

## Article

# Intelligent Resource Allocation Using an Artificial Ecosystem Optimizer with Deep Learning on UAV Networks

Ahsan Rafiq <sup>1</sup>, Reem Alkanhel <sup>2,\*</sup>, Mohammed Saleh Ali Muthanna <sup>3</sup>, Evgeny Mokrov <sup>4</sup>, Ahmed Aziz <sup>5,6</sup> and Ammar Muthanna <sup>4</sup>

- <sup>1</sup> School of Automation, Chongqing University of Posts and Telecommunications, Chongqing 400065, China; l201710003@stu.cqupt.edu.cn
- <sup>2</sup> Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh 11671, Saudi Arabia
- <sup>3</sup> Institute of Computer Technologies and Information Security, Southern Federal University, 347922 Taganrog, Russia; muthanna@sfedu.ru
- <sup>4</sup> Department of Telecommunication Systems, Peoples' Friendship University of Russia (RUDN University), 6 Miklukho-Maklaya, 117198 Moscow, Russia; mokrov-ev@rudn.ru (E.M.); muthanna.asa@sut.ru (A.M.)
- <sup>5</sup> Department of Computer Science, Faculty of Computer and Artificial Intelligence, Benha University, Banha 13511, Egypt; ahmed.aziz@fci.bu.edu.eg
- <sup>6</sup> Department of International Business Management, Tashkent State University of Economics, Tashkent 100066, Uzbekistan
- \* Correspondence: rialkanhal@pnu.edu.sa

**Abstract:** An Unmanned Aerial Vehicle (UAV)-based cellular network over a millimeter wave (mmWave) frequency band addresses the necessities of flexible coverage and high data rate in the next-generation network. But, the use of a wide range of antennas and higher propagation loss in mmWave networks results in high power utilization and UAVs are limited by low-capacity onboard batteries. To cut down the energy cost of UAV-aided mmWave networks, Energy Harvesting (EH) is a promising solution. But, it is a challenge to sustain strong connectivity in UAV-based terrestrial cellular networks due to the random nature of renewable energy. With this motivation, this article introduces an intelligent resource allocation using an artificial ecosystem optimizer with a deep learning (IRA-AEODL) technique on UAV networks. The presented IRA-AEODL technique aims to effectually allot the resources in wireless UAV networks. In this case, the IRA-AEODL technique focuses on the maximization of system utility over all users, combined user association, energy scheduling, and trajectory design. To optimally allocate the UAV policies, the stacked sparse autoencoder (SSAE) model is used in the UAV networks. For the hyperparameter tuning process, the AEO algorithm is used for enhancing the performance of the SSAE model. The experimental results of the IRA-AEODL technique are examined under different aspects and the outcomes stated the improved performance of the IRA-AEODL approach over recent state of art approaches.

**Keywords:** UAV networks; resource allocation; deep learning; artificial ecosystem optimizer; wireless networks



**Citation:** Rafiq, A.; Alkanhel, R.; Muthanna, M.S.A.; Mokrov, E.; Aziz, A.; Muthanna, A. Intelligent Resource Allocation Using an Artificial Ecosystem Optimizer with Deep Learning on UAV Networks. *Drones* **2023**, *7*, 619. <https://doi.org/10.3390/drones7100619>

Academic Editors: Dawei Wang and Ruonan Zhang

Received: 29 August 2023

Revised: 26 September 2023

Accepted: 27 September 2023

Published: 3 October 2023



**Copyright:** © 2023 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

Unmanned aerial vehicle (UAV)-assisted communication presents a line-of-sight (LoS) wireless connection with controllable and flexible utilization [1]. In this regard, UAVs were mainly utilized to enrich the capacity and network coverage for ground users. As well, in wireless powered networks (WPN), UAVs are used as mobile charging stations to deliver radio frequency (RF)-energy supply to lower power user gadgets [2]. As a UAV generally utilizes limited-capacity batteries to carry out tasks, like flying, hovering, and offering services, it was vital to make the trade-offs between their coverage area and energy utilization along with service time [3]. Specifically, UAV-based aerial platforms that provide wireless services have allured the wide industry and research efforts concerning control,

deployment problems, and navigation. To enhance the coverage and energy efficiency for UAV-aided communication networks, resource allocation, namely subchannels, transmit power, and serving users, is essential [4].

Furthermore, consider a multiple-UAV-based wireless communication network (multi-UAV network) where a joint model to optimize trajectory and resource allocation was analyzed as a means to guarantee fairness by optimizing the minimal output throughput among users [5]. In this study, the author to strike tradeoffs between the sum rate and delay of sensing errands for multi-UAV based uplink single cell network devised a hybrid trajectory design and subchannel assignment method [6]. Human interference is constrained for the control design of UAVs because of the maneuverability and versatility of UAVs. Hence, to boost the outcome of UAV-enabled communication networks, machine learning (ML)-based intelligent control of UAVs is a priority [7]. Neural networks (NNs)-based trajectory design is taken into account where concerned from the viewpoint of UAVs' manufactured structures. Likewise, based on reinforcement learning (RL), a UAV routing design method was developed.

To build data distributions, the Gaussian mixture model was used where a weight expectation-related predictive on-demand deployment algorithm of UAV was devised for reducing the transmit power. As previously mentioned, ML is an auspicious power tool to offer potential and autonomous solutions smartly to boost the UAV-assisted communication network. But, several pieces of research focused on the trajectory and deployment models of UAVs in communication networks [8]. However, resource allocation methods like sub-channels and transmit power are taken into account as well as the previous research concentrated on time-independent scenarios. Furthermore, for time-dependent cases, the capacities of ML-based resource allocation techniques were inspected [9]. But, many ML techniques concentrated on multi-or-single UAV scenarios by assuming the accessibility of whole network data for all UAVs.

This article introduces an intelligent resource allocation using an artificial ecosystem optimizer with a deep learning (IRA-AEODL) technique on UAV networks. The presented IRA-AEODL technique aims to effectually allot the resources in the wireless UAV network. In such cases, the IRA-AEODL technique focuses on the maximization of system utility over all users, combined user association, energy scheduling, and trajectory design. To optimally allocate the UAV policies, the stacked sparse autoencoder (SSAE) model is used in the UAV networks. For the hyper-parameter tuning process, the AEO algorithm is used to enhance the performance of the SSAE model. The experimental results of the IRA-AEODL technique are examined under different aspects.

The highlights of this article include the use of unmanned aerial vehicles (UAVs) as a solution for flexible coverage and high data rates in next-generation networks, the challenge of energy consumption and limited battery capacity in UAVs, and the introduction of an intelligent resource allocation technique using an artificial ecosystem optimizer with deep learning (IRA-AEODL) on UAV networks. The research motivation behind this article is to find a solution to the energy cost issue in UAV-aided mmWave networks by utilizing energy harvesting and an intelligent resource allocation technique.

## 2. Related Works

In [10], the authors examine the Resource Allocation (RA) issue in UAV-assisted EH-powered D2D Cellular Networks (UAV-EH-DCNs). The main goal is to enhance power effectiveness and, at the same time, ensure the gratification of Ground Users (GUs). Also, the LSTM network is implemented to ease the rapidity of conjunction by taking out the prior data of GUs' gratification in regulating the present RA policy. Chang et al. [11] suggest an ML-founded policy RA protocol that encompasses RL and DL to devise the maximum strategy of the comprehensive UAV. Then, the authors also introduce a Multi-Agent (MA) DRL system for dispersed employment without being aware of a previous idea of the dynamic behavior of networks. Li et al. [12] suggest a novel DRL-founded Flight Resource Allocation Framework (FRA) to lessen the comprehensive information packet loss in a

sequential activity space. Also, a state classification layer, leveraging LSTM, is established in forecasting network dynamics, outcoming from time-varying airborne channels and power arrivals at the devices on the ground.

In [13], the authors concentrate on a downlink cellular network, where several UAVs play as aerial base stations for the users on the ground over Frequency Division Multiple Access (FDMA). Targeting maximizing both fairness and comprehensive throughput, the authors prototype RA and route design as a Decentralized Partially Observable Markov Decision Process (Dec-POMDP) and suggest MARL as a resolution. In [14], a MADRL-founded approach is introduced to accomplish the optimum long-term network utility while gratifying the customer's device value of service needs. However, considering that the efficacy of every UAV was determined founded on the atmosphere of the network and several other UAV activities, the JTDPDA issue is prototyped as the stochastic game.

In [15], the authors present their IRA-AEODL framework, which combines the Intra-Routing Algorithm (IRA) and Aerial Edge-mounted On-Demand Learning (AEODL). The IRA allows UAVs in the network to organize them for routing, while AEODL leverages machine learning to enhance dynamic route optimization. Afterward, the authors evaluate the performance of their proposed IRA-AEODL network, comparing it against existing UAV network solutions. They perform numerical simulations to evaluate the end-to-end delay, network throughput, and packet delivery ratio. They also analyze the mobile edge computing capabilities of their proposed network.

In [16], the authors examine the anti-jamming issue with integrated channel and energy distribution for UAV networks. Specifically, the authors concentrate on discarding both shared intrusion amongst exterior malevolent jamming and UAVs to optimize the scheme Quality of Experience (QoE) related to energy utilization. Then, the authors suggest a joint MA Layered Q Learning (MALQL) founded anti-jamming transmission protocol in minimizing the huge dimensionality of the activity space and examine the asymptotic convergence of the suggested protocol. In [16], the novelty of this research lies in its ability to address the total energy reduction issue in a non-convex way, while also incorporating several advanced protocols, such as a central MARL protocol and an MA Federated RL protocol, into an MEC scheme with multiple UAVs. By doing so, the authors propose a new and innovative approach that can potentially reduce energy consumption and improve the overall energy efficiency. The author [17] presents a stochastic geometry-based analysis of an integrated aerial-ground network, enabled by multi-UAVs. The novelty of this paper is that the exact distribution of the network throughput is derived and explored under various system parameters. However, the analysis is restricted to Rayleigh fading and a single interfering UAV.

Overall, the literature survey highlights a research gap in the area of resource allocation in UAV-assisted networks. While there have been previous studies focusing on using algorithms such as LSTM, RL, and DRL for efficient resource allocation, there is still a need for further investigation in this area. Furthermore, there is also a need for exploring the use of multi-agent reinforcement learning (MARL) in resource allocation as it has shown promising results in other areas of machine learning. There is also a gap in the evaluation of these proposed resource allocation techniques as most existing studies use simulation-based results rather than real-world implementation and testing. Therefore, further research in this field can contribute to the development of more efficient and adaptive resource allocation policies for UAV-assisted networks.

### 3. The Proposed Model

In this article, we proposed a novel IRA-AEODL technique for efficient resource allocation in UAV networks. A key advantage of the proposed IRA-AEODL technique compared to existing solutions is its ability to maximize system utility over a set of users by combining user association, energy scheduling, and trajectory design. Figure 1 visually demonstrates the overall architecture of the IRA-AEODL approach. Furthermore, a 3D Cartesian coordinate system is used to ensure optimal coverage for each user. The user set

and UAV swarm are represented as  $U$  and  $M$ , respectively, with  $|M| = M$  and  $|U| = U$ . The trajectory of each UAV is modeled through time slots,  $t$ , with  $t \in \{1, 2, T\}$ . Additionally, the constellation of UAVs is assumed to fly at a fixed height  $H$ . Finally, the base station or satellite is responsible for the learning procedure required to ensure optimization within the IRA-AEODL approach. The main motivators for using this technique are (i) the availability and ease of access to unlabeled data; (ii) the potential for notable enhancements in the model's performance by including a significant amount of unlabeled data in training; and (iii) the practical constraints of human resources in terms of labeling data. To assess the efficacy of this approach, we conducted a practical analysis on a genuine dataset which showcased the considerable boost in the overall classification accuracy of the SSAE model through the inclusion of a substantial quantity of unlabeled data in the pre-training stage.

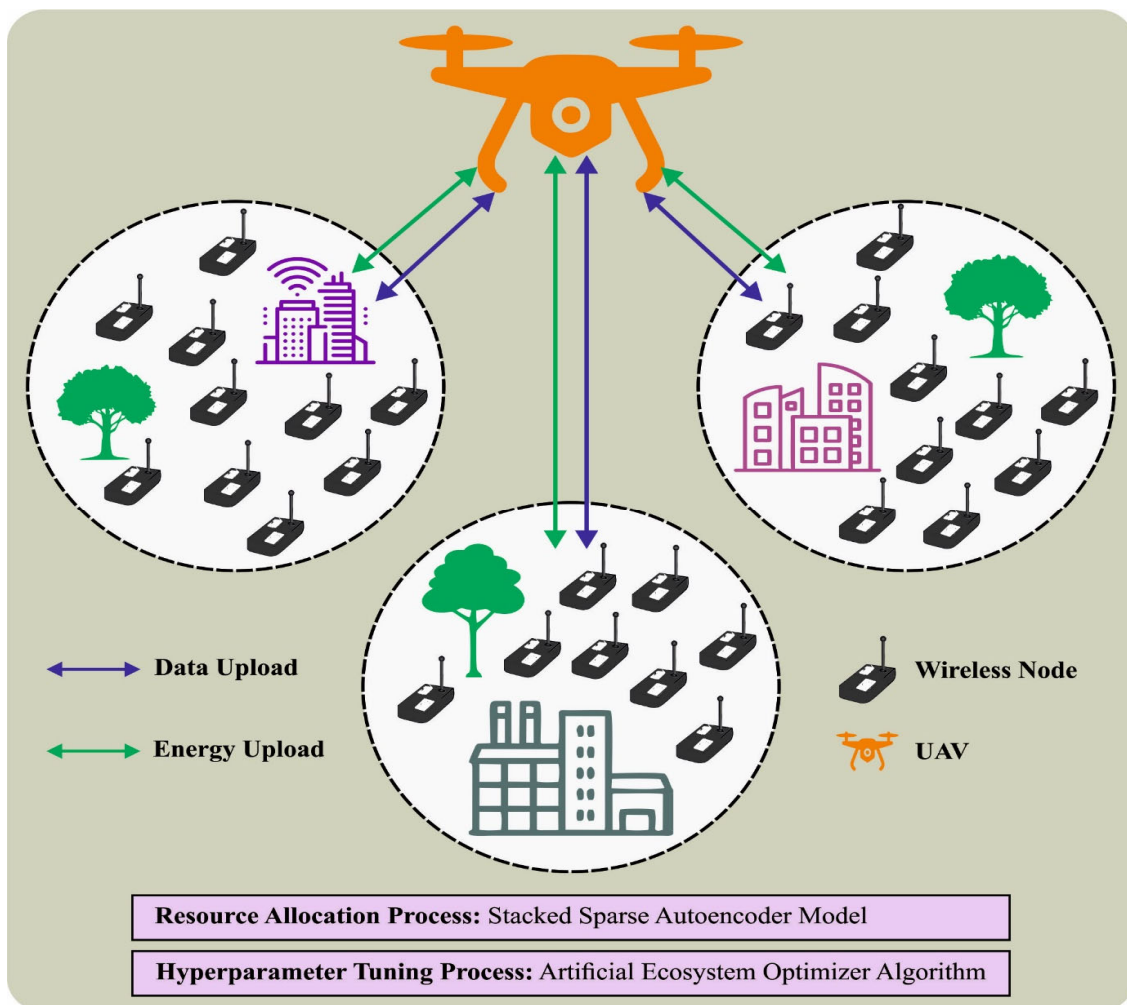


Figure 1. Overall procedure of the IRA-AEODL system.

### 3.1. System Model

Assume  $M > 1$  UAVs share a similar frequency spectrum and a group of  $U > 1GUs$ . The GU set and UAV swarm are represented as  $\mathcal{U}$  and  $M$ , respectively [18]. We have  $|M| = M$  and  $|\mathcal{U}| = U$ . Each UAV provides service to the user in successive time slots. We represent the time slot as  $t$ ;  $t \in \{1, 2, T\}$ . The total period was represented as  $\mathcal{T}$ . In the presented model, take a 3D Cartesian coordinate system but the predetermined position of every GU  $u$  represented by vertical and horizontal coordinates, for example,  $\varphi_u = [x_u, y_u]^T \in \mathbb{R}^{2 \times 1}$ ,  $u \in \mathcal{U}$ . Each UAV is considered to fly at a fixed distance  $d_h = H$  above ground and the coordinate of UAVs  $m$  at  $t$  time was represented as

$\psi_m(t) = [x_m(t), y_m(t)]^T \in \mathbb{R}^{2 \times 1}$ . Assume a base controller is performing the learning procedure that could be BS or satellite. Furthermore, the UAV is capable of communicating within the swam.

Assume each UAV will fly back to the base hence the trajectory needs to fulfil the subsequent constraints

$$\psi_m(1) = \psi_m(T). \quad (1)$$

Moreover, the trajectory of the UAV is also subjected to specific constraints of distance and speed, which are the following:

$$\|\psi_m(t+1) - \psi_m(t)\| \leq V_{\max}, \quad (2)$$

$$\|\psi_m(t) - \psi_j(t)\| \geq S_{\min}, \quad (3)$$

where  $V_{\max}$  denotes the maximal speed of UAV and  $S_{\min}$  represents the minimal inter-UAV distance to prevent specific collision or interference. Consequently, the distance between UAV  $m$  and user  $u$  in the  $t$  time slot is shown below:

$$d_{m,u}(t) = \sqrt{H^2 + \|\psi_m(t) - \psi_u\|^2}. \quad (4)$$

### 3.1.1. Path Loss Model

The UAV is capable of establishing an *LoS* link with *GU*. Since the changes in real-time environments (urban, rural, suburban, and so on) are generally unpredictable, the randomness related to *LoS* and Non-*LoS* (NLoS) in a specific time must be considered while developing the UAV. Consequently, consider the *GU* connection with UAV through the *LoS* connection with specific probability which we represent as *LoS* probability. The *LoS* probability depends on the environment and the location of *GU* and UAV.

$$\rho_{m,u}^{los}(t) = \frac{1}{1 + \zeta_1 \exp[-\zeta_2(\theta_{m,u}(t) - \zeta_1)]}, \quad (5)$$

In Equation (5),  $\zeta_1$  and  $\zeta_2$  denote the constant; the value depends on the environment and carrier frequency.  $\theta_{m,u}(t)$  indicates the elevation angle as follows:

$$\theta_{m,u}(t) = \frac{180}{\pi \operatorname{arctan}\left(\frac{H}{d_{m,u}(t)}\right)}. \quad (6)$$

The *LoS* and NLoS path loss methods between the user  $u$  and UAV  $m$  are shown below:

$$\hat{L}_{m,u}(t) = \begin{cases} \eta_1 \left(\frac{4\pi f_c d_{m,u}(t)}{c}\right)^\alpha, & \text{LoS link,} \\ \eta_2 \left(\frac{4\pi f_c d_{m,u}(t)}{c}\right)^\alpha, & \text{NLoS link,} \end{cases} \quad (7)$$

In Equation (7),  $\eta_1$  and  $\eta_2$  denote the excess coefficient in *LoS* and NLoS links, respectively.  $f_c$  indicates the carrier frequency,  $c$  represents the light speed, and  $\alpha$  shows the path loss exponent. Assuming the UAV and *GU* locations, it is challenging to define whether it is *LoS* or NLoS path loss method that must be utilized in the UAVs technique.

$$L_{m,u}(t) = \rho_{m,u}^{los}(t) \eta_1 \left(\frac{4\pi f_c d_{m,u}(t)}{c}\right)^\alpha + (1 - \rho_{m,u}^{los}(t)) \eta_2 \left(\frac{4\pi f_c d_{m,u}(t)}{c}\right)^\alpha. \quad (8)$$

### 3.1.2. Transmission Model

A binary parameter  $\beta_{m,u}(t)$  is determined as the user association indicator to express the user relationship between GU and UAV, which is

$$\beta_{m,u}(t) = \begin{cases} 1, & \text{if GU } u \text{ associates with UAV } m, \\ 0, & \text{otherwise.} \end{cases} \quad (9)$$

Consider one GU passing through one UAV in a provided time slot, viz.,  $\sum_{m=1}^M \beta_{m,u}(t) \leq 1$ . Furthermore, the transmit power of UAV  $m$  for  $u$  was represented by  $p_{m,u}(t)$  and the channel gain between UAV  $m$  and user  $u$  is represented by  $h_{m,u}(t)$ .

$$R_u(t) = \sum_{m=1}^M \beta_{m,u}(t) \log_2(1 + \gamma_{m,u}(t)), \quad (10)$$

As a result, different UAVs could cause interference with GU  $u$ ,  $\gamma_{m,u}(t)$ , modeled as SINR of the relationship between  $m$  and  $u$ , as follows:

$$\gamma_{m,u}(t) = \frac{p_{m,u}(t)h_{m,u}(t)L_{m,u}^{-1}(t)}{\sum_{j=1}^M p_{j,u}(t)h_{j,u}(t)L_{j,u}^{-1}(t) + \sigma^2} \quad (11)$$

In Equation (11),  $\sigma^2$  denotes the noise variance. It should be such that the transmit power, channel state, and trajectory of the UAV are continuous. Next, after quantizing and partitioning the value into distinct levels within the range, in every  $t$  time slot, the value of this variable is understood as a discrete counterpart.

### 3.2. SSAE-Based Resource Allocation Scheme

To optimally allocate the UAV policies, the SSAE model is used in the UAV networks. The building block of SSAE in the AE is an archetypal NN that learns to map the input  $X$  to output  $Y$  [18]. The entire AE is split into decoder and encoder parts: the encoded part ( $W_X, B_X$ ), which maps the input  $X$  to the code  $I_c$ , and the decoder part ( $W_Y, B_Y$ ), which maps the code to the reconstruction data  $Y$ . The architecture of SSAE was demonstrated in Figure 2, the decoding part is with weighted  $W_Y$  and bias  $B_Y$  and the encoding part is with  $W_X$  weight and  $B_X$  bias. Thus

$$I_c = g_{LS}(W_X X + B_X), \quad (12)$$

$$Y = g_{LS}(W_Y I_c + B_Y), \quad (13)$$

where the output  $Y$  represents the estimate of input  $X$  and  $g_{LS}$  indicates the *log* sigmoid function:

$$g_{LS}(z) = \frac{1}{1 + \exp(-z)}. \quad (14)$$

The SAE is different from the AE model. The sparsity could assist AE to attain the best performance. To minimize the error between the output  $Y$  and the input vector  $X$ , the raw loss function of AE is assumed as follows:

$$J_{raw}(W_x, W_y, B_x, B_y) = \frac{1}{N_S} \|Y - X\|^2, \quad (15)$$

In Equation (15),  $N_S$  denotes the number of training instances. From Equations (12) and (13), the output  $Y$  is formulated as follows

$$Y = g_{AE}(X|W_X, W_Y, B_X, B_Y), \quad (16)$$

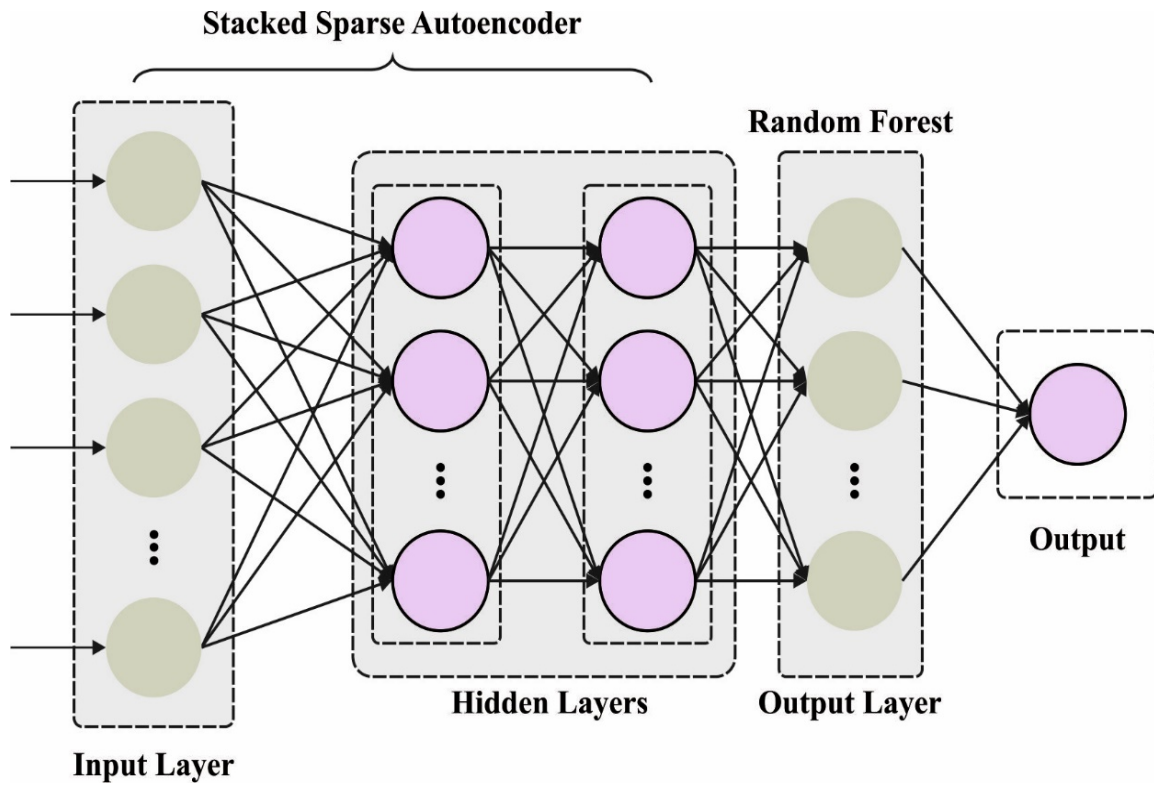


Figure 2. Structure of SSAE.

In Equation (16),  $g_{AE}$  denotes the abstract of the AE function. Thus, Equation (15) is formulated as follows:

$$J_{raw}(W_X, W_Y, B_X, B_Y) = \frac{1}{N_s} \|g_{AE}(X|W_X, W_Y, B_X, B_Y) - X\|^2 \tag{17}$$

To learn a trivial mapping or prevent over-complete mapping, we determine one regularized term  $\Gamma_s$  of the sparsity constraint and one  $L_2$  regularization term  $\Gamma_w$  of the weight ( $W_X, W_Y$ ) and it is expressed below:

$$J(W_X, W_Y, B_X, B_Y) = \frac{1}{N_s} \|g_{AE}(X|W_X, W_Y, B_X, B_Y) - X\|^2 + a_s \times \Gamma_s + a_w \times \Gamma_w \tag{18}$$

In Equation (18),  $a_s$  and  $a_w$  refer to the sparsity and weight regulation factors. The sparsity regularization term can be represented as follows:

$$\Gamma_s = \sum_{j=1}^{|I|} g_{KL}(\rho_j \hat{\rho}_j) = \sum_{j=1}^{|I|} \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho) \log \frac{1-\rho}{1-\hat{\rho}_j} \tag{19}$$

In Equation (19),  $\hat{\rho}_j$  denotes the  $j$ -th neuron's average activation value over each  $N_s$  trained sample,  $|I|$  denotes the number of components of internal code output  $I_C$ ,  $\rho$  indicates the desirable value, termed the sparsity proportion factor, and  $g_{KL}$  represents the Kullback–Leibler divergence function. The weight regularization term can be represented as follows:

$$\Gamma_w = \frac{1}{2} \times \|W_X W_Y\|_2^2 \tag{20}$$

SAE is utilized as a key component and the last SSAE classifiers by subsequent three processes are constructed by the following actions: (i) append the softmax layer at the end of the AI method; (ii) involve preprocessing, input, vectorization, and 2D-FrFE layers; and (iii) stack the available SAE. In the classifier stage, four SAE blocks with many neurons of

( $N_1, N_2, N_3, N_4$ ) are applied. As a result of the trial-and-error method, we apply four SAE blocks. Lastly, the softmax layer with the neuron of  $N_c$  is appended, where  $N_c$  denotes the number of fruit classes.

### 3.3. Hyperparameter Tuning using the AEO Algorithm

For the hyperparameter tuning process, the AEO algorithm is used for enhancing the performance of the SSAE model. The AEO is an innovative nature-inspired metaheuristic algorithm that hinges on the energy transmission model among living creatures that assist to maintain species stability [19]. The three operators that are utilized to obtain solutions are decomposition, production, and consumption. The energy flow in an ecosystem consists of decomposers, producers, and consumers.

#### 3.3.1. Production

In AEO, the producer represents the worse individual in the population. Thus, it needs to be upgraded concerning the optimal individual by the lower and upper boundaries such that it helps others to find other areas. Through the production operator, a new individual is produced, among randomly generated ( $x_{rand}$ ) and the best ( $x$ ) individuals by substituting the prior one. The mathematical representation of the production operator is shown below:

$$x_1(t+1) = (1 - \alpha)x_n(t) + \alpha x_{rand}(t) \quad (21)$$

$$\alpha = \left(1 - \frac{t}{T}\right)r_1 \quad (22)$$

$$x_{rand} = \bar{r}(Ub - Lb) + Lb \quad (23)$$

Here,  $n$  represents the population size,  $T$  signifies the iteration number,  $Ub$  and  $Lb$  denote upper and lower boundaries, and  $r_1$  signifies a random integer that lies between  $[0,1]$ .  $\bar{r}$  and  $\alpha$  denote a random vector within  $[0,1]$  and a linear weight coefficient. The  $\alpha$  coefficient provided in Equation (21) assists to drift the individual linearly from the random location to the optimal individual through iteration.

#### 3.3.2. Consumption

The consumers perform this operation and then the production operator finishes the production. Each consumer may eat an arbitrarily selective consumer taking low energy or a producer for obtaining energy. A Lévy flight is a random walk termed as a consumption factor ( $C$ ) and was determined as follows for enhancing the exploration ability:

$$C = \frac{1}{2} \frac{v_1}{|v_2|} \quad (24)$$

$$v_1 \sim N(0, 1), v_2 \sim N(0, 1) \quad (25)$$

$N(0, 1)$  represents the normal distribution for the mean and SD equivalent to zero and one, respectively. Distinct approaches can be implemented with various kinds of users. A consumer eats only the producer in case of being arbitrarily selective as a herbivore ( $x_2$  and  $x_5$  are herbivore consumers, therefore, consume only producer  $x_1$ ). This strategy was depicted in Equation (26).

$$x_i(t+1) = x_i(t) + C \cdot (x_i(t) - x_1(t)), i \in [2, \dots, n] \quad (26)$$

A consumer only eats another consumer with a high energy level once it can be selective as a carnivore arbitrarily (a consumer in individuals of  $x_2$ – $x_5$  are consumed by consumer  $x_6$  as the last is a carnivore and takes a lower energy level than individuals of  $x_2$ – $x_6$ ). A carnivore performance was demonstrated as follows:

$$x_i(t+1) = x_i(t) + C \cdot (x_i(t) - x_j(t)), +i \in [3, \dots, n] \quad (27)$$



$$j = randi([2i - 1]) \tag{28}$$

Uniquely from the last two performances, a consumer with a high level of energy or producer is arbitrarily eaten by the user when it can be selected as an omnivore arbitrarily (either the producer  $x_1$  or arbitrarily selected users in  $x_2-x_6$  is eaten by  $x_7$  since it can be an omnivore and is the low energy level of  $x_2-x_6$ ).

$$x_i(t + 1) = x_i(t) + C \cdot (r_2 \cdot (x_i(t) - x_1(t))) + (1 - r_2)(x_i(t) - x_j(t)), i \in [3, \dots, n] \tag{29}$$

$$j = randi([2i - 1]) \tag{30}$$

whereas,  $r_2$  implies the random number from the range of zero to one. A searching individual's place was upgraded in terms of both arbitrarily selective and worse individuals from the population utilizing the consumption operator. Hence, it permits the technique for executing a global search.

### 3.3.3. Decomposition

SDecomposition is a vital procedure for taking a suitably working ecosystem. The decomposer breaks down every dead individual continuously from the population for providing needed nutrients for the producer's development. The decomposition feature of  $D$  together with weighted coefficients of  $h$  and  $e$  can be intended for the mathematical model. Individuals' parameters support upgrading the location of  $x_i$  ( $i$ th individual) by the location of  $x_n$  (the decomposer position). Besides, every individual's next position has been permitted for spreading nearby the decomposer (optimum individual). The mathematical formula is provided as follows:

$$x_i(t + 1) = x_n(t) + D \cdot (e \cdot x_n(t) - h \cdot x_i(t)), i \in 1, \dots, n \tag{31}$$

$$D = 3u, u \sim N(0, 1) \tag{32}$$

$$e = r_3 \cdot randi([1 \ 2]) - 1 \tag{33}$$

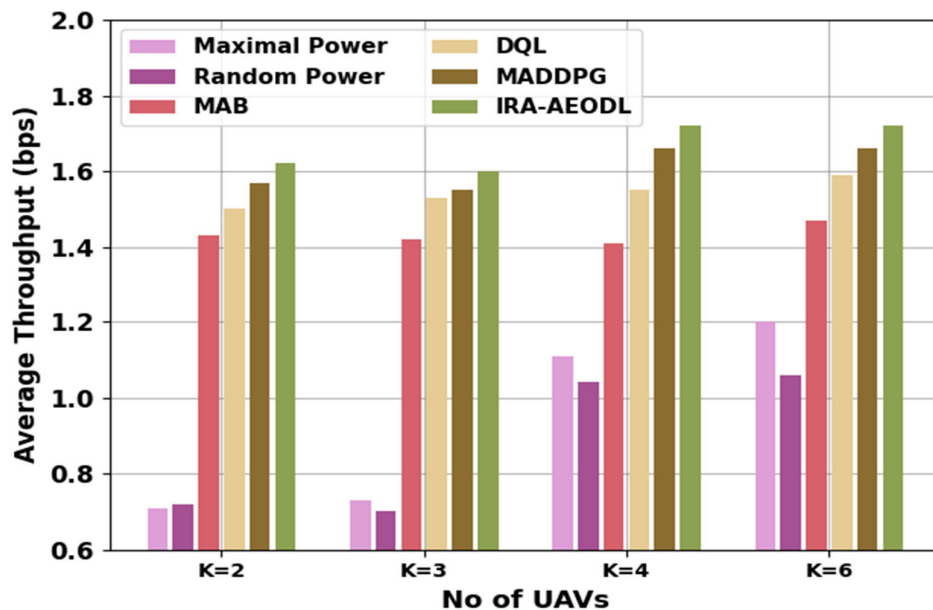
$$h = 2 \cdot r_3 - 1. \tag{34}$$

## 4. Results and Discussion

In this section, the experimental validation of the IRA-AEODL technique is examined under various aspects. Table 1 and Figure 3 report a comparative average throughput (ATHRO) study of the IRA-AEODL technique with recent models [20]. The outcomes indicate the increasing ATHRO values of the IRA-AEODL technique under all K values. For  $K = 2$ , the IRA-AEODL technique obtains a higher ATHO value of 1.62 bps while the MP, RP, MAB, DQL, and MADDPG [21] models accomplish reduced ATHO values of 0.71 bps, 0.72 bps, 1.43 bps, 1.50 bps, and 1.57 bps, respectively. Similarly, with  $K = 6$ , the IRA-AEODL technique reaches improving ATHO of 1.72 bps while the MP, RP, MAB, DQL, and MADDPG models result in reduced ATHO values of 1.20 bps, 1.06 bps, 1.47 bps, 1.59 bps, and 1.66 bps, respectively. The proposed DNN was trained on an offline dataset of simulated UAV-aided mmWave. The parameters of the proposed algorithm were optimized to obtain the best learning performance. The training was conducted for 1000 epochs using Keras and Tensorflow on a Nvidia GTX 1060 GPU. The accuracy comparison of the proposed DNN was conducted against existing state-of-the-art algorithms. The results showed that the proposed IRA-AEODL technique achieved an average improvement in 11.5% over existing algorithms. This accuracy improvement was attributed to the stacked sparse autoencoder's ability to efficiently perform resource allocation and the AEO algorithm's ability to optimize the model.

**Table 1.** ATHRO analysis of the IRA-AEODL approach with other systems under varying UAVs.

No of UAVs	Average Throughput (bps)					
	Maximal Power	Random Power	MAB	DQL	MADDPG	IRA-AEODL
K = 2	0.71	0.72	1.43	1.50	1.57	1.62
K = 3	0.73	0.70	1.42	1.53	1.55	1.60
K = 4	1.11	1.04	1.41	1.55	1.66	1.72
K = 6	1.20	1.06	1.47	1.59	1.66	1.72



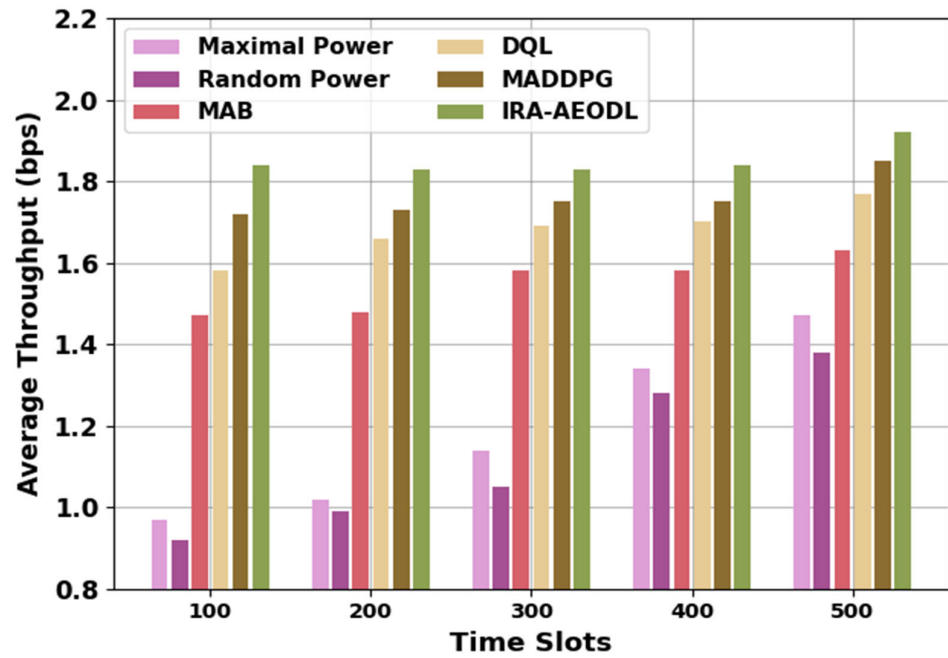
**Figure 3.** ATHRO analysis of the IRA-AEODL approach under varying UAVs.

The proposed model is a Deep Neural Network (DNN) model that has been trained on a dataset of images of different fruits. The DNN architecture uses convolutional layers to extract features from the images, followed by a densely connected set of layers to identify the classes of fruits. The training process will involve feeding the DNN model with labeled images of each of the desired fruit classes. The model will learn the features associated with each class and develop a set of weights that will allow it to recognize which fruits belong to which class. After training has been completed, the model can then be used to classify new images of fruits into their respective classes. Additionally, to improve accuracy, the model can also be fine-tuned using data augmentation techniques, such as randomly adjusting the size and orientation of the images as well as adjusting the brightness and contrast. This can help the model to better recognize the features in different images. Once training and fine-tuning is complete, the DNN can then be tested with a set of validation images to ensure that it is able to accurately classify the different types of fruits. Once satisfactory accuracy has been achieved, the model can then be deployed for use in applications.

Table 2 and Figure 4 demonstrate a comparative ATHRO study of the IRA-AEODL method with recent methods. The results represent the increasing ATHRO values of the IRA-AEODL technique under varying time slots. For 100 time slots, the IRA-AEODL method attains a maximum ATHO value of 1.84 bps whereas the MP, RP, MAB, DQL, and MADDPG methods attain decreased ATHO values of 0.97 bps, 0.92 bps, 1.47 bps, 1.58 bps, and 1.72 bps, respectively. Similarly, with 300-time slots, the IRA-AEODL method attains an increasing ATHO of 1.83 bps while the MP, RP, MAB, DQL, and MADDPG methods resulted in decreased ATHO values of 1.14 bps, 1.05 bps, 1.58 bps, 1.69 bps, and 1.75 bps, respectively.

**Table 2.** ATHRO analysis of the IRA-AEODL approach with other systems under varying time slots.

Average Throughput (bps)						
Time Slots	Maximal Power	Random Power	MAB	DQL	MADDPG	IRA-AEODL
100	0.97	0.92	1.47	1.58	1.72	1.84
200	1.02	0.99	1.48	1.66	1.73	1.83
300	1.14	1.05	1.58	1.69	1.75	1.83
400	1.34	1.28	1.58	1.70	1.75	1.84
500	1.47	1.38	1.63	1.77	1.85	1.92



**Figure 4.** ATHRO analysis of the IRA-AEODL approach under varying time slots.

Table 3 and Figure 5 illustrate a comparative ATHRO study of the IRA-AEODL method with recent models. The results indicate the increasing ATHRO values of the IRA-AEODL technique under varying users. For 100 users, the IRA-AEODL technique obtains a higher ATHO value of 1.84 bps while the MP, RP, MAB, DQL, and MADDPG methods accomplish reduced ATHO values of 1.04 bps, 0.84 bps, 1.49 bps, 1.52 bps, and 1.70 bps, respectively. Similarly, with 300 users, the IRA-AEODL technique reaches an improving ATHO of 2.28 bps while the MP, RP, MAB, DQL, and MADDPG models resulted in reduced ATHO values of 1.43 bps, 1.36 bps, 1.79 bps, 1.91 bps, and 2.06 bps, respectively.

**Table 3.** ATHRO analysis of the IRA-AEODL approach with other systems under varying users.

Average Throughput (bps)						
No. of Users	Maximal Power	Random Power	MAB	DQL	MADDPG	IRA-AEODL
100	1.04	0.84	1.49	1.52	1.70	1.84
200	1.23	1.20	1.63	1.78	1.89	2.08
300	1.43	1.36	1.79	1.91	2.06	2.28
400	1.65	1.55	1.95	2.13	2.33	2.55
500	1.70	1.73	2.10	2.24	2.57	2.61

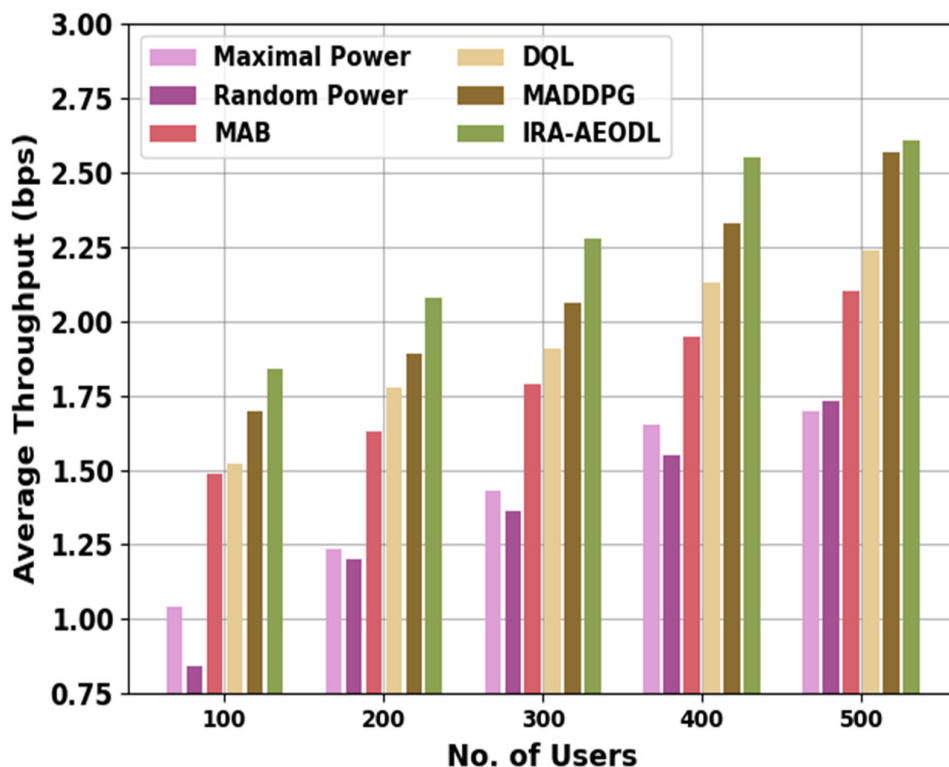


Figure 5. ATHRO analysis of the IRA-AEODL approach under varying users.

Table 4 and Figure 6 depict a comparative ATHRO study of the IRA-AEODL technique with recent models. The outcomes indicate the increasing ATHRO values of the IRA-AEODL technique under varying energy arrival  $E_{max}$ . For 80 energy arrival  $E_{max}$ , the IRA-AEODL technique attains a higher ATHRO value of 1.73 bps while the MAB, DQL, and MADDPG methods obtain minimum ATHRO values of 1.55 bps, 1.66 bps, and 1.71 bps respectively. Similarly, with the 160 energy arrival  $E_{max}$ , the IRA-AEODL technique reaches an improving ATHRO of 1.85 bps while the MAB, DQL, and MADDPG models resulted in reduced ATHRO values of 1.75 bps, 1.80 bps, and 1.83 bps, respectively.

Table 5 and Figure 7 demonstrate a comparative ATHRO study of the IRA-AEODL technique with recent methods. The results indicate the increasing ATHRO values of the IRA-AEODL technique under varying battery capacity (BC). For 3000 BC, the IRA-AEODL technique obtains a higher ATHRO value of 1.74 bps while the MAB, DQL, and MADDPG methods accomplish reduced ATHRO values of 1.55 bps, 1.63 bps, and 1.70 bps, respectively. Similarly, with 5000 BC, the IRA-AEODL technique reaches an improving ATHRO of 1.79 bps while the MAB, DQL, and MADDPG models resulted in reduced ATHRO values of 1.60 bps, 1.67 bps, and 1.79 bps, respectively.

Table 4. ATHRO analysis of the IRA-AEODL approach with other systems under varying energy arrival values  $E_{max}$ .

Energy Arrival $E_{max}$	Average Throughput (bps)			
	MAB	DQL	MADDPG	IRA-AEODL
80	1.55	1.66	1.71	1.73
100	1.64	1.71	1.79	1.81
120	1.65	1.74	1.80	1.83
140	1.69	1.76	1.82	1.85
160	1.75	1.80	1.83	1.85

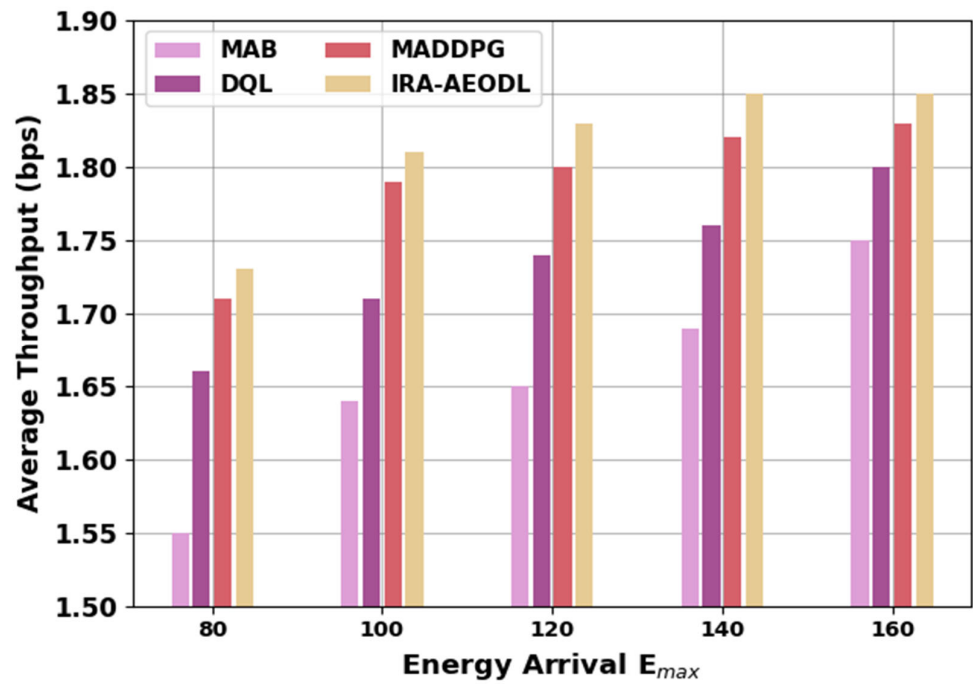


Figure 6. ATHRO analysis of the IRA-AEODL approach under varying energy arrival values  $E_{max}$ .

Table 5. ATHRO analysis of the IRA-AEODL approach with other systems under varying battery capacity.

Average Throughput (bps)				
Battery Capacity (C)	MAB	DQL	MADDPG	IRA-AEODL
3000	1.55	1.63	1.70	1.74
3500	1.56	1.65	1.77	1.78
4000	1.58	1.66	1.77	1.78
4500	1.59	1.67	1.78	1.79
5000	1.60	1.67	1.79	1.79

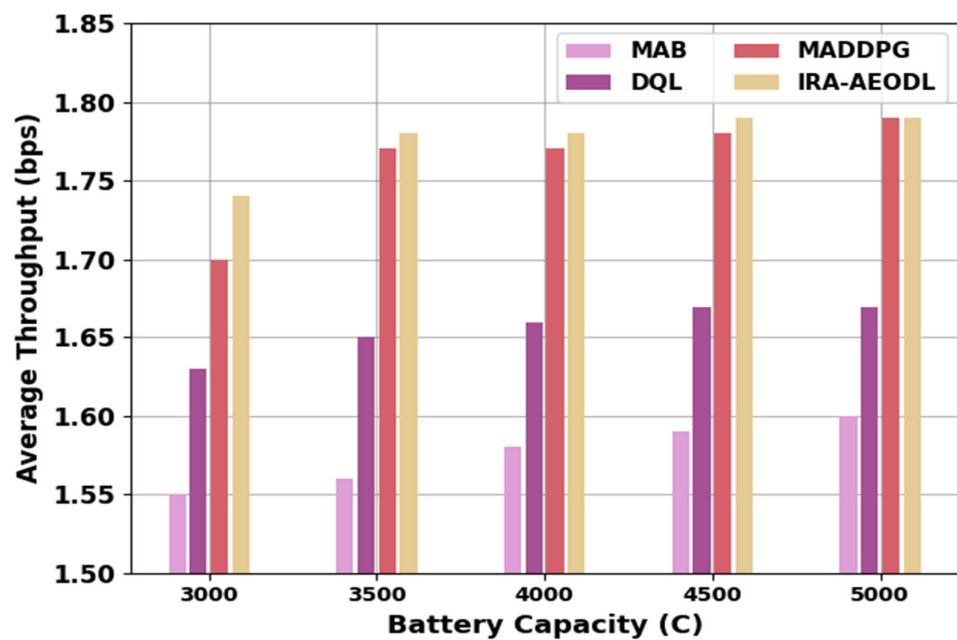
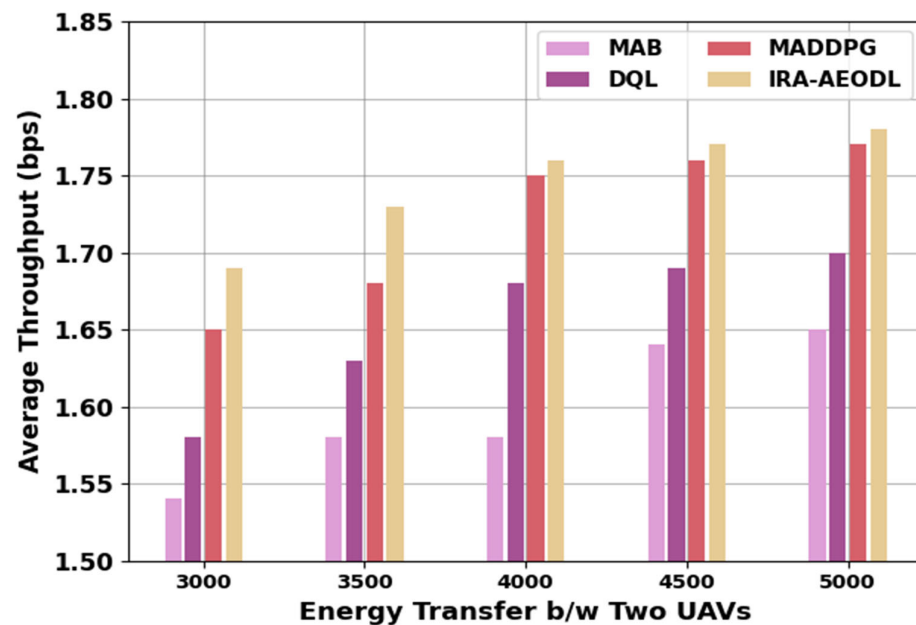


Figure 7. ATHRO analysis of the IRA-AEODL approach under varying battery capacity.

Table 6 and Figure 8 depict a comparative ATHRO study of the IRA-AEODL technique with recent models. The results indicate the increasing ATHRO values of the IRA-AEODL technique under varying Energy Transfer b/w Two UAVs (ETTUAV). For 3000 ETTUAV, the IRA-AEODL technique attains a higher ATHO value of 1.69 bps while the MAB, DQL, and MADDPG methods accomplish reduced ATHO values of 1.54 bps, 1.58 bps, and 1.65 bps, respectively. Similarly, with 5000 ETTUAV, the IRA-AEODL technique reaches an improving ATHO of 1.78 bps while the MAB, DQL, and MADDPG models resulted in reduced ATHO values of 1.65 bps, 1.70 bps, and 1.77 bps, respectively.

**Table 6.** ATHRO analysis of the IRA-AEODL approach with other systems under varying Energy Transfer b/w Two UAVs.

Average Throughput (bps)				
Energy Transfer b/w Two UAVs	MAB	DQL	MADDPG	IRA-AEODL
3000	1.54	1.58	1.65	1.69
3500	1.58	1.63	1.68	1.73
4000	1.58	1.68	1.75	1.76
4500	1.64	1.69	1.76	1.77
5000	1.65	1.70	1.77	1.78

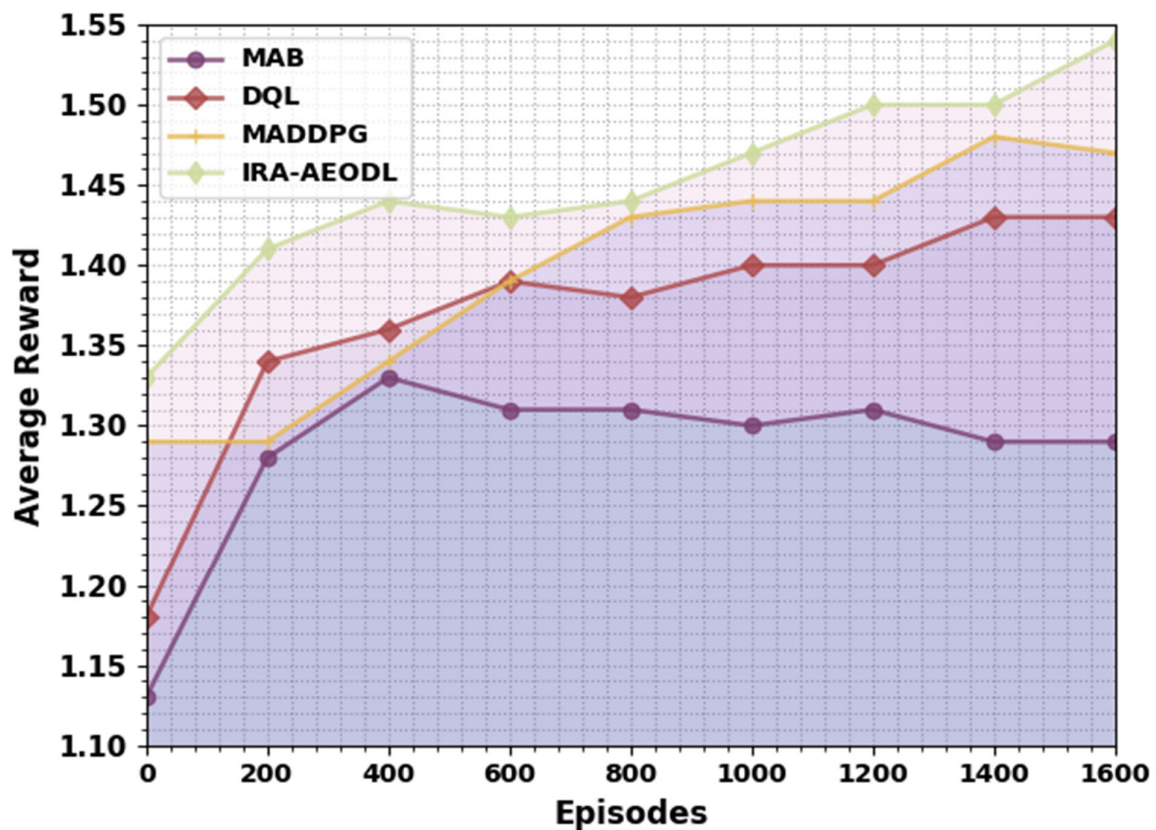


**Figure 8.** ATHRO analysis of the IRA-AEODL approach with other systems under varying ETTUAV.

Finally, the average reward examination of the IRA-AEODL technique with different models takes place in Table 7 and Figure 9. The results demonstrate that the IRA-AEODL technique gains increasing reward values over other models. For instance, with 200 episodes, the IRA-AEODL technique attains an increasing average reward of 1.41 while the MAB, DQL, and MADDPG techniques obtain reducing average rewards of 1.28, 1.34, and 1.29, respectively.

**Table 7.** Average reward analysis of the IRA-AEODL approach with other systems under varying episodes.

Episodes	Average Reward			
	MAB	DQL	MADDPG	IRA-AEODL
0	1.13	1.18	1.29	1.33
200	1.28	1.34	1.29	1.41
400	1.33	1.36	1.34	1.44
600	1.31	1.39	1.39	1.43
800	1.31	1.38	1.43	1.44
1000	1.30	1.40	1.44	1.47
1200	1.31	1.40	1.44	1.50
1400	1.29	1.43	1.48	1.50
1600	1.29	1.43	1.47	1.54

**Figure 9.** Average reward analysis of the IRA-AEODL approach under varying episodes.

Meanwhile, with 800 episodes, the IRA-AEODL technique attains an increasing average reward of 1.44 while the MAB, DQL, and MADDPG methods attain reducing average rewards of 1.31, 1.38, and 1.43, respectively. Eventually, with 1600 episodes, the IRA-AEODL technique attains an increasing average reward of 1.54 while the MAB, DQL, and MADDPG techniques obtain reducing average rewards of 1.29, 1.43, and 1.47, respectively. These results exhibited the superior performance of the IRA-AEODL technique over other existing models on the UAV networks.

## 5. Conclusions

In this article, we introduced a new IRA-AEODL technique for the optimal allocation of resources in UAV networks. The presented IRA-AEODL technique is intended for the effectual allocation of resources in wireless UAV networks. Here, the IRA-AEODL

technique focused on the maximization of system utility over all users, combined trajectory design, user association, and energy scheduling. To optimally allocate the UAV policies, the SSAE model is used in the UAV networks. For the hyperparameter tuning process, the AEO algorithm is used to enhance the performance of the SSAE model. The experimental results of the IRA-AEODL technique are examined under different aspects and the outcomes stated the better performance of the IRA-AEODL approach over recent state of art approaches. In the future, the ensemble learning process can be included to improve the resource allocation performance of the IRA-AEODL technique. In comparison to other learning methods, the proposed algorithm has several advantages such as fast convergence, improved local optimization ability, and well-balanced global or local search ability. With the help of production, consumption, and decomposition operators, the proposed model is able to quickly explore the search space and find the optimal solution. Therefore, the proposed algorithm is vital for hyperparameter tuning as it ensures optimal results, sustainability, and robustness compared to other learning models.

**Author Contributions:** Conceptualization, R.A. and E.M.; methodology, M.S.A.M.; software, R.A.; validation, A.M. and E.M.; formal analysis, A.A.; investigation, M.S.A.M.; resources, A.R. and E.M.; data curation, A.A.; writing—original draft preparation, A.M. and M.S.A.M.; writing—review and editing, R.A. and E.M.; visualization, A.M.; supervision, A.M.; project administration, R.A.; funding acquisition, E.M. and R.A. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia, in part this paper is supported by postdoc fellowship granted by the Institute of Computer Technologies and Information Security, Southern Federal University, project № P.D./22-01-KT, and in part this paper has been supported by the RUDN University Strategic Academic Leadership Program (recipient Ammar Muthanna).

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data are contained within the article and/or available from the corresponding author upon reasonable request.

**Acknowledgments:** The authors express their gratitude to Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2023R323), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia, and in part would like to acknowledge the RUDN University Strategic Academic Leadership Program (recipient Ammar Muthanna).

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

## References

1. Seid, A.M.; Boateng, G.O.; Anokye, S.; Kwantwi, T.; Sun, G.; Liu, G. Collaborative computation offloading and resource allocation in multi-UAV-assisted IoT networks: A deep reinforcement learning approach. *IEEE Internet Things J.* **2021**, *8*, 12203–12218. [[CrossRef](#)]
2. Do, Q.V.; Pham, Q.-V.; Hwang, W.-J. Deep reinforcement learning for energy-efficient federated learning in UAV-enabled wireless powered networks. *IEEE Commun. Lett.* **2021**, *26*, 99–103. [[CrossRef](#)]
3. Dai, Z.; Zhang, Y.; Zhang, W.; Luo, X.; He, Z. A Multi-Agent Collaborative Environment Learning Method for UAV Deployment and Resource Allocation. *IEEE Trans. Signal Inf. Process. Netw.* **2022**, *8*, 120–130. [[CrossRef](#)]
4. Peng, H.; Shen, X. Multi-agent reinforcement learning based resource management in MEC- and UAV-assisted vehicular networks. *IEEE J. Sel. Areas Commun.* **2020**, *39*, 131–141. [[CrossRef](#)]
5. Hu, J.; Zhang, H.; Song, L.; Han, Z.; Poor, H.V. Reinforcement learning for a cellular internet of UAVs: Protocol design, trajectory control, and resource management. *IEEE Wirel. Commun.* **2020**, *27*, 116–123. [[CrossRef](#)]
6. Chen, X.; Liu, X.; Chen, Y.; Jiao, L.; Min, G. Deep Q-Network based resource allocation for UAV-assisted Ultra-Dense Networks. *Comput. Networks* **2021**, *196*, 108249. [[CrossRef](#)]
7. Munaye, Y.Y.; Juang, R.-T.; Lin, H.-P.; Tarekegn, G.B.; Lin, D.-B. Deep reinforcement learning based resource management in UAV-assisted IoT networks. *Appl. Sci.* **2021**, *11*, 2163. [[CrossRef](#)]



8. 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](#)]
9. Qi, W.; Song, Q.; Guo, L.; Jamalipour, A. Energy-Efficient Resource Allocation for UAV-Assisted Vehicular Networks with Spectrum Sharing. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7691–7702. [[CrossRef](#)]
10. Xu, Y.-H.; Sun, Q.-M.; Zhou, W.; Yu, G. Resource allocation for UAV-aided energy harvesting-powered D2D communications: A reinforcement learning-based scheme. *Ad Hoc Netw.* **2022**, *136*, 102973. [[CrossRef](#)]
11. Chang, Z.; Deng, H.; You, L.; Min, G.; Garg, S.; Kaddoum, G. Trajectory design and resource allocation for multi-UAV networks: Deep reinforcement learning approaches. *IEEE Trans. Netw. Sci. Eng.* **2022**, *10*, 2940–2951. [[CrossRef](#)]
12. Li, K.; Ni, W.; Dressler, F. LSTM-characterized deep reinforcement learning for continuous flight control and resource allocation in UAV-assisted sensor network. *IEEE Internet Things J.* **2021**, *9*, 4179–4189. [[CrossRef](#)]
13. Yin, S.; Yu, F.R. Resource allocation and trajectory design in UAV-aided cellular networks based on multiagent reinforcement learning. *IEEE Internet Things J.* **2021**, *9*, 2933–2943. [[CrossRef](#)]
14. 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](#)]
15. Niu, Y.; Yan, X.; Wang, Y.; Niu, Y. An adaptive neighborhood-based search enhanced artificial ecosystem optimizer for UCAV path planning. *Expert Syst. Appl.* **2022**, *208*, 118047. [[CrossRef](#)]
16. Yin, Z.; Lin, Y.; Zhang, Y.; Qian, Y.; Shu, F.; Li, J. Collaborative Multiagent Reinforcement Learning Aided Resource Allocation for UAV Anti-Jamming Communication. *IEEE Internet Things J.* **2022**, *9*, 23995–24008. [[CrossRef](#)]
17. Nie, Y.; Zhao, J.; Gao, F.; Yu, F.R. Semi-distributed resource management in UAV-aided MEC systems: A multi-agent federated reinforcement learning approach. *IEEE Trans. Veh. Technol.* **2021**, *70*, 13162–13173. [[CrossRef](#)]
18. Zhang, S.; Zhu, Y.; Liu, J. Multi-UAV Enabled Aerial-Ground Integrated Networks: A Stochastic Geometry Analysis. *IEEE Trans. Commun.* **2022**, *70*, 7040–7054. [[CrossRef](#)]
19. Zhang, Y.; Satapathy, S.C.; Wang, S. Fruit category classification by fractional Fourier entropy with rotation angle vector grid and stacked sparse autoencoder. *Expert Syst.* **2022**, *39*, e12701. [[CrossRef](#)]
20. Izci, D.; Hekimoğlu, B.; Ekinçi, S. A new artificial ecosystem-based optimization integrated with Nelder-Mead method for PID controller design of buck converter. *Alex. Eng. J.* **2022**, *61*, 2030–2044. [[CrossRef](#)]
21. Domingo, M.C. Power Allocation and Energy Cooperation for UAV-Enabled MmWave Networks: A Multi-Agent Deep Reinforcement Learning Approach. *Sensors* **2022**, *22*, 270. [[CrossRef](#)] [[PubMed](#)]

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