{-\beta(\alpha - U_n^D)}}. \tag{20}$$

In addition, if the utility of a UAV is continuously lower than the aspiration level α for several training rounds, the UAV will update its strategy to L and leave the FL system.

The evolutionary game process is summarized in Algorithm 2.

Algorithm 2 Evolutionary game.

▶ Each UAV executes the following operations independently
Each newcomer UAV submits the staking γ and the values of dataset size $|\mathcal{D}_n|$;
The UAV initializes its strategy as C.
repeat
  The UAV evaluates the historical utility.
  if the utility of the latest b rounds consistently falls below the aspiration level α **then**
    The UAV updates its strategy to L, and leave the FL system
  else
    The UAV updates its strategy using Equation (19) or (20) based on its latest complete record
    The UAV connects to the server and send the request to the server
    The UAV receives the aggregated model ω_{agg} and the participation fee R_n^c should pay from the server or the requirement to supplement the staking γ
    if the UAV receives the requirement to supplement the staking γ **then**
      The UAV supplements the enough staking γ to the server or exit
    else
      if the UAV's Strategy is C **then**
        The UAV trains the model based on aggregated model using its local dataset
        The UAV submits the trained model ω_{new} and R_n^c to the server
        The UAV receives the address associated with the transaction of the uploaded model from the server
      end if
      if the strategy of UAV is D **then**
        End the current process
      end if
    end if
  end if
until The FL training has ended, or the UAV has exited the FL system

▶ The server executes the following operations:
The server receives the request from the UAV;
if the request from the UAV is training request **then**
  The server checks the staking γ of the UAV
  if the rest of staking γ is enough **then**
    The server executes the tip selection algorithm to select tip nodes
    The server aggregates the model in the selected tip nodes to obtain ω_{agg}
    The server sends the aggregated model and the participation fee R_n^c to the UAV
    The server waits the response from the UAV
    if the updated model ω_{new} and R_n^c from the UAV are received in time **then**
      The server distributes R_n^c to the other UAVs according to Equation (7)
      The server packages the model parameter ω_{new} as a transaction
      The server adds the transaction to the DAG blockchain
      The server sends the address of new transaction to the UAV
      The server broadcasts the new transaction to the other servers
    else
      The server deducts the penalty R_n^d from the staking γ
    end if
  else
    The server rejects the request and ask the UAV to supplement the γ
  end if
end if

6. Simulation and Analysis

6.1. Experimental Settings

We perform simulation experiments to evaluate the performance of the proposed incentive mechanism. We modeled an FL system with 100 UAVs. In the experiments, we

assume that all UAVs are equipped with the same hardware configuration, the network connection is stable, and the network link quality is ideal. The parameter settings are summarized in Table 5.

Table 5. Simulation parameter settings.

Parameter	Value	Parameter	Value
c_n	10^{-26} [52]	Model size $\|\omega\|$	67.92 Mbits
ς_n	0.01 gigacycles [53]	Communication round T	100
f_n	2 GHz [44]	Local epochs	5
v	\$0.174 per kwh [53]	Batch size	10
E_{bit}^{up}	3 Joules per Mbit [53]	Learning rate η	0.001
E_{bit}^{down}	3 Joules per Mbit [53]	a	0.8
λ	0.0005	k	0.005
$\overline{\|\mathcal{D}\|}$	500		

Dataset and Model. We conduct the experiments on a widely used image classification dataset, CIFAR-10 [54], which consists of 60,000 color images in 10 different classes, with 6000 images per class. The dataset was divided into 50,000 training images and 10,000 test images. Each image has a resolution of 32 × 32 pixels. We divide all samples into four disjoint superclasses according to their labels: {airplane, automobile}, {bird, cat, deer}, {dog, frog, horse}, and {ship, truck}. Then, the samples in each superclass are allocated to a different number of UAVs. The samples of each superclass are distributed among [20, 30, 30, 20] UAVs. In the same superclass, the samples are allocated to the UAVs using a Dirichlet distribution with a concentration parameter of 1. In each UAV, we split the dataset into training and test sets using an 80:20 ratio. The convolutional neural network (CNN) was adopted as the base model for training. The CNN model consists of two convolutional layers, each followed by a ReLU activation function and a 2 × 2 max pooling layer. The first convolutional layer has 32 filters, while the second convolutional layer has 64 filters. After the convolutional layers, there are two fully connected layers with 512 and 10 neurons, respectively. The Stochastic Gradient Descent (SGD) optimizer is used for training.

6.2. Experimental Evaluation with Different Incentive Mode

We first evaluate the performance of the proposed incentive mechanism. We conduct experimental evaluations under four distinct scenarios for comparative analysis: without an incentive mechanism, only with a reward mechanism, only with a punishment mechanism, and with the proposed incentive mechanism. We evaluate the performance using the following four evaluation metrics. The experiments were repeated five times with different random seeds, and we report the mean and standard deviation of the results.

- Model test accuracy: This metric is defined as the average accuracy of the aggregated model tested on each UAV's local dataset. In each evaluation round, all UAVs obtain the aggregated model from the server based on their latest transactions and evaluate this model using their local test dataset. The evaluation results from all UAVs are averaged to derive the evaluation accuracy for each evaluation round. We use the average accuracy over the latest five rounds as the evaluation result.
- Contribution rate: This metric represents the percentage of UAVs that complete training and publish their models in each communication round. We use the average contribution rate over the latest five rounds as the evaluation result.
- Number of defectors: This metric represents the number of UAVs that select the defection strategy. We use the average number of defectors over the last five rounds as the evaluation result.

- Number of leavers: This metric represents the number of UAVs that exit the FL system. We use the average number of leavers over the latest five rounds as the evaluation result.

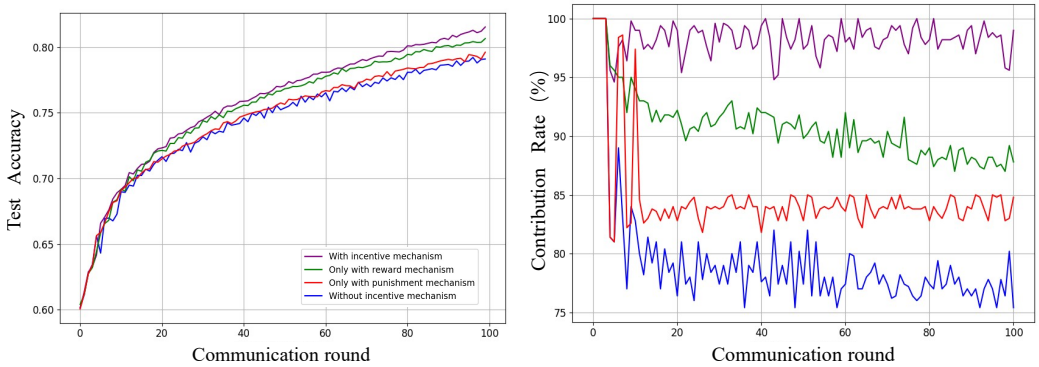
In the experiments, we set $\alpha = 0$, $\beta = 50,000$, and $R = 0.00015$. All UAVs are set to participate in the first three communication rounds with the "C" strategy. Then, the UAVs update their strategy according to their utilities. The experimental results are shown in Table 6.

Table 6. Evaluation results under four distinct scenarios.

Method	Test Accuracy (%)	Contribution Rate (%)	Number of Defectors	Number of Leavers
Without incentive mechanism	$79.02_{\pm 0.36}$	$77.04_{\pm 0.84}$	$19.96_{\pm 0.84}$	$3.00_{\pm 0.00}$
Only with reward mechanism	$80.42_{\pm 0.31}$	$87.80_{\pm 1.38}$	$12.20_{\pm 1.38}$	$0.00_{\pm 0.00}$
Only with punishment mechanism	$79.35_{\pm 0.27}$	$84.08_{\pm 0.27}$	$0.92_{\pm 0.27}$	$15.00_{\pm 0.00}$
With proposed incentive mechanism	$81.25_{\pm 0.41}$	$97.48_{\pm 1.29}$	$2.20_{\pm 1.04}$	$0.32_{\pm 0.44}$

It can be observed that the proposed incentive mechanism achieves the highest contribution rate of $97.48_{\pm 1.29}\%$, significantly outperforming the $77.04_{\pm 0.84}\%$ observed without any incentive mechanism. These results demonstrate the effectiveness of the proposed incentive mechanism in encouraging UAVs to actively contribute to the FL process. The high contribution rate leads to an improvement in model test accuracy, with an increase of approximately 2.23% compared to the scenario without an incentive mechanism. In addition, compared to incentive mechanisms that only rely on either reward or punishment, the proposed mechanism achieves a higher contribution rate and model test accuracy. These results suggest that the combination of reward and punishment is more effective in motivating UAVs to contribute to the FL system.

Figure 7 illustrates the average model test accuracy and contribution rate in each communication round. It can be observed that FL with the proposed incentive mechanism converges faster than others. The contribution rate in the FL framework with the proposed incentive mechanism is higher, and its trend is more stable than that of the others.



(**a**) Model test accuracy in each communication round with different incentive mode.

(**b**) Contribution rate in each communication round with different incentive mode.

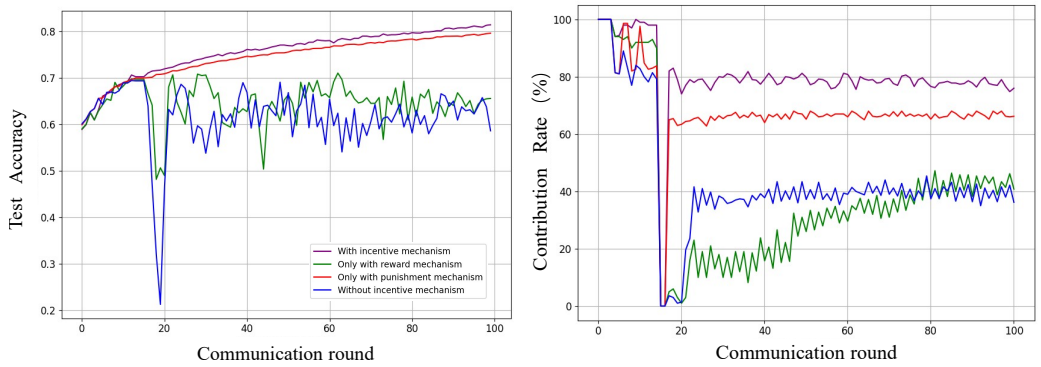
Figure 7. Model test accuracy and contribution rate in each communication round with different incentive mode.

We also simulated the situation in which the UAVs' strategy mutated from "C" to "D". In the experiments, we set $\alpha = 0$, $\beta = 50,000$, and $R = 0.00015$. In the 15th and 16th communication rounds, we allowed the UAVs to change their strategy from "C" to "D". The results are shown in Table 7.

Table 7. Evaluation results under four distinct scenarios under the scenario where the UAVs change their strategy from "C" to "D".

Method	Test Accuracy (%)	Contribution Rate (%)	Number of Defectors	Number of Leavers
Without incentive mechanism	$62.90_{\pm 10.84}$	$38.88_{\pm 10.65}$	$54.32_{\pm 12.61}$	$6.80_{\pm 2.39}$
Only with reward mechanism	$64.64_{\pm 10.56}$	$42.12_{\pm 7.91}$	$48.84_{\pm 3.00}$	$9.04_{\pm 6.91}$
Only with punishment mechanism	$79.47_{\pm 0.25}$	$66.68_{\pm 0.50}$	$18.32_{\pm 0.50}$	$15.00_{\pm 0.00}$
With proposed incentive mechanism	$81.23_{\pm 0.07}$	$76.68_{\pm 0.66}$	$17.32_{\pm 0.30}$	$6.00_{\pm 0.45}$

From Table 7, it can be observed that when the UAV's strategy mutated from "C" to "D" in the 15th and 16th rounds, the proposed incentive mechanism still achieves the highest contribution rate of $76.68_{\pm 0.66}$%, significantly outperforming the $38.88_{\pm 10.65}$% observed without any incentive mechanism. In this scenario, the FL system also realizes a model test accuracy of $81.23_{\pm 0.07}$%. The results also show that the proposed incentive mechanism that includes the punishment mechanism results in a smaller number of defections because the punishment can make the UAVs choosing the "D" strategy obtain a negative utility. The reward scheme can cover the UAVs' training cost, which provides the UAVs with more willingness to choose the "C" strategy. In addition, the standard deviation of the model test accuracy is high in the scenarios without an incentive mechanism and in those only with reward mechanism. This is because in these two situations, the number of UAVs selecting the "C" strategy is reduced to a small level after the 15th and 16th rounds, which can be observed in Figure 8. This reduction in cooperative behavior leads to the failure of the tip selection algorithm. The algorithm struggles to correctly identify and select tip node within the same cluster. As a result, the evaluation result has a high standard deviation.



(**a**) Model test accuracy in each communication round with different incentive mode.

(**b**) Contribution rate in each communication round with different incentive mode.

Figure 8. Model test accuracy and contribution rate in each communication round with different incentive modes under the scenario where the UAVs change their strategy from "C" to "D".

Figure 8 illustrates the average model test accuracy and the contribution rate in each communication round. It can be observed that, in the scenario without a punishment scheme, the contribution rate drops to nearly zero around the 20th round, indicating that the UAVs are more likely to select the defection strategy. However, since almost all UAVs adopted the "D" strategy, their utility becomes low. As a result, some UAVs begin to change their strategy to "C", which leads to an increase in the number of cooperators, although this value remains lower than in the other scenarios.

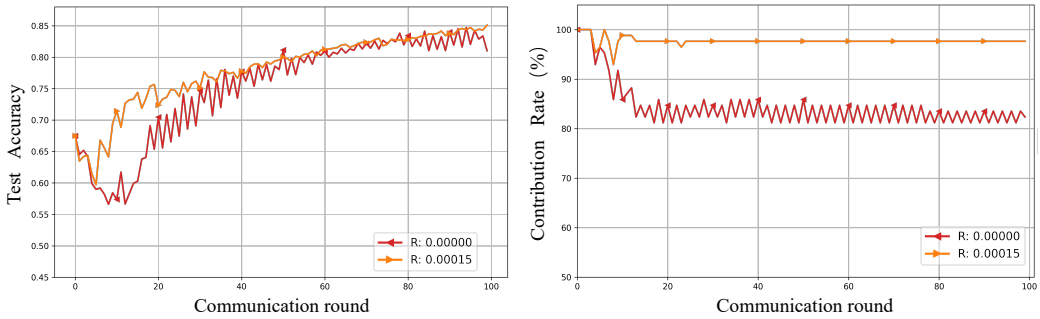
In the scenario with the proposed incentive mechanism and only with the punishment mechanism, the dramatic increase in the number of cooperators indicates that punishment can effectively incentivize cooperative behavior. However, the lack of a reward mechanism in the scenario with only a punishment mechanism causes some UAVs to leave the FL

system. This is because, while punishment can deter non-cooperative behavior, the absence of positive reinforcement in the form of rewards fails to sufficiently motivate continued participation and contribution. As a result, the contribution rate in the scenario only with the punishment mechanism is lower than that in the scenario with the proposed incentive mechanism, which includes both rewards and punishments. The proposed incentive mechanism effectively balances the deterrent effect of punishments with the motivation effect of rewards, encouraging more UAVs to remain in the system and actively participate.

In the scenario with the proposed incentive mechanism, the contribution rate still decreases when the UAVs change their strategy from "C" to "D". This is because there are some UAVs that can benefit from the "D" strategy. This is due to the fact that the current level of punishment might not be sufficient to outweigh the benefits of defecting.

Then, we evaluate the performance of the proposed incentive mechanism using a remote sensing target classification dataset, HelloRS, to demonstrate its applicability in real-world UAV scenarios. In the experiments, we divide all samples into four disjointed superclasses according to their labels: {Forest, CultivatedLand}, {Waters, River, SwimmingPool}, {TennisCourt, BasketballCourt, FootballField}, and {Highway, Pylon}. The samples in each superclass are allocated to different numbers of UAVs, which are [20, 30, 15, 20] UAVs, respectively.

The experimental results are shown in Figure 9. It can be observed that the proposed incentive mechanism achieves higher model test accuracy and a higher contribution rate compared to the scenario without an incentive mechanism.



(**a**) Model test accuracy in each communication round.

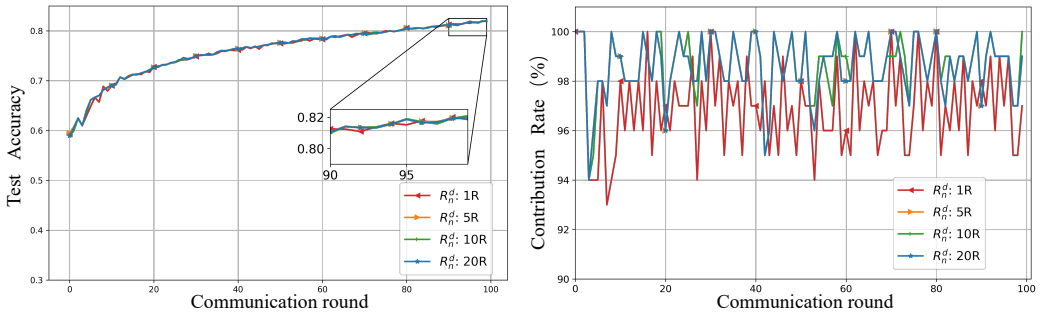
(**b**) Contribution rate in each communication round.

Figure 9. Model test accuracy and contribution rate in each communication round evaluated using the HelloRS dataset.

6.3. Experimental Evaluation with Different Punishment Fee

In the proposed incentive mechanism, the punishment fee is a critical parameter that influences UAV behavior in the FL process. In this section, we evaluate the impact of different punishment fees on the effectiveness of the proposed incentive mechanism. We conduct experiments with varying punishment fees $R_n^d = [1, 5, 10, 20] \cdot R$ to analyze how the punishment fee affects UAVs' behaviors and the overall system performance. In the experiments, we set $\alpha = 0$, $\beta = 50,000$, and $R = 0.00015$. The experimental results are shown in Figure 10.

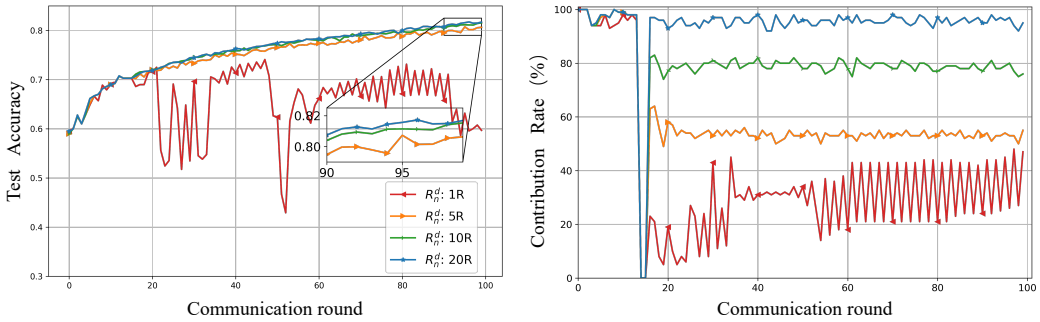
Then, we simulate the situation in which the UAVs' strategy mutated from "C" to "D". In the experiments, we set $\alpha = 0$, $\beta = 50,000$, and $R = 0.00015$. In the 15th and 16th communication rounds, we let the UAVs change their strategy from "C" to "D". The results are shown in Figure 11.



(**a**) Model test accuracy in each communication round with different punishment fee.

(**b**) Contribution rate in each communication round with different punishment fee.

Figure 10. Model test accuracy and contribution rate in each communication round with different punishment fee.



(**a**) Model test accuracy in each communication round with different punishment fee.

(**b**) Contribution rate in each communication round with different punishment fee.

Figure 11. Model test accuracy and contribution rate in each communication round with different punishment fee under the scenario where the UAVs change their strategy from "C" to "D".

From Figures 10 and 11, it can be observed that the impact of different punishment fees on the model test accuracy and the contribution rate is negligible when the UAVs do not change their strategy from "C" to "D". However, when the UAVs change their strategy from "C" to "D", higher punishment fees result in a higher contribution rate, and the model test accuracy becomes similar when the punishment fee satisfies $R_n^d \geq 10 \cdot R$. This is because higher punishment fees effectively deter UAVs from choosing the defection strategy, thereby encouraging more UAVs to cooperate and contribute to the FL process. Consequently, the overall system performance improves, as reflected by the increased contribution rate and model test accuracy. Therefore, we set the punishment fee to $R_n^d = 10 \cdot R$ in the experimental evaluations.

6.4. Experimental Evaluation with Different R, β, and α

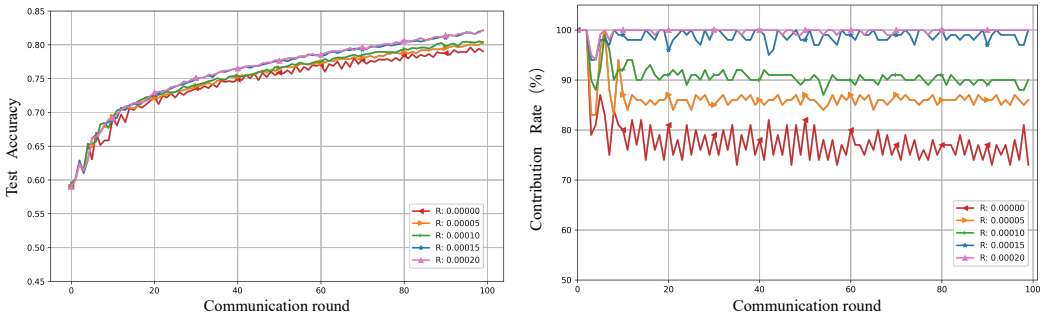
In this section, we evaluate the impact of the parameters R, β, and α on the effectiveness of the proposed incentive mechanism. To comprehensively analyze the influence of these parameters, we employ six evaluation metrics: model test accuracy, contribution rate, the total number of times each UAV chooses the cooperation strategy, the total utility of each UAV, total monetary reward obtained by each UAV, and the total fees (including participation and punishment fees) paid by each UAV. The new metrics are defined as follows:

- Total number of times each UAV chooses the cooperation strategy: this metric denotes the total number of times the cooperation strategy is chosen by each UAV over the entire FL training process.

- Total utility of each UAV: this metric denotes the total utility of each UAV over the entire FL training process.
- Total monetary reward obtained by each UAV: this metric is defined as the total monetary rewards each UAV obtains.
- Total fees paid by each UAV: this metric is defined as the cumulative fees each UAV paid when the UAV selects the cooperation or defection strategy over the entire FL training process.

6.4.1. Evaluation Result with Different Base Participation Fee R

In this section, we evaluate the performance of our proposed incentive mechanism with different base participation fee R. We conducted the experiments with varying R in Equation (6) to analyze how the base participation fee affects the UAVs' behaviors and the overall system performance. In the experiments, we set $\alpha = 0$ and $\beta = 50,000$. We firstly evaluate the model test accuracy and contribution rate, and the corresponding results are shown in Figure 12.



(**a**) Model test accuracy in each communication round with different base participation fee R.

(**b**) Contribution rate in each communication round with different base participation fee R.

Figure 12. Model test accuracy and contribution rate in each communication round with different base participation fee R.

Figure 12 shows that a higher base participation fee generally results in higher final test accuracy, as more UAVs are incentivized to cooperate and contribute to the FL process.

Then, we evaluate the total number of times each UAV selects the cooperation strategy, as shown in Figure 13. In Figure 13, the bar chart represents the local dataset size of each UAV, while the line charts show the total number of times each UAV chooses to cooperate. It can be seen that when $R = 0$, the UAVs with large local datasets exhibit a low level of cooperation, because they obtain low utility from the aggregated model. Consequently, the training process is dominated by the UAVs with small local datasets. As R increase, the reward becomes sufficient to cover the training cost, and UAVs with large local datasets are more willing to train the model. Therefore, the total number of times each UAV chooses the cooperation strategy increases.

Then, we present the total utility of each UAV in Figure 14. To ensure a clearer presentation of the details, we illustrate the part of the figure in the lower subplot. It can be observed that when $R \geq 0.00015$, the utility of most UAVs is greater than 0. This indicates that UAVs can obtain enough monetary reward to compensate for the cost of model training, the cost of model transmitting, and the participation fee. Therefore, UAVs are more willing to participate in FL, which is consistent with the cooperation behavior observed in Figure 13.

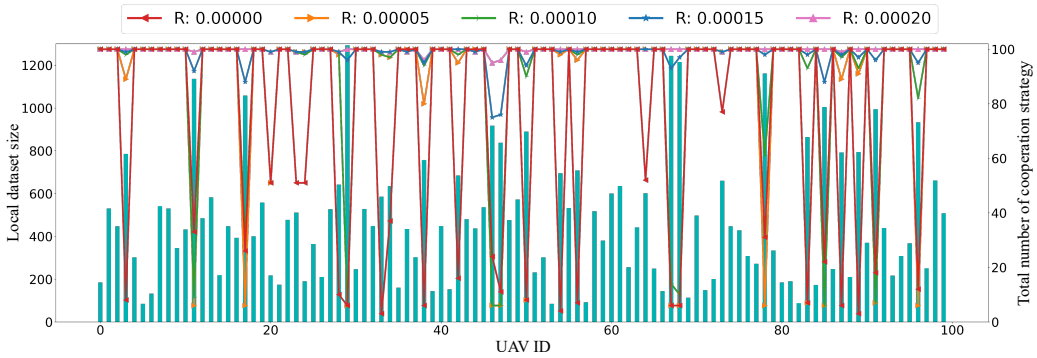


Figure 13. Total number of times each UAV selects cooperation strategy with different base participation fee R.



Figure 14. Total utility of each UAV with different base participation fee R.

Figure 14 indicates that UAVs with larger local datasets obtain lower utility compared to those with smaller datasets. This is primarily because their utility depends mainly on the monetary reward, while the benefit obtained from the aggregated model is relatively small. According to our proposed evaluation function in Equation (12), the aggregated model provides greater utility improvements to UAVs with smaller datasets. Additionally, UAVs with larger local datasets incur higher training costs. As a result, their overall utility is low. In contrast, UAVs with smaller datasets can obtain great benefits from the aggregated models, leading to higher utility and a strong willingness to participate in the FL process.

In Figure 15, we present the total monetary reward obtained by each UAV. The figure shows that as the R value increases, UAVs receive greater monetary rewards. When $R \leq 0.0001$, the total monetary reward for some UAVs approaches zero because they exited the FL training process at the early stage. These UAVs typically possess larger local datasets. The reason for their early exit is that the cost incurred by model training is not fully compensated when the participation fee is low.

The participation fee of the UAVs is shown in Figure 16. The total fees paid by UAVs that have ever selected the defection strategy are higher than those of the other UAVs, whereas the total fees of UAVs that exited the FL training process at an early stage are lower.

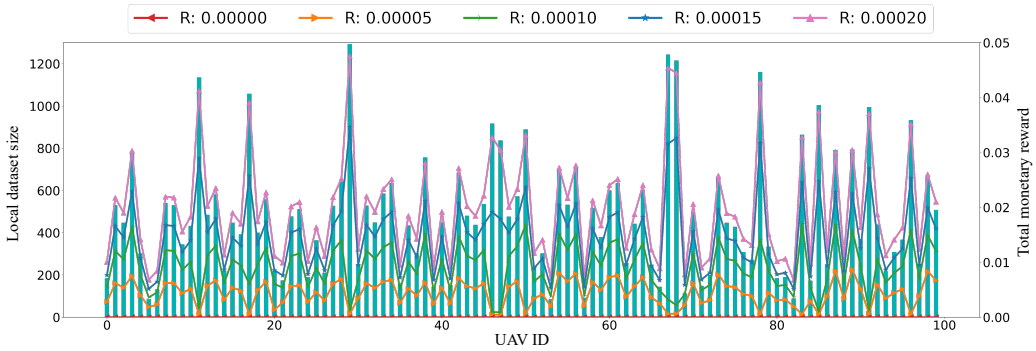


Figure 15. Total monetary reward obtained by each UAV with different base participation fee R.

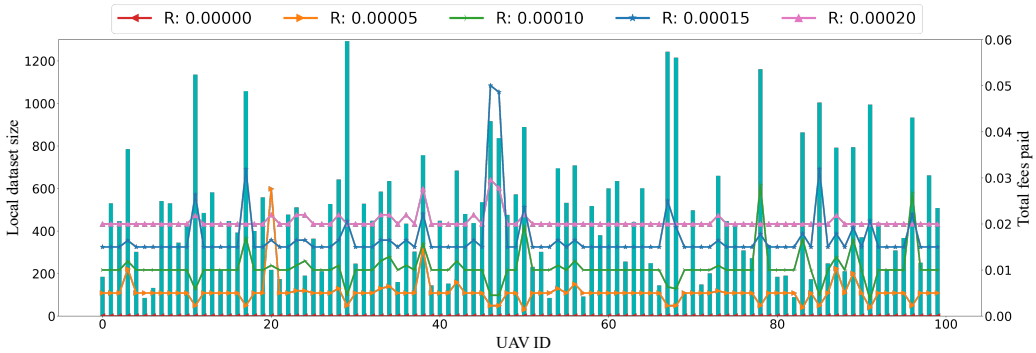
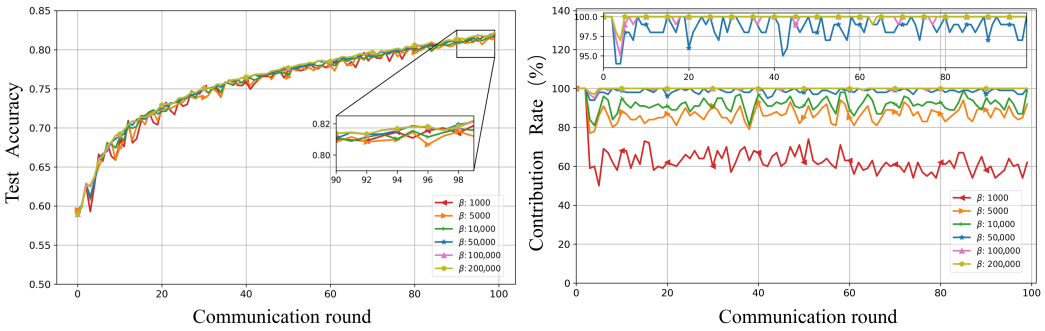


Figure 16. Total fees paid by each UAV with different base participation fee R.

In summary, the experimental results demonstrate that increasing the participation fee R enhances UAVs' participation rate and increases the overall utility of the UAVs. As a result, model accuracy is also improved. It can be concluded that setting R to "0.00015" is a suitable choice for the proposed incentive mechanism, as it effectively motivates most UAVs to participate in the FL process while ensuring high test accuracy and avoiding excessive participation fees.

6.4.2. Evaluation Result with Different Selection Intensity β

In this section, we evaluate the performance of the proposed incentive mechanism with different selection intensities β. In these experiments, we set $\alpha = 0$ and $R = 0.00015$. The evaluation results for model test accuracy and contribution rate are shown in Figure 17.



(**a**) Model test accuracy in each communication round with different selection intensity β.

(**b**) Contribution rate in each communication round with different selection intensity β.

Figure 17. Model test accuracy and contribution rate in each communication round with different selection intensity β.

Figure 17 shows that a higher selection intensity generally leads to higher test accuracy, as more UAVs cooperate and contribute their resources to the FL training. When β is low,

UAVs are less responsive to utility, resulting in a lower willingness to cooperate. In other words, the selection intensity can enhance or reduce the effect of $\Delta = U - \alpha$. Figure 17 shows that when the selection intensity is greater than or equal to 50,000, fluctuations in the contribution rate become relatively small.

The total number of times each UAV selects the cooperation strategy is shown in Figure 18. It can be observed that when $\beta \leq 5000$, the total number of times each UAV chooses to cooperate is independent of the local data size of the UAV, indicating that the incentive mechanism has not fully taken effect, as the low selection intensity makes the UAV less sensitive to the Δ. As β increases, the number of times each UAV chooses to cooperate increases, indicating that the incentive mechanism effectively motivates UAVs to contribute to FL.

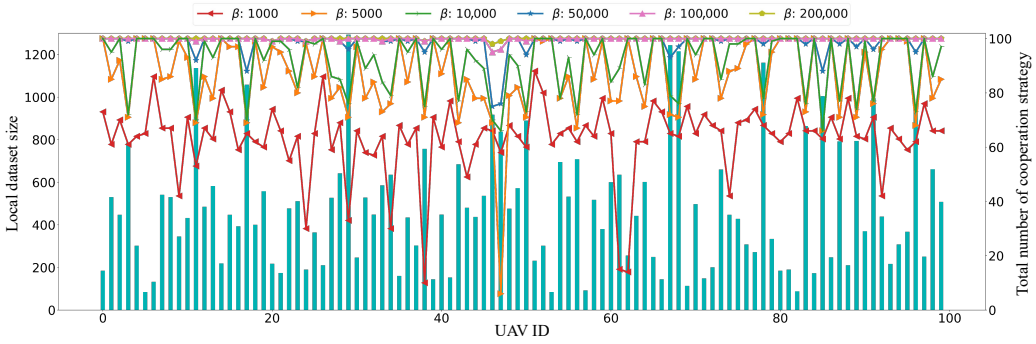


Figure 18. Total number of times each UAV selects cooperation strategy with different selection intensity β.

The total utility of each UAV is shown in Figure 19. It can be observed that with the increase of the β, the utility of the UAVs also increases. This is because a larger β makes UAVs more willing to participate in training, even when their utility is relatively low. As a result, they can earn more monetary rewards, which in turn further enhances their utility. When $\beta \geq 50,000$, the utility of most UAVs is greater than 0.

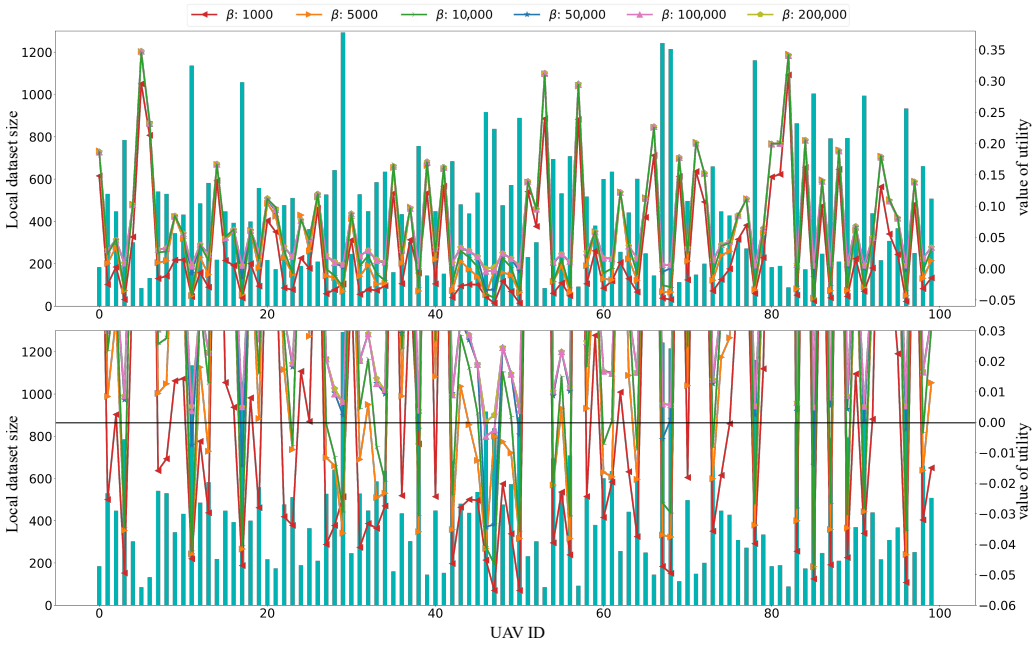


Figure 19. Total utility of each UAV with different selection intensity β.

The total monetary reward obtained by each UAV is shown in Figure 20. It can be observed that as β increases, UAVs obtain higher monetary reward. This is because a higher β promotes UAV cooperation, and more monetary awards are awarded to the UAVs rather than being confiscated as a punishment. In addition, the figure also shows that UAVs with larger local datasets obtain more monetary reward than those with smaller local dataset.

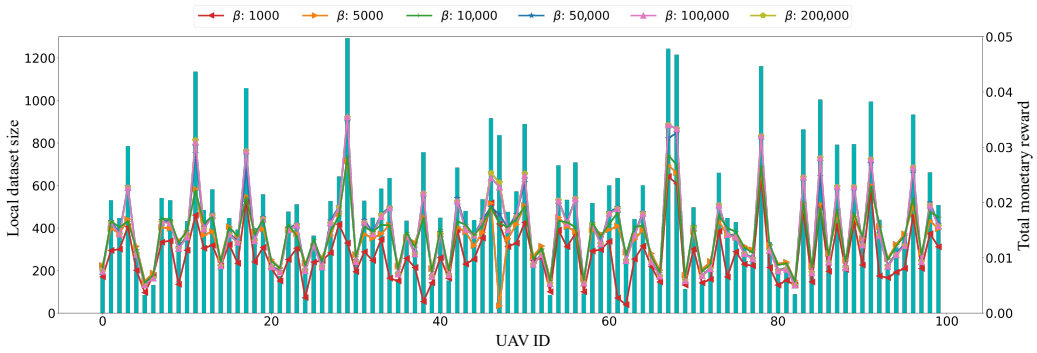


Figure 20. Total monetary reward obtained by each UAV with different selection intensity β.

The evaluation results for the total fees paid by each UAV are shown in Figure 21. The results show that a higher β leads to lower fees. This is because a higher β promotes UAVs to actively participate in FL, resulting in a lower probability of defection. When β is low, UAVs are more likely to choose the defection strategy, which results in higher fees.

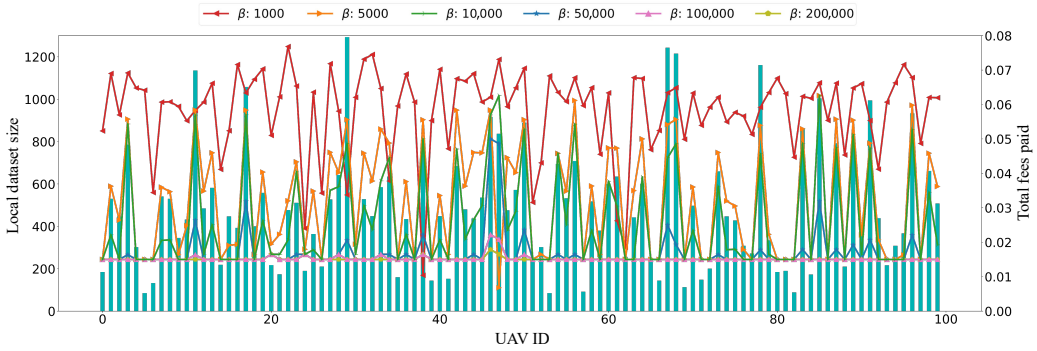


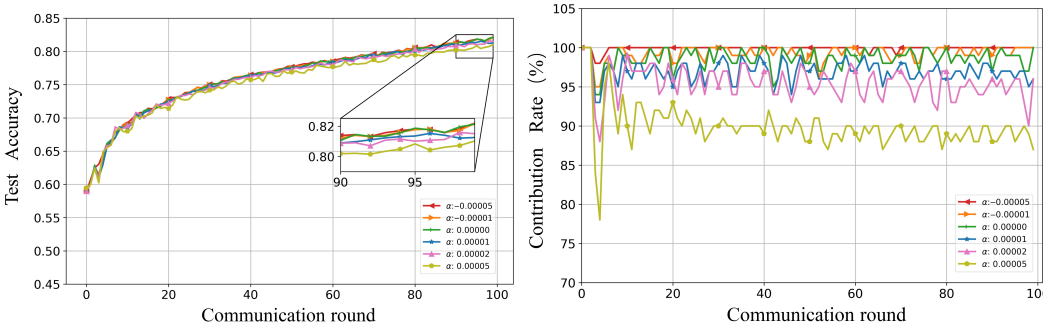
Figure 21. Total fees paid by each UAV with different selection intensity β.

In summary, the experimental results demonstrate that a higher selection intensity β enhances UAV participation rates and increases the overall utility of the UAVs. As a result, when the selection intensity of the UAVs is greater than or equal to 50,000, the proposed incentive mechanism can effectively motivate most UAVs to participate in the FL process while still ensuring high model test accuracy.

6.4.3. Evaluation Result with Different Aspiration Level α

In this section, we evaluate the performance of the proposed incentive mechanism with different aspiration levels α. In the experiments, we set $\beta = 50,000$ and $R = 0.00015$.

Figure 22 shows the test accuracy and the contribution rate in each round with different values of α. The results indicate that a lower aspiration level generally results in higher test accuracy because more UAVs are incentivized to cooperate and contribute to the FL process. When $\alpha = 0.00005$, the contribution rate in each round is low. This is because UAVs with a high aspiration level are more likely to choose the "D" strategy or exit to the FL system, especially those with a large local dataset. In this case, it is necessary to increase the participation fee R to motivate the UAVs to participate in FL.



(**a**) Model test accuracy in each communication round with different aspiration level α.

(**b**) Contribution rate in each communication round with different aspiration level α.

Figure 22. Model test accuracy and contribution rate in each communication round with different aspiration level α.

Figure 23 shows that as the aspiration level α increases, the total number of times each UAV chooses to cooperate decreases, particularly for UAVs with a large local dataset. The is because these UAVs obtain relatively low utility, as illustrated in Figure 24. A higher aspiration level reduces UAV's willingness to participate in FL, which in turn leads to fewer cooperative actions.

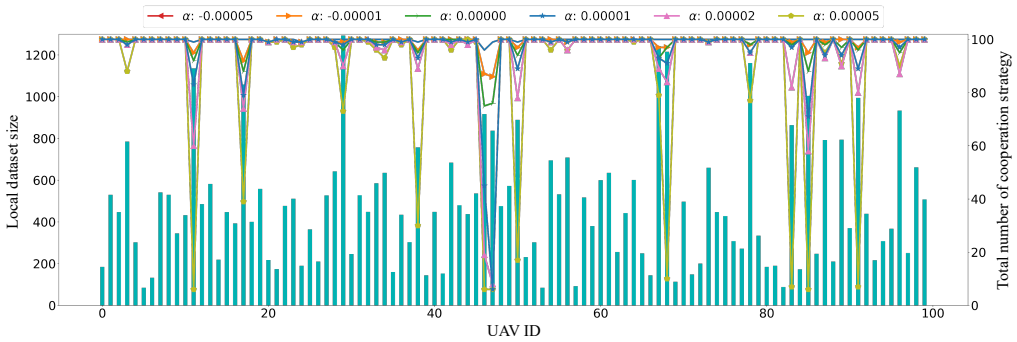


Figure 23. Total number of times each UAV selects the cooperation strategy with different aspiration levels α.

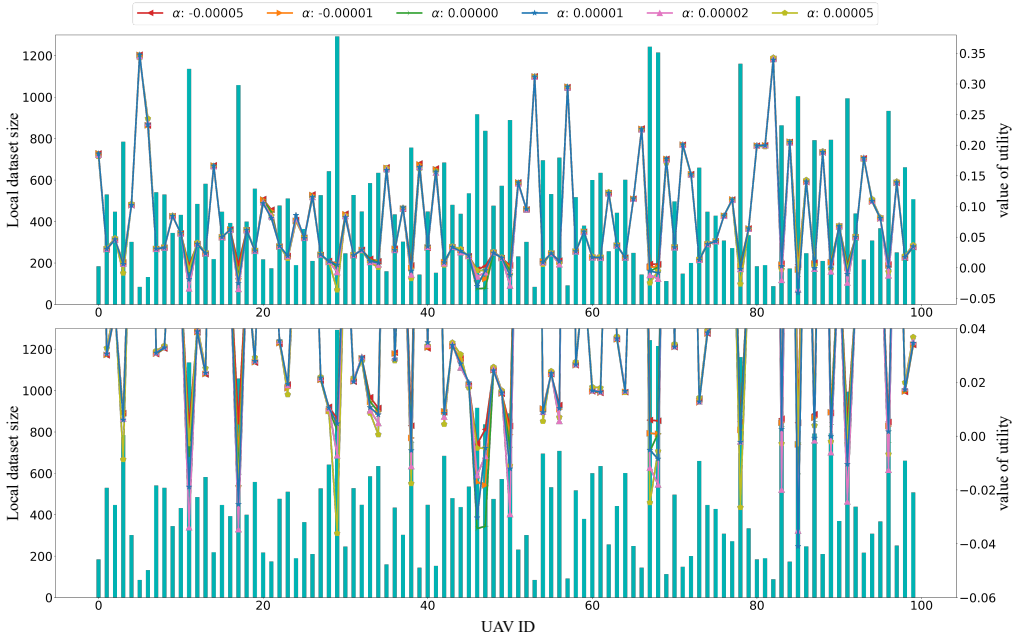


Figure 24. Total utility of each UAV with different aspiration levels α.

Figure 25 shows that when the aspiration level α is low, UAVs obtain more monetary rewards, since UAVs with low aspiration levels are more willing to contribute their resources to the FL. The increase in the number of cooperative UAVs results in a larger total reward being distributed among them. Figure 26 shows that UAVs that select the defection strategy are required to pay higher fees.

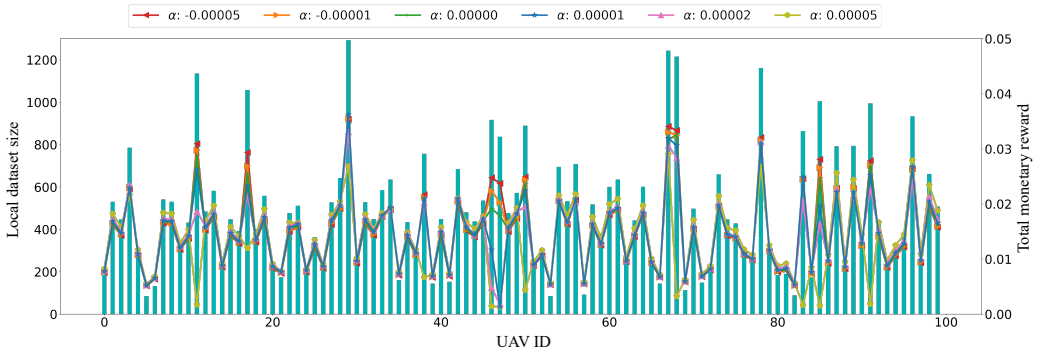


Figure 25. Total monetary reward obtained by each UAV with different aspiration levels α.

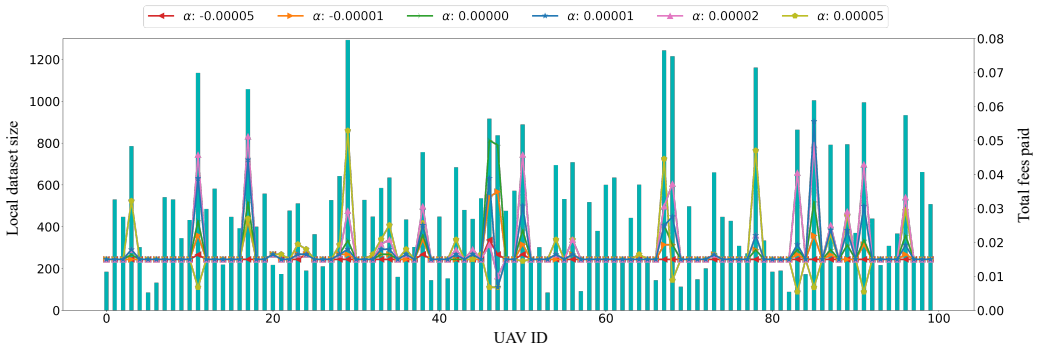


Figure 26. Total fees paid by each UAV with different aspiration levels α.

In summary, the experimental results demonstrate that a lower aspiration level α enhances UAV participation and increases the overall utility of the UAVs. As a result, when the UAVs have a low aspiration level ($\alpha \leq 0$), the proposed incentive mechanism can effectively motivate most of the UAVs to participate in the FL process while ensuring a high model test accuracy. Conversely, if the UAVs have a high aspiration level, it becomes necessary to increase the participation fee R to motivate them to participate in FL.

7. Conclusions

In this paper, a reward-and-punishment-aware incentive mechanism for DAG blockchain-based FL in UAV networks is proposed. Based on the characteristics of the DAG blockchain-based FL, a reward and punishment mechanism-based incentive mechanism is designed to incentivize the UAVs to actively contribute computational and data resources to FL. To evaluate the effectiveness of the proposed incentive mechanism, we formulate the interactions among UAVs as a multi-player evolutionary game, where each UAV adjusts its strategy according to the utility obtained. An aspiration-driven update rule is adopted to model the strategy updates of UAVs. The experimental results demonstrate that the proposed incentive mechanism can effectively motivate UAVs to participate in the FL process and improve the overall performance of the FL. The FL performance and UAV participation behaviors are analyzed under different parameters, including the base participation fee, selection intensity, and aspiration level. The simulation results show that the base participation fee significantly influences the cooperative willingness of UAVs. In general, a higher participation fee leads to stronger cooperative willingness among UAVs

and thereby enhances the overall performance of FL. Meanwhile, the selection intensity affects UAVs' sensitivity to utility, and a higher selection intensity can enhance the cooperative willingness of UAVs. Moreover, a lower aspiration level can improve the cooperation level of UAVs in the system and enhance the performance of FL.

In future work, we plan to explore scenarios in which UAVs have heterogeneous hardware configurations and unstable communication conditions, and to design more flexible incentive mechanisms for these complex settings. The quality evaluation function will also be further refined to better reflect the diverse requirements of UAVs with different local dataset sizes and application scenarios, especially for the parameters used in the function. Additionally, we will investigate the incentive mechanism in scenarios involving malicious UAVs as well as UAVs with varying aspiration levels. Furthermore, we aim to implement the proposed incentive mechanism in real-world UAV networks to validate its effectiveness and practicality.

Author Contributions: Conceptualization, X.X. and H.M.; data curation, X.X.; funding acquisition, Q.L.; investigation, X.X.; methodology, X.X.; project administration, Q.L. and H.M.; software, X.X.; supervision, Q.L. and H.M.; validation, X.X.; visualization, X.X.; writing—original draft, X.X.; writing—review and editing, X.X. and Q.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (Grant No. 62471161), National Cryptography Science Fund (Grant No. 2025NCSF02051), Quantum Science and Technology-National Science and Technology Major Project (Grant No. 2021ZD0300701).

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

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

Abbreviations

The following abbreviations are used in this manuscript:

CNN	Convolutional Neural Network
DAG	Directed Acyclic Graph
FL	Federated Learning
TTP	Trusted Third Party
UAV	Unmanned Aerial Vehicle
MCS	Mobile Crowdsensing Services
P2P	Peer-to-Peer
SGD	Stochastic Gradient Descent

References

1. Xie, L.; Su, Z.; Wang, Y.; Li, Z. A Practical Federated Learning Framework with Truthful Incentive in UAV-assisted Crowdsensing. *IEEE Trans. Inf. Forensics Secur.* **2025**, *20*, 129–144. [CrossRef]
2. Chen, Y.; Ding, Y.; Chen, B.; Si, P.; Lu, W.; Lin, D.; Yang, Z.; Hu, S. Covert Communication towards a Flying Warden in UAV-assisted MEC System. *Chin. J. Aeronaut.* **2025**, *38*, 103568. [CrossRef]
3. Wang, Y.; Su, Z.; Luan, T.H.; Li, R.; Zhang, K. Federated Learning with Fair Incentives and Robust Aggregation for UAV-aided Crowdsensing. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3179–3196. [CrossRef]
4. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; y Arcas, B.A. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, Lauderdale, FL, USA, 20–22 April 2017; pp. 1273–1282.
5. Weidang, L.U.; Kai, F.; Yu, D.; Bo, L.I.; Nan, Z. Research on Security, Privacy, and Energy Efficiency in Unmanned Aerial Vehicle-Assisted Federal Edge Learning Communication Systems. *J. Electron. Inf. Technol.* **2025**, *47*, 1322–1331. [CrossRef]
6. Wang, Z.; Hu, Q. Blockchain-Based Federated Learning: A Comprehensive Survey. *arXiv* **2021**, arXiv:2110.02182. [CrossRef]

7. Xu, C.; Qu, Y.; Xiang, Y.; Gao, L. Asynchronous Federated Learning on Heterogeneous Devices: A Survey. *Comput. Sci. Rev.* **2023**, *50*, 100595. [CrossRef]
8. Kim, H.; Park, J.; Bennis, M.; Kim, S.L. Blockchained On-Device Federated Learning. *IEEE Commun. Lett.* **2020**, *24*, 1279–1283. [CrossRef]
9. Ko, S.; Lee, K.; Cho, H.; Hwang, Y.; Jang, H. Asynchronous Federated Learning with Directed Acyclic Graph-Based Blockchain in Edge Computing: Overview, Design, and Challenges. *Expert Syst. Appl.* **2023**, *223*, 119896. [CrossRef]
10. Lu, Y.; Huang, X.; Zhang, K.; Maharjan, S.; Zhang, Y. Blockchain Empowered Asynchronous Federated Learning for Secure Data Sharing in Internet of Vehicles. *IEEE Trans. Veh. Technol.* **2020**, *69*, 4298–4311. [CrossRef]
11. Yuan, S.; Cao, B.; Peng, M.; Sun, Y. ChainsFL: Blockchain-Driven Federated Learning from Design to Realization. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April, 2021; pp. 1–6. [CrossRef]
12. Beilharz, J.; Pfitzner, B.; Schmid, R.; Geppert, P.; Arnrich, B.; Polze, A. Implicit Model Specialization through Dag-Based Decentralized Federated Learning. In Proceedings of the 22nd International Middleware Conference, Middleware '21, New York, NY, USA, 6–10 December 2021; pp. 310–322. [CrossRef]
13. Cao, M.; Zhang, L.; Cao, B. Toward On-Device Federated Learning: A Direct Acyclic Graph-Based Blockchain Approach. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 2028–2042. [CrossRef]
14. Huang, X.; Yin, H.; Chen, Q.; Zeng, Y.; Yao, J. DAG-based Swarm Learning: A Secure Asynchronous Learning Framework for Internet of Vehicles. *Digit. Commun. Networks* **2023**, *10*, 1611–1621. [CrossRef]
15. Xue, X.; Mao, H.; Li, Q.; Guan, X. Clustered Federated Learning Enhanced by DAG-based Blockchain with Adaptive Tip Selection Algorithm. *Internet Things* **2025**, *31*, 101573. [CrossRef]
16. Xiao, S.; Huang, X.; Deng, X.; Cao, B.; Chen, Q. DB-FL: DAG Blockchain-Enabled Generalized Federated Dropout Learning. *Digit. Commun. Netw.* **2025**, *11*, 886–897. [CrossRef]
17. Ali, A.; Ilahi, I.; Qayyum, A.; Mohammed, I.; Al-Fuqaha, A.; Qadir, J. A Systematic Review of Federated Learning Incentive Mechanisms and Associated Security Challenges. *Comput. Sci. Rev.* **2023**, *50*, 100593. [CrossRef]
18. Qu, Y.; Dai, H.; Zhuang, Y.; Chen, J.; Dong, C.; Wu, F.; Guo, S. Decentralized Federated Learning for UAV Networks: Architecture, Challenges, and Opportunities. *IEEE Netw.* **2021**, *35*, 156–162. [CrossRef]
19. Feng, C.; Liu, B.; Yu, K.; Goudos, S.K.; Wan, S. Blockchain-Empowered Decentralized Horizontal Federated Learning for 5G-Enabled UAVs. *IEEE Trans. Ind. Inform.* **2022**, *18*, 3582–3592. [CrossRef]
20. Liu, Y.; Nie, J.; Li, X.; Ahmed, S.H.; Lim, W.Y.B.; Miao, C. Federated Learning in the Sky: Aerial-Ground Air Quality Sensing Framework with UAV Swarms. *IEEE Internet Things J.* **2021**, *8*, 9827–9837. [CrossRef]
21. Wang, R.; Liu, X.; Xie, L.; Liu, Y.; Su, Z.; Liu, D.; Zhang, H. Privacy-Preserving Incentive Scheme Design for UAV-Enabled Federated Learning. In Proceedings of the 2024 IEEE Wireless Communications and Networking Conference (WCNC), Dubai, United Arab Emirates, 21–24 April 2024; pp. 1–6. [CrossRef]
22. Popov, S. The Tangle. *White Pap.* **2018**, *1*, 30.
23. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System, 2008. Available online: https://ssrn.com/abstract=3440802 (accessed on 5 January 2026).
24. Zheng, Z.; Xie, S.; Dai, H.N.; Chen, X.; Wang, H. Blockchain Challenges and Opportunities: A Survey. *Int. J. Web Grid Serv.* **2018**, *14*, 352. [CrossRef]
25. Schmid, R.; Pfitzner, B.; Beilharz, J.; Arnrich, B.; Polze, A. Tangle Ledger for Decentralized Learning. In Proceedings of the 2020 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), New Orleans, LA, USA, 18–22 May 2020; pp. 852–859. [CrossRef]
26. Wu, Y.; Huang, X.; Yin, H.; Liang, C.; Chen, Q. DAG Blockchain-based Personalized Federated Mutual Learning in Internet of Vehicles. *TechRxiv* **2023**. [CrossRef]
27. Wu, D.; Zang, X.; Chen, J.; Wu, X.; Lu, Y.; Qi, F. An Efficient Decentralized Federated Learning Framework Based on Directed Acyclic Graph. In *Proceedings of the 13th International Conference on Computer Engineering and Networks*; Zhang, Y., Qi, L., Liu, Q., Yin, G., Liu, X., Eds.; Springer: Singapore, 2024; Lecture Notes in Electrical Engineering; pp. 209–220. [CrossRef]
28. Kang, J.; Xiong, Z.; Niyato, D.; Yu, H.; Liang, Y.C.; Kim, D.I. Incentive Design for Efficient Federated Learning in Mobile Networks: A Contract Theory Approach. In Proceedings of the 2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS), Singapore, 28–30 August 2019; pp. 1–5. [CrossRef]
29. Wang, S.; Zhao, H.; Wen, W.; Xia, W.; Wang, B.; Zhu, H. Contract Theory Based Incentive Mechanism for Clustered Vehicular Federated Learning. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 8134–8147. [CrossRef]
30. Yang, D.; Ji, Y.; Kou, Z.; Zhong, X.; Zhang, S. Asynchronous Federated Learning with Incentive Mechanism Based on Contract Theory. In Proceedings of the 2024 IEEE Wireless Communications and Networking Conference (WCNC), Dubai, United Arab Emirates, 21–24 April 2024; pp. 1–6. [CrossRef]

31. Chen, Y.; Zhou, H.; Li, T.; Li, J.; Zhou, H. Multifactor Incentive Mechanism for Federated Learning in IoT: A Stackelberg Game Approach. *IEEE Internet Things J.* **2023**, *10*, 21595–21606. [CrossRef]
32. Guo, W.; Wang, Y.; Jiang, P. Incentive Mechanism Design for Federated Learning with Stackelberg Game Perspective in the Industrial Scenario. *Comput. Ind. Eng.* **2023**, *184*, 109592. [CrossRef]
33. Xiao, G.; Xiao, M.; Gao, G.; Zhang, S.; Zhao, H.; Zou, X. Incentive Mechanism Design for Federated Learning: A Two-Stage Stackelberg Game Approach. In Proceedings of the 2020 IEEE 26th International Conference on Parallel and Distributed Systems (ICPADS), Hong Kong, 2–4 December 2020; pp. 148–155. [CrossRef]
34. Huang, J.; Ma, B.; Wang, M.; Zhou, X.; Yao, L.; Wang, S.; Qi, L.; Chen, Y. Incentive Mechanism Design of Federated Learning for Recommendation Systems in MEC. *IEEE Trans. Consum. Electron.* **2023**, *70*, 2596–2607. [CrossRef]
35. He, W.; Yao, H.; Mai, T.; Wang, F.; Guizani, M. Three-Stage Stackelberg Game Enabled Clustered Federated Learning in Heterogeneous UAV Swarms. *IEEE Trans. Veh. Technol.* **2023**, *72*, 9366–9380. [CrossRef]
36. Sarikaya, Y.; Ercetin, O. Regulating Workers in Federated Learning by Yardstick Competition. In Proceedings of the 13th EAI International Conference on Performance Evaluation Methodologies and Tools, Tsukuba, Japan, 18–20 May 2020; pp. 150–155. [CrossRef]
37. Thi Le, T.H.; Tran, N.H.; Tun, Y.K.; Nguyen, M.N.H.; Pandey, S.R.; Han, Z.; Hong, C.S. An Incentive Mechanism for Federated Learning in Wireless Cellular Networks: An Auction Approach. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 4874–4887. [CrossRef]
38. Jiao, Y.; Wang, P.; Niyato, D.; Lin, B.; Kim, D.I. Toward an Automated Auction Framework for Wireless Federated Learning Services Market. *IEEE Trans. Mob. Comput.* **2021**, *20*, 3034–3048. [CrossRef]
39. Zeng, R.; Zhang, S.; Wang, J.; Chu, X. FMore: An Incentive Scheme of Multi-dimensional Auction for Federated Learning in MEC. In Proceedings of the 2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS), Singapore, 29 November–1 December 2020; pp. 278–288. [CrossRef]
40. Tang, Y.; Liang, Y.; Liu, Y.; Zhang, J.; Ni, L.; Qi, L. Reliable Federated Learning Based on Dual-Reputation Reverse Auction Mechanism in Internet of Things. *Future Gener. Comput. Syst. Int. J. Escience* **2024**, *156*, 269–284. [CrossRef]
41. Pang, J.; Yu, J.; Zhou, R.; Lui, J.C. An Incentive Auction for Heterogeneous Client Selection in Federated Learning. *IEEE Trans. Mob. Comput.* **2023**, *22*, 5733–5750. [CrossRef]
42. Wang, Y.; Su, Z.; Zhang, N.; Benslimane, A. Learning in the Air: Secure Federated Learning for UAV-assisted Crowdsensing. *IEEE Trans. Netw. Sci. Eng.* **2021**, *8*, 1055–1069. [CrossRef]
43. Chaudhuri, A. Sustaining Cooperation in Laboratory Public Goods Experiments: A Selective Survey of the Literature. *Exp. Econ.* **2011**, *14*, 47–83. [CrossRef]
44. Tran, N.H.; Bao, W.; Zomaya, A.; Nguyen, M.N.H.; Hong, C.S. Federated Learning over Wireless Networks: Optimization Model Design and Analysis. In Proceedings of the IEEE INFOCOM 2019—IEEE Conference on Computer Communications, Paris, France, 29 April–2 May 2019; pp. 1387–1395. [CrossRef]
45. Burd, T.D.; Brodersen, R.W. Processor Design for Portable Systems. *J. VLSI Signal Process. Syst. Signal Image Video Technol.* **1996**, *13*, 203–221. [CrossRef]
46. Li, C.C. The Statistical Processes of Evolutionary Theory. *Am. J. Hum. Genet.* **1962**, *14*, 438–439.
47. Imhof, L.A.; Nowak, M.A. Evolutionary Game Dynamics in a Wright-Fisher Process. *J. Math. Biol.* **2006**, *52*, 667–681. [CrossRef]
48. Liu, X.; Pan, Q.; Kang, Y.; He, M. Fixation Probabilities in Evolutionary Games with the Moran and Fermi Processes. *J. Theor. Biol.* **2015**, *364*, 242–248. [CrossRef] [PubMed]
49. Liu, X.; Pan, Q.; Kang, Y.; He, M. Fixation Times in Evolutionary Games with the Moran and Fermi Processes. *J. Theor. Biol.* **2015**, *387*, 214–220. [CrossRef] [PubMed]
50. Stahl, D.O.; Haruvy, E. Aspiration-Based and Reciprocity-Based Rules in Learning Dynamics for Symmetric Normal-Form Games. *J. Math. Psychol.* **2002**, *46*, 531–553. [CrossRef]
51. Platkowski, T. Enhanced Cooperation in Prisoner's Dilemma with Aspiration. *Appl. Math. Lett.* **2009**, *22*, 1161–1165. [CrossRef]
52. Hu, Y.; Huang, H.; Yu, N. Device Scheduling for Energy-Efficient Federated Learning over Wireless Network Based on TDMA Mode. In Proceedings of the 2020 International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, China, 21–23 October 2020; pp. 286–291. [CrossRef]
53. Tang, M.; Wong, V.W. An Incentive Mechanism for Cross-Silo Federated Learning: A Public Goods Perspective. In Proceedings of the IEEE INFOCOM 2021—IEEE Conference on Computer Communications, Vancouver, BC, Canada, 10–13 May 2021; pp. 1–10. [CrossRef]
54. Krizhevsky, A. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.