remote sensing

MDPI

Article

UAV-Assisted Fair Communication for Mobile Networks: A Multi-Agent Deep Reinforcement Learning Approach

Yi Zhou [1,2], Zhanqi Jin [1,2], Huaguang Shi [1,2,*], Zhangyun Wang [1,2], Ning Lu [3] and Fuqiang Liu [4]

[1] School of Artificial Intelligence, Henan University, Zhengzhou 450046, China
[2] International Joint Research Laboratory for Cooperative Vehicular Networks of Henan, Zhengzhou 450046, China
[3] Department of Electrical and Computer Engineering, Queen's University, Kingston, ON K7L 3N6, Canada
[4] College of Electronic and Information Engineering, Tongji University, Shanghai 201804, China
* Correspondence: shihuaguang@henu.edu.cn

Abstract: Unmanned Aerial Vehicles (UAVs) can be employed as low-altitude aerial base stations (UAV-BSs) to provide communication services for ground users (GUs). However, most existing works mainly focus on optimizing coverage and maximizing throughput, without considering the fairness of the GUs in communication services. This may result in certain GUs being underserviced by UAV-BSs in pursuit of maximum throughput. In this paper, we study the problem of UAV-assisted communication with the consideration of user fairness. We first design a Ratio Fair (RF) metric by weighting fairness and throughput to evaluate the tradeoff between fairness and communication efficiency when UAV-BSs serve GUs. The problem is formulated as a mixed-integer non-convex optimization problem based on the RF metric and we propose a UAV-Assisted Fair Communication (UAFC) algorithm based on multi-agent deep reinforcement learning to maximize the fair throughput of the system. The UAFC algorithm comprehensively considers fair throughput, UAV-BSs coverage, and flight status to design a reasonable reward function. In addition, the UAFC algorithm establishes an information sharing mechanism based on gated functions by sharing neural networks, which effectively reduces the distributed decision-making uncertainty of UAV-BSs. To reduce the impact of state dimension imbalance on the convergence of the algorithm, we design a new state decomposing and coupling actor network architecture. Simulation results show that the proposed UAFC algorithm increases fair throughput by 5.62%, 26.57% and fair index by 1.99%, 13.82% compared to the MATD3 and MADDPG algorithms, respectively. Meanwhile, UAFC can also meet energy consumption limitation and network connectivity requirement.

Keywords: Unmanned Aerial Vehicles (UAVs); Multi-Agent Deep Reinforcement Learning (MADRL); fair communication; information sharing mechanism

Citation: Zhou, Y.; Jin, Z.; Shi, H.; Wang, Z.; Lu, N.; Liu, F. UAV-Assisted Fair Communication for Mobile Networks: A Multi-Agent Deep Reinforcement Learning Approach. *Remote Sens.* **2022**, *14*, 5662. https://doi.org/10.3390/rs14225662

Academic Editors: Rui Chen and Nan Cheng

Received: 5 October 2022
Accepted: 5 November 2022
Published: 9 November 2022

1. Introduction

At present, the establishment and realization of mobile communication networks mainly rely on terrestrial base stations and other fixed communication equipment, which requires time-consuming network planning with the consideration of many practical factors. Unmanned Aerial Vehicles (UAVs) with high flexibility, low cost, and wide coverage have aroused widespread concern in academia and industry [1,2], e.g., UAV-assisted base station communications [3], relay communications [4], data collection [5], and secure communications [6]. In the area of UAV-assisted communications, UAVs are mainly regarded as mobile base stations to provide high-quality communication services to ground users (GUs) [7]. The mobility and flexibility of UAV Base Stations (UAV-BSs) can establish communication connections quickly and improve data transmission efficiency and communication range significantly [8–11]. For example, when the ground communication infrastructure is damaged by natural disasters, UAVs can be employed as temporary base stations to provide emergency communication services for GUs.

UAVs as aerial base stations have many advantages compared to terrestrial base stations. (1) UAVs can establish good Line-of-Sight (LoS) links with GUs [12]. (2) For mobile GUs, UAVs can adjust flight trajectory or follow the GUs to provide better communication services [13]. It is worth noting that due to limited communication resources and coverages, UAV-BSs cannot serve all GUs by merely optimizing the locations of UAV-BSs. Thus, UAV-BSs mainly face two challenges in UAV-assisted communication. (1) How to select the GUs to be served by optimizing the locations of the UAVs to maximize the communication efficiency (e.g., throughput). (2) How to adopt a service strategy to ensure fairness among GUs when providing communication services for multiple GUs.

For the problem of maximizing the efficiency of UAV-assisted communication, most studies regard the energy efficiency and throughput as the primary optimization objective. In [14], the authors proposed a centralized multi-agent Q-learning algorithm to maximize the energy efficiency of wireless communication. In [15], a Deep Reinforcement Learning (DRL) algorithm based on Q-Learning and convolutional neural networks is proposed to maximize spectral efficiency. However, maximizing communication efficiency results in the tendency of UAV-BSs to hover approaching a small subset of GUs. This will lead to communication mission interruptions for other GUs due to being out of the communication range of the UAV-BSs. For the fair communication problem, previous works focus on fair coverage. For example, an algorithm based on DRL is proposed in [16] to deploy UAV-BSs. UAV-BSs can provide fair coverage and reduce collisions among UAVs. In [17], a DRL-based control algorithm is proposed to implement energy efficient and fair coverage with energy recharge. A mean-field game model is proposed in [18] to maximize the coverage score while ensuring fair communication range and network connectivity. These studies [16–18] focus on fair coverage of ground areas or users. In the pursuit of regional fairness, UAVs need to serve all cell as much as possible. Thus, focusing only on fair coverage will cause partial task interruption and the degradation of communication efficiency. It is necessary to consider both communication efficiency (throughput) and user fairness in UAV-assisted communications, which has been overlooked in the literature [14–18].

1.1. Related Work

In this subsection, we review relevant work on UAV-assisted communication and point out the inadequacy of these works.

1.1.1. UAV-Assisted Communication

UAV-assisted communication has been extensively researched. For example, Jeong et al. [19] proposed an optimization algorithm based on a concave-convex procedure to maximize the transmission rate by designing the flight trajectory and the transmitting power of the UAV. Yin et al. [20] studied UAV as aerial base stations serving multiple GUs and proposed a deterministic policy gradient algorithm to maximize the total uplink transmission rate. These works [19,20] focus only on the communication performance of UAV-assisted communication networks, the energy consumption of the UAV is also a crucial issue due to the limited onboard energy. In [21], an Actor–Critic-based deep stochastic online scheduling algorithm is proposed to minimize the overall energy consumption of the communication network by optimizing the data transmission and hovering time of the UAV. The proposed algorithm can reduce energy by 29.94% and 24.84% compared to the DDPG and PPO algorithms. Yang et al. [22] investigated UAV-assisted data collection, where the data transmission rate and the energy consumption of the UAV are in conflict with each other during the data collection process. Theoretical expressions for the energy consumption of the UAV and GUs are derived to achieve different Pareto-optimal trade-offs. The results provide a new insight for future energy efficiency of UAV-assisted communication. Zhang et al. [23] considered post-disaster rescue scenarios where energy is limited due to the collapse of the power system. To satisfy energy constraints and obstacle constraints, a safe deep Q-learning-based UAV trajectory optimization algorithm is proposed to maximize uplink throughput. Its weakness is that it cannot be applied to a

larger disaster area due to the limited communication range and on-board energy of single UAV. The above studies [21–23] take full consideration of the energy consumption of the UAV during the trajectory design, enabling the UAV to perform communication services with greater energy efficiency. These studies [19–23] consider single UAV scenarios and the GUs are stationary. Thus, the methods designed in the above references are only applicable to small-scale simple scenarios.

For complex scenarios, it is particularly important that multiple UAVs collaborate with each other to accomplish complex communication tasks. Shi et al. [24] proposed a dynamic deployment algorithm based on hierarchical Reinforcement Learning (RL) to maximize the long-term desired average throughput of the communication network. The proposed algorithm can increase throughput by 40%. To meet the Quality of Experience (QoE) of all GUs with limited system resources and energy, Zeng et al. [25] jointly optimized GUs scheduling, UAVs trajectory, and transmit power to maximize energy efficiency and meet GUs QoE. The proposed algorithm increases energy efficiency by 12.5% compared to the baseline algorithm. Ding et al. [26] modeled the UAVs and GUs as a hybrid cooperative-competitive game problem, maximizing throughput by simultaneously optimizing the trajectory of UAVs and the access of GUs. The above studies [24–26] focus on the multiple UAVs and multiple GUs scenario to maximize throughput and energy efficiency by designing the flight trajectory. However, the optimization objectives of [19–26] focus on communication performance and ignore fairness among GUs. This leads to UAVs allocating more communication resources to GUs with high throughput or following the movement of the GUs. As a result, the UAV-BSs ignore service requests of other critical GUs.

1.1.2. UAV-Assisted Fair Communication

It is also a crucial issue to consider service fairness in UAV-assisted communication. Diao et al. [27] studied the problem of fair perceptual task allocation and trajectory optimization in UAV-assisted edge computing. The non-convex optimization problem is transformed into multiple convex sub-problems for solution by introducing auxiliary variables to minimize energy consumption while meeting fairness. In [28], Ding et al. derived an expression for the energy consumption of UAV. Based on the energy consumption of UAV and the fairness of GUs, a DRL-based algorithm is proposed to maximize the system throughput by jointly optimizing the trajectory and bandwidth allocation. The proposed method increases the fair index by 15.5%. The works in [27,28] study the issue of single UAV-assisted fairness communication and the proposed algorithms are not applicable to multi-UAV scenarios.

For multi-UAV scenarios, Liu et al. [29] studied the fair coverage problem of UAVs and proposed a DRL-based control algorithm to maximize the energy efficiency while guaranteeing communication coverage, fairness, energy consumption, and connectivity. The study improves coverage scores and fairness index by 26% and 206%. A novel distributed control scheme algorithm is proposed in [30] to deploy multiple UAVs in an area to improve coverage with minimum energy consumption and maximum fairness. The proposed algorithm covers 84.6% of the area and improves the fair index by 3% on the same network conditions. Liu et al. [31] investigated how to deploy UAVs to improve service quality for GUs and maximize fair coverage according to the designed fair index. The above studies [29–31] focus on the problem of fair coverage in multi-UAV-assisted communication by optimizing the locations of UAVs to cover the ground area in a fair manner. In contrast with the above studies, we are concerned with maximizing system throughput while considering user fairness in a multi-UAV mobile network.

1.2. Motivation and Contribution

UAV-assisted mobile wireless communication networks have many advantages compared with traditional terrestrial fixed communication infrastructures. However, UAVs as mobile base stations still face the following two problems. (1) Existing research mainly focuses on the communication efficiency. The research objectives aim to maximize communication metrics such as throughput, transmission rate, and energy efficiency by optimizing the flight trajectory and resource allocation of the UAVs. However, these studies ignore the fairness among GUs. For example, service user A can obtain high throughput and user B needs more urgent communication service. If we select throughput as the service goal, it will cause the UAVs to allocate more communication resources to user A maximizing the goal. Thus, the service request of user B is ignored despite the more urgent service request. (2) Another important problem in UAV-assisted communication is the optimization problem of UAV location, which is a typical optimal sequential decision problem. The computational complexity of heuristic and convex optimization algorithms grows exponentially with the increase of numbers of GUs and UAVs, which is not suitable for multi-UAV and multi-user scenarios.

Motivated by communication efficiency and user fairness, this paper investigates fair communication in UAV-BSs-assisted communication systems. We propose an information sharing mechanism based on gated functions and incorporate it into Multi-Agent Deep Reinforcement Learning (MADRL) to obtain near-optimal strategies. The main contributions of this paper are summarized as follows:

- To evaluate the trade-off between fairness and communication efficiency, we design a Ratio Fair (RF) metric by weighing fairness and throughput when UAV-BSs serve GUs. Based on the RF metric, we formulate the UAV-assisted fair communication as a non-convex problem and utilize DRL to acquire a near-optimal solution for this problem;
- To solve the above continuous control problem with an infinite action space, we propose the UAV-Assisted Fair Communication (UAFC) algorithm. The UAFC algorithm establishes an information sharing mechanism based on gated functions to reduce the distributed decision uncertainty of UAV-BSs;
- To address the dimension imbalance and training difficulty due to the high dimension of the state space, we design a novel actor network structure of decomposing and coupling. The actor network utilizes dimension spread and state aggregation to obtain high-quality state information.

The remainder of this paper is organized as follows. Section 2 introduces energy consumption and communication models. In Section 3, we describe the problem and formulate it as a Markov decision process. Section 4 describes the implementation process of the UAFC algorithm. The results and analyses of the experiments are presented in Section 5. We discuss some of the limitations of this paper and future research work in Section 6. Section 7 concludes our paper.

Notations: In this paper, variables are denoted by italicized notation and vectors are denoted by bold notation. $||\mathbf{.}||$ denotes L2 parametric; $\mathbb{R}^W$ denotes a W-dimensional vector space. $\{\mathbf{.}\}$ denotes set. For convenience of reading, the important symbols are listed in Table 1 with the corresponding descriptions.

Table 1. Table of Important Symbols.

Symbol	Description	Symbol	Description
$\mathcal{D}, D, d$	Set, number, and index of UAVs	H	The hover altitude of the UAVs
$\mathcal{K}, K, k$	Set, number, and index of GUs	$d_{\min}$	Minimum safe distance
$\boldsymbol{u}_d(t), \boldsymbol{\omega}_k(t)$	Location of the UAVs and GUs	s_d, a_d	The states and actions of the UAV_d
$dist_d(t), \vartheta_n(t)$	Flight distance and flight direction	r	Reward value
$E_m(t)$	Propulsion energy consumption	$\pi_d^{\mu}, Q_d^{\theta_i}$	Actor and critic networks
$E_c(t)$	Communication energy consumption	$\pi_d^{\mu'}, Q_d^{\theta_i'}$	Target actor and critic networks
$E_{\max}, E_d(t)$	Maximum energy and residual energy of UAV_d	μ_n, μ_n'	Parameters of actor and target actor networks
f_c, P_t	Carrier frequency and transmit power	θ_n^1, θ_n^2	Parameters of critic and target critic networks
X, Y	The parameter of path loss model	λ, u	Learning and updating rate
E_{LoS}, E_{NLoS}	Additional path loss for LoS and NLoS links	M_b, M_r	Buffer size and mini-batch

2. System Model

In this paper, we consider a wireless communication scenario in an area. The scenario contains multi-UAV and mobile GUs. UAVs provide communication services for GUs by optimizing the trajectories, as shown Figure 1. Air-to-Ground (A2G) links exist between UAV-BSs and GUs. K GUs are randomly distributed, and the set of GUs is denoted as $\mathcal{K} \triangleq \{1, 2, \ldots, K\}$. The location of GU_k $(k \in \mathcal{K})$ at time t is denoted as $\boldsymbol{\omega}_k(t) = [x_k(t), y_k(t), 0] \in \mathbb{R}^3$. D UAVs are deployed as mobile base stations to provide communication services for GUs, and the set of UAV-BSs is denoted as $\mathcal{D} \triangleq \{1, 2, \ldots, D\}$. In the three-dimensional Cartesian coordinate system, the location of UAV_d $(d \in \mathcal{D})$ at time t is denoted as $\boldsymbol{u}_d(t) = [x_d(t), y_d(t), z_d(t)] \in \mathbb{R}^3$. To reduce the additional energy overhead in the climbs of UAVs, we assume that UAVs fly at a fixed altitude H, i.e., $z_d = H$ $(d \in \mathcal{D})$.

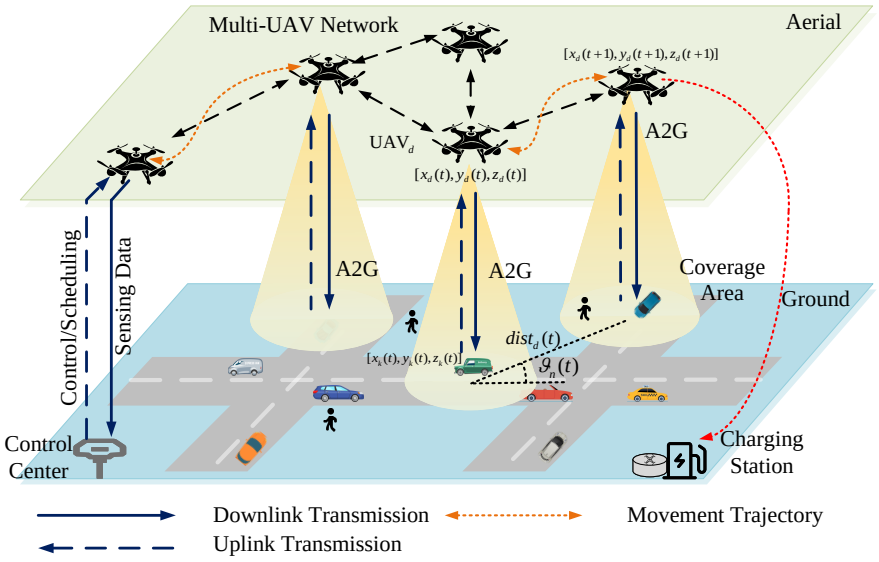


Figure 1. Multi-UAV-assisted communication scenario.

2.1. UAV Movement and Energy Consumption Model

The movement of GUs leads to the change of the channel quality between UAVs and GUs. Thus, the position of the UAV-BSs needs to be optimized to provide better communication services. As shown in Figure 1, the coordinate of UAV_d at time t is denoted as $[x_d(t), y_d(t), z_d(t)]$. UAV_d flies towards the next position according to the moving distance $dist_d(t)$ and the flight angle $\vartheta_d(t)$, where the maximum flying distance and flight angle of UAV_d are denoted as $dist_d(t) \in [0, dist_{\max}]$ and $\vartheta_d(t) \in [0, 2\pi]$, respectively. Therefore, the next position of UAV_d is calculated as

$$\begin{cases} x_d(t+1) = x_d(t) + dist_d(t) \cdot \cos(\vartheta_d(t)), \\ y_d(t+1) = y_d(t) + dist_d(t) \cdot \sin(\vartheta_d(t)). \end{cases} \tag{1}$$

The energy consumption of UAVs mainly depends on propulsion energy consumption and communication energy consumption. According to [32], the propulsion energy consumption is related to the speed and acceleration of the UAV. To simplify the system model and computational complexity, the effect of acceleration on propulsion energy consumption is ignored [33]. In time t, the energy consumption $E_m(t)$ due to the movement of the UAV_d can be expressed as

$$E_m(t) = \int_0^t P_d(\tau) d\tau, \tag{2}$$

where $P_d(\tau)$ is the propulsion power of UAV_d at time τ. The remaining energy of UAV_d (denoted as $E_d(t)$) is calculated as

$$E_d(t) = E_{\max} - (E_m(t) + E_c(t)), \tag{3}$$

where $E_{\max}$ is the maximum energy value of UAV_d being fully charged. The communication energy consumption of UAV_d in the [0, t] period is denoted as $E_c(t)$.

2.2. Communication Model

The transmission links between UAVs and GUs are modeled as a probabilistic channel model [34], and the probability $P_{LoS}(t)$ of establishing LoS connection between UAVs and GUs is given by

$$P_{LoS}(t) = \frac{1}{1 + X \exp\{-Y(\arctan(\frac{z_d(t)}{r_{d,k}(t)}) - X)\}}, \tag{4}$$

where X and Y are the coefficients related to the environment, respectively. $r_{d,k}(t)$ denotes the horizontal distance between UAV_d and GU_k.

The link path loss models between UAV_d and GU_k are given for the LoS link and Non-Line-of-Sight (NLoS) link (denoted as L_{LoS} and L_{NLoS}), respectively, as follows:

$$\begin{aligned} L_{LoS} &= 20 \log(\tfrac{4\pi f_c d_{d,k}(t)}{c}) + E_{LoS}, \\ L_{NLoS} &= 20 \log(\tfrac{4\pi f_c d_{d,k}(t)}{c}) + E_{NLoS}, \end{aligned} \tag{5}$$

where $d_{d,k}(t)$ denotes the distance between UAV_d and GU_k, and f_c denotes the carrier frequency. E_{LoS} and E_{NLoS} denote the additional path loss of the LoS link and NLoS link [35], respectively. $20 \log(\frac{4\pi f_c d_{d,k}(t)}{c})$ is the free space path loss. c is the speed of light.

Therefore, the average path loss between UAV_d and GU_k (denoted as $PL_{d,k}(t)$) is calculated by

$$PL_{d,k}(t) = P_{LoS}(t) \times L_{LoS} + (1 - P_{LoS}(t)) \times L_{NLoS}. \tag{6}$$

In this model, the path loss threshold γ_{dk} is defined, and the link is considered broken when $PL_{d,k}(t) \geq \gamma_{dk}$. Thus, a binary variable $\beta_{d,k}(t)$ is defined, which denotes the association of the UAV_d to the GU_k at time t.

$$\beta_{d,k}(t) = \begin{cases} 1, & if\ \mathrm{UAV}_d\ is\ connected\ to\ \mathrm{GU}_k. \\ 0, & otherwise. \end{cases} \tag{7}$$

The transmission rate between UAV_d and GU_k (denoted as $R_d^k(t)$) is expressed as

$$R_d^k(t) = B_k \beta_{d,k}(t) \log_2(1 + \frac{P_t}{n_0 \times G_{d,k}(t)}), \tag{8}$$

where $G_{d,k}(t) = 10^{PL_{d,k}(t)/10}$ denotes the channel power gain, P_t is the fixed transmit power of the UAV_d, B_k indicates the communication bandwidth allocated to GU_k, and n_0 represents the noise power spectral density.

3. Problem Formulation and Transformation

3.1. Problem Formulation

To maximize the system throughput while guaranteeing user fairness, we design an RF metric to evaluate the trade-off between fairness and communication efficiency when UAVs serve GUs. The aim is to provide communication services for GUs with high communication efficiency. The throughput priority for GU_k (denoted as $f_k(t)$) and is calculated by

$$f_k(t) = \frac{\overline{R}_k(t)}{\overline{T}(t)}, \tag{9}$$

where $\overline{R}_k(t)$ represents the total throughput of GU_k in the period [0, t], and $\overline{T}(t)$ is the throughput of all GUs. $\overline{R}_k(t)$ and $\overline{T}(t)$ are given by Equations (10) and (11), respectively.

$$\bar{R}_k(t) = \int_0^t R_d^k(\tau) d\tau \tag{10}$$

$$\bar{T}(t) = \sum_{k=1}^{K} \bar{R}_k(t) \tag{11}$$

However, UAVs tend to follow GUs with high throughput to maximize the total throughput and ignore service request from other GUs, leading to unfairness among GUs. Due to the unfair behavior of GUs, the communication resource allocation of UAVs is unbalanced, which affects the quality of service for GUs. To maximize throughput while ensuring user fairness, we design the RF metric based on priority and the Jain's index [36] to measure the user fairness, which can be calculated by the following formula

$$W_k(t) = \frac{\left(\sum_{k=1}^{K} f_k(t)\right)^2}{K\left(\sum_{k=1}^{K} f_k(t)^2\right)}. \tag{12}$$

Then, the RF index is employed to obtain a weight coefficient that considers fairness and priority comprehensively, and the total fair throughput is defined as weighted throughput and is denoted as

$$\overline{R}_{toal}(t) = \sum_{k=1}^{K} \int_0^{T_t} W_k(\tau) \overline{R}_k(\tau) d\tau. \tag{13}$$

The optimization objective of this paper is to maximize the fair throughput by optimizing the location of the UAVs (referred to as problem $\mathbb{P}1$). Problem $\mathbb{P}1$ can be formulated as follows:

$$\begin{aligned}
(\mathbb{P}1): &\max_{\{u_d(t)\}_{d\in\mathcal{D}}} \overline{R}_{toal}(t) = \sum_{k=1}^{K}\int_0^{T_t} W_k(\tau)\overline{R}_k(\tau)d\tau \\
s.t.\ & C1: E_d(0) = E_{\max}, E_d(T_t) = E_{\min}, \\
& C2: \sum_{k=1}^{K}\beta_{d,k}(t) \le 1,\ \forall d \in \mathcal{D}, \forall k \in \mathcal{K}, \\
& C3: \beta_{d,k}(t) \in \{0,1\},\ \forall d \in \mathcal{D}, \forall k \in \mathcal{K}, \\
& C4: PL_{d,k}(t) \le \gamma_{dk}, \\
& C5: \| u_i(t) - u_j(t)\|_2 \ge d_{\min}, \forall i,j \in \mathcal{D}, \text{and } i \ne j, \\
& C6: x_d(t), x_k(t) \in [X_{\min}, X_{\max}],\ \forall d \in \mathcal{D}, \forall k \in \mathcal{K}, \\
& C7: y_d(t), y_k(t) \in [Y_{\min}, Y_{\max}],\ \forall d \in \mathcal{D}, \forall k \in \mathcal{K},
\end{aligned} \tag{14}$$

where constraint C1 represents the energy consumption constraint of the UAV-BSs. Constraints C2 and C3 indicate that a GU can only connect to one UAV-BS at time t. Constraint C4 requires that the path loss between the UAV_d and the GU_k should not be greater than the threshold to avoid transmission interruptions. Constraint C5 represents the safe distance between UAV_i and UAV_j. Constraints C6 and C7 represent the movement area constraints of UAVs and GUs.

3.2. Problem Transformation

Since the locations of the UAV-BSs change continuously, the optimization variables are continuous and exist nonlinear coupling. To make the problem $\mathbb{P}1$ trackable, the entire task time is divided into N_t timeslots and the duration of each timeslot is expressed as $\delta_t = T_t / N_t$. Thus, the continuous optimization problem $\mathbb{P}1$ can be transformed into discrete problem $\mathbb{P}2$

$$\begin{aligned}
(\mathbb{P}2): &\max_{\{u_d(n)\}_{d\in\mathcal{D}}} \overline{R}_{toal}(n) = \sum_{n=1}^{N_t}\sum_{k=1}^{K} W_k(n)\overline{R}_k(n)\delta_t \\
s.t.\quad & C1 \sim C7.
\end{aligned} \tag{15}$$

Since the constraints are non-convex, problem $\mathbb{P}2$ is a complex non-convex optimization problem. Traditional heuristic algorithms [37–39] can obtain optimal strategy at the expense of high computational complexity and are not suitable for dynamic environment. DRL is a learning-based approach in which agents obtain optimal strategies by interacting with the environment and it requires little prior experience. Thus, DRL is commonly employed to solve optimal decision problems. However, single-agent DRL algorithms are not applicable to multi-agent problems. The main reason is that a centralized controller is needed to collect global information and control all the agents, leading to the increase of communication costs [40]. To address the problem, the MADRL algorithm can be employed, in which each UAV acts as an agent to learn the optimal collaboration policy.

First, the problem $\mathbb{P}2$ is described as a multi-agent Markov Decision Process (MDP) which consist of five parts $< S, A, P, R, \gamma >$ [41]: The state set S, the action set A, the state transition probability function P, the reward function R, and the reward discount factor γ. The state space, action space, and reward function are designed as follows:

State: In time slot $n \in [0, N_t]$, state $s_d = \{\{\boldsymbol{u}_d(n)\}_{\forall d\in\mathcal{D}}, \{\boldsymbol{\omega}_k(n)\}_{\forall k\in\mathcal{K}}, E_d(n)_{\forall d\in\mathcal{D}}\}$ consists of three parts.

- $\{\boldsymbol{u}_d(n)\}_{\forall d\in\mathcal{D}}$ represents the coordinates of UAV_d at time slot n;
- $\{\boldsymbol{\omega}_k(n)\}_{\forall k\in\mathcal{K}}$ represents the coordinates of the GU_k at time slot n;
- $\{E_d(n)\}_{\forall d\in\mathcal{D}}$ represents the remaining energy of UAV_d at time slot n.

Action: In time slot $n \in [0, N_t]$, action $a_d = \{dist_d(n), \vartheta_d(n)\}_{\forall d\in\mathcal{D}}$ consists of two parts;

- $dist_d(n) \in [0, V_d(t)\delta_t]$ represents the distance that UAV_d flies in time slot n. $V_d(t)$ represents the maximum flight speed;
- $\vartheta_d(n) \in [0, 2\pi]$ represents the direction of UAV_d in time slot n.

Reward: Since the goal of the action taken by the agents is to maximize the system reward, the setting of the reward function plays an important role in MADRL. The reward mainly includes the following three components:

- Fair throughput $r_1 = \sum_{k=1}^{K} W_k(n)\overline{R}_k(n)\delta_t$: In the UAV-assisted fair communication problem, to trade off the user fairness and communication efficiency, we define the weighted sum of the fairness index and throughput as fair throughput and as part of the reward function. $W_k(n)$ is an RF metric utilized to weigh communication efficiency and communication fairness.
- Coverage reward $r_2 = \sum_{d=1}^{D}\sum_{k=1}^{K} e_{d,k}$: To accelerate the convergence of the UAFC algorithm, we design the coverage reward of the UAV in the reward function. The coverage reward is proportional to the number of GUs covered by the UAVs. $e_{d,k} = 1$ indicates that GU_k is covered by UAV_d, and $e_{d,k} = 0$ otherwise. Note that the coverage range is not strictly a communication range, and covering more GUs only provides a direction for the UAVs to search for the optimal strategy.
- Punishment: The UAVs will receive large negative reward when one of the following requirements are fulfilled:
  (1) The UAVs fly out of the mission boundary area, i.e., $x_{d,k}(t) \notin [X_{\min}, X_{\max}]$ or $y_{d,k} \notin [Y_{\min}, Y_{\max}]$, where $X_{\min}$, $X_{\max}$, $Y_{\min}$, and $Y_{\max}$ represent the values of the abscissa and ordinate of the mission area, respectively;
  (2) UAV_i and UAV_j collide with each other, i.e., $\| u_i(t) - u_j(t)\|_2 \leq d_{\min}$, where $d_{\min}$ represents the safety distance threshold;
  (3) The remaining energy of the UAV_d is lower than the threshold, i.e., $E_d \leq E_{\min}$. A binary variable $\xi_i \in \{0, 1\}$ is employed to indicate whether violation occurs in the above condition. $\xi_i = 1(i \in \{1, 2, 3\})$ means that violation occurs and a fixed penalty p_i $(i \in \{1, 2, 3\})$ will be given to the UAVs.

  In summary, the reward function is formulated as

$$r = r_1 + r_2 - \xi_1 p_1 - \xi_2 p_2 - \xi_3 p_3. \tag{16}$$

In MDP, the UAVs aim to maximize the reward function by optimizing policy π and thus the problem P2 is rephrased as

$$\begin{aligned} &\max_{\pi} \mathrm{E}\Big(\sum_{n=1}^{N_t} r | \pi, s, a\Big) \\ &s.t. C1 \sim C7. \end{aligned} \tag{17}$$

4. The UAFC Algorithm

Since multi-agent systems are sensitive to the change in the training environment [42], the policies obtained by agents may fall into local optimization. The Multi-Agent Twin Delayed Deep Deterministic policy gradient (MATD3) algorithm [43] is based on the Actor–Critic architecture and incorporates policy smoothing technique in the actor network. The target policy smoothing technique is utilized to compute the target Q value, which is beneficial to improving the accuracy of the target Q value and ensure the stability of the training process. Thus, the proposed UAFC algorithm employs the MATD3 algorithm as the basic algorithm and adopts the MADRL framework with centralized training and distributed execution [44] as shown in Figure 2. In the centralized training stage, the MATD3 algorithm learns a policy by jointly modeling all agents. Specifically, the observations of all the agents are employed as input to the actor network, which outputs the joint actions of the agents. Thus, the problem of environment non-stationarity is solved according to

centralized training. In the distributed execution stage, the UAVs cannot fully obtain the state information of the environment and other agents due to the limited perception ability. Thus, the unknown state information results in the uncertainty of strategy and makes it challenging for the agent to obtain the optimal strategy quickly. To reduce the distributed decision-making uncertainty of UAVs, the information-sharing based on gated functions is designed in the UAFC algorithm.

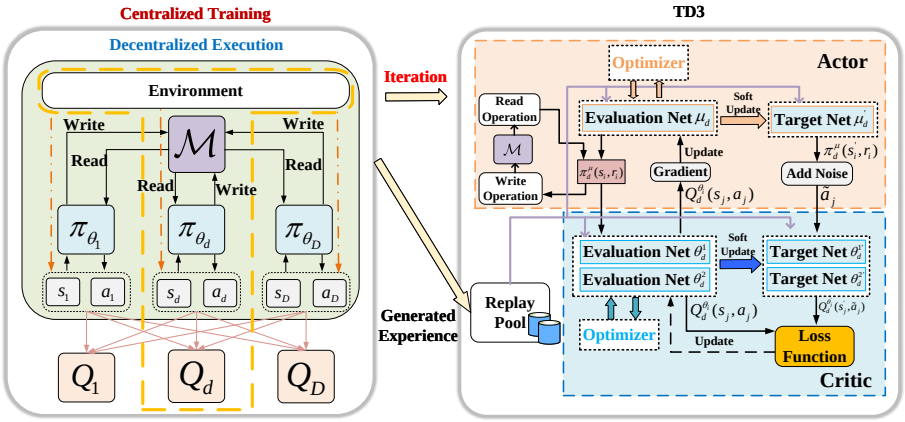


Figure 2. The architecture of the UAFC algorithm.

4.1. MATD3 Algorithm

As shown in Figure 2, agents adopt the TD3 algorithm. Two main techniques are introduced to enhance the performance of the TD3 algorithm: clipped double-Q learning and target policy smoothing.

- Clipped Double-Q Learning: The TD3 algorithm consists of an actor network with parameter $\boldsymbol{\mu}_d$ and two critical networks with network parameters $\boldsymbol{\theta}_d^1$ and $\boldsymbol{\theta}_d^2$, respectively. We assume that the actions, states, and rewards of all agents are accessible during training. The actor network makes decisions based on the local state information, and the critic network utilizes the state–action pair to learn two centralized evaluation functions $Q_d^{\theta i}(s(t), a(t))(i \in \{1,2\})$ to evaluate the policy. To avoid the overestimate of the Q value in a single critical network, the Q value is updated with the minimum value of the two critic networks. Thus, the target values y_i can be formulated as

$$y_i = r_i + \delta \min_{i=1,2} Q_d^{\theta i}(s^{'}, \tilde{a}),\ i = 1,2. \tag{18}$$

  where $s^{'}$ indicates next moment state, $\tilde{a}$ denotes the action generated by the target actor network.
- Target Policy Smoothing: Furthermore, clipped Gaussian noise ξ is added to the actor network to prevent overfitting of the Q value, which can achieve smoother state–action estimation and the modified target action.

4.2. Information Sharing Mechanism Based on Gated Functions

In addition, to reduce the uncertainty of distributed decision-making. The information-sharing mechanism based on gated functions is designed, which enables UAVs to establish state information sharing through a central memory $\mathcal{M}$ with a storage capacity of M [45]. The memory is used to store the collective state information $m \in \mathbb{R}^M$ of the UAVs. As shown in Figure 2, with the information sharing mechanism, the strategy of each UAV becomes $s_d \times \mathcal{M}(d \in \mathcal{D})$. The policy is determined by observation s_d and the information in the memory. Each UAV accesses the central memory to retrieve information shared by other UAVs before taking action. The neural networks are utilized to build policy networks for DRL. Furthermore, the gated functions are employed to characterize the information interaction between the agent and memory.

4.2.1. Encoding and Reading Operations

The encoding operation and reading operation are shown in Figure 3. Each UAV maps its own state vector to an embedding vector (denoted as $\mathbf{e}_d$) representing the state information and is given by

$$\mathbf{e}_d = \varphi^{enc}_{\boldsymbol{\theta}^e_d}(s_d), \tag{19}$$

where $\varphi^{enc}_{\boldsymbol{\theta}^e_d}$ is a neural network with network parameters $\boldsymbol{\theta}^e_d$.

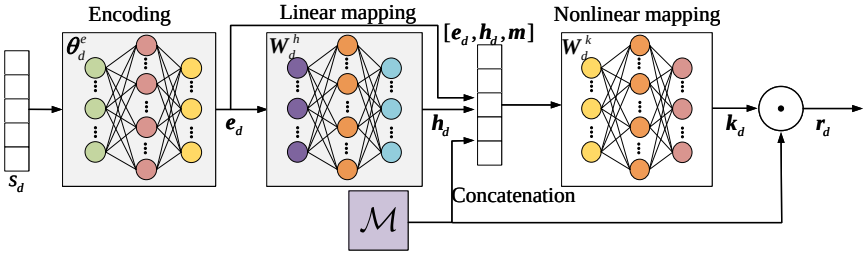


Figure 3. Encoding and reading operations.

The UAVs perform the reading operation to extract the associated information stored in $\mathcal{M}$ after encoding the current information. A latent vector $\boldsymbol{h}_d$ is generated to learn the temporal and spatial dependency information of the embedded vector $\mathbf{e}_d$

$$\boldsymbol{h}_d = \boldsymbol{W}^h_d \boldsymbol{e}_d, \quad \boldsymbol{h}_d \in \mathbb{R}^H, \boldsymbol{W}^h_d \in \mathbb{R}^{H \times E}, \tag{20}$$

where $\boldsymbol{W}^h_d$ denotes the network parameters of the linear mapping. H denotes the dimension of the context vector and E denotes the dimension of the embedding vector. The state embedding vector $\boldsymbol{e}_d$, the context vector $\boldsymbol{h}_d$, and content $\boldsymbol{m}$ in current memory $\mathcal{M}$ contain different information, respectively.

$\boldsymbol{e}_d$, $\boldsymbol{h}_d$, and $\boldsymbol{m}$ are employed jointly as input to learn a gated mechanism. $\boldsymbol{k}_d$ is utilized as a weighting factor to adjust the information reading from the memory and is given by

$$\boldsymbol{k}_d = \sigma(\boldsymbol{W}^k_d[\boldsymbol{e}_d, \boldsymbol{h}_d, \boldsymbol{m}]), \quad \boldsymbol{k}_d \in [0,1]^M, \boldsymbol{W}^k_d \in \mathbb{R}^{M \times (E+H+M)}, \tag{21}$$

where $[\boldsymbol{e}_d, \boldsymbol{h}_d, \boldsymbol{m}]$ denotes the concatenation operation of the vectors and $\sigma(\boldsymbol{.})$ conducts the calculation of the sigmoid activation function. M represents the dimension of the content $\boldsymbol{m}$. Thus, the information reading from $\mathcal{M}$ (denoted as $\boldsymbol{m}_d$) is given by

$$\boldsymbol{m}_d = \boldsymbol{m} \odot \boldsymbol{k}_d, \tag{22}$$

where $\odot$ indicated the Hadamard product.

4.2.2. Writing Operation and Action Selection

The writing operation regulates the keeping and discarding of the information through gated functions, and the framework is shown in Figure 4. UAV_d obtains a candidate storage vector $\boldsymbol{c}_d$ based on the state embedding vector $\boldsymbol{e}_d$ and the shared information $\boldsymbol{m}$ by nonlinear mapping

$$\boldsymbol{c}_d = \tanh(\boldsymbol{W}^c_d[\boldsymbol{e}_d, \boldsymbol{m}]) \quad \boldsymbol{c}_d \in [-1,1]^M, \boldsymbol{W}^c_d \in \mathbb{R}^{M \times (E+M)}, \tag{23}$$

where $\boldsymbol{W}^c_d$ is the network parameter. The input gate $\boldsymbol{g}_d$ is employed to regulate the contents of the candidate, and $\boldsymbol{f}_d$ is utilized to decide the information to be kept. These operations can be expressed as

$$\begin{aligned} \boldsymbol{g}_d &= \sigma(\boldsymbol{W}^g_d[\boldsymbol{e}_d, \boldsymbol{m}]) \quad \boldsymbol{g}_d \in [0,1]^M, \boldsymbol{W}^g_d \in \mathbb{R}^{M \times (E+M)}, \\ \boldsymbol{f}_d &= \sigma(\boldsymbol{W}^f_d[\boldsymbol{e}_d, \boldsymbol{m}]) \quad \boldsymbol{f}_d \in [0,1]^M, \boldsymbol{W}^f_d \in \mathbb{R}^{M \times (E+M)}. \end{aligned} \tag{24}$$

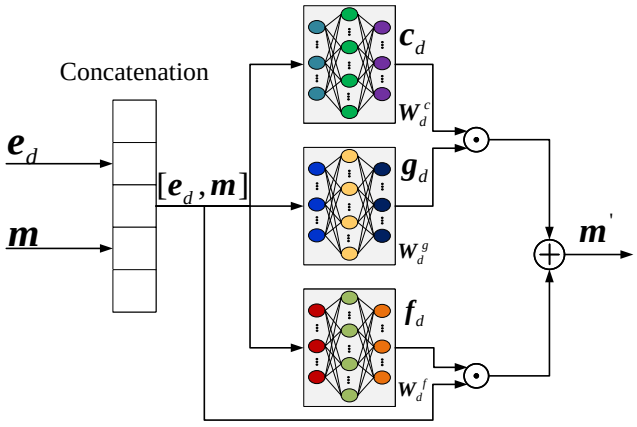


Figure 4. The writing operation framework.

Then, UAV$_d$ finally generates newly updated information $m^{'}$ by weighting the historical state information and real-time state information, and it is calculated as

$$m^{'} = g_d \odot c_d + f_d \odot m. \tag{25}$$

After completing the reading and writing operations, UAV$_d$ obtains the action a_d, which depends on the current state and the information reading from the $\mathcal{M}$

$$a_d = \pi_d^{\mu}(s_d, m_d). \tag{26}$$

According to the above description, the pseudo code of the reading and writing operation based on gated functions is given in Algorithm 1.

Algorithm 1 Memory-Based Reading and Writing Operations

Input: State information of UAVs: $s_d = \{\{u_d(n)\}_{\forall d \in \mathcal{D}}, \{\omega_k(n)\}_{\forall k \in \mathcal{K}}, E_d(n)_{\forall d \in \mathcal{D}}\}$;
Output: Decisions of UAVs: $a_d = \{dist_d, \theta_d\}_{\forall d \in \mathcal{D}}$;
1: Initialize the state s_d, memory $\mathcal{M}$;
2: Initialize each actor networks of UAV$_d$ with weights μ_d and $\mu_d^{'}$, respectively;
3: **for** $d = 1$ to D **do**
4: Obtain state s_d and the share information m;
5: Set $m_d = m$;
6: Generate observation encoding e_d according to Equation (19);
7: Generate read vector m_d according to Equation (22);
8: Generate new message $m^{'}$ according to Equation (25);
9: Update information in memory;
10: Select action $a_d = \pi_d^{\mu}(s_d, m_d)$ according to Equation (26);
11: **end for**

Both reading and writing operations in Algorithm 1 are the core of the information sharing mechanism. The agents utilize gated functions to select the required information from the memory based on own observations. Thus, unknown state information can be obtained through reading operation. The read information and observations are jointly used as input to the policy network. Hence, the actions depend on observations and the state information of other agents. With the dynamic changes of both agents and environments, the information in the memory needs to be dynamically updated. The writing operation regulates the keeping and discarding of the information through gated functions. As a result, Algorithm 1 enables the sharing of state information among UAVs and avoids policy uncertainty due to the partial state information.

4.3. The Architecture of Actor Network

Furthermore, the actor network of the UAFC algorithm consists of more than one network. The input of actor network can be divided into three categories:

(1) The remaining energy of UAVs ($\boldsymbol{s}_e$). It determines whether the UAVs perform the mission.

(2) The location of UAVs and GUs ($\boldsymbol{s}_l$). They determine whether the UAVs should move to optimal location to provide great communication services.

(3) The information read from memory ($\boldsymbol{m}_d$). They can help UAVs create optimal policies.

The final actions of the UAVs depend on the comprehensive impact of these three categories of input information. If we directly input all the state information and share information into an actor network, it may hardly output desirable policy due to the imbalance and high dimension of state information. Thus, we design a novel actor network architecture of decomposing and coupling. The architecture decouples the input vector into three categories. Then, it expands the dimension of part state information ($\boldsymbol{s}_e$) and aggregates three parts of information as a total input vector. This method of state dimension spread and state aggregation can address the dimension imbalance problem and reduce state dimension to generate higher-quality policy.

The actor network architecture is shown in Figure 5. It aims to avoid the crash and service interruption of the UAVs due to insufficient power. Thus, the energy state of dimension size D is very important. Furthermore, the energy state information dimension is much smaller than the position information dimension of the UAVs and GUs. There exists a dimension imbalance problem, which makes the algorithm difficult to converge. The dimension spread and linear mapping are utilized to process energy state, location state, and the information read from memory to obtain three state vectors with the same dimension, respectively. After the state decomposing and linear mapping, the input dimension is reduced and the vectors are denoted as N_e, N_l, and N_d. Then, network 4 combines N_e, N_l, and N_d into a new vector and as the input, and outputs the final action.

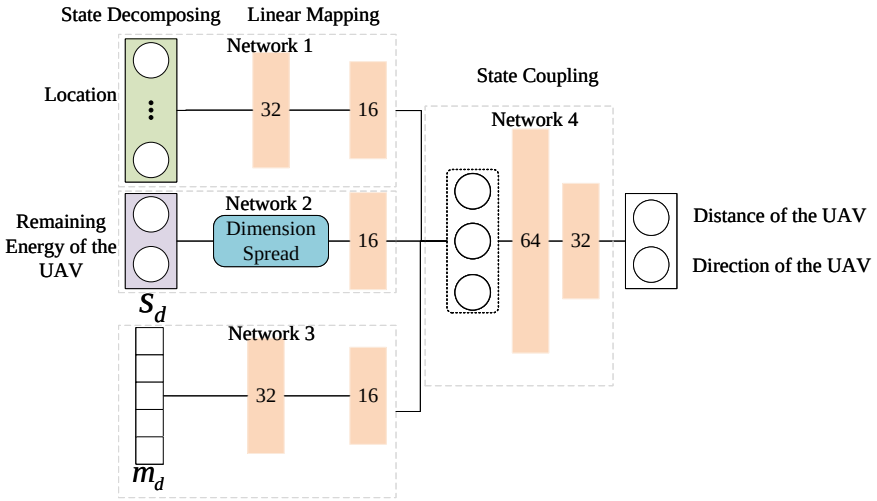


Figure 5. The network architecture of the actor in the UAFC algorithm.

4.4. Training of UAV-Assisted Fair Communication

Algorithm 2 summarizes the UAFC algorithm for UAVs-assisted fair communication. First, the training data are randomly sampled from the experience replay pool. s_j and $s_j^{'}$ are input into the evaluation and target critic network to generate state-action value function $Q_d^{\theta_i}$ and target state-action value function target $Q_d^{\theta_i{}'}$, respectively. The loss function is constructed according to $Q_d^{\theta_i}$ and $Q_d^{\theta_i{}'}$ to train the critic network.

Algorithm 2 UAFC Algorithm

Input: State information of UAVs: $s_d = \{\{\boldsymbol{u}_d(n)\}_{\forall d \in \mathcal{D}}, \{\boldsymbol{\omega}_k(n)\}_{\forall k \in \mathcal{K}}, E_d(n)_{\forall d \in \mathcal{D}}\}$;
Output: Decisions of UAVs: $a_d = \{dist_d, \theta_d\}_{\forall d \in \mathcal{D}}$;

1: ▷ **Parameter initialization**
2: Initialize actor and critic networks parameters μ_d, $\mu_d^{'}$, $\{\theta_n^i\}_{i=1,2}$ and $\{\theta_d^{i'}\}_{i=1,2}$, respectively;
3: Initialize replay buffer B;
4: **for each** *episode* **do**
5: ▷ **Action generation**
6: Obtain the action of UAV_d from **Algorithm** 1;
7: Set $s = (s_1, s_2, \ldots, s_D)$and $\Phi = (\boldsymbol{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_D)$;
8: ▷ **Experience storage**
9: UAV_d take selected actions $a = (a_1, a_2, \ldots, a_D)$;
10: UAV_d obtain the reward R, state s transfers to new $s^{'}$;
11: The experience $(s, a, \Phi, R, s^{'})$ is stored in replay pool B;
12: ▷ **Parameter updating**
13: **for** $d = 1$ to D **do**
14: Sample a random mini-batch of $(s_j, a_j, \phi_j, R_j, s_j^{'})$ from B;
15: Update weights $\{\theta_d^i\}_{i=1,2}$ of evaluation critic networks by minimizing loss function $Loss(\theta_d^i)$ according to Equation (28);
16: Update weights μ_d of evaluation actor network according to Equation (30);
17: Update the weights of the three target networks according to Equations (31) and (32);
18: **end for**
19: **end for**

Initialization (lines 2–3): During the centralized training phase, the actor network and critic network parameters are randomly initialized. Furthermore, the two storage spaces of the experience replay pool and the memory are initialized.

Generate action (lines 4–7): Each UAV through the current observation value s_d and the information $\boldsymbol{m}_d$ of other UAVs obtains the action according to the policy function $\pi_d^{\mu}(s_d, \boldsymbol{m}_d)$.

Experience storage (Lines 9–11): The experience of each UAV can be expressed as a tuple $(s_d, a_d, \boldsymbol{m}_d, R_d, s_d^{'})$. After performing the action, UAV_d obtains the reward R_d, and the current state will transfer to the new state at the next moment. Finally, the experience $(s_d, a_d, \boldsymbol{m}_d, R_d, s_d^{'})$ is stored into replay pool B with a capacity of M_r.

Parameter update (lines 13–17): During the training process, experience $(s_j, a_j, \boldsymbol{m}_j, R_j, s_j^{'})$ of size M_b is randomly sampled from the experience replay pool. The evaluation actor network generates a policy $\pi_d^{\mu}(s_j, \boldsymbol{m}_j)$ according to s_j and $\boldsymbol{m}_j$. The parameters of the evaluation actor network are updated according to the following policy gradient [46]

$$\nabla_{\mu_d} J(\mu_d) = \frac{1}{M_b} \sum_{j=1}^{M_b} \nabla_{\mu_d} \pi_d^{\mu}(s_d^j, m_d^j) \nabla_{a_d} Q_d^{\theta_1}(s_j, a_1^j, a_2^j, \ldots, a_D^j)|_{a_d = \pi_d^{\mu}(s_d^j, \boldsymbol{m}_d^j)}. \tag{27}$$

Based on the policy $\pi_d^{\mu}(s_j, m_j)$, two Q values, i.e., $Q_d^{\theta_1}(s_j, \pi_d^{\mu}(s_j, m_j))$ and $Q_d^{\theta_2}(s_j, \pi_d^{\mu}(s_j, m_j))$, are obtained by two evaluation critic networks. The parameters of the critic networks are updated by minimizing the loss function $Loss(\theta_d^i)$

$$Loss(\theta_d^i) = \frac{1}{M_b} \sum_{j=1}^{M_b} [y_i - Q_d^{\theta_i}(s_j, a_j)]^2, \quad i = 1, 2. \tag{28}$$

According to the above loss function, each UAV updates three evaluation networks

$$\theta_d^i \leftarrow \theta_d^i - \lambda \cdot \nabla_{\theta_d^i} L(\theta_d^i), \quad i = 1, 2\,, \tag{29}$$

$$\mu_d \leftarrow \mu_d - \lambda \cdot \nabla_{\mu_d} J(\mu_d)\,, \tag{30}$$

where λ denotes the learning rate, and the target network parameters are updated as follows

$$\mu_d^{'} = u \cdot \mu_d + (1-u) \cdot \mu_d^{'}, \tag{31}$$

$$\theta_d^{j^{'}} = u \cdot \theta_d^i + (1-u) \cdot \theta_d^{j^{'}}\,, \quad i = 1, 2 \tag{32}$$

where u denotes the updating rate.

4.5. Complexity Analysis

We evaluate the efficiency of the UAV-assisted fair communication algorithm by complexity analysis. The non-linear mapping of states to actions is achieved by a deep neural network during the offline training and online execution phases. The actor and critic networks contain J-th layer and F-th layer neural networks, respectively. Thus, the time complexity of the UAFC algorithm (denoted as T_{UAFC}) is given by

$$\begin{aligned} T_{UAFC} &= 2 \times \sum_{j=1}^{J} U_{actor,j} \cdot U_{actor,j+1} + 4 \times \sum_{f=1}^{F} U_{critic,f} \cdot U_{critic,f+1} \\ &= O(\sum_{j=1}^{J} U_{actor,j} \cdot U_{actor,j+1} + \sum_{f=1}^{F} U_{critic,f} \cdot U_{critic,f+1})\,, \end{aligned} \tag{33}$$

where $U_{actor,j}$ represents the number of neurons in the j-th layer of the actor network, and $U_{critic,f}$ represents the number of neurons in the f-th layer of the critic network.

A matrix of $P \times Q$ and a bias of Q exist in a fully connected neural network. Therefore, the number of storage unit required by a fully connected neural network is $(P+1) \times Q$, and thus, the space complexity is $O(G)$. In addition, it is also necessary to allocate storage space to the experience replay pool and memory to store information in the process of training, and the space complexities are $O(M_r)$ and $O(M)$, respectively. Hence, the space complexity of the UAFC algorithm (denoted as S_{UAFC}) is formulated as

$$\begin{aligned} S_{UAFC} &= \sum_{j=1}^{J} (U_{actor,j} + 1) \cdot U_{actor,j+1} + 2 \times \sum_{f=1}^{F} (U_{critic,f} + 1) \cdot U_{critic,f+1} + M_r + M \\ &= \underbrace{O(\sum_{j=1}^{J} U_{actor,j} \cdot U_{actor,j+1} + \sum_{f=1}^{F} U_{critic,f} \cdot U_{critic,f+1})}_{O(G)} + O(M_r) + O(M)\,. \end{aligned} \tag{34}$$

In the distributed execution stage, only the trained actor network is needed. Thus, the space complexity of the execution phase is

$$O(\sum_{j=1}^{J} U_{actor,j} \cdot U_{actor,j+1}) + O(M), \tag{35}$$

and the time complexity is

$$O(\sum_{j=1}^{J} U_{actor,j} \cdot U_{actor,j+1}). \tag{36}$$

5. Performance Evaluation

In this section, we introduce the detailed settings of the algorithm and simulation parameters, and conduct extensive simulation experiments to verify the effectiveness of the UAFC algorithm.

5.1. Simulation Settings

We verify the performance of the UAFC algorithm through extensive experiments. The experimental platform is built based on Intel Core i9-11900H, NVIDIA GeForce RTX3090, and Tensorflow-CPU-1.14. GUs are randomly deployed in a target area (500 m × 500 m) and move in random directions and speeds. The UAVs initializes their position randomly to provide communication services for GUs. The experimental parameters are shown in Table 2. The two metrics are chosen for performance evaluation: A novel fairness index W_k and fair throughput are expressed as Equation (12) and Equation (13), respectively.

Table 2. Simulation Settings.

Parameters	Values
Number of UAVs (D)	$\{2, 3\}$
Number of Gus (K)	{10∼15}
Carrier frequency (f_c)	2.4 GHz
Maximum and minimum energy of UAVs ($E_{\max}$, $E_{\min}$)	500 KJ, 50 KJ
The parameters of channel model (X, Y)	4.88, 0.33
Additional path loss for LoS and NLoS (E_{LoS}, E_{NLoS})	1.6, 2.1
The hover altitude and minimum safe distance of the UAVs (H,$d_{\min}$)	100 m, 10 m
Learning rate(λ)	0.001
Buffer size and mini-batch (M_b, M_r)	60,000, 256
Memory capacity (M)	256
Discount factor (γ)	0.99
Updating rate (u)	0.01
Penalty value ($p_i, i \in \{1, 2, 3\}$)	$\{500, 100, 100\}$

5.2. Training Results

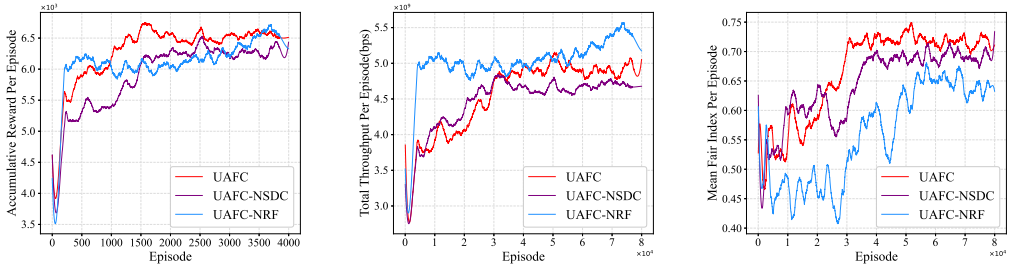
To verify the impact of RF metric and state decomposing and coupling on algorithm performance. We compare the accumulative reward, fair throughput, and fair index of the UAFC algorithm with the UAFC-NSDC and UAFC-NRF algorithms.

- No state decomposing and coupling (UAFC-NSDC): Compared with the UAFC algorithm, this algorithm directly involves complete state information without employing state decomposing and coupling.
- No RF (UAFC-NRF): The UAFC-NRF algorithm maximizes throughput while ignoring the fairness of the GUs in communication services.

From Figure 6a, we can observe that the UAFC algorithm achieves higher accumulative reward than the other two algorithms. This is because the UAFC algorithm takes into consideration the GUs fairness to serve more GUs and obtain higher coverage reward. Furthermore, the UAFC algorithm utilizes state decomposing and coupling to eliminate the influences of state dimension imbalance. Thus, UAVs can obtain high-quality policies to achieve great reward.

From Figure 6b, we can observe that the UAFC-NRF algorithm converges to optimal value quickly. This is due to the fact that the UAFC-NRF algorithm ignores the fairness among GUs where the UAVs tend to hover close to partial GUs to achieve higher throughput. Compared to the UAFC-NRF algorithm, the throughput values of both UAFC and UAFC-NSDC are lower, since these two algorithms trade off the communication efficiency and fairness. To ensure fair communication services for GUs, the throughputs of UAVs are sacrificed, especially in the case of a limited number of UAVs.

The mean fairness index is employed as the evaluation indicator. From Figure 6c, we can observe that the mean fair index of the UAFC algorithm outperforms the other two algorithms, because UAFC considers the fairness of GUs and the state information is involved into the actor network after state decomposing and coupling.



(**a**) (**b**) (**c**)

Figure 6. The training process of the UAFC, UAFC-NSDC, and UAFC-NRF algorithm. (**a**) Accumulative reward per episode. (**b**) Total fair throughput per episode. (**c**) Mean fair index per episode.

5.3. Performance Comparisons with Two Related Algorithms

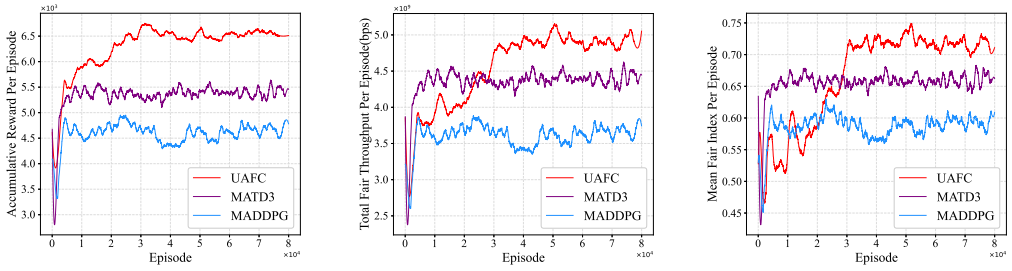
In this subsection, we compare the accumulative reward, fair throughput, and fair index of the UAFC algorithm with another two existing algorithms.

- MADDPG: The MADDPG algorithm in [47] is utilized as the benchmark for designing the trajectory of UAV-BSs to maximize throughput without considering fairness.
- MATD3: The MATD3 algorithm in [48] is employed as a UAV trajectory planning and resource allocation algorithm based on MATD3 to minimize the time delay and energy consumption of UAV tasks.

Figure 7 shows the convergence curves of the UAFC algorithm and the three baseline algorithms for accumulative reward value, total fair throughput, and mean fair index, respectively.

- Figure 7a shows the results of accumulative reward. The reward value of UAFC converges to around 6500 at 30,000 episodes. UAFC outperforms both MATD3 and MADDPG. This is because the information sharing mechanism makes full use of the shared states information among UAVs. The information is conducive to finding the optimal policy and avoiding falling into the local optimum. Furthermore, the training curves of the three algorithms have obvious oscillations. This is because MADRL is different from supervised learning, since it has no clear label information.
- Figure 7b shows the convergence curve of fair throughput. We can observe that the total fair throughput of the MATD3 algorithm is better than the UAFC algorithm in the first 25,000 episodes. This is due to the fact that the actor network of UAFC contains reading and writing operations, which are implemented through a multi-layer fully connected (FC) layer neural network. In the FC network, the gradient becomes smaller as the hidden layers propagate backwards. This means that neurons in the previous hidden layers learn more slowly than neurons in the later hidden layers. Thus, the UAFC is harder to train than that of MATD3. The total fair throughput curve of UAFC converges at 40,000 episodes and outperforms both MADDPG and MATD3 as the training time increases. This is because: (1) UAVs share state information with each other to obtain the optimal policy; (2) State information is processed by state decomposing and coupling. Thus, the input of actor network gains a low-dimension state vector which includes complete state information and reading information from $\mathcal{M}$.
- The experimental results of the three algorithms on mean fair index are shown in Figure 7c. In the first 20,000 episodes, the UAFC algorithm fluctuates greatly and the numerical value is lower than that of both MATD3 and MADDPG algorithms. By decoupling and coupling the input of the actor network, the overall actor network

has more layers and more complex structures. Furthermore, the distribution of GUs is changing, and it is difficult to obtain the best strategy for UAVs cooperative search. Both MATD3 and MADDPG converge to the local optimal value quickly, which can also be seen from the final convergence value. The fairness index keeps increasing and it finally converges to 0.72. Compared with the MADDPG and MATD3 algorithms, the mean fairness index of the UAFC algorithm is improved by 18.13% and 6.31%, respectively.



(**a**) (**b**) (**c**)

Figure 7. The training process of the UAFC, MATD3, and MADDPG algorithms. (**a**) Accumulative reward per episode. (**b**) Total fair throughput per episode. (**c**) Mean fair index per episode.

To demonstrate the performance of the UAFC algorithm more intuitively, we present the result of the UAFC algorithm and the four compared algorithms regarding the evaluation metrics in Table 3. It can be seen that the UAFC algorithm outperforms the comparison algorithm in terms of reward function and fair index. Table 4 shows a comparison of the results on reward function, equity throughput, and fair index. Compared with UAFC-NRF, UAFC has a 9.09% decrease in fair throughput. The main reason is that the UAFC-NRF algorithm does not consider fairness among GUs. As a result, the UAV can always serve GUs with high throughput. The results also show that the UAFC algorithm obtains a higher fair index by sacrificing part of the throughput.

Table 3. Results of five algorithms on reward function, fair throughput, and fair index.

Algorithm	Reward	Fair Throughput	Fair Index
UAFC	**6290.319**	4558.494	**0.667**
UAFC-NSDC	5902.323	4465.863	0.645
UAFC-NRF	6098.576	**5014.304**	0.559
MATD3	5323.756	4341.374	0.654
MADDPG	4588.57	3601.502	0.586

Table 4. Comparison of results on reward function, fair throughput, and fair index.

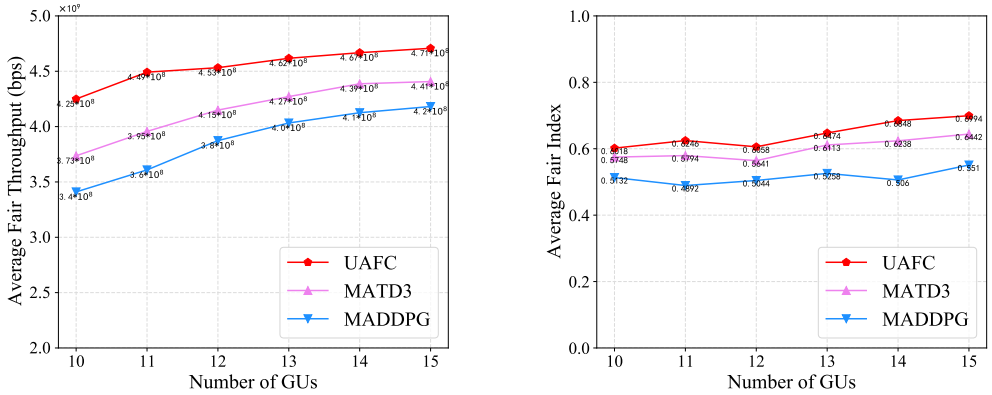
Algorithm	Reward	Fair Throughput	Fair Index
UAFC VS UAFC-NSDC	6.57%	2.07%	3.41%
UAFC VS UAFC-NRF	3.14%	−9.09%	19.32%
UAFC VS MATD3	18.15%	5.62%	1.99%
UAFC VS MADDPG	37.09%	26.57%	13.82%

Figure 8 shows the impact of GU numbers on system performance. The results indicate the following:

- From Figure 8a, we can observe that the average fair throughput of the three algorithms increases as the number of GUs increases. As the number of GUs served by the UAVs will increase as the number of GUs increases, there is an upward tendency in the average fair throughput. It is worth noting that as the number of GUs increases,

the performance of the UAFC algorithm on fair throughput outperforms both MADDPG and MATD3. This is because each UAV can obtain the state information of other UAVs, and perform state decomposing and coupling. Thus, the UAVs can obtain complete state information of the environment and other UAVs;

- Figure 8b shows the changing trend of the fairness index of the three algorithms with the increase of GUs. As the number of mobile GUs increases, the fairness of the three algorithms does not change much. The fairness index of the MADDPG algorithm is the lowest. This is because the absence of target policy smoothing regularization in the actor network of MADDPG leads to convergence to a local optimum. Furthermore, the UAFC algorithm numerically outperforms the MATD3 algorithm in fairness index. This is due to the fact that the UAVs can provide fair communication services for GUs in a collaborative way by sharing the states information of the UAVs.

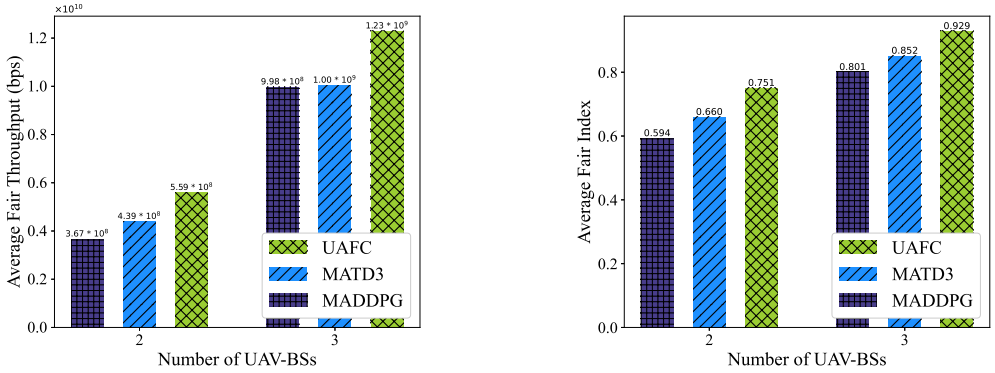


(**a**) (**b**)

Figure 8. The impact of the number of GUs on system performance. (**a**) Average fair throughput. (**b**) Average fair index.

Figure 9 shows the impact of number of UAV-BSs on system performance. From the results, we can conclude that:

- More GUs can be served as the number of UAVs increases, thus, the fair throughput of the three algorithms gradually increases. The performance of the UAFC algorithm outperforms both MADDPG and MATD3 in terms of fair throughput. Thus, the UAFC algorithm can provide fair communication services. This is due to the fact that: (1) The information sharing mechanism is vital and reduce the distributed decision-making uncertainty of UAVs. Thus, the UAFC algorithm still performs well when the number of UAVs increases; (2) The dimension of state information increases with the number of UAVs. The state decomposing and coupling can address the dimension imbalance problem and reduce state dimension to generate higher-quality policy;
- As shown in Figure 9b, the fair index of the three algorithms increases as the number of UAVs increases. The reason is that more UAVs means more extensive coverage and the ability to cover almost all GUs. When the number of UAVs is 3, the UAFC algorithm improves the fairness index by 16.4% and 9.6% compared to the MADDPG and MATD3 algorithms. This is because the information mechanism and actor network architecture can help UAVs make decisions.



(**a**) (**b**)

Figure 9. The impact of the number of UAV-BSs on system performance. (**a**) Average fair throughput. (**b**) Average fair index.

6. Discussion

In this section, we focus on some of the limitations of this paper. The main limitations of this paper are: (1) We model the UAV-assisted fair communication problem as a complex non-convex optimization problem that is an NP-hard problem. It is difficult to find an analytic solution. We utilize a DRL algorithm to solve it. The DRL algorithm cannot obtain an optimal solution, but it can be trained to obtain an approximate optimal solution. In addition, experimental results show that our algorithm is more effective than other algorithms. (2) UAVs can flexibly adjust their positions to establish good LoS communication links with GUs and provide reliable wireless communication environment. However, in some complex environments (e.g., urban scenarios), it is inevitable that the LoS link between the UAVs and the GUs will be blocked by high buildings or trees, affecting the quality of communication. In this paper, we assume that the links between the UAVs and the GUs is unaffected by obstructions. (3) In addition, UAVs carry very limited energy due to their limited size. This paper considers the residual energy consumption of UAVs only as a constraint and does not design methods to extend the flight time of UAVs, such as utilizing wireless power transfer technology to recharge the UAVs. In future work, we consider building more realistic mathematical models and designing more accurate solution algorithms.

7. Conclusions

UAV-assisted communication has been expected to be a suitable method for wireless communication. In this paper, we have studied the problem of UAV-assisted communication with the consideration of user fairness. First, a novel metric to evaluate the trade-off between fairness and communication efficiency is presented to maximize fair system throughput while ensuring user fairness. Then, the UAV-assisted fair communication problem is modeled as a mixed-integer non-convex optimization problem. We reformulated the problem as an MDP and proposed a UAFC algorithm based on MADRL. Further, inspired by the communication among agents, the information sharing mechanism based on gated functions is designed to reduce the distributed decision-making uncertainty of UAVs. To solve the problem of state dimension imbalance, a new actor network architecture is designed to reduce the impact of dimension imbalance and dimensional catastrophe on policy search through dimensional expansion and linear mapping techniques. Finally, we have verified the effectiveness of the proposed algorithm through extensive experiments. Simulation results show that the proposed UAFC algorithm increases fair throughput by 5.62%, 26.57% and fair index by 1.99%, 13.82% compared to the MATD3 and MADDPG algorithms. Intelligent Reflecting Surface (IRS) is a new technology in 6G that has received widespread academic attention. Combining IRS and wireless power information trans-

mission technology is a good option to further improve the performance of UAV-assisted communication. In future, we will extend this paper to design a novel algorithm based on IRS.

Author Contributions: Conceptualization, Y.Z.; methodology, Y.Z.; software, Z.J.; validation, Z.J., H.S. and Z.W.; formal analysis, H.S.; investigation, Z.J.; resources, Y.Z.; writing—original draft preparation, Y.Z.; writing—review and editing, H.S., N.L. and F.L.; visualization, Z.W.; supervision, N.L.; project administration, F.L.; funding acquisition, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by National Natural Science Foundation of China (No. 62176088), the Program for Science & Technology Development of Henan Province (Nos. 212102210412, 222102210067, 222102210022), and Young Elite Scientist Sponsorship Program by Henan Association for Science and Technology (No. 2022HYTP013).

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

Abbreviations

UAVs	Unmanned Aerial Vehicles
GUs	Ground Users
RF	Ratio Fair
UAFC	UAV-Assisted Fair Communication
MADRL	Multi-Agent Deep Reinforcement Learning
UAV-BSs	UAV Base Stations
DRL	Deep Reinforcemrnt Learning
A2G	Air-to-Ground
LoS	Line-of-Sight
NLoS	Non-Line-of-Sight
MDP	Markov Decision Process
MATD3	Multi-Agent Twin Delayed Deep Deterministic policy gradient

References

1. Xiao, Z.; Zhu, L.; Liu, Y.; Yi, P.; Zhang, R.; Xia, X.G.; Schober, R. A survey on millimeter-wave beamforming enabled UAV communications & networking. *IEEE Commun. Surv. Tutor.* **2022**, *24*, 557–610.
2. Liu, Y.; Yan, J.; Zhao, X. Deep reinforcement learning based latency minimization for mobile edge computing with virtualization in maritime UAV communication network. *IEEE Trans. Veh. Technol.* **2022**, *71*, 4225–4236. [CrossRef]
3. Zhang, C.; Zhang, L.; Zhu, L.; Zhang, T.; Xiao, Z.; Xia, X. 3D Deployment of multiple UAV-mounted base stations for UAV communications. *IEEE Trans Commun.* **2021**, *69*, 2473–2488. [CrossRef]
4. Zhong, X.; Guo, Y.; Li, N.; Chen, Y. Joint optimization of relay deployment, channel allocation, and relay assignment for UAVs-aided D2D networks. *IEEE Trans Commun.* **2020**, *28*, 804–817. [CrossRef]
5. Zhang, J.; Zeng, Y.; Zhang, R. Multi-antenna UAV data harvesting: Joint trajectory and communication optimization. *J. Commun. Inf. Netw.* **2020**, *5*, 86–99. [CrossRef]
6. Lu, W.; Ding, Y.; Gao, Y.; Su, H.; Wu, Y.; Zhao, N.; Gong, Y. Resource and trajectory optimization for secure communications in dual unmanned aerial vehicle mobile edge computing systems. *IEEE Trans Commun.* **2022**, *18*, 2704–2713. [CrossRef]
7. Al-Ahmed, S.A.; Shakir, M.Z.; Zaidi, S.A.R. Optimal 3D UAV base station placement by considering autonomous coverage hole detection, wireless backhaul and user demand. *J. Commun. Netw.* **2020**, *22*, 467–475. [CrossRef]
8. Hao, C.; Chen, Y.; Mai, Z.; Chen, G.; Yang, M. Joint optimization on trajectory, transmission and time for effective data acquisition in UAV-enabled IoT. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7371–7384. [CrossRef]
9. Liu, Y.; Xiong, K.; Lu, Y.; Ni, Q.; Fan, P.; Letaief, K.B. UAV-aided wireless power transfer and data collection in rician fading. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3097–3113. [CrossRef]
10. Li, X.; Yao, H.; Wang, J.; Xu, X.; Jiang, C.; Hanzo, L. A near-optimal UAV-aided radio coverage strategy for dense urban areas. *IEEE Trans. Veh. Technol.* **2019**, *68*, 9098–9109. [CrossRef]
11. Zhang, X.; Duan, L. Energy-saving deployment algorithms of UAV swarm for sustainable wireless coverage. *IEEE Trans. Veh. Technol.* **2020**, *69*, 10320–10335. [CrossRef]
12. Alkama, D.; Ouamri, M.A.; Alzaidi, M.S.; Shaw, R.N.; Azni, M.; Ghoneim, S.S.M. Downlink performance analysis in MIMO UAV-cellular communication with LOS/NLOS propagation under 3D beamforming. *IEEE Access* **2022**, *10*, 6650–6659. [CrossRef]
13. Wu, Z.; Yang, Z.; Yang, C.; Lin, J.; Liu, Y.; Chen, X. Joint deployment and trajectory optimization in UAV-assisted vehicular edge computing networks. *J. Commun. Netw.* **2022**, *24*, 47–58. [CrossRef]

14. Wang, L.; Zhang, H.; Guo, S.; Yuan, D. Deployment and association of multiple UAVs in UAV-assisted cellular networks with the knowledge of statistical user position. *IEEE Wirel. Commun.* **2022**, *21*, 6553–6567. [CrossRef]
15. Wang, C.; Deng, D.; Xu, L.; Wang, W. Resource scheduling based on deep reinforcement learning in UAV assisted emergency communication networks. *IEEE Trans. Commun.* **2022**, *70*, 3834–3848. [CrossRef]
16. Abeywickrama, H.V.; He, Y.; Dutkiewicz, E.; Jayawickrama, B.A.; Mueck, M. A Reinforcement learning approach for fair user coverage using UAV mounted base stations under energy constraints. *IEEE Open J. Veh. Technol.* **2020**, *1*, 67–81. [CrossRef]
17. Qi, H.; Hu, Z.; Huang, H.; Wen, X.; Lu, Z. Energy Efficient 3-D UAV Control for Persistent Communication Service and Fairness: A Deep Reinforcement Learning Approach. *IEEE Access* **2020**, *8*, 53172–53184. [CrossRef]
18. Chen, D.; Qi, Q.; Zhuang, Z.; Wang, J.; Liao, J.; Han, Z. Mean Field Deep Reinforcement Learning for Fair and Efficient UAV Control. *IEEE Internet Things J.* **2021**, *8*, 813–828. [CrossRef]
19. Jeong, C.; Chae, S.H. Simultaneous wireless information and power transfer for multiuser UAV-enabled IoT networks. *IEEE Internet Things J.* **2021**, *8*, 8044–8055. [CrossRef]
20. Yin, S.; Zhao, S.; Zhao, Y.; Yu, F.R. Intelligent trajectory design in UAV-aided communications with reinforcement learning. *IEEE Trans. Veh. Technol.* **2019**, *68*, 8227–8231. [CrossRef]
21. Yuan, Y.; Lei, L.; Vu, T.X.; Chatzinotas, S.; Sun, S.; Ottersten, B. Energy minimization in UAV-aided networks: Actor-critic learning for constrained scheduling optimization. *IEEE Trans. Veh. Technol.* **2021**, *70*, 5028–5042. [CrossRef]
22. Yang, D.; Wu, Q.; Zeng, Y. Energy tradeoff in ground-to-UAV communication via trajectory design. *IEEE Trans. Veh. Technol.* **2018**, *67*, 6721–6726. [CrossRef]
23. Zhang, T.; Lei, J.; Liu, Y.; Feng, C.; Nallanathan, A. Trajectory optimization for UAV emergency communication with limited user equipment energy: A safe-DQN approach. *IEEE Trans. Green Commun. Netw.* **2021**, *5*, 1236–1247. [CrossRef]
24. Shi, W.; Li, J.; Wu, H.; Zhou, C.; Cheng, N.; Shen, X. Drone-cell trajectory planning and resource allocation for highly mobile networks: A hierarchical DRL approach. *IEEE Internet Things J.* **2021**, *8*, 9800–9813. [CrossRef]
25. Zeng, F.; Hu, Z.; Xiao, Z.; Jiang, H.; Zhou, S.; Liu, W.; Liu, D. Resource allocation and trajectory optimization for QoE provisioning in energy-efficient UAV-enabled wireless networks. *IEEE Trans. Veh. Technol.* **2020**, *69*, 7634–7647. [CrossRef]
26. Ding, R.; Xu, Y.; Gao, F.; Shen, X. Trajectory design and access control for air–ground coordinated communications system with multiagent deep reinforcement learning. *IEEE Internet Things J.* **2022**, *9*, 5785–5798. [CrossRef]
27. Diao, X.; Zheng, J.; Cai, Y.;Wu, Y.; Anpalagan, A. Fair data allocation and trajectory optimization for UAV-assisted mobile edge computing. *IEEE Commun. Lett.* **2019**, *23*, 2357–2361. [CrossRef]
28. Ding, R.; Gao, F.; Shen, X.S. 3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 7796–7809. [CrossRef]
29. Liu, C.H.; Chen, Z.; Tang, J.; Xu, J.; Piao, C. Energy-efficient UAV control for effective and fair communication coverage: A deep reinforcement learning approach. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 2059–2070. [CrossRef]
30. Nemer, I.A.; Sheltami, T.R.; Belhaiza, S.; Mahmoud, A.S. Energy-efficient UAV movement control for fair communication coverage: A deep reinforcement learning approach. *Sensors* **2022**, *22*, 1919. [CrossRef]
31. Liu, Y.; Huangfu, W.; Zhou, H.; Zhang, H.; Liu, J.; Long, K. Fair and energy-efficient coverage optimization for UAV placement problem in the cellular network. *IEEE Trans Commun.* **2022**, *70*, 4222–4235. [CrossRef]
32. Zeng, Y.; Xu, J.; Zhang, R. Energy minimization for wireless communication with rotary-wing UAV. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 2329–2345. [CrossRef]
33. Cheng, Z.; Hong, L. Energy minimization in internet-of-things system based on rotary-wing UAV. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 1341–1344.
34. Al-Hourani, A.; Kandeepan, S.; Lardner, S. Optimal LAP altitude for maximum coverage. *IEEE Wirel. Commun. Lett.* **2014**, *3*, 569–572. [CrossRef]
35. Al-Hourani, A.; Kandeepan, S.; Jamalipour, A. Modeling air-to-ground path loss for low altitude platforms in urban environments. In Proceedings of the 2014 IEEE Global Communications Conference, Austin, TX, USA, 8–12 December 2014.
36. Jain, R.K.; Chiu, D.M.W.; Hawe, W.R. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. *arXiv* **1998**, arXiv:cs/9809099.
37. Lin, L.; Goodrich, M.A. Hierarchical heuristic search using a gaussian mixture model for UAV coverage planning. *IEEE Trans Cybern.* **2014**, *44*, 2532–2544. [CrossRef]
38. Wang, H.;Wang, J.; Ding, G.; Chen, J.; Gao, F.; Han, Z. Completion time minimization with path planning for fixed-wing UAV communications. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 3485–3499. [CrossRef]
39. Dong, L.; Liu, Z.; Jiang, F.; Wang, K. Joint optimization of deployment and trajectory in UAV and IRS-assisted IoT data collection system. *IEEE Internet Things J.* **2022**, *9*, 21583–21593. [CrossRef]
40. Zhang, W.; Yang, D.; Wu, W.; Peng, H.; Zhang, N.; Zhang, H.; Shen, X. Optimizing federated learning in distributed industrial IoT: A multi-agent approach. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3688–3703. [CrossRef]
41. Meshgi, H.; Zhao, D. Opportunistic scheduling for a two-way relay network using markov decision process. *IET Commun.* **2016**, *10*, 1846–1854. [CrossRef]
42. Papoudakis, G.; Christianos, F.; Rahman, A.; Albrecht, S.V. Dealing with non-stationarity in multi-agent deep reinforcement learning. *arXiv* **2019**, arXiv:1906.04737.

43. Ackermann, J.; Gabler, V.; Osa, T.; Sugiyama, M. Reducing overestimation bias in multi-agent domains using double centralized critics. *arXiv* **2019**, arXiv:1910.01465.
44. Yuan, T.; Neto, W.d.R.; Rothenberg, C.E.; Obraczka, K.; Barakat, C.; Turletti, T. Dynamic controller assignment in software defined internet of vehicles through multi-agent deep reinforcement learning. *IEEE Trans. Netw. Serv. Manag.* **2021**, *18*, 585–596. [CrossRef]
45. Pesce, E.; Montana, G. Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication. In Proceedings of the Conference and Workshop on Neural Information Processing Systems, Montreal, QC, Cananda, 3–8 December 2018.
46. Ding, F.; Xu, L.; Meng, D.; Jin, X.B.; Alsaedi, A.; Hayat, T. Gradient estimation algorithms for the parameter identification of bilinear systems using the auxiliary model. *J. Comput. Appl. Math.* **2020**, *369*, 112575–112588. [CrossRef]
47. Zhao, N.; Liu, Z.; Cheng, Y. Multi-agent deep reinforcement learning for trajectory design and power allocation in multi-UAV networks. *IEEE Access* **2020**, *8*, 139670–139679. [CrossRef]
48. Zhao, N.; Ye, Z.; Pei, Y.; Liang, Y.C.; Niyato, D. Multi-agent deep reinforcement learning for task offloading in UAV-assisted mobile edge computing. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 6949–6960. [CrossRef]