Hindawi
Mobile Information Systems
Volume 2019, Article ID 7575037, 12 pages
https://doi.org/10.1155/2019/7575037

Research Article

WiFi Offloading Algorithm Based on Q-Learning and MADM in Heterogeneous Networks

Lin Sun [1,2] **and Qi Zhu** [1,2]

[1] *Jiangsu Key Laboratory of Wireless Communications, Nanjing University of Posts and Telecommunications, Nanjing 210003, China*
[2] *Engineering Research Center of Health Service System Based on Ubiquitous Wireless Networks, Nanjing University of Posts and Telecommunications, Ministry of Education, Nanjing, China*

Correspondence should be addressed to Qi Zhu; zhuqi@njupt.edu.cn

Received 28 March 2019; Revised 18 November 2019; Accepted 10 December 2019; Published 27 December 2019

Academic Editor: Alessandro Bazzi

This paper proposes a WiFi offloading algorithm based on Q-learning and MADM (multiattribute decision making) in heterogeneous networks for a mobile user scenario where cellular networks and WiFi networks coexist. The Markov model is used to describe the changes of the network environment. Four attributes including user throughput, terminal power consumption, user cost, and communication delay are considered to define the user satisfaction function reflecting QoS (Quality of Service), and Q-learning is used to optimize it. Through AHP (Analytic Hierarchy Process) and TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution) in MADM, the intrinsic connection between each attribute and the reward function is obtained. The user uses Q-learning to make offloading decisions based on current network conditions and their own offloading history, ultimately maximizing their satisfaction. The simulation results show that the user satisfaction of the proposed algorithm is better than the traditional WiFi offloading algorithm.

1. Introduction

With the popularity of smart devices, cellular data traffic is growing at an unprecedented rate. Cisco visual network index [1] predicts that global mobile data traffic will reach 49 exabytes per month in 2021, which is equivalent to six times that of 2016. In order to solve the problem of data traffic explosion, we can add cellular BS (base station) or upgrade the cellular network into networks such as LTE (long-term evolution), LTE-A (LTE-Advanced), and WiMAX release 2 (IEEE 802.16m), but this is usually not economical, which requires expensive CAPEX (capital expenditure) and OPEX (operating expense) [2]. In addition, the limited licensed band is another bottleneck to improve network capacity [3]. As a result, mobile data offloading technology [4] has gradually become a mainstream in 5G, and WiFi offloading is one of the most effective offloading solutions.

WiFi offloading technology transfers part of the cellular network load to WiFi network through the WiFi AP (access point), by which we can solve the congestion in licensed band, achieve load balancing, and fully utilize unlicensed spectrum resources. Due to the effectiveness of WiFi offloading, many literatures have studied it. Li et al. [5] considered the coexistence of WiFi and LTE-U on unlicensed bands and offloaded LTE-U services to WiFi networks, establishing multiple targets for maximizing LTE-U user throughput while optimizing WiFi user throughput. To solve the problem, the authors used the Pareto optimization algorithm to get the optimal value. In [6], a satisfaction function reflecting the user communication rate is defined in the scenario of overlapping WiFi network and cellular network, and a resource block allocation matrix is constructed. Based on the accurate potential game theory, the best response algorithm is used to optimize the total system satisfaction. Cai et al. [7] proposed an incentive mechanism to compensate cellular users who are willing to delay their traffic for WiFi offloading. The authors calculated the optimal compensation value according to the available

attribute parameters in the scenario and modeled the problem into two stages. In the first stage of the Stackelberg game, the operator announces that it would provide users with uniform compensation to delay its cellular services. In the second phase, each user decides whether to join the delayed offloading based on the compensation, network congestion, and estimation of the waiting cost for WiFi connection. From the perspective of operators, Kang et al. [8] formulated mobile data offloading problem as a utility maximization problem. The authors established an integer programming problem and obtained a mobile data offloading scheme by considering the relaxed condition. The authors further proved that when the number of users is large, the proposed centralized data offloading scheme is near optimal. Jung et al. [9] proposed a user-centric, network-assisted WiFi offloading model. In this model, heterogeneous networks are responsible for collecting network information and users make offloading decisions based on this information to maximize their throughput. In the heterogeneous network scenario composed of LTE and WiFi, aiming at maximizing the minimum energy efficiency of users, a closed expression is proposed in [10] to calculate the number of users to be offloaded, and these users with the smallest SINR (signal to interference and noise ratio) are offloaded into WiFi network. According to the above references, the most challenging problem in WiFi offloading is how to make an offloading decision, that is, how to choose the most suitable WiFi AP for communication. Fakhfakh and Hamouda [11] aimed to minimize the residence time of the cellular network and optimized it by Q-learning. The reward function considers SINR, handover delay, and AP load. By offloading cellular services to the best WiFi AP nearby, operators can greatly increase their network capacity, and users' QoS will also increase. However, the above references only make an immediate offloading decision based on the current network conditions, without considering the user's previous access history. In addition, most of the references only perform an offloading decision for the optimization of one particular attribute, such as throughput or energy efficiency, without considering multiple network attributes for comprehensive decision making.

In this paper, for the mobile user scenario where the cellular base station and the WiFi AP coexist, considering the current network conditions and the access history, a Q-learning scheme is used to make the offloading decision. By considering its own access history, users will accumulate the experience of offloading, which will not only avoid offloading to the poor network that was previously accessed but also actively select the best WiFi AP according to the maximum discounted cumulative reward, which in turn increases user's QoS. In this paper, four attributes including user throughput, terminal power consumption, user cost, and communication delay are considered and the reward function in Q-learning is defined by TOPSIS. In addition, if the service type is different, the importance of each network attribute will be different. We use AHP to define the weight of each network attribute according to the specific service type. The mobile terminal collects various attributes of the heterogeneous network, and the user continuously updates his discounted cumulative reward in combination with the instant reward and the experience reward until convergence. After the convergence, the user can make the best offloading decision in each state.

The rest of this paper is arranged as follows. Section 2 gives the system model of WiFi offloading in heterogeneous networks. Section 3 builds the Q-learning model, defines the reward function model based on AHP and TOPSIS, and gives the specific steps of the WiFi offloading algorithm. In Section 4, the simulation results are presented and analysed. Finally, Section 5 concludes the paper.

2. System Model

The system model in this paper is shown in Figure 1. A cellular base station is located in the center of the cell with a radius equal to r_{cell}. There are N_{AP} WiFi APs in the cell, which are represented as $\text{AP}_k, k \in \{1, 2, \ldots, N_{\text{AP}}\}$. The cell is covered by overlapping cellular network and WiFi network. These networks are divided into valid networks and invalid networks. When the throughput of the user accessing a certain network is greater than a threshold, we regard this network as a valid network; otherwise, it is considered as an invalid network. The mobile multimode terminal is the agent of Q-learning, and it can perform data transmission through both cellular network and WiFi network. The agent moves straightly inside the cell, marking its passing position as $\text{Posi}_i, i \in \{1, 2, \ldots, N_{\text{p}}\}$, where N_{p} represents the total number of positions the user has passed. Due to the movement of the agent, the network environment such as channel quality and available bandwidth is constantly changing, which will cause the network attribute of the user to change. This paper regards the four network attributes of the agent in different locations as the state in Q-learning, including throughput, power consumption, cost, and delay. In addition, we consider the offloading decision as the action choice in Q-learning and offload mobile data if agent chooses WiFi network.

Figure 2 shows the algorithm structure based on Q-learning. The agent first collects the network environment information, filters out invalid networks, and calculates four attributes of user throughput (TP), terminal power consumption (PC), user cost (C), and communication delay (D) of the valid network. The AHP algorithm is used to calculate the weights of the four attributes under different services, and the instant rewards obtained by selecting each network under the current state are calculated by TOPSIS. In combination with the instant reward and the experience reward, the Q-learning iteration is performed and the Q-table is updated. As a result, the offloading decision is made based on the discounted cumulative reward in Q-table.

This paper reflects the performance of the network from four aspects of throughput, power consumption, cost, and delay. The throughput reflects the rate of wireless transmission. According to the large-scale fading model of the wireless channel in [12], combined with the small-scale fading model, when the distance between the agent and the cellular BS or WiFi AP is d, the pass loss is defined as

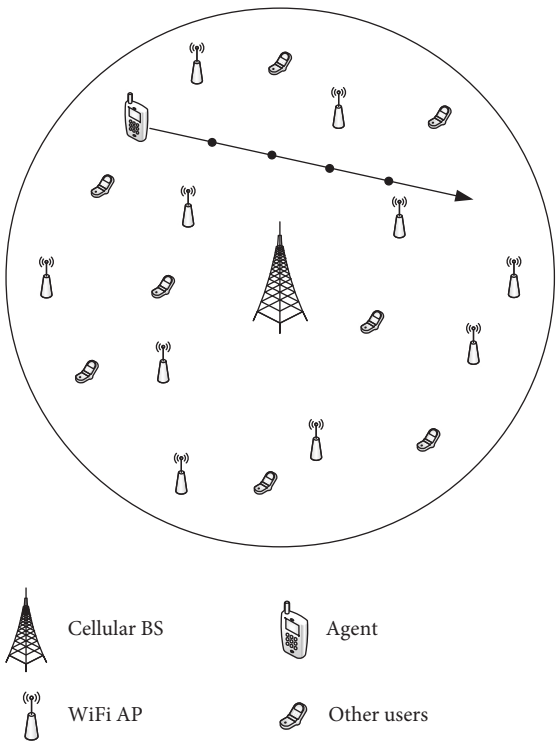

Figure 1: System model of WiFi offloading. The system model consists of a cellular BS, a few WiFi APs, one moving agent, and some other users.

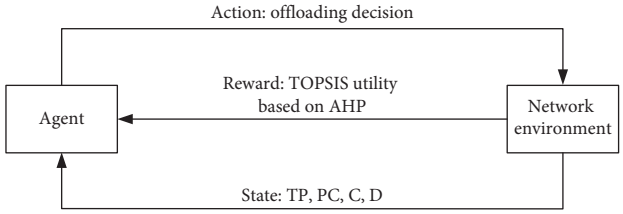

Figure 2: Algorithm structure based on Q-learning.

$$L = L_0 + 10\alpha \log_{10}\left(\frac{d}{d_0}\right) + F_{\text{Rayleigh}}(\theta, \beta), \tag{1}$$

where d_0 is the reference distance, L_0 is the path loss when the distance between agent and BS or AP is d_0, α is the path loss exponent, and $F_{\text{Rayleigh}}(\theta, \beta)$ is the Rayleigh fading of the Gaussian distribution with a mean of θ and a variance of β. The signal power P_i^r received at BS or AP from agent d away at the i–th position is expressed as

$$P_i^r = P_i^t - L, \tag{2}$$

where P_i^t is the transmit power of the terminal which is not fixed. By the Shannon capacity formula [13], we can get the throughput of the agent accessing a network at the i–th position:

$$V_i^{\text{TP}} = W \times \log_2\left(1 + \frac{P_i^r}{N_0 \times W}\right), \tag{3}$$

where N_0 is the additive white Gaussian noise power spectral density and $W$is the available bandwidth of the agent. Since the available bandwidth of the network is constantly changing and each AP or BS provides services to other users in addition to the agent at the same time, which affects the available network bandwidth of the agent, this paper uses the Markov model to describe the change of W and quantizes the continuous W into N_{markov} states. The available bandwidth is transferred to the two adjacent states with the probability of p_{tr} or remains unchanged with the probability of $1 - p_{\text{tr}}$.

Power consumption is an important attribute to be considered for the operation of mobile terminals. According to [14], it is assumed that the minimum received power threshold of BS or AP is $P_{\min}^r$. When the transmit power of the terminal is too small, BS and AP will not receive the uplink signal of the terminal. To ensure the normal transmission of data, we define the minimum transmit power $P_{\min}^t$of the terminal as

$$P_{\min}^t = P_{\min}^r + L. \tag{4}$$

The actual transmit power P_i^t of the terminal must be greater than $P_{\min}^t$. In this paper, the power consumption of the agent accessing a network at the i–th location is expressed as

$$V_i^{\text{PC}} = P_0 + P_i^t, \tag{5}$$

where P_0 is the fixed operating power consumption of the terminal and P_i^t is the transmitting power of the terminal.

The operator charges the agent whether he accesses the cellular BS or a WiFi AP. In this paper, the unit price costed per second after the agent accesses a network in i–this defined as V_i^C, which is used to represent a relative price of two networks. It is usually cheaper if the user chooses to offload.

Communication delay is also an important indicator for users to evaluate the network. In this paper, the transmission delay after the agent accesses a network in i–th location is defined as V_i^D. Because of CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), the delay time is longer when the user accesses WiFi, which makes V_i^D bigger than accessing the BS.

This paper considers the above four network attributes to calculate the satisfaction Φ_j^{sat} of the agent in the whole mobile scenario.

Firstly, we calculate the average of the four network attributes at N_{p} locations; that is, $V_{\text{ave}}^{\text{TP}} = \sum_i V_i^{\text{TP}}/N_{\text{p}}$, $V_{\text{ave}}^{\text{PC}} = \sum_i V_i^{\text{PC}}/N_{\text{p}}$, $V_{\text{ave}}^{C} = \sum_i V_i^{C}/N_{p}$, and $V_{\text{ave}}^{D} = \sum_i V_i^{D}/N_{\text{p}}$.

Then, we normalize the four values using the method in [15]:

$$u = \begin{cases} \dfrac{U - U_{\min}}{U_{\max} - U_{\min}}, & \text{when } U \text{ is a positive attribute,} \\ \dfrac{U_{\max} - U}{U_{\max} - U_{\min}}, & \text{when } U \text{ is a negative attribute,} \end{cases} \tag{6}$$

where $U_{\max}$ is the maximum possible value of the attribute and $U_{\min}$ is the minimum possible value of the attribute. For user satisfaction, the greater the throughput is, the better satisfaction the agent gets, which is a positive attribute. On the other hand, the other three attributes are kept as small as possible, belonging to the negative attribute. The normalized values of the four network attributes are expressed as $V_{\text{ave}}^{\text{tp}} = (V_{\text{ave}}^{\text{TP}} - V_{\min}^{\text{TP}})/(V_{\max}^{\text{TP}} - V_{\min}^{\text{TP}})$, $V_{\text{ave}}^{\text{pc}} = (V_{\max}^{\text{PC}} - V_{ave}^{\text{PC}})/(V_{\max}^{\text{PC}} - V_{\min}^{\text{PC}})$, $V_{\text{ave}}^{c} = (V_{\max}^{\text{C}} - V_{\text{ave}}^{C})/(V_{\max}^{\text{C}} - V_{\min}^{C})$, and $V_{\text{ave}}^{d} = (V_{\max}^{D} - V_{\text{ave}}^{D})/(V_{\max}^{D} - V_{\min}^{D})$.

Combining the attribute weight data of different services obtained by using the AHP algorithm, the satisfaction of the user in the entire mobile scenario is defined as the sum of the weighted normalized attribute values:

$$\begin{aligned}\Phi_j^{\text{sat}} = {} & w_j^{\text{tp}} \times V_{\text{ave}}^{\text{tp}} + w_j^{\text{pc}} \times V_{\text{ave}}^{\text{pc}} + w_j^{c} \times V_{\text{ave}}^{c} \\ & + w_j^{d} \times V_{\text{ave}}^{d}, \quad j \in \{1, 2\},\end{aligned} \tag{7}$$

where j is the user service type, $j = 1$ is the streaming media service, $j = 2$ is the conversation service, and $w_j^{\text{tp}}, w_j^{\text{pc}}, w_j^{c}$, and w_j^{d} are the AHP weights of the throughput, power consumption, cost, and delay when the service type is j.

The optimization goal of this paper is to find out the best offloading decision of the user to maximize the satisfaction of the entire mobile scenario:

$$\begin{aligned}\prod^{*} = {} & \arg\max_{\Pi \in \Omega} \left(\Phi_j^{\text{sat}}\right) \\ \text{s.t.} \quad & c1\text{: } 0 < w_j^h < 1 \; h \in \{\text{tp}, \text{pc}, c, d\} \\ & c2\text{: } \sum_h w_j^h = 1 \; h \in \{\text{tp}, \text{pc}, c, d\} \\ & c3\text{: } P_i^t > P_{\min}^t \; i \in \left\{1, 2, \ldots, N_p\right\},\end{aligned} \tag{8}$$

where $\Omega = A_1 \otimes A_2 \otimes \cdots \otimes A_{N_{\text{p}}}$ is the total action space of the user during the whole movement process in which A_i is the action set when the agent passes position i. It is the Cartesian product of the action set of the user passing N_{p} positions, and Π^* is the optimal offloading strategy of the whole moving process. In equation (8), $c1$ and $c2$ indicate that the weight of each network attribute is limited to 0 to 1 and the sum is 1; $c3$ indicates that the user's transmit power is greater than the minimum transmit power at each position. However, because the action space is very large and the network environment such as available bandwidth is constantly changing, the traditional method is difficult to solve this optimization problem, so we use Q-learning to solve it.

3. WiFi Offloading Algorithm Based on Q-Learning and MADM

For the mobile user scenario where the cellular BS and the WiFi AP coexist, we propose a WiFi offloading algorithm based on Q-learning and MADM. Considering the current network conditions and the access history, the Q-learning algorithm is used to make the offloading decision, which will not only avoid offloading to the poor network that was previously accessed but also actively select the best WiFi AP according to the maximum discounted cumulative reward. MADM is an effective decision-making method when we need to consider a variety of factors. According to [16], attribute weight and network utility value are of great importance in MADM. We use two MADM algorithms in this paper, called AHP and TOPSIS. AHP is used to define the weight of each network attribute according to the specific service type. TOPSIS is used to obtain the instant reward of Q-learning based on the network utility. The agent collects various attributes of the heterogeneous network and continuously updates his discounted cumulative reward in combination with the instant reward and the experience reward. After the convergence, the user can make the best offloading decision in each state.

3.1. Q-Learning. Q-learning is one of the widely used reinforcement learning algorithms that treat learning as a process of trying, evaluation, and feedback. Q-learning consists of three elements, including state, action, and reward. The state set is denoted as S and the action set is denoted as A, and the purpose of Q-learning is to obtain the optimal action selection strategy Π^* to maximize the agent's discounted cumulative reward [11]. In state $s \in S$, the agent selects an action $a \in A$ from the action set to act on the environment. After the environment accepts the action, the environment changes and generates an instant reward $\text{Rw}(s, a)$ feedback to the agent. Then, the agent will select the next action $a' \in A$ based on the reward and his own experience, which will in turn affect the discounted cumulative reward $\text{Rc}(s)$ and state s' of the next moment. It has been proved that for any given Markov decision process, Q-learning can be used to obtain an optimal action selection strategy Π^* for each state s, maximizing the discounted cumulative reward for each state [17].

The discounted cumulative reward $\text{Rc}(s)$ for state s is

$$\text{Rc}(s) = \text{Rw}(s, a) + \delta \sum_{s \in S} P\left(s' \mid s, a\right) \text{Rc}\left(s'\right), \tag{9}$$

where $\text{Rw}(s, a)$ is the instant reward obtained by the agent selecting action a in state s, $\delta \in (0, 1)$ is the discount factor, and $P(s' \mid s, a)$ is the probability when agent performs action a and transmits from state s to s'. According to Bellman's theory [18], when the discounted cumulative reward is maximum, the optimal action selection decision under state s can be obtained:

$$\text{Rc}(s)^{*} = \max\left[\text{Rw}(s, a) + \delta \sum_{s \in S} P\left(s' \mid s, a\right) \text{Rc}\left(s'\right)\right]. \tag{10}$$

The optimal action selection decision is

$$\pi^{*}(s) = \arg\max_{a \in A}\left[\text{Rw}(s, a) + \delta \sum_{s \in S} P\left(s' \mid s, a\right) \text{Rc}\left(s'\right)\right]. \tag{11}$$

Since $\text{Rw}(s, a)$ and $P(s' \mid s, a)$ are still unknown, the agent can learn these values during the Q-learning process of trial, evaluation, and feedback. We use Q function to

represent the discounted cumulative reward when agent selects a in state s:

$$Q(s,a) = \mathrm{Rw}(s,a) + \delta \sum_{s \in S} P\left(s' \mid s,a\right)\mathrm{Rc}\left(s'\right). \tag{12}$$

This paper uses Q-learning to solve the problem of WiFi offloading and proposes a WiFi offloading algorithm based on Q-learning and multiattribute decision making. In this paper, the multimode terminal moving inside the cell is regarded as the agent. The state, action, and reward of Q-learning are mapped in the following, respectively:

(1) State set S: the location that agent passes and the network environment around the location, that is, $S = \left\{s_i = (\mathrm{Posi}_i, \mathrm{Envi}_i) \mid i \in \left\{1, 2, \ldots, N_{\mathrm{p}}\right\}\right\}$, where Posi_i represents the location of the agent and Envi_i represents the network attributes of location i, including throughput, power consumption, cost, and delay

(2) Action set A: the process of selecting an action is regarded as an offloading decision, that is, $A = \{a_k, k \in \{0, 1, 2, \ldots, N_{\mathrm{AP}}\}\}$, where a_0 indicates that the terminal accesses the cellular BS and $a_k, k \in \{1, 2, \ldots, N_{\mathrm{AP}}\}$ indicates that the terminal is offloaded to the WiFi AP corresponding to the subscript

(3) Reward function $\mathrm{Rw}(s,a)$: the utility value of the TOPSIS algorithm is used to represent the instant reward that the user obtains after attempting to access a certain network

3.2. AHP Algorithm. This paper uses AHP to calculate the user's subjective assessment of the importance of each network attribute under different service types. AHP is one of the MADM algorithms using qualitative and quantitative calculations, which is widely used in network evaluation and strategy selection. According to [15], AHP has five steps: (1) establishing a hierarchical model; (2) constructing a paired comparison matrix; (3) calculating attribute weights; (4) checking consistency; and (5) selecting network. However, this paper only needs to use AHP to calculate the weight of different network attributes, so steps (1) and (5) are omitted. The specific steps are as follows:

Step 1: construct the paired comparison matrix according to the user service type j and the attributes to be analysed. Since this paper considers four attributes of throughput, power consumption, cost, and delay, the paired comparison matrix B can be expressed as

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ b_{31} & b_{32} & b_{33} & b_{34} \\ b_{41} & b_{42} & b_{43} & b_{44} \end{bmatrix}, \tag{13}$$

where b_{mn} represents the ratio of the importance degree between m and n network attributes. We assume b_{mn} as an integer from 1 to 9 or a reciprocal of them to evaluate the relative importance between different attributes. Furthermore, we have $b_{mn} = 1/b_{nm}$, and the value on the diagonal is 1.

Step 2: calculate the weight of each network attribute in the service type scenario. According to [19], B is a positive reciprocal matrix which has multiple eigenvalues and eigenvector pairs (λ, V):

$$B \times V = \lambda \times V, \tag{14}$$

where λ is a certain feature value of B and V is a feature vector corresponding to λ. The feature vector corresponding to the largest eigenvalue λ^* is selected and normalized into $[w_j^{\mathrm{tp}}, w_j^{\mathrm{pc}}, w_j^{c}, w_j^{d}]^{\mathrm{T}}$, which is also the AHP weight of the four attributes.

Step 3: check the consistency of the paired comparison matrix. Normally, the most accurate AHP weight cannot be obtained at one time because the paired comparison matrix may be inconsistent if $b_{mn} \neq b_{mk}/b_{kn}$, so the weight calculated in Step 2 is not accurate. It is necessary to check consistency of comparison matrix to ensure the subjective weight reasonable [15]. This paper uses the consistency ratio CR to measure the rationality of B:

$$\mathrm{CR} = \frac{(\lambda^* - N)}{N - 1}\frac{1}{\mathrm{RI}}, \tag{15}$$

where N is the number of network attributes and is the order of matrix B. RI is the index of average random consistency, and it is fixed if comparison matrix order is known [15], as is shown in Table 1.

According to the theory of AHP, if the consistency ratio $\mathrm{CR} > 0.1$, then B is unacceptable, and it is necessary to return to Step 1 to adjust B until $\mathrm{CR} > 0.1$. Finally, the accurate AHP weights of the four network attributes can be obtained (Table 1).

3.3. TOPSIS Algorithm. This paper uses TOPSIS to calculate the instant reward Rw obtained by the terminal accessing the cellular network or WiFi network. TOPSIS is also a MADM algorithm, the principle of which is to calculate and sort the proximity of candidate solutions to ideal solutions. In the Q-learning model, the action set contains all possible network choices; however, this is not a candidate network set because before the TOPSIS algorithm, this paper has filtered the invalid network whose actual throughput is less than the throughput threshold $V_{\mathrm{th}}^{\mathrm{TP}}$. So, we use TOPSIS to calculate the reward corresponding to the candidate network. Assume that the filtered candidate network set is $\{\mathrm{Net}_1, \ldots, \mathrm{Net}_l, \ldots, \mathrm{Net}_L\}$, which are the L valid actions extracted from the action set A, and the reward corresponding to the filtered invalid action is 0. The specific steps for calculating the Q-learning reward using the TOPSIS algorithm are as follows:

Table 1: Average random consistency with respect to matrix order.

Matrix order	RI
1	0.00
2	0.00
3	0.58
4	0.90
5	1.12
6	1.24
7	1.32
8	1.41
9	1.45

Step 1: establish a standardized decision matrix H. Constructing a candidate network attribute matrix X using the network attribute values calculated in Section 2:

$$X = \begin{bmatrix} V_{\text{Net}_1}^{\text{TP}} & V_{\text{Net}_1}^{\text{PC}} & V_{\text{Net}_1}^{C} & V_{\text{Net}_1}^{D} \\ \cdots & \cdots & \cdots & \cdots \\ V_{\text{Net}_l}^{\text{TP}} & V_{\text{Net}_l}^{\text{PC}} & V_{\text{Net}_l}^{C} & V_{\text{Net}_l}^{D} \\ \cdots & \cdots & \cdots & \cdots \\ V_{\text{Net}_L}^{\text{TP}} & V_{\text{Net}_L}^{\text{PC}} & V_{\text{Net}_L}^{C} & V_{\text{Net}_L}^{D} \end{bmatrix} = (x_{ln})_{L\times N}, \tag{16}$$

where l represents the number of the candidate network and n represents the number of the network attribute. Normalize each column to obtain a standardized decision matrix $H = (h_{ln})_{L\times N}$, where h_{ln} is the normalization of x_{ln}:

$$h_{ln} = \frac{x_{ln}}{\sum_{l\in\{1,2,\ldots,L\}} x_{ln}}. \tag{17}$$

Step 2: establish a weighted decision matrix Y. Each attribute is weighted by the AHP weight $[w_j^{tp}, w_j^{pc}, w_j^{c}, w_j^{d}]^{\text{T}}$ obtained in Section 3.2, which is represented by $[w_1, w_2, w_3, w_4]^{\text{T}}$, and the attribute value of each column in H is multiplied by the corresponding AHP weight to obtain $Y = (y_{ln})_{L\times N}$:

$$y_{ln} = w_n h_{ln}. \tag{18}$$

Step 3: calculate the proximity of each candidate solution and two extreme solutions. First, determine the ideal solution and the least ideal solution. Since throughput is a positive attribute and power consumption, cost, and delay are negative attributes, the ideal solution Solution^{+} is

$$\begin{aligned} \text{Solution}^{+} &= \left[\max_l y_{l1}, \min_l y_{l2}, \min_l y_{l3}, \min_l y_{l4}\right] \\ &= \left[y_1^{+}, y_2^{+}, y_3^{+}, y_4^{+}\right]. \end{aligned} \tag{19}$$

On the contrary, the least ideal solution is:

$$\begin{aligned} \text{Solution}^{-} &= \left[\min_l y_{l1}, \max_l y_{l2}, \max_l y_{l3}, \max_l y_{l4}\right] \\ &= \left[y_1^{-}, y_2^{-}, y_3^{-}, y_4^{-}\right]. \end{aligned} \tag{20}$$

Calculate the Euclidean distances between the l-th candidate network and Solution^{+} and Solution^{-} to get ED_l^{+} and ED_l^{-}:

$$\begin{aligned} \text{ED}_l^{+} &= \sqrt{\sum_{n\in\{1,2,3,4\}} \left(y_{ln} - y_n^{+}\right)^2}, \\ \text{ED}_l^{-} &= \sqrt{\sum_{n\in\{1,2,3,4\}} \left(y_{ln} - y_n^{-}\right)^2}. \end{aligned} \tag{21}$$

Step 4: calculate the instant reward after the user selects a candidate network. In this paper, Rw_l is expressed by the relative proximity of the candidate network to the ideal solution:

$$\text{Rw}_l = \frac{\text{ED}_l^{-}}{\text{ED}_l^{+} + \text{ED}_l^{-}}. \tag{22}$$

The larger ED_l^{-} is, the smaller ED_l^{+} is and the closer Rw_l is to 1, indicating the candidate solution is closer to ideal solution and the reward is larger. Conversely, the smaller ED_l^{-} is, the larger ED_l^{+} is, indicating that the network accessed by the agent is poor and Rw_l is closer to 0.

In summary, the reward function of the paper is as follows:

$$\text{Rw}(s,a) = \begin{cases} \dfrac{\text{ED}_l^{-}}{\text{ED}_l^{+} + \text{ED}_l^{-}}, & \text{valid action}, \\ 0, & \text{invalid action}. \end{cases} \tag{23}$$

3.4. Algorithm Steps. In order to maximize the satisfaction of mobile users in the cell, this paper considers the four attributes of throughput, power consumption, cost, and delay, uses AHP to calculate the weight of each attribute, defines the reward function by TOPSIS, and relies on Q-learning to iterate until convergence. The best offloading strategy in each state can finally be obtained. In Q-learning, the Q value will be updated with the user learning:

$$\begin{aligned} Q_t(s,a) &= (1-\mu) Q_{t-1}(s,a) \\ &\quad + \mu\left[\text{Rw}_t(s,a) + \delta \max_{a'\in A} Q_{t-1}(s',a')\right], \end{aligned} \tag{24}$$

where $\mu \in (0,1)$ is the learning rate. The larger μ is, the less the Q value of the previous training is retained and the more important is the instant reward $\text{Rw}_t(s,a)$ and the experience reward $\max_{a'\in A} Q_{t-1}(s',a')$. δ is the discount factor of the experience reward, and s' is the state that the agent transfers into.

In addition, this paper also introduces the ε-greedy algorithm. In each action selection of Q-learning, the agent explores with a small probability ε, that is, randomly selects a network to

Input: state set S, action set A, paired comparison matrix B, candidate network attribute matrix X, and iteration limit Z
Output: trained Q-table, best action selection strategy Π^*, and user satisfaction Φ_j^{sat}
(1) Calculate attribute weights based on B
(2) For $s \in S$, $a \in A$
(3) $Q(s, a) = 0$
(4) End For
(5) Randomly choose $s_{\text{ini}} \in S$ as the initialization state
(6) While iteration $< Z$
(7) For each state
(8) If rand $< \varepsilon$
(9) Randomly choose an action
(10) Else
(11) Select the action corresponding to the maximum Q value in this state.
(12) End If
(13) Perform a
(14) Calculate $\text{Rw}_t(s, a)$ according to equation (23)
(15) Observe the next state s'
(16) Update the Q-table according to equation (24)
(17) End For
(18) End While
(19) Record the action corresponding to the maximum Q value in each state into Π^*
(20) Calculate user satisfaction Φ_{sat}

Algorithm 1: WiFi offloading algorithm based on Q-learning and MADM in heterogeneous networks.

offload. Without ε-greedy algorithm, it is possible that the cumulative reward of a suboptimal action becomes bigger and bigger, which makes the user choose this action and increase the cumulative reward again, instead of finding a better one. In other words, the core of ε-greedy is to explore. The reason why the ε-greedy algorithm performs better is that it continuously explores the probability of finding the optimal action. Although it is possible to reduce the user satisfaction in the next period of time, hoping that in the future, we can make better action choices and ultimately get the most user satisfaction. Based on the above analysis, Algorithm 1 gives the WiFi offloading algorithm based on Q-learning and MADM.

4. Numerical and Simulation Results

As shown in Figure 1, the simulation scenario is established in a circular cell with a radius r_{cell} of 500 m. The cellular BS is located in the cell center, and N_{AP} WiFi AP is randomly distributed inside the cell. The additive white Gaussian noise power spectral density N_0 is -174 dBm/Hz, and reference distance d_0 is 1 m. In $F_{\text{Rayleigh}}(\theta, \beta)$, mean $\theta = 0$ and variance $\beta = 5$ dB. Furthermore, the learning rate μ of the Q-learning is set to 0.8, the discount factor of the experience reward δ is set to 0.1, and ε in $\varepsilon-$greedy is set to 0.01. In AHP, when network attribute number $N = 4$, the consistency index RI = 0.9 [15]. The paired comparison matrices B of different services are shown in Table 2, and they are recognized results based on the general needs of each service, which are given by experts' opinions. The remaining parameters are shown in Table 3.

Firstly, we analyse the performance of this algorithm under stream service. According to AHP algorithm, the weight vector corresponding to throughput, power consumption, cost, and delay is obtained as $[w_1^{\text{tp}}, w_1^{\text{pc}}, w_1^{\text{c}}, w_1^{d}]^{\text{T}} = [0.4891, 0.1896, 0.2321, 0.0893]$. When the user conducts streaming media services like watching a video, the most important thing is throughput and the least is delay. Because a video usually has a large size such as 500 MB, 1 GB, or more, we need the throughput to be big enough to support the cache of the video. The user equipment only needs to read the data precached in it to perform the service, which is not real-time. So stream service does not need low delay.

Figure 3 shows the convergence comparison between the invalid action filtering and nonfiltering in the WiFi offloading algorithm under stream service. Advance filtering means that this paper filters the invalid network whose actual throughput is less than the throughput threshold $V_{\text{th}}^{\text{TP}}$ before Q-learning. Assume $N_{\text{AP}} = 30$, and the total number of positions N_{p} passed by the user is equal to 10. The two cases are subjected to Q-learning in the same experimental scenario, and the convergence was observed. Since the action selection in Q-learning is discontinuous, user satisfaction will jump when changing the action selection strategy. As can be seen from Figure 3, after filtering out the invalid network whose throughput is less than the threshold $V_{\text{th}}^{\text{TP}}$ in advance, the convergence speed of the Q-learning can be greatly accelerated.

Figures 4 and 5 show the comparison between this paper's algorithm, Fakhfakh and Hamouda's algorithm [11], and RSS (received signal strength) algorithm based on user satisfaction, throughput, power consumption, cost, and delay under stream service. We repeatedly scatter APs 1000 times to eliminate randomness. The number of user-passed positions N_{p} is equal to 10, and the number of WiFi AP is changed from 20 to 60. As can be seen from Figure 4, the WiFi offloading algorithm in this paper is superior to the

Table 2: Comparison matrices corresponding to stream service and conversation service.

Network attribute	Stream				Conversation			
	TP	PC	C	D	TP	PC	C	D
TP	1	3	2	5	1	2	1	1/9
PC	1/3	1	1	2	1/2	1	1/3	1/9
C	1/2	1	1	3	1	3	1	1/9
D	1/5	1/2	1/3	1	9	9	9	1

Table 3: Simulation parameters of cellular network and WiFi network in this paper.

Simulation parameters	Cellular network	WiFi network
User cost V_i^C (/s)	0.8	0.1
Communication delay V_i^D (ms)	25 to 50	100 to 150
Bandwidth W (MHz)	4 to 6	10 to 12
Path loss L_0 at d_0 (dB)	5.27	8
Terminal fixed power consumption P_0 (mW)	10	10
Minimum received power $P^r_{\min}$ (dBm)	−110	−100
User throughput threshold $V^{\mathrm{TP}}_{\mathrm{th}}$ (kb/s)	10	12
Path loss exponent α	3.76	4

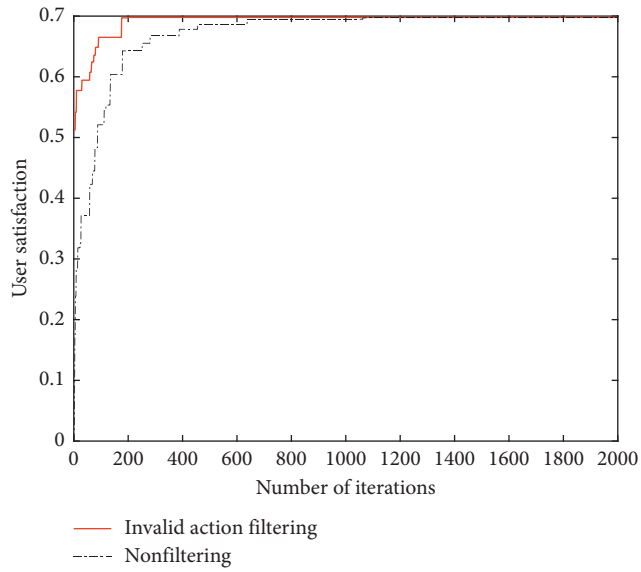

Figure 3: Convergence comparison between the invalid action filtering and nonfiltering, recorded 2000 iterations under stream service.

other two algorithms in user satisfaction. The main difference between this paper and [11] is the reward function of the Q-learning. Fakhfakh and Hamouda's algorithm [11] aims to minimize the residence time of the cellular network and optimize it by Q-learning, but its reward function only considers SINR, handover delay, and AP load, without considering the attributes directly related to user QoS, such as terminal power consumption, user cost, and communication delay. The RSS algorithm only considers the received signal strength of the terminal, and the terminal automatically accesses network with the largest RSS, so the user satisfaction is lower. The Q-learning algorithm in this paper not only considers the attributes directly related to user QoS but also uses two MADM algorithms to obtain the intrinsic relationship of these attributes. It establishes a more reasonable Q-learning reward function and obtains the best user satisfaction. As can be seen from Figure 5, the algorithm in this paper is similar to [11] in terms of user throughput. This is because Fakhfakh and Hamouda's algorithm [11] regards SINR as the most important aspect of the reward function, which directly affects throughput. Since the simulation is based on the stream service, the weight of throughput accounts for almost half of all the attributes, so the two algorithms perform similarly in throughput. Since the other two algorithms do not consider power consumption and cost, the algorithm performs better on these two network attributes. The RSS algorithm selects the network with the highest receiving power to access. In this scenario, as long as the terminal is not too far away from the cellular BS, RSS of the cellular network will be the largest, so the number of WiFi offloading is reduced. Since the WiFi network uses the unlicensed frequency band, the bandwidth

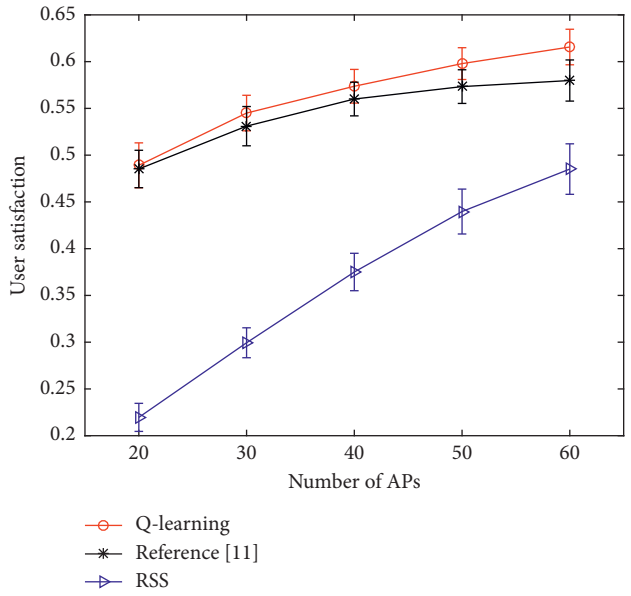

Figure 4: User satisfaction comparison under stream service. The error bars represent the standard deviation for the user satisfaction of 1000 times scatter of WiFi APs.

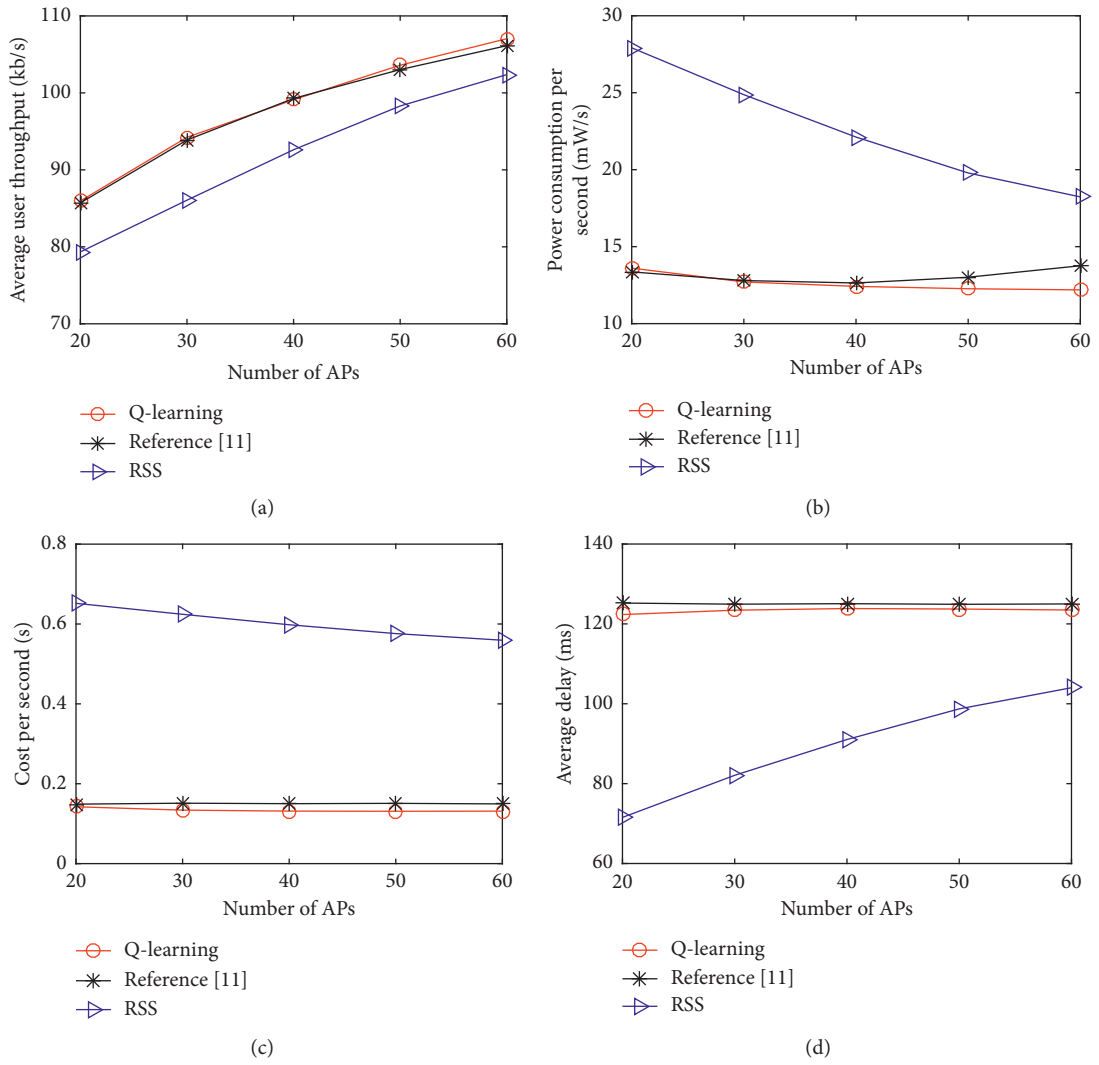

Figure 5: Comparisons of throughput, power consumption, cost, and delay under stream service.

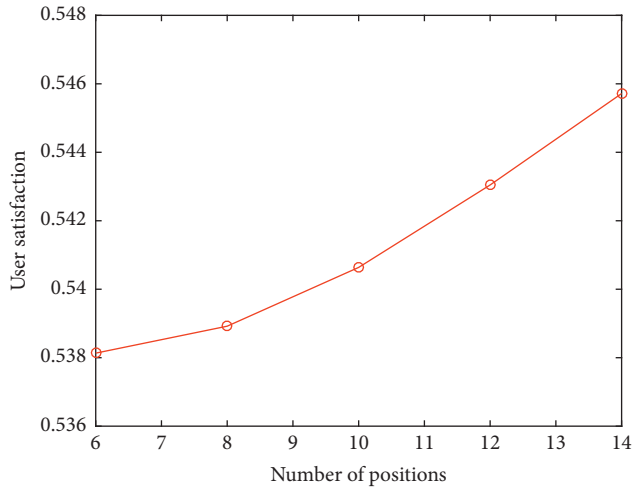

Figure 6: Plot of user satisfaction with respect to the number of positions passed by the agent. The number of WiFi APs is 30, and the service type is stream.

available to the user is usually larger than accessing the cellular network. As a result, the throughput of it becomes less. Because the delay of cellular network is usually lower than WiFi network, the RSS algorithm performs best on the delay attribute. However, since the weight of the delay attribute in the stream service is very low, the user does not pay attention to the delay of the precached data when watching video or listening to music. As a result, although the algorithm in this paper is not as good as the RSS algorithm in delay, user satisfaction is much higher than it.

Figure 6 shows the user satisfaction against the number of positions passed by agent after repeatedly scattering AP 1000 times to eliminate randomness. The number of WiFi AP $N_{\mathrm{AP}} = 30$, and the terminal passes through 6, 8, 10, 12, and 14 positions, respectively. It can be seen that the more the positions, the higher the user satisfaction because as the number of positions increases, the states of Q-learning will increase, and the chances of agent actively selecting the optimal network to offload will also increase, so the satisfaction will also become higher.

Figures 7 and 8 show the comparison between this paper's algorithm, Fakhfakh and Hamouda's algorithm [11], and RSS algorithm based on user satisfaction, throughput, power consumption, cost, and delay under conversation service. The number of user-passed positions N_{p} is equal to 10, and the number of WiFi AP is changed from 20 to 60. According to AHP algorithm, the weight vector is obtained as $[w_2^{\mathrm{tp}}, w_2^{\mathrm{pc}}, w_2^{c}, w_2^{d}]^{\mathrm{T}} = [0.0955, 0.0534, 0.1084, 0.7427]$, which indicates that when the user chooses conversation service like making a voice call, the most important attribute is communication delay while the other three attributes are less important. When we make a voice call, it will drastically reduce the QoS if the time we wait is too long. As can be seen from Figure 7, the WiFi offloading algorithm in this paper is superior to the other two algorithms in user satisfaction. Fakhfakh and Hamouda's algorithm [11] does not consider the communication delay, so the satisfaction is the worst. As

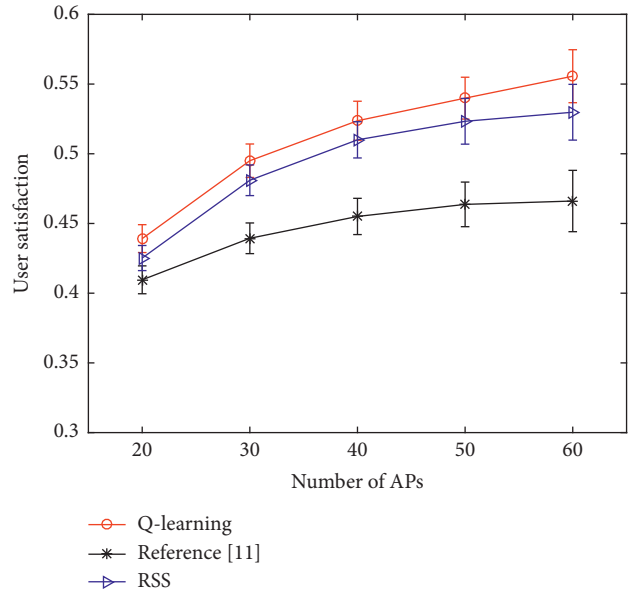

Figure 7: User satisfaction comparison under conversation service. The error bars represent the standard deviation for the user satisfaction of 1000 times scatter of WiFi APs.

is mentioned above, RSS algorithm usually makes the terminal access the cellular BS which has a bigger transmit power and a lower delay, so the satisfaction is better than [11]. As can be seen from Figure 8, the WiFi offloading algorithm in this paper is superior to the RSS algorithm in throughput, power consumption, and cost, while the communication delay performance is near RSS algorithm. In this paper, delay is the most important attribute under conversation service, so the delay performance nears RSS algorithm. We also consider other attributes, which makes a few users offload to WiFi network, so the delay of this algorithm is slightly higher than the RSS algorithm.

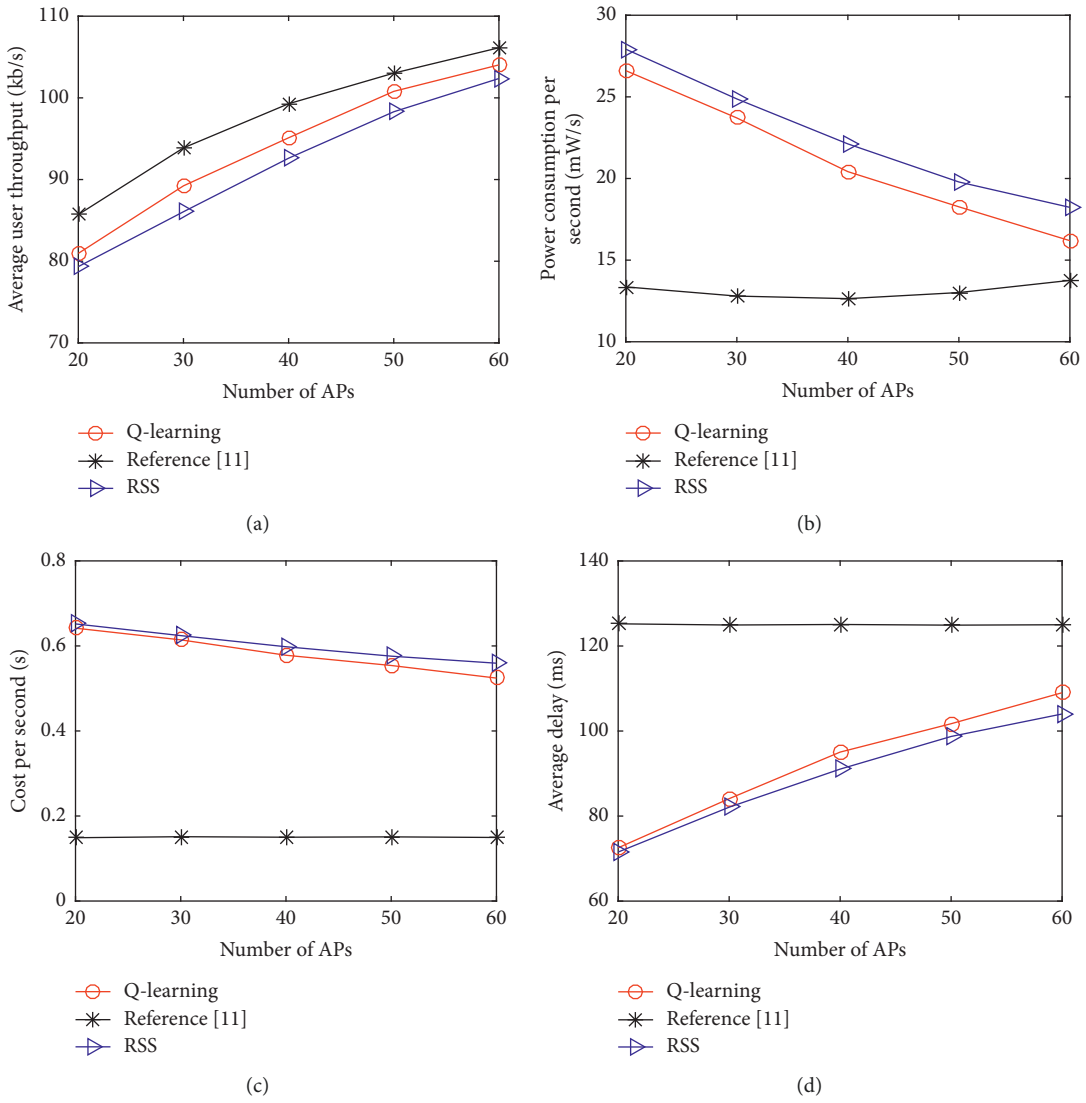

Figure 8: Comparisons of throughput, power consumption, cost, and delay under conversation service.

5. Conclusion

In the heterogeneous network scenario where cellular network and WiFi network overlap, this paper establishes a model of mobile terminal WiFi offloading, and the Markov model is used to describe the change of available bandwidth. Four network attributes of user throughput, terminal power consumption, user cost, and communication delay are considered to define a user satisfaction function. The AHP algorithm is used to calculate the attribute weights, and the TOPSIS algorithm is used to obtain the instant rewards when the user accesses the cellular network or offloads to the WiFi network. Using the Q-learning algorithm, combined with instant rewards and experience rewards to update the discounted cumulative rewards, the user can make the optimal offloading decision and get the maximum satisfaction in each passing position. The simulation results show that the proposed algorithm can converge under limited times, and compared with the comparison algorithm, the algorithm has a great improvement in user satisfaction.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61971239 and 61631020).

References

[1] Cisco White Paper, *Cisco Visual Networking Index—Global Mobile Data Traffic Forecast, Update, 2016–2021*, Cisco Systems, Prague, Czech Republic, 2017.

[2] D. Ho, G. S. Park, and H. Song, "Game-theoretic scalable offloading for video streaming services over LTE and WiFi networks," *IEEE Transactions on Mobile Computing*, vol. 17, no. 5, pp. 1090–1104, 2018.

[3] Q. Chen, G. Yu, H. Shan, A. Maaref, G. Y. Li, and A. Huang, "Cellular meets WiFi: traffic offloading or resource sharing?," *IEEE Transactions on Wireless Communications*, vol. 15, no. 5, pp. 3354–3367, 2016.

[4] A. Aijaz, H. Aghvami, and M. Amani, "A survey on mobile data offloading: technical and business perspectives," *IEEE Wireless Communications*, vol. 20, no. 2, pp. 104–112, 2013.

[5] Z. Li, C. Dong, A. Li, and H. Wang, "Traffic offloading from LTE-U to WiFi: a multi-objective optimization approach," in *Proceedings of the 2016 IEEE International Conference on Communication Systems (ICCS)*, IEEE, Shenzhen, China, December 2016.

[6] J. Xu, S. Wu, L. Xu, N. Zhang, and Q. Zhang, "Green-oriented user-satisfaction aware WiFi offloading in HetNets," *IET Communications*, vol. 12, no. 5, pp. 501–508, 2018.

[7] S. Cai, L. Duan, J. Wang et al., "Incentive mechanism design for delayed WiFi offloading," in *Proceedings of the ICC 2015—2015 IEEE International Conference on Communications*, June 2015.

[8] X. Kang, Y.-K. Chia, S. Sun, and H. F. Chong, "Mobile data offloading through a third-party WiFi access point: an operator's perspective," *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5340–5351, 2014.

[9] B. H. Jung, N. O. Song, and D. K. Sung, "A network-assisted user-centric WiFi-offloading model for maximizing per-user throughput in a heterogeneous network," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 4, pp. 1940–1945, 2014.

[10] U. Sethakaset, Y. K. Chia, and S. Sun, "Energy efficient WiFi offloading for cellular uplink transmissions," in *Proceedings of the 2014 IEEE 79th Vehicular Technology Conference (VTC Spring)*, IEEE, Seoul, Korea, May 2015.

[11] E. Fakhfakh and S. Hamouda, "Optimised Q-learning for WiFi offloading in dense cellular networks," *IET Communications*, vol. 11, no. 15, pp. 2380–2385, 2017.

[12] S. Kunarak and R. Suleesathira, "Predictive RSS with fuzzy logic based vertical handoff algorithm in heterogeneous wireless networks," in *Proceedings of the 2010 10th International Symposium on Communications & Information Technologies*, IEEE, Tokyo, Japan, June 2010.

[13] J. I. Pelaez, E. A. Martinez, and L. G. Vargas, "Consistency in positive reciprocal matrices: an improvement in measurement methods," *IEEE Access*, vol. 6, pp. 25600–25609, 2018.

[14] L. Zhang and Q. Zhu, "Network selection algorithm based on multi-radio parallel transmission for heterogeneous wireless networks," *Journal of Signal Processing*, vol. 30, no. 10, pp. 1176–1184, 2014.

[15] L. Zhang and Q. Zhu, "Multiple attribute network selection algorithm based on AHP and synergetic theory for heterogeneous wireless networks," *Journal of Electronics (China)*, vol. 31, no. 1, pp. 29–40, 2014.

[16] H.-W. Yu and B. Zhang, "A hybrid MADM algorithm based on attribute weight and utility value for heterogeneous network selection," *Journal of Network and Systems Management*, vol. 27, no. 3, pp. 756–783, 2019.

[17] C. J. C. H. Watkins and P. Dayan, "Technical note: Q-Learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[18] A. E. Shafie, T. Khattab, H. Saad, and A. Mohamed, "Optimal cooperative cognitive relaying and spectrum access for an energy harvesting cognitive radio: reinforcement learning approach," in *Proceedings of the 2015 International Conference on Computing, Networking and Communications (ICNC)*, IEEE, Anaheim, CA, USA, February 2015.

[19] A. Bazzi, B. M. Masini, A. Zanella, and D. Dardari, "Performance evaluation of softer vertical handovers in multiuser heterogeneous wireless networks," *Wireless Networks*, vol. 23, no. 1, pp. 159–176, 2017.