Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8891020, 14 pages
https://doi.org/10.1155/2021/8891020

Research Article

Novel Bee Colony Optimization with Update Quantities for OFDMA Resource Allocation

Ming Sun,[1] **Yujing Huang**,[1] **Shumei Wang**,[2] **and Yaoqun Xu**[2]

[1]*College of Computer and Control Engineering, Qiqihar University, Qiqihar 161006, China*
[2]*School of Computer and Information Engineering, Harbin University of Commerce, Harbin 150028, China*

Correspondence should be addressed to Ming Sun; snogisunming@163.com

Received 4 January 2021; Revised 18 June 2021; Accepted 5 July 2021; Published 21 July 2021

Academic Editor: Marta Cimitile

In recent years, large usage of wireless networks puts forward challenge to the utilization of spectrum resources, and it is significant to improve the spectrum utilization and the system sum data rates in the premise of fairness. However, the existing algorithms have drawbacks in efficiency to maximize the sum data rates of orthogonal frequency division multiple access (OFDMA) systems in the premise of fairness threshold. To address the issue, a novel artificial bee colony algorithm with update quantities of nectar sources is proposed for OFDMA resource allocation in this paper. Firstly, the population of nectar sources is divided into several groups, and a different update quantity of nectar sources is set for each group. Secondly, based on the update quantities of nectar sources set for these groups, nectar sources are initialized by a greedy subcarrier allocation method. Thirdly, neighborhood searches and updates are performed on dimensions of nectar sources corresponding to the preset update quantities. The proposed algorithm can not only make the initialized nectar sources maintain high levels of fairness through the greedy subcarrier allocation but also use the preset update quantities to reduce dimensions of the nectar sources to be optimized by the artificial bee colony algorithm, thereby making full use of both the local optimization of the greedy method and the global optimization of the artificial bee colony algorithm. The simulation results show that, just in the equal-power subcarrier allocation stage, the proposed algorithm can achieve the required fairness threshold and effectively improve the system sum data rates.

1. Introduction

It has become an undisputed fact that spectrum resources are limited wireless resources. Wireless networks have brought great changes to the world and people's lifestyles because of their strong scalability and mobility. All kinds of electronic products, such as mobile phones and tablet computers, have gradually become necessities of life, and the rapid increase of the demand for wireless networks causes spectrum resources to become more and more precious. Therefore, how to improve the utilization of spectrum resources to meet the various requirements of wireless networks for the sum data rates, power transmissions, energy efficiency, and so on has become an important issue at present [1–5].

Orthogonal frequency division multiple access (OFDMA) is a widely used modulation technology in wireless communications, which divides the bandwidth of a high-speed data stream into several mutually orthogonal low-speed data streams and effectively resists the frequency selectivity weakening [6–8]. Besides, the performance of OFDAM systems, such as signal detection and bit-error-rate evaluation, can be further enhanced by deep neural networks [9, 10]. OFDMA systems can adaptively allocate wireless resources such as subcarriers and power based on the rate adaptive (RA) criterion to improve the transmission performance of the system [8]. In the RA criterion, diversity gain is still an effective technical means to improve the performance of OFDMA systems. Balancing the tradeoff between the sum data rates and fairness in multiuser OFDMA systems has been widely concerned [7, 8, 11–28]. Fairness has been considered in different forms, such as rate proportional fairness [11], capacity-outage fairness [27], and delay-outage fairness [28]. The rate proportional fairness is the constraint of users' rate proportionality while the capacity/delay-outage fairness is the constraint of the percentage of users satisfying minimum capacity/delay performance. This paper focuses on the former.

Various techniques such as greedy algorithms [11, 15, 23], evolutionary algorithms [7, 12–14, 18, 19, 21, 22, 25, 26], game theory [24], and deep neural networks [16, 17, 20, 27] have been applied for resource allocations with fairness in OFDMA systems. For example, Shen et al. [11] proposed a greedy subcarrier allocation algorithm based on the data rate proportional fairness. The algorithm allocates subcarriers according to a predetermined data rate proportional coefficient with equal power, which can achieve a high level of fairness. However, the algorithm proposed by Shen et al. neglects the improvement of the sum data rates because of blindly pursuing a high level of fairness. Inspired by Shen et al., Wong et al. [15] first approximated the data rate proportional coefficient as the subcarrier number proportional coefficient and then allocated subcarriers, which improves the system sum data rates in the way of reducing the level of fairness. Yuan et al. [19, 26] used the artificial bee colony algorithm (ABC) to allocate optimal power for further improvement of the system sum data rates. However, the works in [19, 26] improve the system sum data rates just by simply sacrificing fairness, which lacks certain flexibility.

Since the fairness of OFDMA resource allocation can affect the system sum data rates, setting the fairness threshold for resource allocation can not only ensure fairness but also improve the flexibility of resource allocation. For example, Zhang and Zhao [14] first used an algorithm based on the fairness threshold to allocate equal-power subcarriers to achieve a rough compromise between the system sum data rates and the fairness and then used particle swarm optimization (PSO) to allocate power to ensure the required fairness threshold. However, characteristics of the greedy optimization of the algorithm based on the fairness threshold make it difficult to ensure the fairness threshold just in the equal-power subcarrier allocation stage. At the same time, in the process of allocating power to ensure the fairness threshold by PSO, improper inertia weights can easily lead PSO to be premature and hence reduce the system sum data rates. As another example, Sharma and Anpalagan [12] and Sharma and Madhukumar [13] directly use an ABC algorithm and a genetic algorithm (GA), for which the fairness threshold can be set, for subcarrier allocation. Compared with the greedy subcarrier allocation algorithm based on the fairness threshold in [14], both the ABC algorithm [12] and the GA [13] can allocate subcarriers globally and optimally. However, dimensions of variables to be optimized by both the ABC algorithm and the GA increase rapidly with the number of subcarriers increasing. The large-scale subcarrier allocation not only increases the difficulty of optimization but also puts forward challenges to the optimization performance of both the ABC algorithm and the GA. The optimization performance of both the ABC algorithm and the GA is easily decayed especially when subcarriers are relatively insufficient in allocation to users [21]. In order to avoid the reduction of the system sum data rates caused by ensuring the fairness threshold as much as possible, Sun et al. [21] adopted a method, which combines a greedy algorithm of local optimization with a Hungarian algorithm of global optimization. In addition, an ABC algorithm was adopted in [21] to ensure the required fairness threshold and obtain better resource allocation solutions than those in [7, 12]. However, the method in [21] does not make full use of the fairness threshold in the subcarrier allocation stage, so it is still insufficient to improve the utilization of subcarriers. In addition, although the above algorithms in [12–14] can flexibly adjust the fairness and the system sum data rates by setting a fairness threshold, few algorithms can be superior to the Shen algorithm [10] in both the fairness and the system sum data rates at the same time, which shows the shortcomings of the above algorithms in [12–14].

Deep neural networks perform slightly poorer but with lower computational complexity and latency than nondeep learning algorithms [16, 17, 20, 27]. However, deep neural networks need environment-specific design and training data with labels for supervised learning [16, 17, 20, 27]. Evolutionary algorithms have advantages over global optimization [29, 30] and can be used to generate environment-specific training data with labels for training supervised-learning-based deep neural networks in our future study. Motivated by these, we apply ABC to maximize the system sum data rates while ensuring the fairness threshold. The contributions of this paper are stated as follows.

(1) In order to maximize the system sum data rates more effectively in the premise of fairness threshold, this paper proposes a novel ABC algorithm with update quantities of nectar sources (ABC-UQ) for the OFDMA resource allocation by combining a greedy subcarrier allocation method based on the data rate proportional fairness with the ABC algorithm

(2) The following measures are taken in our proposed ABC-UQ algorithm. Firstly, the proposed ABC-UQ divides the population of nectar sources into several groups and sets a different update quantity of nectar sources for each group. Secondly, based on the update quantities of nectar sources set for these groups, the proposed ABC-UQ initializes nectar sources by using the greedy subcarrier allocation method based on the data rate proportional fairness. Thirdly, the proposed ABC-UQ performs neighborhood searches and updates on dimensions of nectar sources corresponding to the update quantities

(3) Because of the measures taken as above, our proposed ABC-UQ can make full use of the advantages of the greedy algorithm in local optimization and the ABC algorithm in global optimization to maximize the system sum data rates in the premise of fairness threshold, listed as the following two aspects. For one thing, based on the preset update quantities of nectar sources, the greedy subcarrier allocation method based on the data rate proportional fairness is used to initialize nectar sources, which not only makes the initialized nectar sources maintain a higher level of fairness but also reduces dimensions to be optimized in nectar sources. For another, the ABC algorithm is used for the global optimization

on the reduced dimensions in nectar sources, which is helpful in ensuring the fairness threshold and enhancing the utilization of subcarriers and the system sum data rates. Simulation results show that, compared with the existing algorithms [12–14, 21] using the fairness threshold for enhancing the system sum data rates, the proposed algorithm is more efficient in improving the system sum data rates in the premise of fairness threshold just in the equal-power subcarrier allocation stage

The rest of the paper is structured as follows. Section 2 provides the formulation of the OFDMA resource allocation with constraint of the fairness threshold. The proposed ABC-UQ for the OFDMA resource allocation is described in detail in Section 3. Section 4 presents the simulation results, and finally, Section 5 concludes this paper.

2. System Model

It is assumed that a multiuser OFDMA downlink system has one base station, K users, and N subcarriers. In addition, the transmission power of the base station transmitter is assumed to be P_T, and the base station is assumed to completely know the channel state information. After a series of processing such as the transmitter adaptive modulation, IFFT and parallel-to-serial conversion, and adding cyclic prefix, the data is transmitted through Rayleigh fading channel with slow time variation and total bandwidth of B and then sent to users after a series of processing including cyclic prefix removal, serial-to-parallel and FFT conversion, and the receiver adaptive demodulation [19].

Because of the constraint requirements of the fairness threshold, the resource allocation problem of the multiuser OFDMA is modeled as follows:

$$\max_{p_{k,n},c_{k,n}} \sum_{k=1}^{K}\sum_{n=1}^{N}\frac{B}{N}c_{k,n}\log_2\left(1+\frac{p_{k,n}\left|h_{k,n}\right|^2}{N_0B/N}\right), \tag{1}$$

$$\sum_{k=1}^{K}c_{k,n}=1,\ c_{k,n}\in\{0,1\},\forall k,n, \tag{2}$$

$$\sum_{k=1}^{K}\sum_{n=1}^{N}c_{k,n}p_{k,n}\le P_T,\ p_{k,n}\ge 0,\forall k,n, \tag{3}$$

$$F=\frac{\left(\sum_{k=1}^{K}R_k/\lambda_k\right)^2}{K\sum_{k=1}^{K}\left(R_k/\lambda_k\right)^2}\ge\varepsilon, \tag{4}$$

where Equation (1) is the objective function of the multiuser OFDMA resource allocation problem; Equation (2) indicates that each subcarrier is uniquely allocated to one single user; Equation (3) represents the total transmit power constraint of subcarriers; Equation (4) represents the fairness threshold constraint required by the system; B is the total bandwidth available to the system; N_0 is the additive white Gaussian noise power density; P_T is the total transmission power of the base station; $p_{k,n}$ represents the transmission power of subcarrier n allocated to user k; $H_{k,n}$ represents the channel gain of user k on subcarrier n; $c_{k,n}$ indicates whether subcarrier n is allocated to user k; if subcarrier n is allocated to user k, $c_{k,n}=1$; otherwise, $c_{k,n}=0$; and R_k is the data rate of user k, which can be expressed as

$$R_k=\sum_{n=1}^{N}\frac{B}{N}c_{k,n}\log_2\left(1+\frac{p_{k,n}\left|h_{k,n}\right|^2}{N_0B/N}\right), \tag{5}$$

where λ_k is the predetermined data rate proportional coefficient, $F\in(0,1]$ is the fairness achieved by the system and F is equal to 1 only if $R_1:R_2:\cdots:R_K=\lambda_1:\lambda_2:\cdots:\lambda_K$, and ε is the required fairness threshold.

It can be seen from Equations (1)–(5) that the multiuser OFDMA resource allocation problem is an NP-hard problem, and the simultaneous joint allocation of subcarriers and power may take a large computation cost [8]. However, the computation cost can be reduced by allocating subcarriers and power separately [6]. For this reason, the equal-power subcarrier allocation is used in this paper to solve the OFDMA resource allocation problem (1)–(5), i.e., $p_{k,n}=P_T/N$. So, the multiuser OFDMA resource allocation problem solved in this paper can be transformed as follows:

$$\max_{c_{k,n}} \sum_{k=1}^{K}\sum_{n=1}^{N}\frac{B}{N}c_{k,n}\log_2\left(1+\frac{p_T\left|h_{k,n}\right|^2}{N_0B}\right), \tag{6}$$

$$\sum_{k=1}^{K}c_{k,n}=1,\ c_{k,n}\in\{0,1\},\forall k,n, \tag{7}$$

$$g(c)=F-\varepsilon\ge 0, \tag{8}$$

$$R_k=\sum_{n=1}^{N}\frac{B}{N}c_{k,n}\log_2\left(1+\frac{p_T\left|h_{k,n}\right|^2}{N_0B}\right), \tag{9}$$

where c is a subcarrier allocation matrix composed by $c_{k,n}$.

3. Proposed ABC-UQ for Resource Allocation

ABC [29, 31] is a metaheuristic algorithm that imitates bees to find and collect nectar, in which the position of nectar sources corresponds to that of the solution of optimization problems, while the process of searching and collecting nectar sources by bees corresponds to that of solving optimization problems. When the ABC algorithm is used for subcarrier allocation, the scale of dimensions of nectar sources is determined by the number of subcarriers [12]. However, the ABC algorithm has drawbacks in slow convergence speed and weak exploitation capacity [29, 31], and the large-scale subcarrier allocation would undoubtedly increase the difficulty of bee colony optimization, which easily decays the optimization solutions.

In order to use the ABC algorithm to guarantee the fairness threshold and improve the subcarrier utilization and the system sum data rates more effectively, this paper applies a greedy subcarrier allocation method based on the data rate proportional fairness to initialize the population of nectar sources and proposes a novel artificial bee colony algorithm with update quantities of nectar sources (ABC-UQ) for the resource allocation. The proposed ABC-UQ first divides the population of nectar sources into several groups and sets a different update quantity of nectar sources for each group. After that, the proposed ABC-UQ uses the greedy subcarrier allocation method based on the data rate proportional fairness to initialize the population of nectar sources with the help of the update quantities of nectar sources set for those groups. At last, the proposed ABC-UQ performs neighborhood searches and updates on dimensions of nectar sources corresponding to the preset update quantities.

The proposed ABC-UQ for the resource allocation includes several parts, such as initialization for the population of nectar sources, fitness calculation for nectar sources, phase of employed bees, phase of onlooker bees, storage of the optimal nectar source, and phase of scout bees. In the following, each part is described detailedly.

3.1. Initialization for Population of Nectar Sources. In this paper, a nectar source variable is defined as a vector with dimension D equal to the number of subcarriers N. Denote S as the population size of nectar sources, x_i as the ith individual of nectar sources, and $x_{i,d}$ as the element of the dth dimension in x_i, where $x_{i,d} \in [1, K]$, $i \in \{1, 2, \cdots, S\}$, $d \in \{1, 2, \cdots, D\}$.

The initialization for the population of nectar sources is described in the following steps.

Step 1. According to the population size S, the population of nectar sources is divided into $\bar{S}$ groups and sets a different update quantity of nectar sources $N_{\bar{s}}$ for each group $\bar{s}$, where $\bar{S} \in \{2, 3, \cdots, S\}$, $\bar{s} \in \{1, 2, \cdots, \bar{S}\}$, $N_{\bar{s}} \in \{1, 2, \cdots, N-1\}$.

Step 2. Obtain the minimal update quantity of nectar sources $N_{\bar{s}}^{\min}$ and $\bar{N} = N - N_{\bar{s}}^{\min}$. Then, the greedy subcarrier allocation method based on data rate proportional fairness is used to generate $\bar{N}$ binary allocation pairs, which is described in detail in Steps 3–5. According to the generation order of the binary allocation pairs, the generated $\bar{N}$ binary allocation pairs are denoted as $\Phi_{\bar{n}} = \langle n_{\bar{n}}, k_{\bar{n}} \rangle$ $(\bar{n} = 1, 2, \cdots, \bar{N})$, where $n_{\bar{n}} \in \{1, 2, \cdots, N\}$ represents a subcarrier and $k_{\bar{n}} \in \{1, 2, \cdots, K\}$ represents a user.

Step 3. $R_1 = R_2 = \cdots = R_K = 0$, $\Omega = \{1, 2, \cdots, N\}$, $\bar{n} = 1$, $k = 1$.

Step 4. while ($k \le K$and $\bar{n} \le \bar{N}$) {
$n = \arg\ \max_{n \in \Omega}\{|h_{k,n}|^2\}$
$n_{\bar{n}} = n$
$k_{\bar{n}} = k$,
$\Phi_{\bar{n}} = \langle n_{\bar{n}}, k_{\bar{n}} \rangle$
$\bar{n} = \bar{n} + 1$
$R_k = R_k + \log_2(1 + p_T|h_{k,n}|^2/(N_0 B))$
$\Omega = \Omega - \{n\}$
$k = k + 1$}.

Step 5. while ($\bar{n} \le \bar{N}$) {
$k = \arg\ \min_{k \in \{1,2,\cdots,K\}}\{R_k/\lambda_k\}$
$n = \arg\ \max_{n \in \Omega}\{|h_{k,n}|^2\}$
$n_{\bar{n}} = n$
$k_{\bar{n}} = k$
$\Phi_{\bar{n}} = \langle n_{\bar{n}}, k_{\bar{n}} \rangle$
$\bar{n} = \bar{n} + 1$
$R_k = R_k + \log_2(1 + p_T|h_{k,n}|^2/(N_0 B))$
$\Omega = \Omega - \{n\}$}.

Step 6. Use the binary allocation pairs $\Phi_{\bar{n}} = \langle n_{\bar{n}}, k_{\bar{n}} \rangle$ $(\bar{n} = 1, 2, \cdots, \bar{N})$ generated in Step 2 to initialize the nectar sources of each group, shown as Steps 7-9.

Step 7. For each group $\bar{s}$ $(\bar{s} \in \{1, 2, \cdots, \bar{S}\})$ and its corresponding update quantity of nectar sources $N_{\bar{s}}$, compute $\bar{N}_{\bar{s}} = N - N_{\bar{s}}$. Because $N_{\bar{s}} \ge N_{\bar{s}}^{\min}$, $N - N_{\bar{s}} \le N - N_{\bar{s}}^{\min}$, i.e., $\bar{N}_{\bar{s}} \le \bar{N}$ comes into existence.

Step 8. Initialize $\bar{N}_{\bar{s}}$ dimensions of each nectar source x_j in each group $\bar{s}$ by using the first $\bar{N}_{\bar{s}}$ binary allocation pairs, i.e., $\Phi_m = \langle n_m, k_m \rangle$ $(m = 1, 2, \cdots, \bar{N}_{\bar{s}})$, shown as

$$x_{j,n_m} = k_m \left(m = 1, 2, \cdots, \bar{N}_{\bar{s}}\right), \tag{10}$$

where $k_m \in \{1, 2, \cdots, K\}$, $n_m \in \{1, 2, \cdots, N\}$.

Step 9. Initialize the remaining $N_{\bar{s}}$ dimensions of each nectar source x_j in each group $\bar{s}$ by using random real numbers in $[1, K]$, shown as

$$x_{j,\bar{n}_l} = 1 + r_l(K-1)\ (l = 1, 2, \cdots, N_{\bar{s}}), \tag{11}$$

where $r_l \in [0, 1]$ is a uniformly distributed random real number and $\bar{n}_l \in \Omega_{\bar{s}}$, where the set $\Omega_{\bar{s}}$ contains the remaining $N_{\bar{s}}$ dimensions of each nectar source x_j in each group $\bar{s}$, expressed as $\Omega_{\bar{s}} = \{1, 2, \cdots, N\} - \{n_1, n_2, \cdots, n_{\bar{N}_{\bar{s}}}\}$. Obviously, $\Omega_{\bar{s}}$ is just the set corresponding to the preset update quantity $N_{\bar{s}}$ of each nectar source x_j in the group $\bar{s}$.

The following takes the initialization for the population of nectar sources with the number of subcarriers $N = 16$ and the number of users $K = 4$ as an example. In Step 1, the population size of nectar sources is set as $S = 12$, and the number of groups of nectar sources is set as $\bar{S} = 6$ $(\bar{s} = 1, 2, \cdots, 6)$, and each group contains 2 nectar sources, and the update quantities of nectar sources for the six groups are set as $N_1 = 1, N_2 = 2, N_3 = 4, N_4 = 6, N_5 = 7,$ and $N_6 = 8$, respectively. In Step 2, the minimal update quantity of nectar sources $N_{\bar{s}}^{\min} = 1$, and $\bar{N} = 15$, and then, the 15 binary allocation pairs are sequentially generated by the greedy subcarrier allocation method based on the data rate proportional fairness, assumed to be $\Phi_1 = \langle 11, 2 \rangle$, $\Phi_2 = \langle 4, 1 \rangle$, $\Phi_3 = \langle 6, 4 \rangle$, $\Phi_4 = \langle 16, 3 \rangle$, $\Phi_5 = \langle 13, 4 \rangle$, $\Phi_6 = \langle 7, 3 \rangle$, $\Phi_7 = \langle 5, 1 \rangle$, $\Phi_8 = \langle 14, 4 \rangle$, $\Phi_9 = \langle 12, 2 \rangle$, $\Phi_{10} = \langle 15, 3 \rangle$, $\Phi_{11} = \langle 3, 1 \rangle$, $\Phi_{12} = \langle 2, 2 \rangle$, $\Phi_{13} = \langle 1, 4 \rangle$, $\Phi_{14} = \langle 8, 3 \rangle$, $\Phi_{15} = \langle 10, 2 \rangle$. In Step 6, initialize $\bar{N}_{\bar{s}}$ dimensions of each nectar source in each group $\bar{s}$ by using the first $\bar{N}_{\bar{s}}$ binary allocation pairs, where $\bar{N}_1 = 15$, $\bar{N}_2 = 14$, $\bar{N}_3 = 12$, $\bar{N}_4 = 10$, $\bar{N}_5 = 9$, and $\bar{N}_6 = 8$. After that, initialize the remaining $N_{\bar{s}}$ dimensions of each nectar source in each

Table 1: Results of initialization for the population of nectar sources with the number of subcarriers $N = 16$ and the number of users $K = 4$.

Update quantities	Nectar sources	Subcarrier n															
$N_{\bar{s}}$	$x_{i,n}$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$N_1 = 1$	$i = 1$	4	2	1	1	1	4	3	3	**3.11**	2	2	2	4	4	3	3
	$i = 2$	4	2	1	1	1	4	3	3	**3.67**	2	2	2	4	4	3	3
$N_2 = 2$	$i = 3$	4	2	1	1	1	4	3	3	**3.65**	**3.95**	2	2	4	4	3	3
	$i = 4$	4	2	1	1	1	4	3	3	**1.64**	**2.93**	2	2	4	4	3	3
$N_3 = 4$	$i = 5$	**2.76**	2	1	1	1	4	3	**2.19**	**3.29**	**2.93**	2	2	4	4	3	3
	$i = 6$	**3.84**	2	1	1	1	4	3	**3.79**	**2.42**	**3.05**	2	2	4	4	3	3
$N_4 = 6$	$i = 7$	**2.02**	**1.51**	**3.6**	1	1	4	3	**2.53**	**2.07**	**2.59**	2	2	4	4	3	3
	$i = 8$	**3.06**	**3.03**	**1.38**	1	1	4	3	**2.05**	**1.94**	**1.53**	2	2	4	4	3	3
$N_5 = 7$	$i = 9$	**3.66**	**2.18**	**3.74**	1	1	4	3	**3.81**	**3.73**	**3.74**	2	2	4	4	**2.51**	3
	$i = 10$	**2.36**	**2.82**	**1.07**	1	1	4	3	**1.38**	**2.23**	**1.78**	2	2	4	4	**2.12**	3
$N_6 = 8$	$i = 11$	**2.60**	**3.63**	**3.89**	1	1	4	3	**3.78**	**1.48**	**1.42**	2	**1.61**	4	4	**1.55**	3
	$i = 12$	**1.72**	**1.44**	**2.95**	1	1	4	3	**3.78**	**1.66**	**3.86**	2	**2.85**	4	4	**3.31**	3

Sets corresponding to the bold areas of each group are $\Omega_1 = \{9\}$, $\Omega_2 = \{9, 10\}$, $\Omega_3 = \{1, 8, 9, 10\}$, $\Omega_4 = \{1, 2, 3, 8, 9, 10\}$, $\Omega_5 = \{1, 2, 3, 8, 9, 10, 15\}$, and $\Omega_6 = \{1, 2, 3, 8, 9, 10, 12, 15\}$, respectively. Elements of the bold areas in each nectar source are initialized by Equation (11), and the rest elements are initialized by Equation (10).

group $\bar{s}$ by using random real numbers in $[1, K]$. Results of initialization for the population of nectar sources are shown in Table 1.

It can be seen from Table 1 that the sets $\Omega_{\bar{s}}$ ($\bar{s} = 1, 2, 3, 4, 5, 6$) are corresponding to the preset update quantities ($N_{\bar{s}} = 1, 2, 4, 6, 7, 8$) of the six groups of nectar sources. In addition, initialization values corresponding to dimensions in $\Omega_{\bar{s}}$ are the elements in the bold areas in Table 1. Obviously, it can enable the initialized nectar sources to maintain a high level of fairness by using the greedy subcarrier allocation method based on the data rate proportional fairness shown in Steps 3–5 for initialization. The nectar sources with smaller update quantities are more likely to have high fairness. For example, the nectar sources x_1 and x_2 in the group $\bar{s} = 1$ with the update quantity $N_{\bar{s}} = 1$ are more likely to have high fairness than x_{11} and x_{12} in the group $\bar{s} = 6$ with the update quantity $N_{\bar{s}} = 8$. However, it is helpful to keep the diversity of the population of nectar sources by setting different update quantities for different groups of nectar sources.

3.2. Fitness Calculation for Nectar Sources. Because the multiuser OFDMA resource allocation problem (6)–(9) is a constrained optimization problem, the fitness function should be designed to consider feasible solutions satisfying the constraints and infeasible solutions not satisfying the constraints and can guide the ABC algorithm to transit from the space of infeasible solutions to that of feasible solutions [31, 32].

Based on the above, the fitness function designed in this paper is shown as below:

$$\mathrm{Fit}(x_i) = \begin{cases} \dfrac{1}{1 + f(\bar{x}_i)}, & \text{if } f(\bar{x}_i) \geq 0, \\ 1 + |f(\bar{x}_i)|, & \text{if } f(\bar{x}_i) < 0, \end{cases} \tag{12}$$

where $\mathrm{Fit}(x_i)$ is the fitness function of nectar source x_i and $\bar{x}_i$ is the vector obtained by rounding the elements of nectar source x_i, and $f(\bar{x}_i)$ is the objective function of the vector $\bar{x}_i$, expressed as follows:

$$f(\bar{x}_i) = \begin{cases} f_0 + |g(X_i)|, & \text{if } g(X_i) < 0, \\ -T(X_i), & \text{if } g(X_i) \geq 0, \end{cases} \tag{13}$$

where f_0 is a positive constant and $g(\cdot) = F - \varepsilon$ is the function of the fairness threshold constraint, and X_i is the K-row and N-column subcarrier allocation matrix transformed from $\bar{x}_i$. The transformation from the vector $\bar{x}_i$ to the matrix X_i is shown as

$$[X_i]_{k,n} = \begin{cases} 1, & \text{if } [\bar{x}_i]_n = k, \\ 0, & \text{if } [\bar{x}_i]_n \neq k, \end{cases} \tag{14}$$

where $[X_i]_{k,n}$ is the element in the kth row and the nth column of the subcarrier allocation matrix X_i and $[\bar{x}_i]_n$ is the element in dimension n of the vector $\bar{x}_i$. $T(X_i)$ is expressed as

$$T(X_i) = \sum_{k=1}^{K} \sum_{n=1}^{N} \frac{B}{N} [X_i]_{k,n} \log_2 \left(1 + \frac{P_T |h_{k,n}|^2}{N_0 B} \right). \tag{15}$$

Firstly, it can be seen from Equations (12) and (13) that the fitness function (12) is determined by the objective function (13), while the objective function (13) is affected by feasible solutions satisfying the fairness threshold constraint and infeasible solutions not satisfying the fairness threshold constraint. From that, it can be concluded that the fitness value of a feasible solution is greater than that of an infeasible

Update quantities $N_{\bar{s}}$	Nectar sources $x_{i,n}$	Subcarrier$_n$ 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
N_1=1	q=1	4	2	1	1	1	4	3	3	3.11	2	2	2	4	4	3	3
		↑	↑	↑					↑	↑	↑		↑			↑	
N_6=8	i=12	1.72	1.44	2.95	1	1	4	3	3.78	1.66	3.86	2	2.85	4	4	3.31	3

Figure 1: Schematic diagram of the neighborhood searches and updates by taking the nectar sources 12 and 1 in Table 1 as an example.

solution. Secondly, it is not difficult to find out from Equations (12), (13), and (15) that among the feasible solutions satisfying the fairness threshold constraint, the preferred feasible solution has a larger system sum data rate and a larger fitness value. Finally, it can be known from Equations (12) and (13) that among the infeasible solutions that do not satisfy the fairness threshold constraint, the superior infeasible solution also has a larger fitness value than the inferior infeasible solution. The above analyses show that the designed fitness function accords with the principle that superior solutions should have larger fitness, which is conducive to guiding the ABC algorithm to transit from the space of infeasible solutions to that of feasible solutions and can search for better feasible solutions in the space of feasible solutions.

3.3. Phase of Employed Bees. In solving optimization problems by the ABC algorithm, it is beneficial to improve optimization solutions by reducing dimensions to be optimized in nectar sources. Hence, according to the preset update quantity $N_{\bar{s}}$ of nectar sources in the group $\bar{s}$, this paper proposes to perform neighborhood searches and updates just on $N_{\bar{s}}$ dimensions of each nectar source in the group $\bar{s}$. Note that, $\Omega_{\bar{s}}$ is the set corresponding to the preset update quantity $N_{\bar{s}}$ of each nectar source in the group $\bar{s}$, and $|\Omega_{\bar{s}}| = N_{\bar{s}}$.

In this paper, the following neighborhood searches and updates of nectar sources for phases of employed bees and onlooker bees are proposed, shown as

$$v_{i,n} = \begin{cases} x_{i,n} + \varphi_{i,n}\left(x_{i,n} - x_{q,n}\right), \\ \text{if}\left(r_n < \delta\right) \text{ and } \left(n \in \Omega_{\bar{s}(i)}\right), \\ x_{i,n}, \text{ otherwise}, \end{cases} \tag{16}$$

$$v_{i,n} = \max\left(\min\left(v_{i,n}, K\right), 1\right), \tag{17}$$

where Equation (17) is used to limit $v_{i,n}$ generated by Equation (16) to the range of $[1, K]$, $\bar{s}(i) \in \{1, 2, \cdots, \bar{S}\}$ is the group to which the ith nectar source belongs, $\Omega_{\bar{s}(i)}$ is the set of dimensions corresponding to the preset update quantity of nectar sources in the group $\bar{s}(i)$, $x_{q,n}$ is the element on the n th dimension of the qth nectar source, q is a randomly generated integer satisfying $q \in \{1, 2, \cdots, S\}$ and $q \neq i$, $\varphi_{i,n} \in [-1, 1]$ is a uniformly distributed random real number, $v_{i,n}$ is the newly generated element of the nth dimension of the ith nectar source, $r_n \in [0, 1]$ is a uniformly distributed random real number, and $\delta \in [0, 1]$ is the predetermined real number.

The following takes the nectar sources $i = 12$ and $i = 1$ in Table 1 as an example to explain the neighborhood searches and updates described in Equation (16). For convenience, the neighborhood searches and updates described in Equation (16) are shown in Figure 1, where the nectar sources $i = 12$ and $q = 1$ correspond to the nectar sources $i = 12$ and $i = 1$ in Table 1, respectively, and the elements in the shadow areas of the nectar source $i = 12$ corresponding to the set $\Omega_{\bar{s}(12)}$ ($\Omega_{\bar{s}(12)} = \Omega_6 = \{1, 2, 3, 8, 9, 10, 12, 15\}$) are those that the neighborhood searches and updates are performed on, and the sign "↑" is used to indicate the elements which take part in the neighborhood searches and updates in the nectar source $q = 1$.

Equations (16) and (17) suggest that the neighborhood searches and updates for nectar sources in the group $\bar{s}$ are performed just on dimensions in the set $\Omega_{\bar{s}}$ corresponding to the preset update quantity $N_{\bar{s}}$ of these nectar sources. Because $|\Omega_{\bar{s}}| = N_{\bar{s}}$ and $N_{\bar{s}} < N$, the neighborhood searches and updates described in Equations (16) and (17) can effectively reduce dimensions of nectar sources to be optimized compared to the ABC algorithm proposed by Sharma and Anpalagan [12] and benefit to the enhancement of optimization of the ABC algorithm.

In the phase of employed bees, the employed bees perform the neighborhood searches and updates for each nectar source x_i and generate a new nectar source v_i for x_i by using Equations (16) and (17) and calculate the fitness $\text{Fit}(x_i)$ for x_i and the fitness $\text{Fit}(v_i)$ for v_i by using Equations (12)–(15). In addition, the employed bees take comparisons between $\text{Fit}(x_i)$ and $\text{Fit}(v_i)$ to determine whether to adopt the new nectar source v_i and simultaneously modify the nonimprovement number Bas_i, shown as

$$\begin{cases} x_i = v_i, \text{Bas}_i = 0, & \text{if } \text{Fit}(v_i) > \text{Fit}(x_i), \\ x_i = x_i, \text{Bas}_i = \text{Bas}_i + 1, & \text{if } \text{Fit}\,(v_i) \leq \text{Fit}(x_i). \end{cases} \tag{18}$$

3.4. Phase of Onlooker Bees. In the phase of onlooker bees, the probability $p(x_i)$ for selecting nectar sources x_i calculated by onlooker bees is expressed as

$$p(x_i) = \begin{cases} \dfrac{1}{2}\left(1 + \dfrac{\text{Fit}(x_i)}{\sum_{s=1}^{S}\text{Fit}(x_s)}\right), & \text{if } \text{Fit}(x_i) > 1, \\ \dfrac{1}{2}\left(1 - \dfrac{\text{Fit}(x_i)}{\sum_{s=1}^{S}\text{Fit}(x_s)}\right), & \text{if } \text{Fit}(x_i) \leq 1. \end{cases} \tag{19}$$

The onlooker bees use the probability of Equation (19) to select S nectar sources by the way of roulette. Then, similar to the phase of employed bees, the onlooker bees perform the neighborhood searches and updates for each nectar source x_i and generate a new nectar source v_i for x_i by using

Table 2: Fairness obtained by various algorithms as the fairness threshold is set as $\varepsilon = F_{\text{Shen}}$.

K	$\lambda_1:\lambda_2:\cdots:\lambda_K=1:1:\cdots:1$				$\lambda_1:\lambda_2:\cdots:\lambda_K=8:1:\cdots:1$				$\lambda_1:\lambda_2:\cdots:\lambda_K=16:1:\cdots:1$			
	Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ	Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ	Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ
6	0.9992	0.9993	0.9993	0.9993	0.9958	0.9961	0.9734	0.9961	0.9892	0.9900	0.9113	0.9896
7	0.9989	0.9990	0.9985	0.9990	0.9954	0.9957	0.9635	0.9957	0.9890	0.9896	0.9104	0.9896
8	0.9987	0.9987	0.9972	0.9987	0.9948	0.9952	0.9681	0.9951	0.9884	0.9892	0.9236	0.9889
9	0.9983	0.9983	0.9959	0.9984	0.9940	0.9943	0.9543	0.9941	0.9869	0.9875	0.9166	0.9865
10	0.9979	0.9979	0.9922	0.9980	0.9936	0.9939	0.9584	0.9937	0.9864	0.9869	0.9259	0.9862
11	0.9975	0.9976	0.9886	0.9977	0.9931	0.9934	0.9540	0.9932	0.9859	0.9865	0.9264	0.9859
12	0.9970	0.9971	0.9862	0.9972	0.9928	0.9932	0.9487	0.9928	0.9865	0.9871	0.9252	0.9866
13	0.9966	0.9967	0.9820	0.9968	0.9918	0.9922	0.9517	0.9917	0.9859	0.9868	0.9300	0.9855
14	0.9958	0.9958	0.9750	0.9960	0.9899	0.9902	0.9495	0.9893	0.9851	0.9859	0.9276	0.9845
15	0.9949	0.9950	0.9680	0.9952	0.9889	0.9892	0.9474	0.9889	0.9859	0.9869	0.9275	0.9859
16	0.9944	0.9945	0.9663	0.9945	0.9873	0.9875	0.9427	0.9870	0.9852	0.9860	0.9240	0.9847

Equations (16) and (17) and calculate the fitness $\text{Fit}(x_i)$ for x_i and the fitness $\text{Fit}(v_i)$ for v_i by using Equations (12)–(15). Following that, the employed bees use Equation (18) to determine whether to adopt the new nectar source v_i and modify the nonimprovement number Bas_i at the same time. That is, if $\text{Fit}(x_i) > \text{Fit}(v_i)$, then $x_i = v_i$ and $\text{Bas}_i = 0$; otherwise, x_i remains unchanged and $\text{Bas}_i = \text{Bas}_i + 1$.

3.5. Storage of Optimal Nectar Source. Before the phase of scout bees, the nectar source with the largest fitness from the population of nectar sources was obtained, and the obtained nectar source was saved as the current optimal solution.

3.6. Phase of Scout Bees. Same as [12], a fixed interval of cycles, i.e., ρ, is used to generate the scout bees. If the current iteration number of ABC, Cycle, satisfies Cycle modulo ρ equal to zero, then check whether the largest nonimprovement number is larger than the predetermined constant, Limit. If Bas_i attached with the nectar source x_i is the largest nonimprovement number and larger than Limit, then abandon the original nectar source x_i, and create a new nectar source belonging to the same group as the original nectar source x_i, and set $\text{Bas}_i = 0$. What is more, if the fitness of the newly generated nectar source is greater than that of the optimal solution, the newly generated nectar source is saved as the current optimal solution.

Finally, judge whether the current iteration number, Cycle, is equal to the maximal iteration number, Maxcycle. If Cycle is equal to Maxcycle, stop calculating and output the optimal solution as well as its corresponding fairness and the system sum data rate; otherwise, Cycle = Cycle + 1 and continue to run by moving to the phase of employed bees.

4. Simulations and Analyses

In the simulations, it is assumed that the wireless channel of the multiuser OFDMA system is a six-path Rayleigh fading channel with exponential attenuation, where the multipaths of exponential attenuation are set as 0, -8.69, -17.37, -26.06, -34.74, and -43.43 dB [7]. In addition, the number of subcarriers N is set as 64, and the total bandwidth and transmit power of the base station are set as 1 MHz and 1 W, respectively. Results of both the fairness and the sum data rates are average values taken with 200 different experiments.

4.1. Simulations on Fairness Threshold $\varepsilon = F_{Shen}$. In order to verify that the proposed ABC-UQ in this paper has the ability to guarantee a higher fairness threshold and obtain larger sum data rates simultaneously, the fairness F_{Shen} obtained by the Shen algorithm [10] is used as the fairness threshold, i.e., $\varepsilon = F_{\text{Shen}}$, since the Shen algorithm tends to achieve higher fairness than other algorithms [21, 26]. For comparisons, the proposed ABC-UQ, ABC-OSA [12], and PSO-EQ [14] are used to solve the resource allocation with the fairness threshold constraint ($F - F_{\text{Shen}} \geq 0$). The parameters of the proposed ABC-UQ are set as the following: $S = 60$, $f_0 = 1000$, $\delta = 0.6$, $\rho = 12$, $\text{Limit} = 10$, $\text{Maxcycle} = 1000$, $S' = 6$, $N_1 = 1$, $N_2 = 4$, $N_3 = 6$, $N_4 = 8$, $N_5 = 10$, and $N_6 = 12$, and there are 10 nectar sources in each group for the proposed ABC-UQ. In addition, the data rate proportional coefficient is set as $\lambda_1:\lambda_2:\cdots:\lambda_K = 1:1:\cdots:1$, $\lambda_1:\lambda_2:\cdots:\lambda_K = 8:1:\cdots:1$, and $\lambda_1:\lambda_2:\cdots:\lambda_K = 16:1:\cdots:1$, and the number of users K varies from 6 to 16. To be fair, the population size and the maximal iteration number in PSO-EQ are set the same as the above. The fairness and the sum data rates obtained in the simulations are listed in Tables 2 and 3.

It can be seen from Tables 2 and 3 that the proposed ABC-UQ can surpass the Shen algorithm in both the fairness and the sum data rates at the same time. This makes our proposed ABC-UQ different from the previous algorithms in [7, 12–15, 19, 21, 26] which are superior to the Shen algorithm in only one single aspect of the fairness or the sum data rates. ABC-OSA simply applies an improved ABC for subcarrier allocation but still lacks an effective mechanism to improve the optimization efficiency and performance of ABC for the resource allocation with the fairness threshold constraint. Therefore, with users increasing, both the fairness and the sum data rates obtained by ABC-OSA become inferior to even lower and lower than the Shen algorithm. Different

Table 3: Sum data rates obtained by various algorithms as the fairness threshold is set as $\varepsilon = F_{\text{Shen}}$.

K	$\lambda_1:\lambda_2:\cdots:\lambda_K = 1:1:\cdots:1$ Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ	$\lambda_1:\lambda_2:\cdots:\lambda_K = 8:1:\cdots:1$ Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ	$\lambda_1:\lambda_2:\cdots:\lambda_K = 16:1:\cdots:1$ Shen	Proposed ABC-UQ	ABC-OSA	PSO-EQ
6	6.7521	6.7620	6.0765	5.0167	6.3974	6.4125	6.0292	4.6196	6.2310	6.2593	6.0008	4.4852
7	6.7743	6.7835	5.9236	4.6187	6.5422	6.5643	5.9861	4.5059	6.3984	6.4292	5.9924	4.3756
8	6.7658	6.7759	5.7334	4.3100	6.6257	6.6547	5.8688	4.3838	6.5083	6.5470	5.9028	4.3974
9	6.8319	6.8387	6.0783	4.2254	6.5043	6.5392	5.9465	4.0943	6.3006	6.3493	5.9914	4.0499
10	6.7527	6.7635	6.0173	3.9591	6.5036	6.5432	6.0887	3.9509	6.3416	6.3904	6.0490	3.9988
11	6.8644	6.8718	5.9319	4.0581	6.6729	6.7181	6.0291	3.9823	6.5180	6.5692	6.0748	4.2105
12	6.8817	6.8896	5.8397	3.9750	6.7380	6.7801	5.769	4.0723	6.5706	6.6152	5.8154	4.2431
13	6.8593	6.8683	5.9789	3.8879	6.7369	6.7846	6.0962	3.9729	6.5550	6.6127	6.0902	4.1121
14	6.8752	6.8856	5.8702	4.2726	6.6658	6.7106	6.0309	3.7581	6.4395	6.5034	5.9803	3.9005
15	6.8901	6.8982	5.8957	3.8421	6.7944	6.8307	5.9402	3.8370	6.6409	6.7135	5.9824	3.9419
16	6.8838	6.8928	5.9583	3.8014	6.7806	6.8168	6.0025	3.7733	6.6502	6.7200	5.9301	3.8404

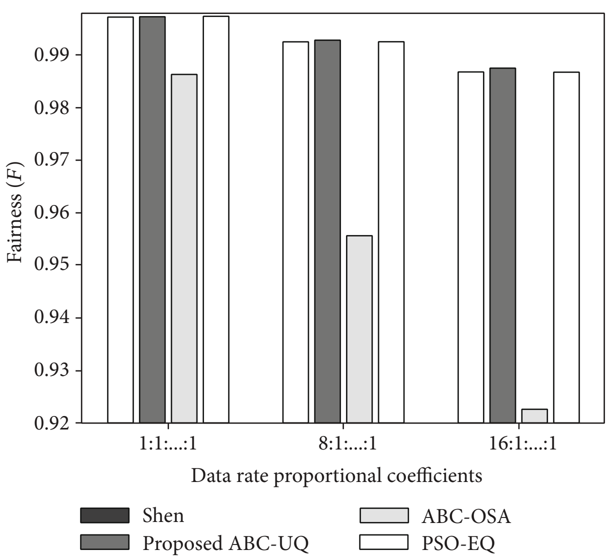

Figure 2: Average fairness obtained by various algorithms at the fairness threshold $\varepsilon = F_{\text{Shen}}$ and different data rate proportional coefficients.

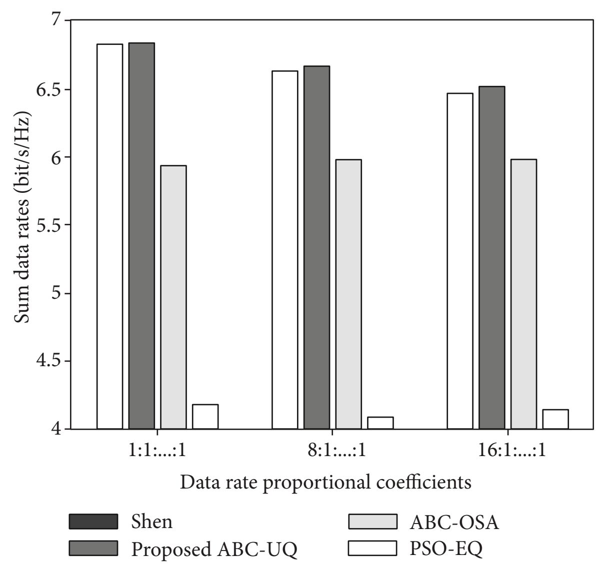

Figure 3: Average sum data rates obtained by various algorithms at the fairness threshold $\varepsilon = F_{\text{Shen}}$ and different data rate proportional coefficients.

from ABC-OSA, PSO-EQ uses the fairness threshold-based greedy algorithm to allocate subcarriers and applies PSO-based power allocation to guarantee the fairness threshold and maximize the sum data rates. Hence, fairness obtained by PSO-EQ can exceed the Shen algorithm. However, the sum data rates obtained by PSO-EQ is lower than that of the Shen algorithm. That is, PSO-EQ cannot be superior to the Shen algorithm in both the fairness and the sum data rates simultaneously.

In order to show the influence of different data rate proportional coefficients on the optimization of various algorithms, the average values of the fairness and the sum data rates of each algorithm with different data rate proportional coefficients obtained from Tables 2 and 3 are plotted in Figures 2 and 3. It can be seen from Figures 2 and 3 that the average values of both the fairness and the sum data rates obtained by various algorithms decrease with the increase of λ_1 of the data rate proportional coefficients. The main reason is that, with the increase of λ_1 of the data rate proportional coefficients, the subcarrier resources cannot better meet the requirements of users. However, Figures 2 and 3 show that the proposed ABC-UQ is superior to the Shen algorithm, ABC-OSA, and PSO-EQ in the average values of both the fairness and the sum data rates at different data rate proportional coefficients.

The above simulations on the fairness threshold $\varepsilon = F_{\text{Shen}}$ indicate that the proposed ABC-UQ can obtain both the higher fairness and the larger sum data rates than other algorithms at the same time.

4.2. Simulations on Fairness Threshold $\varepsilon = 0.96$. In this subsection, the fairness threshold ε is set as 0.96 to further demonstrate that, as the required fairness threshold is slightly

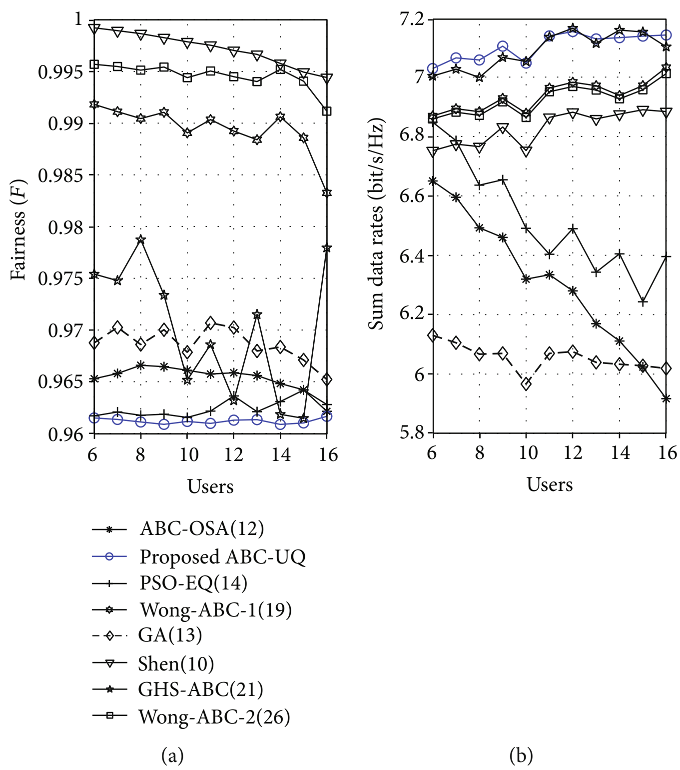

Figure 4: Fairness and sum data rates of various algorithms at the fairness threshold $\varepsilon = 0.96$ and $\lambda_1 : \lambda_2 : \cdots : \lambda_K = 1 : 1 : \cdots : 1$: (a) fairness; (b) sum data rates.

lower, the proposed ABC-UQ is more effective to improve the sum data rates. In the following simulations, parameters of the proposed ABC-UQ remain the same as the above. For comparisons, the algorithms including the proposed ABC-UQ, Shen [10], ABC-OSA [12], GA [13], PSO-EQ [14], Wong-ABC-1 [19], GHS-ABC [21], and Wong-ABC-2 [26] are used to solve the resource allocation with the fairness threshold constraint ($F - 0.96 \geq 0$). Note that Shen [10], Wong-ABC-1 [19], and Wong-ABC-2 [26] do not have the parameter of the fairness threshold ε, which means that Shen [10], Wong-ABC-1 [19], and Wong-ABC-2 [26] cannot improve the system sum data rates flexibly with the required fairness threshold.

Both the fairness and the sum data rates of various algorithms at different user numbers K and different data rate proportional coefficients $\lambda_1 : \lambda_2 : \cdots : \lambda_K$ are shown in Figures 4–6. It can be seen from Figures 4–6 that, because of the increase of λ_1 of the data rate proportional coefficients, subcarrier resources are becoming more and more insufficient relative to the requirements of users, so that the sum data rates obtained by various algorithms are becoming lower and lower. However, with the increase of λ_1, the advantage of the proposed ABC-UQ over other algorithms in sum data rates becomes more and more obvious in the premise of the fairness threshold. Although the proposed ABC-UQ obtains lower sum data rates than the GHS-ABC [21] in some cases at $\lambda_1 = 1$, the proposed ABC-UQ obtains larger sum data rates than the GHS-ABC [21] at $\lambda_1 = 8$ and 1.

Because PSO-EQ [14] combines the fairness threshold-based greedy subcarrier allocation and PSO-based power allocation, it can guarantee the required fairness threshold for the resource allocation. However, the premature PSO leads to the sum data rates lower than other algorithms.

At $\lambda_1 = 1$ where subcarrier resources are relatively sufficient, ABC-OSA [12] and GA [13] can guarantee the fairness threshold at the expense of reducing the sum data rates. However, at $\lambda_1 = 8$ and 16 where subcarrier resources are relatively insufficient, ABC-OSA [12] and GA [13] can neither guarantee the fairness threshold nor improve the sum data rates. The main reason is that both ABC-OSA [12] and GA [13] do not take more effective measures for subcarrier allocation.

Simulation results in Figures 4–6 show that, compared with the existing algorithms [12–14, 21] using the fairness threshold for enhancing the system sum data rates, the proposed algorithm is more efficient in improving the system sum data rates in the premise of fairness threshold just in the equal-power subcarrier allocation stage. Due to the lack of the fairness threshold ε, the algorithms including Shen [10], Wong-ABC-1 [19], and Wong-ABC-2 [26] can achieve higher fairness but cannot effectively improve the subcarrier utilization and the sum data rates, which suggests that Shen [10], Wong-ABC-1 [19], and Wong-ABC-2 [26] are inefficient in improving the utilization of spectrum resources.

In order to further reveal differences between the proposed ABC-UQ and other algorithms, iterations of the fairness and the sum data rates of optimal solutions at $K = 16$, $\varepsilon = 0.96$, and $\lambda_1 : \lambda_2 : \cdots : \lambda_K = 16 : 1 : \cdots : 1$ are plotted in Figure 7. As can be seen from Figures 7(c)–7(f), at the beginning of iterations, the fairness level of the optimal solutions of both ABC-OSA [12] and PSO-EQ [14] is far lower than the

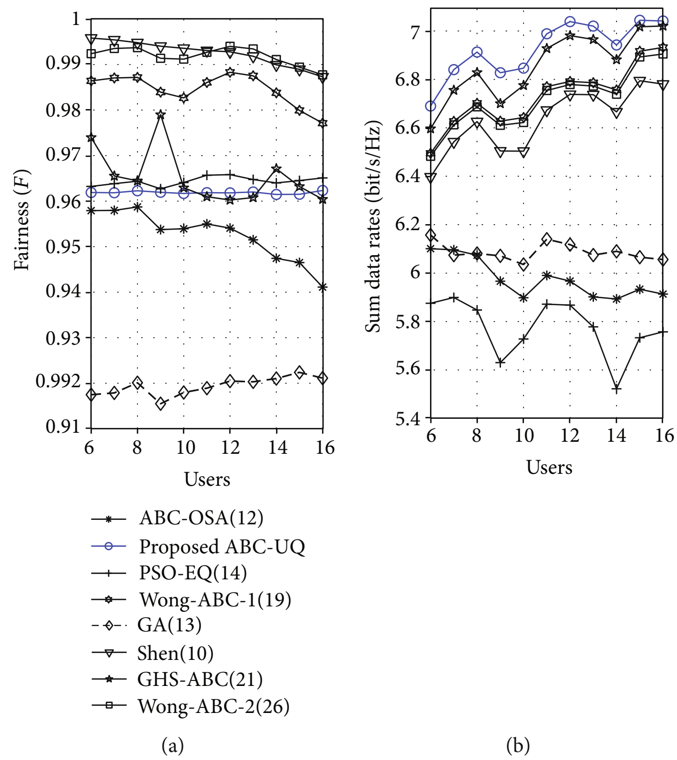

Figure 5: Fairness and sum data rates of various algorithms at the fairness threshold $\varepsilon = 0.96$ and $\lambda_1 : \lambda_2 : \cdots : \lambda_K = 8 : 1 : \cdots : 1$: (a) fairness; (b) sum data rates.

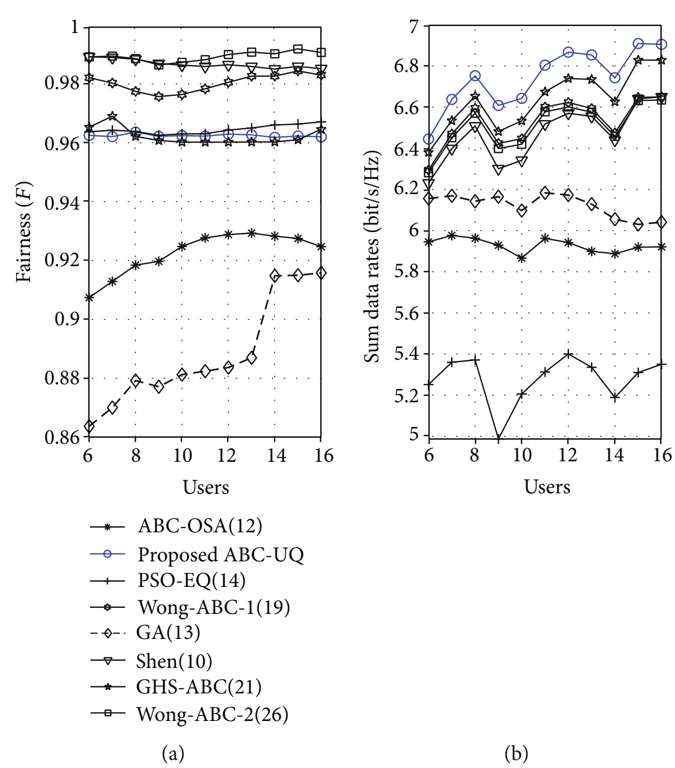

Figure 6: Fairness and sum data rates of various algorithms at the fairness threshold $\varepsilon = 0.96$ and $\lambda_1 : \lambda_2 : \cdots : \lambda_K = 16 : 1 : \cdots : 1$: (a) fairness and (b) sum data rates.

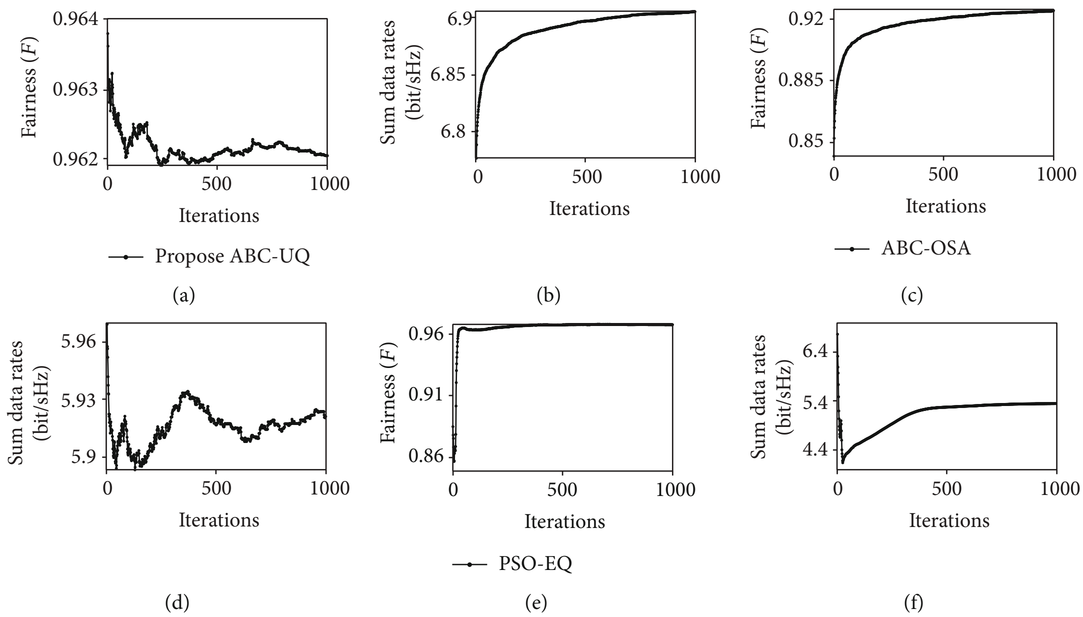

Figure 7: Iterations of fairness and sum data rates of optimal solutions obtained by the proposed ABC-UQ, ABC-OSA, and PSO-EQ at $K = 16$, $\varepsilon = 0.96$, and $\lambda_1 : \lambda_2 : \cdots : \lambda_K = 16 : 1 : \cdots : 1$: (a) iterations of fairness obtained by ABC-UQ, (b) iterations of sum data rates obtained by ABC-UQ, (c) iterations of fairness obtained by ABC-OSA, (d) iterations of sum data rates obtained by ABC-OSA, (e) iterations of fairness obtained by PSO-EQ, and (f) iterations of sum data rates obtained by PSO-EQ.

required fairness threshold, and then, the fairness level of the optimal solutions increases with the iterations increasing, which results in the sum data rates decreasing first and then increasing. Since PSO-EQ [14] can achieve the required fairness threshold in Figure 7(e), the sum data rates can be optimized by PSO-EQ [14] so that the sum data rates increase steadily. However, due to the premature PSO, the increasing of the sum data rates becomes stagnated at about 500 iterations in Figure 7(f). Since ABC-OSA [12] cannot achieve the required fairness threshold in Figure 7(c), it can only optimize the fairness as much as possible while not optimizing the sum data rates at all so that the sum data rates become unstable in Figure 7(d). Different from ABC-OSA [12] and PSO-EQ [14], the proposed ABC-UQ effectively combines the greedy subcarrier allocation method based on the data rate proportional fairness with ABC, so that the proposed ABC-UQ can make the fairness achieve the required fairness threshold at the beginning of iterations, as shown in Figure 7(a). As a result, the sum data rates optimized by the proposed ABC-UQ increase steadily in Figure 7(b).

Simulations and analyses in Subsections 4.2 and 4.3 show that the proposed ABC-UQ can not only guarantee the fairness threshold but also effectively improve the subcarrier utilization and the sum data rates.

4.3. Simulations on Optimal Update Quantity to Ensure Fairness Threshold. In this subsection, relationships between the update quantities of the optimal solutions and the required fairness thresholds are analyzed by simulations. In the following simulations, the population of nectar sources is set as $S = 60$, and the number of groups for the population is set as $S' = 30$, and each group contains 2 nectar sources. The corresponding update quantities for the groups are set as $1, 2, \ldots, 30$, respectively. In addition, the required fairness thresholds vary from 0.86 to 0.98, and the rest parameters remain unchanged. As the number of users K is equal to 6 and 16, the relationships between the average update quantities of the optimal solutions and the required fairness thresholds are plotted in Figures 8 and 9.

Figures 8 and 9 suggest that whether the number of users K is equal to 6 or 16, the average update quantities of the optimal solutions decrease gradually with the increase of the fairness thresholds. The main reason can be described as follows. According to Section 3, as the update quantity is equal to $N_{\bar{s}}$, there are $N - N_{\bar{s}}$ elements of the nectar sources generated by the greedy subcarrier allocation based on the data rate proportional fairness. Since the greedy subcarrier allocation based on the data rate proportional fairness can maintain high fairness, the fairness is more likely to be high as $N - N_{\bar{s}}$ is large. That is, the fairness is more likely to be high as the update quantity $N_{\bar{s}}$ decreases. Therefore, as the required fairness threshold increases, the optimal solutions tend to have lower update quantities.

Besides the fairness thresholds, both λ_1 of the data rate proportional coefficients and the number of users K have effects on the average update quantities of the optimal solutions. On the one hand, as can be seen from Figures 8(a)–8(c) and 9(a)–9(c), as λ_1 of the data rate proportional coefficients varies from 1 to 16, the average update quantities of the optimal solutions gradually decrease for both $K = 6$ and 16. On the other hand, comparing Figure 8(a) with Figure 9(a), Figure 8(b) with Figure 9(b), and Figure 8(c) with Figure 9(c), it can be found out that when the number of users K varies from 6 to 16, the average update quantities of the optimal solutions at the corresponding data rate proportional coefficients also gradually decrease. The main

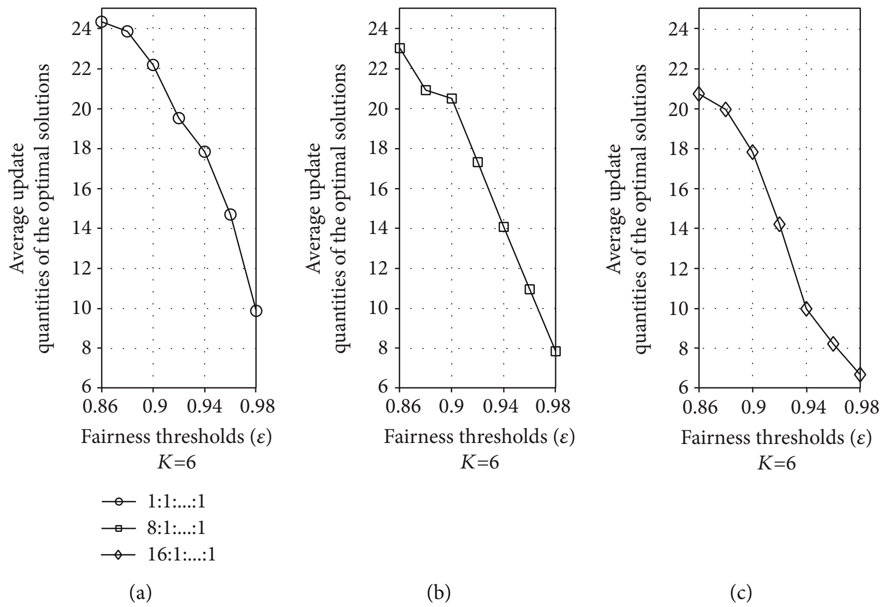

Figure 8: Relationships between the average update quantities of the optimal solutions and the fairness thresholds at the number of users $K = 6$ and different data rate proportional coefficients with (a) 1:1:…:1, (b) 8:1:…:1, and (c) 16:1:…:1.

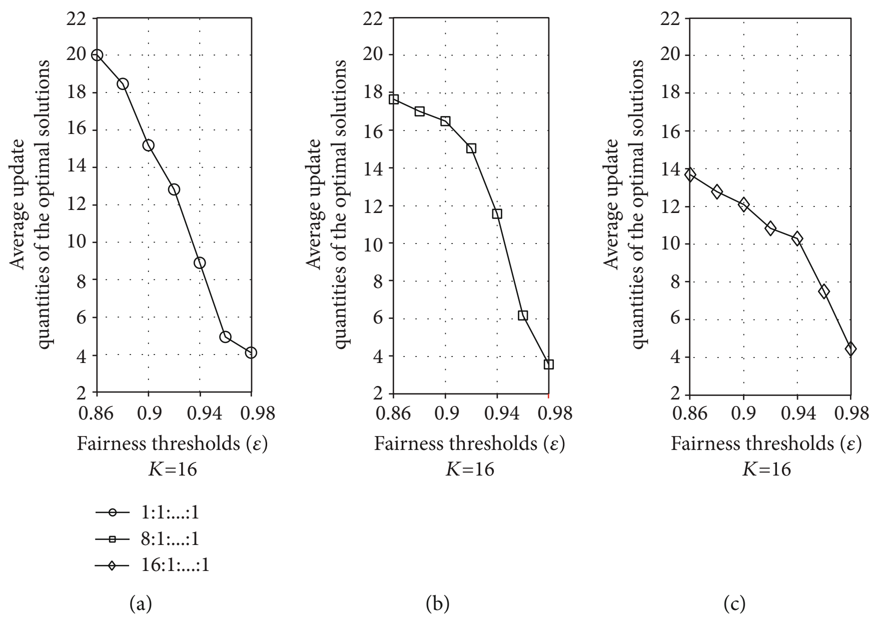

Figure 9: Relationships between the average update quantities of the optimal solutions and the fairness thresholds at the number of users $K = 16$ and different data rate proportional coefficients with (a) 1:1:…:1, (b) 8:1:…:1, and (c) 16:1:…:1.

reason is that, whether λ_1 of the data rate proportional coefficients varies from 1 to 16 or the number of users K varies from 6 to 16, both of them make the subcarrier resources relatively insufficient in allocation to users, so that the proposed ABC-UQ has to reduce the update quantities to ensure the required fairness threshold in the process of searching the optimal solution.

5. Conclusions

This paper proposes a novel ABC-UQ for resource allocation to improve the sum data rates effectively in the premise of fairness threshold. First, the proposed ABC-UQ divides the population of nectar sources into several groups and sets a different update quantity for each group. Then, based on the update quantities set for the groups, the proposed ABC-UQ applies the greedy subcarrier allocation method based on the data rate proportional fairness for initialization of nectar sources of ABC and performs neighborhood searches and updates on the corresponding dimensions of nectar sources. By combining the advantages of the greedy subcarrier allocation in local optimization and ABC in global optimization, the proposed ABC-UQ can not only make nectar sources maintain a higher fairness level but also reduce dimensions to be optimized in nectar sources. Results of simulations on different fairness thresholds indicate that the proposed ABC-UQ can achieve the required fairness thresholds and effectively improve the sum data rates just in the equal-power subcarrier allocation stage. In addition, small update quantities are more conducive to maintaining a high degree of fairness than large update quantities. As the data rate proportional coefficients or the number of users makes the subcarrier resources relatively insufficient in allocation, the proposed ABC-UQ with smaller update quantities can guarantee the required fairness thresholds and improve the subcarrier utilization and the sum data rates.

Data Availability

The data used in the experiments are generated randomly with Rayleigh distribution. All the data included in this study are available from the corresponding author (Ming Sun, e-mail: snogisunming@163.com) upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 61872204 and 71803095, the National Statistical Science Research Project 2020LY074, the Joint guiding project of Natural Science Foundation of Heilongjiang Province under Grant LH2019F038, the Young Innovative Talents Program of Basic Business Special Project of Heilongjiang Provincial Education Department under Grant 135309340, and the Science and Technology Project of Qiqihar under Grant GYGG-201915.

References

[1] J. Yu, S. Han, and X. Li, "A robust game-based algorithm for downlink joint resource allocation in hierarchical OFDMA femtocell network system," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 7, pp. 2445–2455, 2020.

[2] J. M. Alostad, "Design of power and resource management in OFDMA networks using sleep mode selection technique," *Computer Networks*, vol. 180, article 107411, 2020.

[3] Z. Mohammadian, M. J. Dehghani, and M. Eslami, "Efficient resource allocation algorithms for high energy efficiency with fairness among users in OFDMA networks," *Engineering Science and Technology, an International Journal*, vol. 23, no. 5, pp. 982–988, 2020.

[4] T.-T. Nguyen, Q. V. Pham, V. D. Nguyen, J. H. Lee, and Y. H. Kim, "Resource allocation for energy efficiency in OFDMA-enabled WPCN," *IEEE Wireless Communications Letters*, vol. 9, no. 12, pp. 2049–2053, 2020.

[5] Z. Yang, W. Xu, and Y. Li, "Fair non-orthogonal multiple access for visible light communication downlinks," *IEEE Wireless Communications Letters*, vol. 6, no. 1, pp. 66–69, 2017.

[6] D. Qin and T. Zhou, "Joint resource optimization for orthogonal frequency division multiplexing based cognitive amplify and forward relaying networks," *Sensors*, vol. 20, no. 7, article 2074, 2020.

[7] Z. Yin, S. Zhuang, Z. Wu, and B. Ma, "Rate adaptive based resource allocation with proportional fairness constraints in OFDMA systems," *Sensors*, vol. 15, no. 10, pp. 24996–25014, 2015.

[8] M. Sun, K. Y. Lee, Y. Xu, and W. Bai, "Hysteretic noisy chaotic neural networks for resource allocation in OFDMA system," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 2, pp. 273–285, 2018.

[9] J.-H. Ro, W. S. Lee, M. G. Kang, D. K. Hong, and H. K. Song, "A strategy of signal detection for performance improvement in clipping based ofdm system," *Computers, Materials & Continua*, vol. 64, no. 1, pp. 181–191, 2020.

[10] D. Singh Kapoor and A. Kumar Kohli, "Intelligence-based channel equalization for 4x1 SFBC-OFDM receiver," *Intelligent Automation & Soft Computing*, vol. 26, no. 3, pp. 439–446, 2020.

[11] Zukang Shen, J. G. Andrews, and B. L. Evans, "Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints," *IEEE Transactions on Wireless Communications*, vol. 4, no. 6, pp. 2726–2737, 2005.

[12] N. Sharma and A. Anpalagan, "Bee colony optimization aided adaptive resource allocation in OFDMA systems with proportional rate constraints," *Wireless Networks*, vol. 20, no. 7, pp. 1699–1713, 2014.

[13] N. Sharma and A. S. Madhukumar, "Genetic algorithm aided proportional fair resource allocation in multicast OFDM systems," *IEEE Transactions on Broadcasting*, vol. 61, no. 1, pp. 16–29, 2015.

[14] C. Zhang and X. Zhao, "Adaptive resource allocation in multiuser OFDM system based on fairness threshold," *Journal of Communication*, vol. 32, no. 12, pp. 65–71, 2011.

[15] I. C. Wong, Z. Shen, B. L. Evants, and J. G. Andrews, "A low complexity algorithm for proportional resource allocation in OFDMA systems," in *IEEE Workshop on Signal Processing Systems, 2004. SIPS 2004*, pp. 1–6, Austin, TX, USA, October 2004.

[16] S. Sritharan, H. Weligampola, and H. Gacanin, "A study on deep learning for latency constraint applications in beyond 5G wireless systems," *IEEE Access*, vol. 8, pp. 218037–218061, 2020.

[17] S.-M. Tseng, Y. F. Chen, C. S. Tsai, and W. D. Tsai, "Deep-learning-aided cross-layer resource allocation of OFDMA/-NOMA video communication systems," *IEEE Access*, vol. 7, pp. 157730–157740, 2019.

[18] L. Jiang and R. Song, "A low-complexity resource allocation scheme for OFDMA multicast systems with proportional fairness," *China Communications*, vol. 15, no. 1, pp. 1–11, 2018.

[19] J. Yuan, F. Zhang, J. Wang, Y. Wang, J. Lin, and Y. Pang, "OFDMA adaptive resource allocation based on fairness and penalty function," *Systems Engineering and Electronics*, vol. 40, no. 2, pp. 427–434, 2018.

[20] Y. Shen, Y. Shi, J. Zhang, and K. B. Letaief, "LORM: learning to optimize for resource management in wireless networks with few training samples," *IEEE Transactions on Wireless Communications*, vol. 19, no. 1, pp. 665–679, 2020.

[21] M. Sun, K. Zhai, W. Cao, Y. Wang, and Y. Xu, "Hybrid OFDMA resource allocation scheme for ensuring required level of proportional fairness," *Mathematical Problems in Engineering*, vol. 2020, Article ID 5201545, 19 pages, 2020.

[22] P. S. Swapna and S. S. Pillai, "Low complexity resource allocation techniques for symmetrical services in OFDMA systems," *Wireless Personal Communications*, vol. 116, no. 4, pp. 3217–3234, 2021.

[23] Y. Shen, X. Huang, B. Yang, and S. Wang, "Low complexity resource allocation algorithms for chunk based OFDMA multi-user networks with max-min fairness," *Computer Communications*, vol. 163, pp. 51–64, 2020.

[24] C. K. Tan, T. C. Chuah, and S. Tan, "A coalitional game-based algorithm for OFDMA resource allocation in multicast cognitive radio networks," *Wireless Personal Communications*, vol. 80, no. 1, pp. 415–427, 2015.

[25] C. Liao, J. Wu, J. Du, and L. Zhao, "Ant colony optimization inspired resource allocation for multiuser multicarrier systems," in *2017 9th International Conference on Wireless Communications and Signal Processing (WCSP)*, pp. 2472–7628, Nanjing, China, October 2017.

[26] J. Yuan, J. Wang, F. Zhang, Y. Wang, J. Lin, and Y. Pang, "Adaptive resource allocation for multiuser OFDM based on artificial bee colony algorithm," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 49, no. 2, pp. 624–630, 2019.

[27] S.-M. Tseng, C. S. Tsai, and C. Y. Yu, "Outage-capacity-based cross layer resource management for downlink NOMA-OFDMA video communications: non-deep learning and deep learning approaches," *IEEE Access*, vol. 8, pp. 140097–140107, 2020.

[28] S. Khakurel, L. Musavian, H. V. Vu, and T. Le-Ngoc, "QoS-aware utility-based resource allocation in mixed-traffic multi-user OFDM systems," *IEEE Access*, vol. 6, pp. 21646–21657, 2018.

[29] P. Liu, X. Liu, Y. Luo, Y. Du, Y. Fan, and H. Feng, "An enhanced exploitation artificial bee colony algorithm in automatic functional approximations," *Intelligent Automation & Soft Computing*, vol. 25, no. 2, pp. 385–394, 2019.

[30] M. Jayalakshmi and S. Nagaraja Rao, "Discrete wavelet transmission and modified PSO with ACO based feed forward neural network model for brain tumour detection," *Computers, Materials & Continua*, vol. 65, no. 2, pp. 1081–1096, 2020.

[31] D. Karaboga and B. Akay, "A modified artificial bee colony (ABC) algorithm for constrained optimization problems," *Applied Soft Computing*, vol. 11, no. 3, pp. 3021–3031, 2011.

[32] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics & Engineering*, vol. 186, no. 2-4, pp. 311–338, 2000.