


## Article

# A Fuzzy-Logic-Based Load Balancing Scheme for a Satellite–Terrestrial Integrated Network

Yuehong Gao, Haotian Yang \*, Xiaoqi Wang, Yihao Chen, Chenyang Li and Xin Zhang 

School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

\* Correspondence: haotianyang@bupt.edu.cn

**Abstract:** With the development of communication systems, users are becoming more widely distributed and require higher speed networks. A satellite–terrestrial integrated network could provide seamless coverage for these users. In previous studies of load balancing, initial access and load balancing are decided on based on signal reception and are performed reactively after the overloading occurs, which may not work well in satellite–terrestrial integrated networks. Therefore, this paper proposes a fuzzy-logic-based load balancing scheme. In this scheme, a fuzzy evaluation metric to pre-evaluate the user’s impact on overload is presented. The fuzzy logic system is constructed based on adaptive neuro fuzzy system, which takes the user’s signal reception, speed and data requirement as inputs. Then, the fuzzy-logic- and reinforcement-learning-based access is proposed to give an access decision for all users in the network to prevent overloading. Due to the large dimensions of action space, the reinforcement learning model is trained by the proposed fuzzy, deep, deterministic policy gradient. Next, the fuzzy-logic-based offloading algorithm is proposed to balance load after overloading. A simulation platform is established to evaluate the performance. Simulation results indicate that the proposed scheme can ensure load balance for a longer time than base line schemes while ensuring data rate of users.

**Keywords:** load balancing; fuzzy logic; reinforcement learning; satellite–terrestrial network



**Citation:** Gao, Y.; Yang, H.; Wang, X.; Chen, Y.; Li, C.; Zhang, X. A Fuzzy-Logic-Based Load Balancing Scheme for a Satellite–Terrestrial Integrated Network. *Electronics* **2022**, *11*, 2752. <https://doi.org/10.3390/electronics11172752>

Academic Editor: Dimitris Kanellopoulos

Received: 27 July 2022

Accepted: 29 August 2022

Published: 1 September 2022

**Publisher’s Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 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 (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Coinciding with the development of 5th generation (5G) technology, users have higher data requirements and are widely distributed [1]. Due to the uneven distribution of data requirements, the frequency resource utilization in any system will be also unevenly distributed. Cells with high resource utilization are called overloaded cells. These cells have less available resources, which may affect the quality of service (QoS) for users. Load balancing means to transfer some traffic from overloaded cells to other available cells to balance the frequency resources among cells. It is an important method to optimize resource utilization and improve QoS. Reference [2] reviews the historical developments of load balancing and provides guidance and a roadmap for developing load balancing. However, there is a growing demand for high-rate communication in remote areas with sparsely deployed terrestrial cells. In these areas, the frequency resources are insufficient, such that the cells are more likely to overload. What is more, neighboring cells found by traditional load balancing schemes may not suit being the offloading target cells due to the limitation of coverage. Since neighboring cells are not always available, reference [3] suggests using cells with high coverage to achieve load balancing. According to the architecture of the future network [4], non-terrestrial cells will provide coverage for these suburban, rural and island areas. Therefore, this paper studies an effective scheme for a satellite–terrestrial integrated network (STIN) to solve the overloading problem.

According to the different radio access technologies (RAT) in networks, load balancing is divided into intra-RAT load balancing and inter-RAT load balancing. Some work has

focused on intra-RAT load balancing: In [5], a load assignment policy and a target selection policy are proposed, which utilize the matching theory. In [6], traffic was transferred from overloaded cells to the neighboring cells with less load to guarantee seamless handover in 5G systems. In [7], for giant low-Earth-orbit satellite networks, the authors designed two different handover methods for users with predictable handover times and users with unpredictable handover times to ensure load balance. In [8], the authors used fuzzy logic to adjust the cell individual offset (CIO) parameter in the handover process, so that overloaded cells were more likely to trigger handover and the neighboring cells were more likely to accept handover users. In [9], the authors also focused on CIO, but they introduced reinforcement learning to suit dynamically changing environments. These schemes focus on the optimization of handover events. Without considering the characteristics of satellite coverage, these schemes may not be suitable for the STIN [10].

For inter-RAT load balancing, a heterogeneous network consisting of small cells and traditional macro cells is one of the typical research scenarios. The authors of [11] proposed a two-step mechanism based on two biases for RAT selection. The authors of [12] proposed an algorithm that adapts handover margins and time to trigger. Reference [13] proposes two different versions of simulated annealing to improve load balancing and spectral performance. On the other hand, STIN, which is expected to achieve seamless coverage and transmission, has also introduced many new challenges for current systems [14]. The authors of [15] used a content popularity and Stakelberg game model to propose an effective scheme for load balancing between unmanned aerial vehicle cells and macro cells. Reference [16] proposes an efficient scheme for load balancing by using Knapsack and Zipf. Reference [17] formulates the problem as a constrained, multi-objective linear programming problem to maximize the utilization efficiency between satellites. The authors of [18] analyzed the transmission characteristics of terrestrial and back-haul links to propose a greedy-based user association algorithm and a matching algorithm with user grouping for balancing the load by performing multiple iterations between users and cells. In [19], the authors noted that the current methods adopt the greedy strategy, which leads to the load imbalance problem in cells. Thus, they defined a load coefficient and added it to the reward function to make handover decisions while balancing loads. Reference [20] proposes a load balancing scheme based on a load measurement metric for both a terrestrial network (TN) and a non-terrestrial network (NTN). However, the metric ignores the impact of user changes in the future, which may result in poor performance over a long period of time.

Since most existing load-balancing methods are usually performed reactively after the overload occurs [21], it would be worthwhile to design a method of active load balancing that acts before overloading occurs. Fuzzy logic is an effective pre-evaluation method [22]. Thus, in this paper, the fuzzy evaluation metric (FEM) is proposed to evaluate the impact of users on overloading for both TN and NTN. Then, we propose a joint load balancing scheme to solve the problem of overloading in STIN. The joint scheme consists of two parts. To reduce the tendency of overload before it occurs, an access algorithm is proposed: deep reinforcement learning (DRL) is a widely utilized tool to make selection decisions in a dynamic environment [23]. However, the neural network will not learn effectively with too many states and actions to be explored. Therefore, the fuzzy-logic- and reinforcement-learning-based access algorithm (FLRL-AC) is proposed. Then, to further ensure load balance, an offloading algorithm is proposed to offload users after overloading occurs: Existing studies on offloading usually only take the received signal as a single metric. To take into account the impact of dynamic changes of users, a fuzzy-logic-based offloading algorithm (FL-OL) is proposed. Finally, the fuzzy-logic-based load balancing scheme (FL-LB) is proposed, which is the combination of FLRL-AC and FL-OL.

The rest is organized as follows. Section 2 describes the considered network structure and the problems and solutions discussed in this paper. Numerical and simulation results, which demonstrate and verify the analysis, are presented in Section 3. Section 4 describes

advantages and future research areas of this study. Finally, our concluding remarks are made in Section 5.

## 2. Materials and Methods

This section firstly describes the structure and problems to be solved in STIN. Then the proposed solutions are described: The FEM is proposed to pre-evaluate the impact of users on overloading. In the proposed FL-LB, the access algorithm runs in a centralized control cell and determines initial access in order to prevent overloading before it occurs. The offloading algorithm runs in each cell and determines how to offload users in the overloaded cells. The proposed metric and algorithms are described in detail in the following subsections.

### 2.1. Network Structure

We consider an  $L \times L$  area in the STIN network. There are in total  $N$ , cells including  $N - 1$  terrestrial cells, and one satellite hanging overhead. Terrestrial cells are modeled with reference to [24] and distributed sparsely in hexagonal cellular cell mode with the inter site distance (ISD)  $D$ . The satellite is modeled as a geostationary Earth orbit (GEO) satellite [25]. The coverage diameter of a single beam of the satellite is usually 50 to 250 km, which is very large compared with TN coverage. Therefore, in the system we assume the central beam of the satellite covers the whole TN area. TN and NTN cells differ in bandwidth, defined as  $B_g$  and  $B_s$ .  $M$  users are randomly distributed, requiring data transfer at a random rate with an average of  $R_{avg}$ . Users move at a fixed speed  $v$  and with random angles.

TN channel is modeled as a downlink channel with additive white Gaussian noise and Rayleigh fading. The maximum achievable rate is expressed as

$$C_{m,n} = B \frac{n_{RB}}{N_{RB}} \log_2 \left( 1 + \frac{p_s |h|^2}{\Gamma(\epsilon) d_{m,n}^2 N_0} \right), \quad (1)$$

where  $C_{m,n}$  is bit rate between user  $m$  and cell  $n$ ,  $n_{RB}$  is the number of assigned resource blocks (RB),  $B$  is the maximum usable bandwidth of the cell,  $N_{RB}$  is the total number of RB,  $p_s$  is the transmission power and  $\epsilon$  is the target block error rate (BLER).

$$\Gamma(\epsilon) = -\frac{2 \lg(5\epsilon)}{3} \quad (2)$$

represents the signal to noise ratio (SNR) margin to meet the desired target BLER with the QAM constellation [26].

$$h = \frac{1}{\sqrt{2}} (N(\mu_0, \sigma_0) + j \cdot N(\mu_0, \sigma_0)) \quad (3)$$

is the channel fading coefficient where  $N(\mu, \sigma)$  means a Gaussian random number in Equation (4).  $d_{m,n}$  is the distance between user  $m$  and cell  $n$ .  $N_0$  is the noise power.

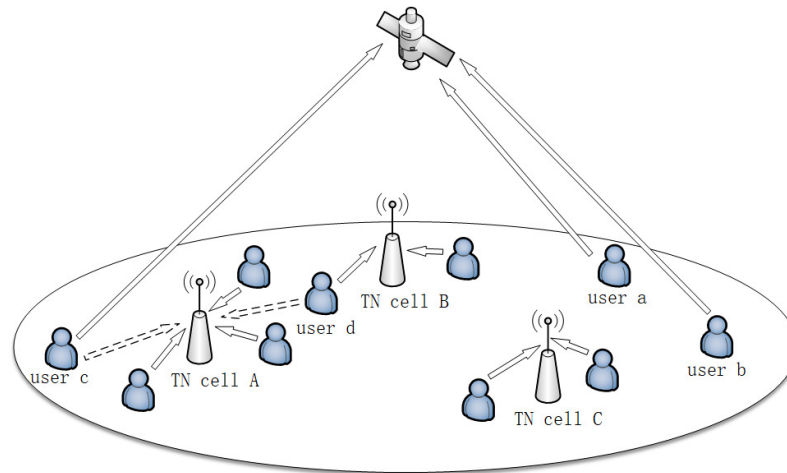
$$X \sim N(\mu, \sigma) : f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4)$$

The altitude of GEO is 35,536 km. It has a typical reflector antenna with a circular aperture [25]. We consider the simulation area is in the coverage area of the central beam and that the elevation angle of the beam's center is  $90^\circ$ . The NTN channel is modeled with reference to [27], and the inter-beam interference is modeled with reference to [28], and they are not described in detail due to our being limited in space.

### 2.2. Problem Description

The process of load balancing in STIN with sparsely deployed TN cells is shown in Figure 1. Some users are on the edge of the area, having poor signal reception and requiring

more bandwidth. Some users cannot find a better cell to handover. These make TN cells likely to overload in this network. An example is shown in Figure 1. The solid arrows point to cells that users are currently accessing. The dashed arrows point to cells that users previously accessed. User *a* and user *b* are users on the edge of the area that initially accesses the NTN cell. When the TN cell A suffers overloading, user *c* is selected to be offloaded to the NTN cell, and user *d* is selected to be offloaded to the adjacent TN cell B for relatively better QoS. Thus, the problem is split into two sub-problems: the access algorithm decides which cell users initially access, and the offloading algorithm decides which users are offloaded and how to offload them.



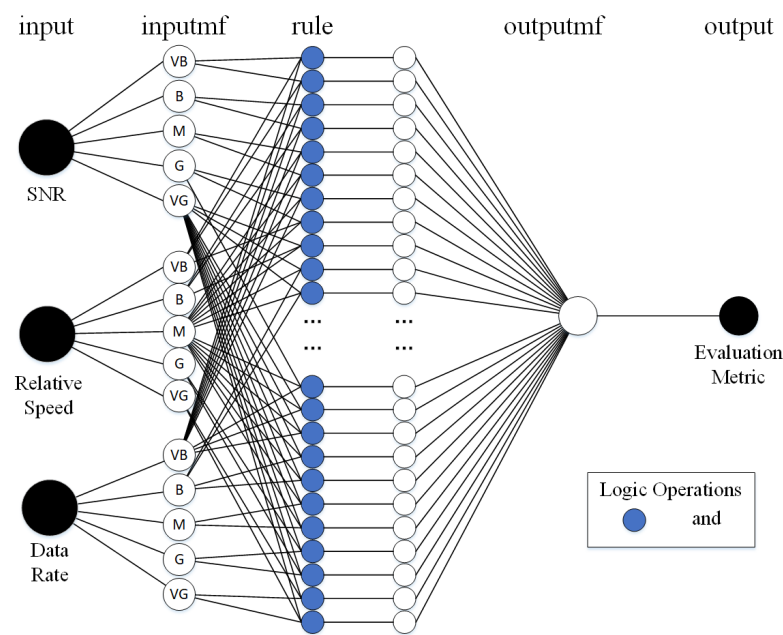
**Figure 1.** Diagram of load balancing in a satellite–terrestrial integrated network.

For the access algorithm, according to the link budget, in a scenario where cells are sparsely deployed, such as the rural scenario in 3GPP [24], the average SNR is 9.21 dB. In the coverage range of the central beam of a GEO satellite, the average SNR is only 2.95 dB. If the received signal quality is used to determine initial access, only a few users will access the NTN cell. As a result, NTN resources are not effectively utilized and TN cells are more likely to overload. To limit access to the NTN to suitable users, we aim to consider not only received signal, but also the impact of users on cell overloading. Therefore, fuzzy logic is utilized to propose an overload evaluation metric. Then, to deal with dynamic changes in the environment and determine the appropriate number of users allowed to access the NTN, the FLRL-AC is proposed.

For the offloading algorithm, even if the initial access is optimized, overloading may still occur after a certain period of time due to users' movements, especially when the user density in the environment is large. Additionally, if the FLRL-AC is utilized to make a global re-access decision at this time, the QoS of users in the cells which are not overload is affected. Therefore, the FL-OL is proposed for those overloaded cells. The FL-OL offloads the most suitable users to the most suitable cells by utilizing FEM.

### 2.3. The Fuzzy Evaluation Metric

Existing load balancing methods are usually performed reactively after the overloading occurs. Active load balancing adapts the network in advance to prevent overloading and improve performance. Therefore, this paper firstly proposes a metric to pre-evaluate the impact of users on overloading, which helps the networks to make further decisions. Considering the differences in carrier frequency and bandwidth between TN and NTN, and the difficulty of explicitly evaluating overload tendency, fuzzy control is utilized [29] for the metric. It provides a unified measurement to evaluate the impact of users on overloading in STIN. An adaptive neuro fuzzy network (ANFN) is utilized to build the fuzzy system. The training network is shown in Figure 2. In the following, the structure of each layer of the network is described.



**Figure 2.** Adaptive neuro fuzzy network structure.

In the input layer, SNR, relative speed and data rate are selected as the inputs by considering the Shannon formula  $C = B \log_2(1 + SNR)$ : SNR and the target rate. Channel with low SNR needs more bandwidth to transmit data at the same target rate, which is more likely to cause overload. In addition, if the user is getting closer to the cell, SNR between them will get better, otherwise it will get worse. Additionally, if the SNR is the same, users with higher data rates will require more bandwidth. Measurements of SNR and data rate requirements are in the same way in both RAT. The movement of user could be expressed by the relative speed between user and cell. Since the satellite is far from the Earth, the moved distance of the user in a short period of time can be ignored relative to the satellite height. Thus, the speed relative to the satellite is considered to be 0 in this paper. It is difficult to directly judge whether SNR, relative speed and data rate are high or not. Therefore, we take them as the three inputs in the “input” layer.

The “inputmf” layer uses membership functions to convert input values into fuzzy values. Commonly utilized membership functions are triangular, trapezoidal, Gaussian and bell-shaped. Since the relationship between inputs and output is not linear, the Gaussian membership function was selected.

$$f(x, c, \sigma) = e^{-\frac{(x-c)^2}{2\sigma^2}}, \quad (5)$$

where  $c$  determines the center position of the function.  $\sigma$  determines the width of the function. Both  $c$  and  $\sigma$  are trained by the ANFN. The fuzzy system has three inputs, and there are  $P$ ,  $Q$  and  $R$  fuzzy concepts for each input, respectively. The membership degrees of inputs to different fuzzy concepts could be calculated via the membership functions. In this paper, both  $P$ ,  $Q$  and  $R$  are set to 5, based on five fuzzy concepts: very bad (VB), bad (B), medium (M), good (G) and very good (VG). For each input, there are five membership functions subjecting to the same distribution with different parameters.

The “rule” layer pairs  $P$  fuzzy concepts of the first input,  $Q$  fuzzy concepts of the second input and  $R$  fuzzy concepts of the third input to obtain  $P \times Q \times R$  fuzzy rules. T-S fuzzy reasoning is utilized in the proposed fuzzy system. For the  $l$ th rule, the first input  $x$  is  $X_i$ , the second input  $y$  is  $Y_j$  and the third input  $z$  is  $Z_k$ . The mapping result  $u_l$  is calculated by

$$u_l = p_l x_i + q_l y_j + r_l z_k + c_l, \quad (6)$$



where  $l = (i - 1)QR + (j - 1)R + k$ ;  $i \in [1, P], j \in [1, Q], k \in [1, R]$ .  $p_l, q_l, r_l$  and  $c_l$  are parameters of the  $l$ th rule trained by ANFN;  $X_i, Y_j$  and  $Z_k$  are the  $i$ th,  $j$ th and  $k$ th fuzzy concepts of the three inputs;  $x_i, y_j$  and  $z_k$  are the membership degree values calculated by the corresponding membership functions.

The “outputmf” layer uses weighted average method to consider the influence of each fuzzy rule comprehensively. The output fuzzy evaluation metric  $f$  is calculated by

$$f = \frac{\sum w_l u_l}{\sum w_l} = \frac{\sum x_i y_j z_k u_l}{\sum x_i y_j z_k}, \quad (7)$$

where  $l = (i - 1)QR + (j - 1)R + k$ ,  $i \in [1, P], j \in [1, Q], k \in [1, R]$ .  $w_l$  is the weight of the  $l$ th rule calculated by the product method.

In order to train the ANFN, multiple groups of user trajectories; SNR; relative speeds and data rate requirements at each time and location; and the average required bandwidth for a period of time, were generated via simulation. The ANFN was trained with these simulation data. The smaller the value is, the greater the impact is.

#### 2.4. The Fuzzy-Logic- and Reinforcement-Learning-Based Access Algorithm

Traditionally, users access the cell with the best reference signal receiving power (RSRP) [30]. According to the link budget, this may not work well in the STIN studied in this paper. In order to reduce the occurrence of overloading, to reduce the frequency of calling offloading algorithms and ensure QoS, the intelligence of reinforcement learning is introduced to make access decisions. In the following, the proposed FLRL-AC is described.

Reinforcement learning is a common method for intelligent decision. It obtains learning information and updates model parameters by calculating the rewards for actions in the current state of the environment. Reinforcement learning is divided into two categories: One is value learning, which uses a neural network to approximate the optimal action value function, such as Q-Learning or a deep Q-network. The other is policy learning, which uses a neural network to approximate the policy function, such as the actor–critic method. In this paper, reinforcement learning is used to select the initial access cell for each user so that the action dimensions are  $N^M$ . Due to exponential expansion and large dimensions of action space, a deep deterministic policy gradient (DDPG) [31] which is compatible with a large dimension state and actions is utilized in the proposed algorithm. FEM is utilized in DDPG in order to further reduce the difficulty of training and enable the decision to have a better impact on the future state, which is called fuzzy deep deterministic policy gradient (FDDPG) in this paper. In the following, we will describe the Markov decision process of the problem and the FDDPG training process.

(1) State space: The state space describes the environment. It reflects the relative positions, relative motions and channel states between cells and users. Thus, the state at time  $t$  is defined as

$$s^t = (f^t(1, 1), f^t(1, 2), \dots, f^t(M, N)), \quad (8)$$

where  $f^t(m, n)$  means the FEM between user  $m$  and cell  $n$  at time  $t$ .

(2) Action space: To make better access decisions, the index of cell is set as the action. In Equation (9),  $AC^t(m) \in [0, N]$  means the index of cell which user  $m$  accesses at time  $t$ .

$$a^t = (AC^t(1), AC^t(2), \dots, AC^t(M)). \quad (9)$$

(3) Reward function: The proposed access decision method aims at maximizing the total reward of the access selections for all users. The discounted total reward is

$$R^t = \sum_{T=0}^{\infty} \gamma^T \cdot r^{t+T} = \gamma^T (a_1 \cdot r_1^{t+T} + a_2 \cdot r_2^{t+T} + a_3 \cdot r_3^{t+T}), \quad (10)$$

where  $\gamma$  is the reward discount and  $r^t$  is the per-time reward. In this paper, the per-time reward is composed of three parts,  $r_1^t$ ,  $r_2^t$  and  $r_3^t$ . For the first part,

$$r_1^t = \frac{\sum_{m=1}^M f^t(m, AC^t(m))}{M}, \quad (11)$$

the design goal is to reduce the overloading tendency in the future. Thus,  $r_1^t$  equals the average value of the FEM of all users. At meanwhile, for the second part,

$$r_2^t = \sum_{n=1}^N O^t(n), \quad (12)$$

we hope to keep load balanced at the current time by minimizing the number of overloaded cells.  $r_2^t$  equals the overload penalty for all cells, where

$$O^t(n) = \begin{cases} 1, & \text{if } \eta_n^t \geq \Theta \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

indicates whether cell  $n$  is overloaded at time  $t$ .  $\eta_n^t = \frac{B_n^t}{B_C}$  is the radio resource utilization ratio (RRUR) of cell  $n$  at time  $t$ , where  $B_n^t$  is the occupied portion of the bandwidth of cell  $n$  and  $B_C$  is the total bandwidth resources of the cell. For terrestrial cells,  $B_C$  is equal to  $B_g$ . Additionally, for the satellite,  $B_C$  is equal to  $B_s$ .  $\Theta$  is the overload threshold. For TN cells, the threshold is  $\Theta_g$ . Additionally, for the NTN cell, the threshold is  $\Theta_s$ . In order to minimize the resource utilization ratio and balance the resource utilization, the third part of the reward is defined as

$$r_3^t = \sum_{n=1}^{N-1} \eta_n^t + w_s \cdot \eta_N^t + \frac{\sum_{n=1}^N \left( \eta_n^t - \frac{\sum_{n=1}^N \eta_n^t}{N} \right)^2}{N}, \quad (14)$$

where the weighted value of the sum and the variance of RRUR is included in  $r_3^t$ . Due to the large bandwidth, the NTN resources are enough to load many users. If there is no additional limit on accessing NTN, the agent will tend to let as many users as possible access the NTN to avoid overloading in TN cells. In order to prevent the transmission delay of users from being affected, we use the additional weight  $w_s$  to increase the impact of resource utilization of the NTN cell.

(4) The training process of the FDDPG algorithm: As in DDPG, FDDPG has two components, actor and critic. The actor network defined as  $\mu(s^t)$  takes  $s^t$  as input and returns action  $a^t$ . The critic network defined as  $Q(s^t, a^t)$  returns long-term reward based on states and actions.  $Q(s^t, a^t)$  can be expressed as

$$Q(s^t, a^t) = E[r^t | s^t, a^t] \approx E[r^t + \gamma Q(s^{t+1}, \mu(s^{t+1}))] \quad (15)$$

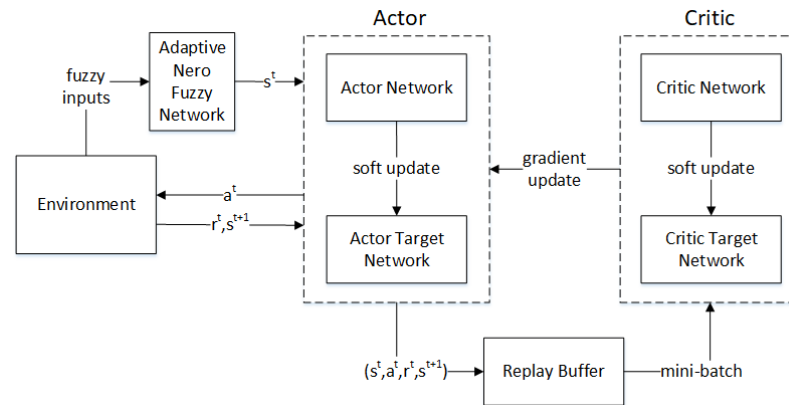
according to the Bellman equation, where  $E[\cdot]$  means expectation and  $\gamma$  means the reward discount.

DDPG combines actor–critic and DQN, so there are four networks in total. The actor network  $\mu(s^t)$  and actor target network  $\mu'(s^t)$  have the same structure, but different parameters  $\theta^\mu$  and  $\theta^{\mu'}$  and different update frequencies. The critic network  $Q(s^t, a^t)$  and critic target network  $Q'(s^t, a^t)$  have the same structure, but different parameters  $\theta^Q$  and  $\theta^{Q'}$  and different update frequencies. For the activation function, the linear rectification function (ReLU) is utilized in hidden layers and the hyperbolic tangent function is utilized in output layers.

Figure 3 shows the training structure of the proposed fuzzy reinforcement learning. In each episode during the training process, users' positions and velocities are randomly reset to reset the environment. In each training step, the three fuzzy inputs are obtained from the environment and the state  $s^t$  is obtained based on Equation (8). In order to speed up training,  $s^t$  is normalized to get

$$\hat{s}^t = (s^t(m, n) - \text{mod}(s^t(m, n), \beta)) \quad (16)$$

with the normalization coefficient  $\beta$ , where  $m \in [1, M], n \in [1, N]$ .



**Figure 3.** The fuzzy deep deterministic policy gradient training structure.

To explore new states, the output of the actor network is added by random noise. The action  $a^t$  is obtained by

$$a^t = \mu(\hat{s}^t | \theta^\mu) + N(0, var), \quad (17)$$

in which the noise is a Gaussian random number with mean value of 0 and variance of  $var$ . After performing  $a^t$  on the environment,  $r^t$  and the next state  $\hat{s}^{t+1}$  can be obtained from the output of the environment. To break the association between data,  $(\hat{s}^t, a^t, r^t, \hat{s}^{t+1})$  is stored in a replay buffer.  $\theta^\mu$ ,  $\theta^{\mu'}$ ,  $\theta^Q$  and  $\theta^{Q'}$  are updated by sampling a mini-batch with size  $K$  from the replay buffer. The loss function of the critic network is defined as

$$\begin{aligned} L(\theta^Q) &= E_{\mu'}[(y^t - Q(\hat{s}^t, a^t | \theta^Q))^2], \\ y^t &= r^t + \gamma Q'(\hat{s}^{t+1}, \mu'(\hat{s}^{t+1}) | \theta^{Q'}), \end{aligned} \quad (18)$$

which is the temporal difference error between the outputs of  $\theta^Q$  and  $\theta^{Q'}$ . Thus, the gradient of critic network is calculated by

$$\nabla_a Q(s, a | \theta^Q) |_{s=\hat{s}^t, a=\mu(\hat{s}^t | \theta^\mu)}. \quad (19)$$

By applying the chain rule to the expected return from the start distribution  $J$  with respect to the actor parameters [31], the actor is updated by Equation (20).  $\theta^Q$  and  $\theta^\mu$  can be updated via gradient descent method.

$$\nabla_{\theta^\mu} J \approx \frac{1}{K} \sum_t [\nabla_a Q(s, a | \theta^Q) |_{s=\hat{s}^t, a=\mu(\hat{s}^t | \theta^\mu)} \nabla_{\theta^\mu} \mu(s | \theta^\mu) |_{s=\hat{s}^t}]. \quad (20)$$

The weights of target networks  $\theta^{Q'}$  and  $\theta^{\mu'}$  are updated based on the weights of  $\theta^Q$  and  $\theta^\mu$ , as in Equations (21) and (22). The detailed training process is shown in Algorithm 1.

$$\theta^{Q'} \leftarrow \theta^Q + (1 - \tau)\theta^{Q'}. \quad (21)$$



$$\theta^{\mu'} \leftarrow \theta^{\mu} + (1 - \tau)\theta^{\mu'}. \quad (22)$$

---

**Algorithm 1.** Fuzzy reinforcement learning training algorithm.
 

---

**Require:** Training episodes  $E_{max}$ , training steps for each episode  $T_{max}$ , learning rate of actor network  $\alpha_A$ , learning rate of critic network  $\alpha_C$ , initial exploration rate  $v_{init}$ , exploration discount  $v_{dis}$ , minimum exploration rate  $v_{min}$ , replay buffer size  $G$ , mini-batch size  $K$ , reward discount  $\gamma$ , update rate  $\tau$ .

- 1: Randomly initialize the weights of actor network and critic network as  $\theta^{\mu}$  and  $\theta^Q$ , the initial weight of actor target network  $\theta^{\mu'}$  is same as  $\theta^{\mu}$  and the initial weight of critic target network  $\theta^{Q'}$  is same as actor network  $\theta^Q$
  - 2: Initialize the empty replay buffer, initialize exploration rate  $var$  as  $v_{init}$
  - 3: **for** each episode  $ep$  in range  $(1, E_{max})$  **do**
  - 4:   Randomly set users' positions and speeds and data requirements to reset environment
  - 5:   **for** each step  $t$  in range  $(1, T_{max})$  **do**
  - 6:     Get fuzzy inputs from the environment, get  $s^t$  and  $\hat{s}^t$  from Equations (8) and (16)
  - 7:     Get  $a^t$  from Equation (17)
  - 8:     Perform  $a^t$  to the environment and get  $r^t$  from Equations (11)–(14)
  - 9:     Get  $s^{t+1}$  and  $\hat{s}^{t+1}$  from Equations (8) and (16)
  - 10:    Store  $(\hat{s}^t, a^t, r^t, \hat{s}^{t+1})$  in replay buffer
  - 11:    **if** the replay buffer is full **then**
  - 12:     Replace a data randomly by  $(\hat{s}^t, a^t, r^t, \hat{s}^{t+1})$
  - 13:     Update exploration rate,  $var = \max(var \cdot v_{dis}, v_{min})$
  - 14:     Sample a mini-batch of size  $K$  from the replay buffer
  - 15:     Update  $\theta^Q$  by Equation (19)
  - 16:     Update  $\theta^{\mu}$  by Equation (20)
  - 17:     Update target networks by Equations (21) and (22)
  - 18:    **end if**
  - 19:   **end for**
  - 20: **end for**
  - 21: Training completed, save the actor network
- 

### 2.5. The Fuzzy-Logic-Based Offloading Algorithm

The algorithm above provides the access policy for all users to prevent cells from overloading. However, with the irregular movements of users, some cells could still overload after a long enough period of time. At this moment, to ensure the service quality both at the time of offloading and in the future, FL-OL is proposed to select appropriate users to be offloaded. In the following, the proposed FL-OL is described.

Consider a set of cells  $S = \{s_1, s_2, \dots, s_N\}$ . Whenever the resource utilization rate  $\eta_i$  of the TN cell  $s_i \in S, i \neq N$  is higher than the threshold  $\Theta_g$ , it is considered as an overloaded TN cell. Consider a set of users  $U^i = \{u_1^i, u_2^i, \dots, u_{M_i}^i\}$  which are served by cell  $s_i$ .  $M_i$  is the total number of users served by the cell  $s_i$ .  $f_j^i$  is the FEM between cell  $s_i$  and user  $u_j^i$ ;  $j \in [1, M_i]$ .  $u_j^i$  is the user with the minimum value of the evaluation metric  $f_{min} = f_j^i$ . Calculate the FEM  $f_j^{i'}$  between cell  $s_{i'}, i' \in [1, N], i' \neq i$  and user  $u_j^i$ . Due to the long distance between users and the GEO satellite, relative speed can be ignored for the NTN cell  $s_N$ . Thus,  $f_j^N$  is only influenced by user's position and the randomness of the channel. If there is any  $f_j^{i'}$  higher than  $f_{min}$ , select the cell  $s_{\hat{i}}$  with the maximum value of the evaluation metric. If  $\hat{i} \neq N$ , check whether  $\eta_{\hat{i}}$  is higher than  $\Theta_g$  after offloading  $u_j^i$  to  $s_{\hat{i}}$ . If not,  $s_{\hat{i}}$  is selected as the target cell. Otherwise, it means that user  $u_j^i$  is at the outer edge of the area and there is no more suitable TN cell for this user. Thus,  $u_j^i$  is offloaded to the NTN cell on the premise that  $\eta_N$  is not higher than  $\Theta_s$ . The algorithm will continue to find new  $u_j^i$  with

new  $f_{min}$  and offload until the current cell  $s_i$  is no longer overloaded. The whole process is summarized by Algorithm 2 as follows.

---

**Algorithm 2.** The fuzzy logic offloading algorithm.

---

**Require:** Resource utilization rate  $\eta_k, k \in [1, N]$  of cells in  $S = \{s_1, s_2, \dots, s_N\}$

```

1: while  $\eta_i \geq \Theta_g, i \in [1, N - 1]$  do
2:   Get users  $U^i = \{u_1^i, u_2^i, \dots, u_{M_i}^i\}$  severed by  $s_i$ 
3:   for  $u_j^i$  in  $U^i, j \in [1, M_i]$  do
4:     Get  $f_j^i$  by FEM and save in a set  $F$ 
5:   end for
6:   Sort  $F$  in ascending order and get the first user  $u_j^i$  with  $f_{min} = f_j^i$ 
7:   for  $i' \in [1, N - 1], i' \neq i$  do
8:     Calculate  $\eta_{i'}$  assuming  $u_j^i$  is offloaded to  $s_{i'}$ 
9:     if  $f_j^{i'} > f_{min}$  and  $\eta_{i'} < \Theta_g$  then
10:      Save  $f_j^{i'}$  in a set  $F'$ 
11:    end if
12:  end for
13:  if  $F'$  is not empty then
14:    Sort  $F'$  in descending order and get the first cell  $s_{i'}$ 
15:    Offload  $u_j^i$  to  $s_{i'}$ 
16:  else if  $\eta_N < \Theta_s$  then
17:    Offload  $u_j^i$  to NTN cell
18:  else
19:    break;
20:  end if
21:  Update  $\eta_k, k \in [1, N]$ 
22: end while

```

---

### 3. Results

In this section, the simulation scenario is presented and parameters are set to train the networks both by the proposed scheme and several typical algorithms, including DDPG, proximal policy optimization (PPO) [32] and the adaptive multi-RAT mobile offloading algorithm (AMMO) [20]. Simulation results show that compared with baseline schemes, the proposed scheme solves the overloading problem in STIN more effectively.

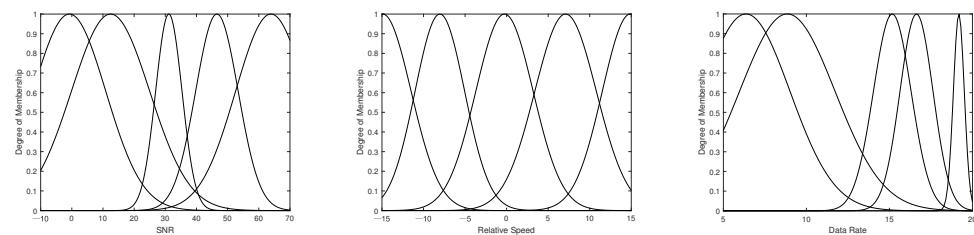
#### 3.1. Simulation Environment

Referring to the rural scenario in 3GPP [24],  $D$  was set to 1732 m. We supposed there were three TN cells in the area; thus,  $N = 4$ . The width of the area  $L$  was set to 3500 m. To compare the performance of the proposed scheme with different user densities, 5 to 30 users per TN cell were deployed, which means  $M$  was set from 15 to 90. The TN cells worked at 4 GHz with 30 MHz bandwidth. The GEO satellite worked in the Ka band (20 GHz) with 400 MHz bandwidth. The threshold of TN was set to 85%. Considering that the simulation area only accounts for a small part of the satellite coverage area, the threshold of NTN was set to 30%. There was no signal interference between the cells of TN and NTN. Other parameters mentioned in the previous sections were configured as in Table 1.

**Table 1.** Parameter configuration.

Parameter	Value
$L$	3500 m
$N$	3
$M$	15~90
$D$	1732 m
$B_g$	30 MHz
$B_s$	400 MHz
$\epsilon$	0.1
$p_s$	46 dBm
$N_0$	$5 \times 10^{-5}$ W
$\mu_0$	0
$\theta_h$	1
$v$	10 m/s
$R_{avg}$	10 Mbps
$P$	5
$Q$	5
$R$	5
$\Theta_g$	85%
$\Theta_s$	30%

Membership functions of the three inputs trained by ANFN are shown in Figure 4. The performances of the proposed scheme based on FEM are described in the following subsections.

**Figure 4.** Input membership functions.

### 3.2. Performance of the Fuzzy-Logic- and Reinforcement-Learning-Based Access Algorithm

The FDDPG model utilized in FLRL-AC was trained by Algorithm 1 and the hyper-parameters in Table 2. In order to reflect the improvements of FDDPG, two other baseline algorithms were utilized to train the neural networks. PPO is based on an actor–critic network similar to that of DDPG, which can solve the problem of continuous control. It balances the difficulty of implementation, the complexity of sampling and the effort required for debugging. For these reasons, it is widely used as a default reinforcement learning algorithm for new problems. The other baseline algorithm was DDPG without the ANFN, in which the state function and reward function are defined as

$$s^t = (\zeta^t(1,1), \zeta^t(1,2), \dots, \zeta^t(M,N)), \quad (23)$$

$$r_1^t = \frac{\sum_{m=1}^M \zeta(m, AC^t(m))}{M}, \quad (24)$$

instead of Equations (8) and (11); and  $\zeta^t(m,n)$  means SNR between user  $m$  and cell  $n$  at time  $t$ .

**Table 2.** Training parameters.

Parameter	Value
$E_{max}$	10,000
$T_{max}$	100
$a_1$	0.1
$a_2$	0.3
$a_3$	0.2
$w_s$	2
$\alpha_A$	0.001
$\alpha_C$	0.002
$\beta$	5
$v_{init}$	10
$v_{dis}$	0.999997
$v_{min}$	0.01
$G$	5000
$K$	64
$\gamma$	0.001
$\tau$	0.01

The environment was reset and users were dropped randomly in each drop. Users accessed cells based on different algorithms at time slot 0 and moved in later time slots. The overload ratio (OLR) is defined as

$$OLR = \frac{N_{ol}}{N_{total}}, \quad (25)$$

where  $N_{ol}$  is the number of time slots in which overloading occurs and  $N_{total}$  is the total number of the simulation slots.

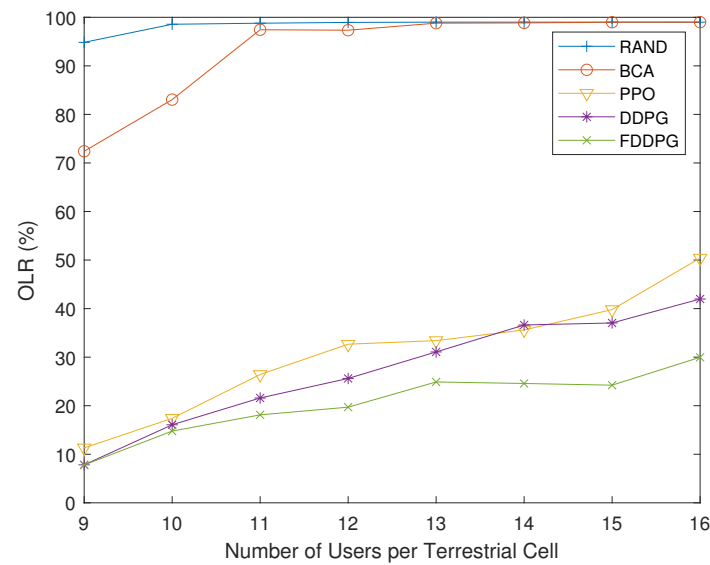
Results with different access algorithms are shown in Figure 5. RAND means users access randomly, which is considered as the lower limit. BCA [30] means users access with the best channel state. As the number of users increases, the OLR increases for all algorithms. The purpose of designing FDDPG and FLRL-AC was to pre-process the factors affecting overloading by ANFN and to reduce the training difficulty of the network in a high-dimensional state and action space. Additionally, the FEM calculated by ANFN gives a more accurate evaluation of the impact of users on overloading. Therefore, taking the statistics of FEM as the reward function could enable the agent to make better decisions. The advantages brought by FDDPG are not obvious when the number of users is small and the training dimensions are not too high. Therefore, the three curves of PPO, DDPG and FDDPG are close. As the number of users increases, the results of BCA become unacceptable because the resources of NTN cell are not effectively utilized, resulting in overloading of TN cells. PPO and DDPG have similar performances to the two baseline algorithms, and the gain brought about by FDDPG becomes more and more obvious with the increase users: up to 29%.

Users who achieve their data requirements are called satisfied users. The satisfied ratio (SR) is defined as

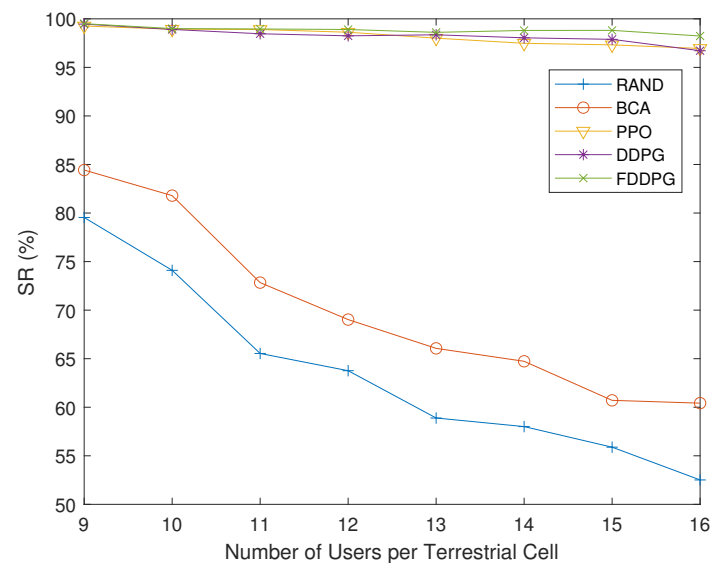
$$SR = \frac{M_s}{M_{total}}, \quad (26)$$

where  $M_s$  is the number of satisfied users and  $M_{total}$  is the total number of users.

Results in Figure 6 show that with RAND and BCA, many users fail to achieve their target rates. With the three access algorithms based on reinforcement learning, there are always more than 97% of users who achieve their data rate requirements, which again proves that active load balancing can significantly improve QoS.



**Figure 5.** The overload ratio with different algorithms.

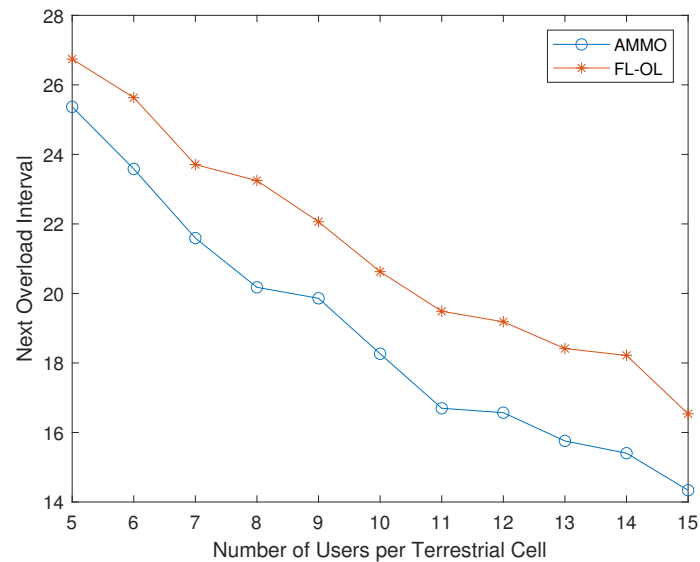


**Figure 6.** The satisfied ratio with different algorithms.

### 3.3. Performance of the Fuzzy-Logic-Based Offloading Algorithm

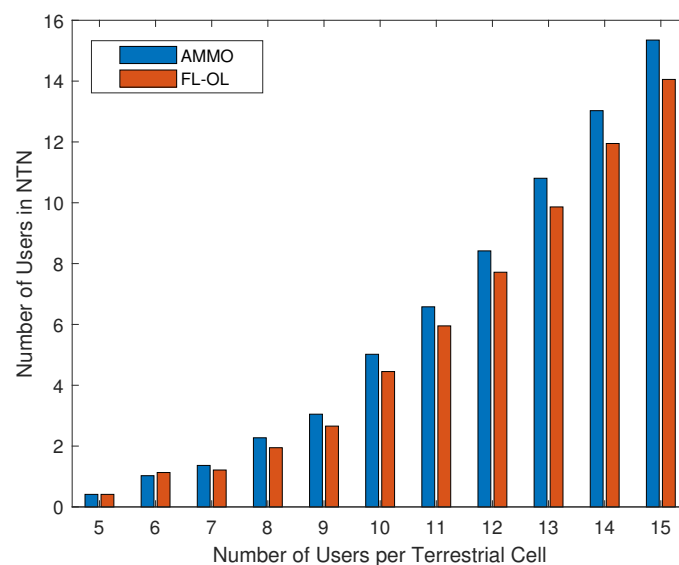
In order to evaluate the performance of the offloading algorithm, a special user dropping method was used. Two of the cells with more users were dropped to emulate overloading. Additionally, users accessed cells based on the basic algorithm, BCA. The proposed FL-OL is compared with AMMO. Whenever overloading occurred, the two offloading algorithms were utilized, respectively, to offload users to other available cells. After a period of time, the cell could overload again due to the dynamic changes of the environment. The time interval of the first or second overload time is called the next overload interval. Although the current signal reception quality of some users is poor, they may be moving toward the current serving cell and away from adjacent cells. Such users are likely to be selected if the offloaded users are selected only according to RSRP, which may bring a heavy load to the adjacent cell. The rate requirement is also an important factor. Low rate users have little impact on the overloading and should have a lower priority for being offloaded. Unlike AMMO, FL-OL takes the above factors into account by introducing FEM and should be able to prolong the time with a balanced load for the system.

Figure 7 shows the next overload interval for the two algorithms and different numbers of users. The next overload interval of both algorithms decreases with the increase in users due to limited frequency resources. FL-OL makes the next overload interval longer by about 17%. It proves our inference that by considering future changes, FL-OL prolongs the time that the system maintains load balance. In the long run, it will reduce the number of times of calling the offloading algorithm and the number of times of handover.



**Figure 7.** Next overload interval with different algorithms.

With an increase in users and limited TN frequency resources, it is more difficult to balance load when only relying on TN cells. The number of users served by each satellite is increasing. However, there are always less users offloaded to NTN with the proposed FL-OL due to it making more effective use of the TN's resources. As shown in Figure 8, FL-OL reduces users affected by the delay of long-distance transmission of satellite-user link by up to 14% compared with the existing algorithm.

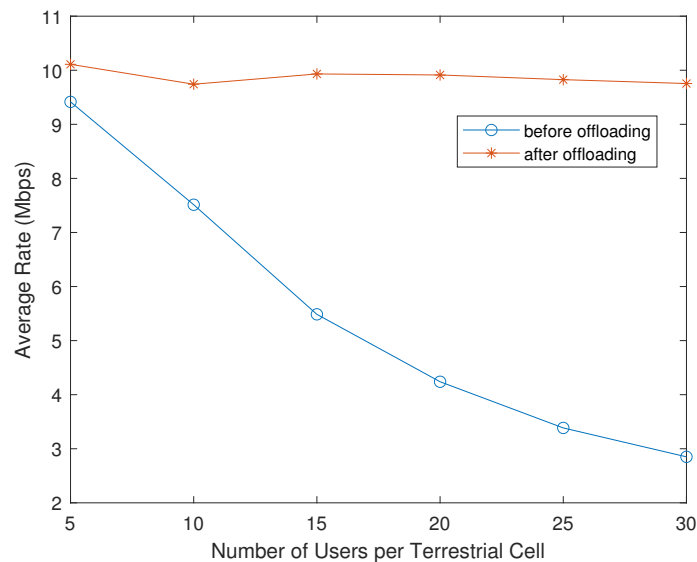


**Figure 8.** Number of users served by a satellite with different algorithms.

Furthermore, to verify the effectiveness of the proposed algorithm, Figure 9 shows the average data rate before and after offloading. With an increase in users, the limited



resources of the overloaded cell struggle more difficult to meet users' rate requirements so that the average rate decreases significantly. After performing offloading, the average rates increased by 10% to 70% with different user numbers, and users basically met their target data rates. The results show that overloading will significantly affect QoS, and once again verifies that the scheme of reducing the overloading through active load balancing in this paper is meaningful.



**Figure 9.** Average rate with different algorithms.

### 3.4. Performance of the Fuzzy-Logic-Based Loading Balancing Scheme

In previous subsections, the performances of the two parts of the FL-LB were evaluated. Combining the two parts, the performance of FL-LB is evaluated in this subsection. Different access algorithms and offloading algorithms were paired to obtain the following four baseline joint schemes. BCA~RSRP means to access based on BCA and offload based on RSRP. BCA~FL-OL means to access based on BCA and offload based on FL-OL. FLRL-AC~RSRP means to access based on FLRL-AC and offload based on RSRP. Finally, FLRL-AC~FL-OL is the FL-LB proposed in this paper, which means to access based on FLRL-AC and offload based on FL-OL.

Results in Figures 10 and 11 show that BCA~RSRP had the worst performance with all user numbers because it does not consider any dynamic changes in the environment. FLRL-AC~RSRP performed better with smaller numbers of users, and BCA~FL-OL performed better with larger number of users. This is because in the two parts of the FL-LB, the access algorithm plays a more important role when the number of users is small. No matter how good the access decision is, overloading will occur when the number of users is too large. Therefore, a better offloading algorithm should be used for the overloading cells in time. This is the reason why the FL-LB proposed in this paper retains the reactive load balancing part. Finally, FL-LB had the best performance with all user numbers. Therefore, FL-LB can effectively reduce the occurrence of overloading in a longer term and improve QoS for users in STIN.

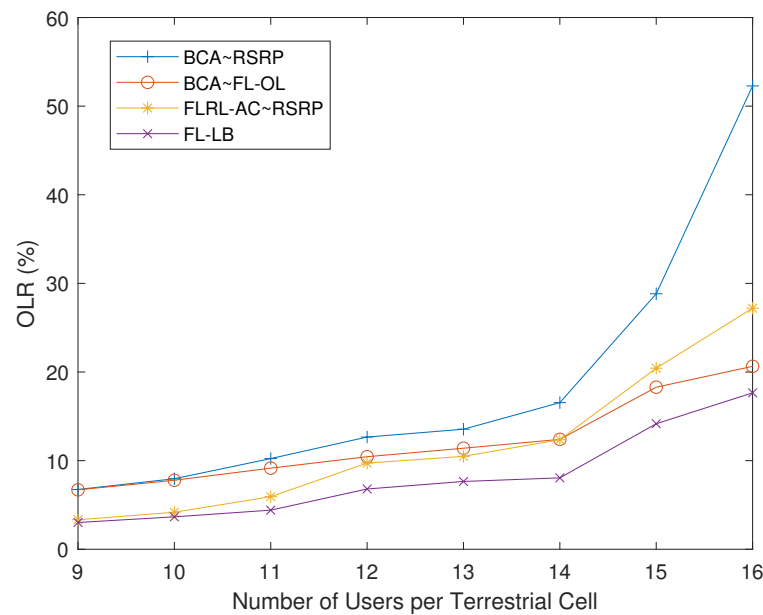


Figure 10. The overload ratio with different joint schemes.

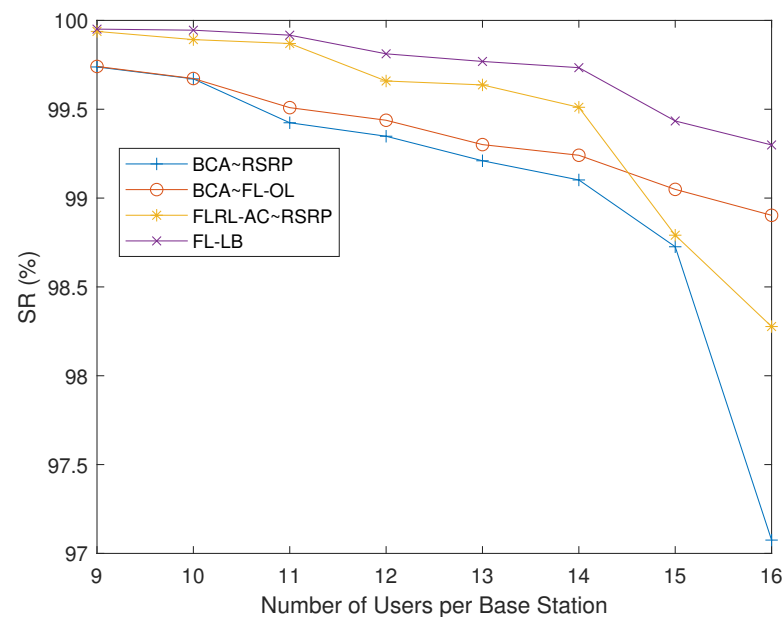


Figure 11. The satisfied ratio with different joint schemes.

#### 4. Discussion

Contributions of this paper are presented in this section. Existing load balancing schemes usually focus on the offloading algorithms, which are executed after overloading. Note that once overloading occurs, data rates of users decrease significantly. Therefore, we utilized FEM to learn and evaluate users' impacts on overloading in the future, and propose an active load balancing scheme to ensure data rates. Furthermore, the proposed FDDPG adds an adaptive neuro fuzzy network before the original DRL network. Compared with the widely utilized PPO and DDPG, FDDPG pre-filters the data relations that the DRL network needs to learn, thereby reducing the training difficulty of the DRL network and obtaining better training results. On the other hand, the proposed FL-OL makes FL-LB retain the ability of reactive load balancing on the basis of active load balancing. When the ANFN is trained, the complexity of calculating FEM is only related to the number of input

variables and the number of membership functions in the ANFN. Therefore, the computational complexity of calculating FEM will not increase with increases in base stations and users. Compared with the existing method, FL-OL maintains the same complexity  $O(MN)$  while utilizing FEM to consider the impact of offloading selection on overload in the future and extending the next overload interval.

However, FDDPG still needs more than  $10^6$  training steps even after pre-filtering by FEM. This is because the proposed scheme hopes to run on the central control cell and give the access decision for all of the users. This inevitably leads to a large state space and action space. On the other hand, compared with the existing algorithms that only focus on reactive load balancing, the FLRL-AC in FL-LB obviously increases the computational complexity, even though it is thought acceptable compared with the QoS gain. In order to solve these problems, multi-agent deep reinforcement learning (MADRL) based load balancing is a future research area. If FLRL-AC is distributed to each cell, the complexity of training will be greatly reduced. Additionally, in this case, since the decision dimension is the same, the access algorithm and the offloading algorithm can be more deeply combined so that to reduce the additional computing overhead brought by the access algorithm.

## 5. Conclusions

Existing studies on load balancing in STIN only considered a single metric of signal reception. When users move and require data randomly, the baseline schemes may not have acceptable performance in the long term. Active load balancing methods could obtain performance gains compared with the existing reactive methods. Considering the randomness of users in the future and the difficulty of explicitly evaluating overload tendency, we proposed an overload tendency evaluation metric based on fuzzy logic. Then, the overloading problem in STIN was solved by the proposed FL-LB: An access algorithm for all users called FLRL-AC was proposed, which prevents overloading while considering the characteristics of NTN. An offloading algorithm for the already overloaded cells called FL-OL was proposed, which balances load between cells. The fuzzy logic network is trained by ANFN, and the neural network in FLRL-AC is trained by FDDPG. Results show that FL-LB reduces the possibility of overloading before it occurs and makes cells maintain a longer load balance after offloading to ensure QoS for users.

**Author Contributions:** Conceptualization, Y.G. and H.Y.; methodology, Y.G. and H.Y.; software, H.Y. and X.W.; validation, Y.C. and C.L.; formal analysis, Y.G.; investigation, H.Y., X.W., Y.C. and C.L.; resources, X.Z. and Y.G.; data curation, H.Y.; writing—original draft preparation, H.Y. and Y.G.; visualization, H.Y.; supervision, Y.G.; project administration, Y.G. and X.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** All relevant data presented in the article are stored according to institutional requirements, and as such are not available online. However, all data used in this manuscript can be made available upon request to the authors.

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

## References

1. 3GPP TR 38.913; Study on Scenarios and Requirements for Next Generation Access Technologies. 3GPP: Sophia Antipolis, France, 2020.
2. Gures, E.; Shayea, I.; Ergen, M.; Azmi, M.H.; El-Saleh, A.A. Machine Learning-Based Load Balancing Algorithms in Future Heterogeneous Networks: A Survey. *IEEE Access* **2022**, *10*, 37689–37717. [[CrossRef](#)]
3. Tey, F.J.; Wu, T.Y.; Wu, Y.; Chen, J.L. Generative Adversarial Network for Simulation of Load Balancing Optimization in Mobile Networks. *J. Internet Technol.* **2022**, *23*, 297–304.

4. Yang, P.; Xiao, X.; Xiao, M.; Li, S.Q. 6G Wireless Communications: Vision and Potential Techniques. *IEEE Netw.* **2019**, *33*, 70–75. [\[CrossRef\]](#)
5. Park, J.; Kim, Y.; Lee, J.R. Mobility Load-Balancing Method for Self-Organizing Wireless Networks Inspired by Synchronization and Matching with Preferences. *IEEE Trans. Veh. Technol.* **2018**, *67*, 2594–2606. [\[CrossRef\]](#)
6. Zreikat, A.I. Load Balancing Call Admission Control Algorithm (CACA) Based on Soft-Handover in 5G Networks. In Proceedings of the 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 26–29 January 2022.
7. Zhang, T.T.; Yang, L.T.; Dong, T.; Yin, J.; Liu, Z.H.; Wang, Z.W. A Multi-Attribute Decision Handover Strategy for Giant LEO Mobile Satellite Networks. In Proceedings of the 6th International Conference on Smart Computing and Communication (SmartCom), Online, 29–31 December 2021.
8. Chen, Y.S.; Chang, Y.J.; Tsai, M.J.; Sheu, J.P. Fuzzy-Logic-Based Handover Algorithm for 5G Networks. In Proceedings of the 2021 IEEE Wireless Communications and Networking Conference (WCNC), Nanjing, China, 29 March–1 April 2021.
9. Asghari, M.Z.; Ozturk, M.; Hämäläinen, J. Reinforcement Learning Based Mobility Load Balancing with the Cell Individual Offset. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Helsinki, Finland, 25–28 April 2021.
10. Giuseppi, A.; Maaz, S.S.; Santis, E.; Ho Won, S.; Kwon, S.; Choi, T. Design and Simulation of the Multi-RAT Load-balancing Algorithms for 5G-ALLSTAR Systems. In Proceedings of the 2020 International Conference on Information and Communication Technology Convergence (ICTC), Jeju, Korea, 21–23 October 2020.
11. Ghatak, G.; De Domenico, A.; Coupechoux, M. Coverage Analysis and Load Balancing in HetNets With Millimeter Wave Multi-RAT Small Cells. *IEEE Trans. Wirel. Commun.* **2018**, *17*, 3154–3169. [\[CrossRef\]](#)
12. Hatipoğlu, A.; Başaran, M.; Yazici, M.A.; Durak-Ata, L. Handover-based Load Balancing Algorithm for 5G and Beyond Heterogeneous Networks. In Proceedings of the 2020 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Brno, Czech Republic, 5–7 October 2020.
13. Agarwal, B.; Ruffini, M.; Muntean, G.-M. Reduced Complexity Optimal Resource Allocation for Enhanced Video Quality in a Heterogeneous Network Environment. *IEEE Trans. Wirel. Commun.* **2022**, *17*, 2892–2908. [\[CrossRef\]](#)
14. Zhao, Z.X.; Du, Q.H.; Wang, D.W.; Tang, X.; Song, H.B. Overview of Prospects for Service-Aware Radio Access towards 6G Networks. *Electronics* **2022**, *11*, 1262. [\[CrossRef\]](#)
15. Furqan, M.; Iqbal, S.; Wasim, M.; Huang, Y. Load Balancing for Future 5G Network Communication: Performance and Trade-off. In Proceedings of the 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), Dubai, United Arab Emirates, 11–12 December 2019.
16. Furqan, M.; Ali, Z.; Jan, Q.; Nazir, S.; Iqbal, S.; Huang, Y. An Efficient Load-Balancing Scheme for UAVs in 5G Infrastructure. *IEEE Syst. J.* **2022**, 1–12. [\[CrossRef\]](#)
17. Huang, Y.; Feng, B.; Dong, P.; Tian, A.; Yu, S. A Multi-objective Based Inter-Layer Link Allocation Scheme for MEO/LEO Satellite Networks. In Proceedings of the 2022 IEEE Wireless Communications and Networking Conference (WCNC), Austin, TX, USA, 10–13 April 2022.
18. Dai, C.-Q.; Li, S.; Wu, J.; Chen, Q. Distributed User Association With Grouping in Satellite–Terrestrial Integrated Networks. *IEEE Internet Things J.* **2022**, *9*, 10244–10256. [\[CrossRef\]](#)
19. Wu, D.F.; Huang, C.H.; Yin, Y.B.; Huang, S.D.; Ashraf, W.A.; Guo, Q.Q.; Zhang, L. LB-DDQN for handover decision in satellite-terrestrial integrated networks. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 5871114. [\[CrossRef\]](#)
20. Shahid, S.M.; Seyoum, Y.T.; Won, S.H.; Kwon, S. Load Balancing for 5G Integrated Satellite–Terrestrial Networks. *IEEE Access* **2020**, *8*, 132144–132156. [\[CrossRef\]](#)
21. Huang, M.; Chen, J. Joint Load balancing and Spatial-temporal Prediction Optimization for Ultra-Dense Network. In Proceedings of the 2022 IEEE Wireless Communications and Networking Conference (WCNC), Austin, TX, USA, 10–13 April 2022.
22. Zhu, A.; Ma, M.; Guo, S.; Yang, Y. Adaptive Access Selection Algorithm for Multi-Service in 5G Heterogeneous Internet of Things. *IEEE Trans. Netw. Sci. Eng.* **2022**, *70*, 1630–1644. [\[CrossRef\]](#)
23. Fan, K.; Feng, B.; Zhang, X.; Zhang, Q. Network Selection Based on Evolutionary Game and Deep Reinforcement Learning in Space-Air-Ground Integrated Network. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 1802–1812. [\[CrossRef\]](#)
24. 3GPP TR 38.901; Study on Channel Model for Frequencies from 0.5 to 100 GHz. 3GPP: Sophia Antipolis, France, 2019.
25. 3GPP TR 38.821; Solutions for NR to Support Non-Terrestrial Networks (NTN). 3GPP: Sophia Antipolis, France, 2021.
26. Yu, S.; Wang, X.; Langar, R. Computation Offloading for Mobile Edge Computing: A Deep Learning Approach. In Proceedings of the 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, Canada, 8–13 October 2017.
27. 3GPP TR 38.811; Study on New Radio (NR) to Support Non-Terrestrial Networks. 3GPP: Sophia Antipolis, France, 2020.
28. 3GPP R1-1909515; Summary on Simulation Assumptions for NTN. 3GPP: Sophia Antipolis, France, 2019.
29. Gao, Y.H.; Yang, H.T.; Chen, L.; Yang, H.W.; Yin, L. Selection Algorithm of eMBB/URLLC Multiplexing Schemes Based on Fuzzy Logic. *J. Beijing Univ. Posts Telecommun.* **2021**, *44*, 15–20, 34.
30. 3GPP TR 38.321; Medium Access Control (MAC) Protocol Specification. 3GPP: Sophia Antipolis, France, 2022.
31. Lillicrap, T.; Hunt, J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous Control with Deep Reinforcement Learning. *arXiv* **2015**, arXiv:1509.02971.
32. Schulman, J.; Wolski, P.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal Policy Optimization Algorithms. *arXiv* **2017**, arXiv:1707.06347.