A Q-learning-based Heterogeneous Wireless Network Selection Algorithm

Chen-Wei Feng[1,2] Lian-Fen Huang[1,*] Pei-Zhi Ye[1]
Yu-liang Tang[1] Han-Chieh Chao[3,4]

[1] Department of Communication Engineering, Xiamen University
Xiamen 361005, China
lfhuang@xmu.edu.cn, * Corresponding Author
peizhiye@139.com, tyl@xmu.edu.cn

[2] Department of Communication Engineering, Xiamen University of Technology
Xiamen 361024, China
cwfeng@xmut.edu.cn

[3] Institute of Computer Science & Information Engineering and Department of Electronic Engineering
National Ilan University
I-Lan, Taiwan

[4] Department of Electrical Engineering, National Dong Hwa University
Hualien, Taiwan
hcc@niu.edu.tw

Received 7 October 2013; Revised 25 November 2013 ; Accepted 16 December 2013

Abstract: A Q-learning-based algorithm is presented for heterogeneous wireless network selection. The proposed algorithm can select the appropriate network for access according to different traffic types, terminal mobility and network load status. Simulation results show that the proposed algorithm has an efficient learning ability to achieve autonomous radio resource management, which effectively improves the spectrum utility and reduces the blocking probability.

Keywords: heterogeneous wireless networks; selection algorithm; Q-learning

1 Introduction

As wireless communication technology develops a variety of wireless access technologies will coexist in the future communication environment. It is essential to coordinate heterogeneous wireless network resources due to overlapping network coverage, different traffic needs as well as complementary technical features. A lot of Joint Radio Resource Management (JRRM) [1] methods are presented to realize load balancing [2] and heterogeneous network selection [3]. However, many existing algorithms are neither autonomous for network access nor adaptable to the dynamic wireless network environment. The network should have the self-learning ability to constantly revise its strategy according to the actual environmental situation to achieve network resources self-management.

Reinforcement Learning (RL) [4] is a learning algorithm in which the learning agent learns through its interactions. The objective is learning what action to take at each state to maximize a specific metric. The agent achieves an optimal decision policy by repeatedly interacting with the controlled environment and evaluating its performance through a reward. RL is widely used in robotics and automatic control [5]. RL has been introduced into resource management in wireless communication systems [6-9] for its flexibility and adaptability. Q-learning is a RL method in which the learning agent incrementally builds a Q-function that attempts to estimate the discounted future costs for taking an action in the agent's current state. Existing research on Q-learning has been carried out in heterogeneous wireless network selection.

Paper [10-11] studied the joint Q-learning algorithm for network admission control, but the traffic attributes are not distinguished. Paper [12] discusses Q-learning for autonomous joint management of resources, which considers the traffic attributes but without the difference in terminal mobility. Paper [13] distinguishes both, but

the utility function does not involve the bandwidth request which may influence the resource allocation. Moreover, the algorithm proposed lacks comparison with other algorithms.

This paper improves the joint selection Q-learning based heterogeneous wireless network algorithm which selects the appropriate network to access according to the traffic type, terminal mobility and network load status. Simulation results show that the algorithm reduces the system blocking probability and effectively improves the spectrum.

This paper is organized as follows. The Q-learning strategy model is introduced in section 2. The process used to realize this scene with Q-Learning is presented in section 3. The proposed algorithm's performance is assessed in section 4 through simulations. Section 5 concludes the paper.

2 Q-learning Strategy

In reinforcement learning systems a machine or various systems with learning ability are referred to as the agent. The learning agent aims at learning an optimal control strategy by repeatedly interacting with the controlled environment in such a way that its performance evaluated by a scalar reward (cost) obtained from the environment is maximized (minimized) [14].

The basic reinforcement learning model consists of the following elements:

1) The set of possible state $S = \{s_1, s_2, \cdots s_m\}$.

2) The set of possible action $A = \{a_1, a_2, \cdots, a_n\}$.

3) Reward (payoff) r.

4) The strategy of the agent $\pi : S \to A$.

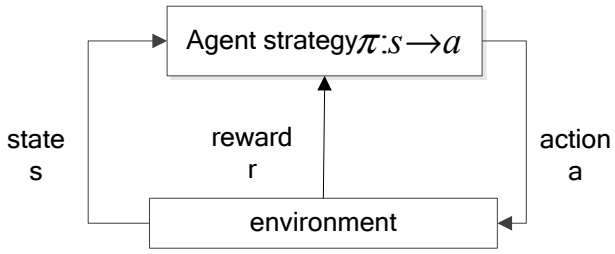

Fig. 1. An illustration of agent-environment interaction

The interaction between the agent and the environment is shown in Fig.1 [7]. The agent senses the environmental state $s \in S$ at each time, then chooses an action $a \in A$ to perform based on the strategy π. As a result the environment makes a transition to a new state $s' \in S$ and thereby generates a reward (payoff) $r(s,a)$ according to the effect of the action. The reward is passed back to the agent and the process is repeated. The task of the learner is to find an optimal strategy $\pi^*(s) \in A$ for each state that maximizes the total expected cumulative return over time.

A variety of RL algorithms exist. A particular algorithm that appears to be suitable for network selection is called Q-learning [15]. The agent will update its strategy according to Equation (1) for the next time.

$$Q_{t+1}(s,a) = (1-\alpha)Q_t(s,a) + \alpha(r_t + \gamma \max_{a' \in A} Q_t(s',a')) \ . \quad \textbf{(1)}$$

In the above equation, $\alpha \in [0,1)$ is the learning rate. If the learning rate is decreased to zero in a suitable way, then as $t \to \infty$, $Q_t(s,a)$ converges to $Q^*(s,a)$ with probability 1. The optimal strategy π^* is the one with the maximum Q-value: $\pi^*(s) = \arg\max_{a \in A} Q^*(s,a)$.

3 Q-Learning Algorithm for Network Selection

3.1 System Model

The overlapping coverage of a heterogeneous wireless network in this paper consists of UMTS and WLAN as shown in Fig.2.

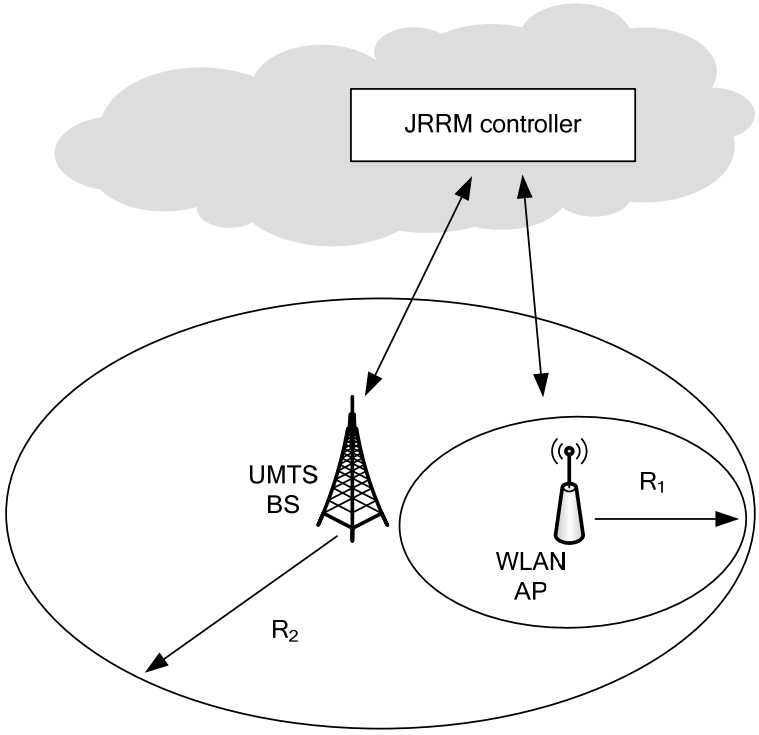

Fig.2. The heterogeneous networks model

There is a JRRM controller with learning ability where the Q-learning algorithm is running. The JRRM controller selects the appropriate network to access according to the traffic type, the terminal mobility, the network load status and other conditions.

Only two types of traffic are considered in this paper, voice traffic and data traffic. Since the UMTS network and WLAN network have different features, WLAN network is more suitable for data traffic with high bandwidth, and the UMTS network is more suitable for voice traffic because of its real-time characteristic. However, from the viewpoint of terminal mobility, the UMTS network is more suitable for high-speed mobile terminals, yet the WLAN network is more suitable for low-speed or stationary terminals.

3.2 Problem Mapping

The system state, action and reward must be carefully defined before the Q-learning algorithm can be applied to system selection in a heterogeneous network.

(1) State

The system state designated as $s \in S$ represents the load characteristics of all available networks, but also the different traffic types and terminal mobility. Even if the same load conditions exist in two networks, the optimal network will be different due to these two attributes. The main state set is defined in this article as follows:

$$S = \{v, m, l\} \ . \tag{2}$$

Where v denotes the traffic type, only voice and data traffic are considered in this paper, so the value is 1 or 2, respectively, applied to both voice and data traffic.

Similarly, terminal mobility designated as m is briefly divided into two states, stationary and movement, and the value is 1 or 2 respectively.

l represents network load conditions in the system, designated as the ratio of resources used to the total resources in each network. For simplicity, the ratios are quantized into several levels in order to construct a Q value table.

Note that a discrete series of arrival or the end of session events will affect the network status, such as the network load conditions, etc. However, the JRRM controller will not trigger when the session ends, because the JRRM Controller processes only access requests. The set of states is only associated with the arrival of a session, so the algorithm for network selection is used only when an access request comes.

(2) Action

In a wireless heterogeneous network the JRRM controller selects the appropriate network for access according to the state and learning experience. Only UMTS and WLAN are considered in this paper, so it is relatively simple to set the actions defined as follows:

$$A = \{1, 2\} \ . \tag{3}$$

where 1 is the UMTS network and 2 is the WLAN network.

(3) Reward

The reward $r(s,a)$ assesses the immediate payoff incurred due to the acceptance of a call in state s. Different call accessing networks will give rise to different influence to the system performance because of the traffic type and terminal mobility. If the traffic type and terminal mobility match the chosen network, the cumulative spectrum utility is the maximum, otherwise it will be smaller. In order to balance the network load status the reward function is defined as follows:

$$r = \beta \times \left(\eta(v,k) + \eta(m,k) \right) \ . \tag{4}$$

where $\eta(v,k)$ is the matching coefficient of the traffic v and the network k, $\eta(m,k)$ is the matching coefficient of the mobile state m and the network k. β is the load factor, which means the ratio of the remaining resources to the total resources in the chosen network. The exact values will be given in the following simulation section.

Equation (4) can be described simply as follows: If the chosen network can match better with the traffic type and the terminal mobility, the matching value obtained is larger. However, if the match is poor the corresponding value obtained is smaller. However simply maximizing matching values may cause severe resource imbalances in the system, especially when particular continuous sessions reach the network. Thus, the JRRM controller will timely guide the request session to the lower load network according to the load factor β playing a certain role in load balancing.

Meanwhile, in order to evaluate network performance reflecting the gains obtained by the user, the cumulative spectrum utility U is defined as follows:

$$U = \sum_j b_j (\eta_j(v,k) + \eta_j(m,k)) \ . \tag{5}$$

where b_j is the bandwidth session j is allocated.

3.3 Algorithm Implementation

The Q-learning algorithm is built into the system to consider the network load condition comprehensively, i.e., the type of traffic and the terminal mobility. Q-learning is an on-line learning scheme composed of two aspects: strategy update and Q-value update.

1. Strategy update: In order to learn an optimum decision strategy, at each decision epoch the agent will select a network randomly with probability $\varepsilon \in [0,1]$, and with probability $(1-\varepsilon)$ it will select a network based on the stored Q-values.

2. Q-value update: Q-values can be obtained from the Q-value look-up table corresponding to state and the network will be selected with the maximum Q-value. After transition to the next state the Q-values will be updated according to equation (1).

The algorithm procedure is described as follows:

1) Initialize. Set Q as 0, the discount factor γ, the initial learning rate α_0 and the initial probability exploration ε_0 .The exact values of the related parameters are listed in Table 1.

2) Acquire the current state S. The JRRM controller will collect the related state including the resources used in each network, the traffic type, the bandwidth requested and terminal mobility when session arrives.

3) Choose an action from A. Choose an action to perform according to the action function of the current state $Q_t(s,a)$, based on ε -greedy strategy.

4) Obtain the reward r based on Equation (4) and the state s' of the next instant. The reward value is 0 if the session access request is rejected by the network.

5) Update $Q_t(s,a)$ according to equation (1).

6) Update the parameters. After each iteration, the learning rate α and exploring probability ε must be updated to satisfy the convergence requirement. These two parameters are set to reduce to 0 according to a function inverse to the learning process.

7) Return to 2).

4 Simulation Results and Analysis

This paper considers that the new session occurs in the overlapping coverage area shown in Fig.2. The time interval for each arriving session is subject to an exponential distribution with a mean of 20s.The call-holding time obeys an exponential distribution with a mean of 80s.Changing the user intensity coefficient u simulates how busy the network is. The larger u is set, the more sessions occur. There are only two types of traffic in the area, real-time voice traffic and non-real-time data traffic that are uniformly distributed. The voice traffic bandwidth requirement is set to 1 to 2 resource blocks. The data traffic bandwidth requirement is set to 3 to 5 resource blocks. The network load is uniformly quantized into 10 levels for building the Q-value table. Other simulation parameters are set in Table 1.

Table 1. Simulation parameters

		UMTS	WLAN
Total resource blocks		50	100
Matching coefficient $\eta(v,k)$	voice	5	1
	data	1	5
Matching coefficient $\eta(m,k)$	still	5	1
	moving	1	5
Other parameters		Discount factor $\gamma = 0.8$ Initial learning rate $\alpha_0 = 0.5$ Initial exploring probability $\varepsilon_0 = 0.5$	

The simulation evaluates the blocking probability and the cumulative spectrum utility of the JRRM Q-learning algorithm (QLA). This paper also assesses the JRMM load balancing (LBA) algorithm and random access algorithm (RAA) without considering the JRRM concept for comparison.

Figures 3 and 4 show the blocking probability and cumulative spectrum utility of each algorithm with the change in the user intensity coefficient. As can be seen from Fig.3 the network gradually becomes busy because of the increasing user intensity coefficient, so the blocking probability gradually becomes larger. All algorithms show the same trends in Fig.2. The RAA algorithm has the highest blocking probability because the status of the available resources in the network is not considered. Network selection based on the RAA algorithm is very blind, which is likely to select a fully loaded network, resulting in increasing blocking probability. The session request is always connected to the low-load network based on the LBA algorithm, so each network is not easily saturated. This greatly reduces the network blocking probability. Network load is one of the status parameters in the QLA algorithm, which only partly takes the load balancing between different networks into account with the traffic type and terminal mobility to match the network. Therefore, the blocking probability is worse than that in the LBA algorithm which fully features load balancing.

Figure 4 shows that as the user intensity coefficient becomes larger the total number of sessions will also increase. According to the cumulative spectrum utility definition, the system will accumulate the benefits arising from session access, so the cumulative spectrum utility gradually becomes larger. Among the three algorithms the QLA algorithm cumulative spectrum utility performance is the best. The session will access the proper net-

work to make reasonable use of system resources based on the QLA algorithm because the traffic and mobile matching properties are taken into consideration. Although the blocking probability based on the LBA algorithm is better than that of the QLA algorithm, the LBA algorithm ignores the traffic type and terminal mobility influence, resulting in inefficient use of resources. The RAA algorithm has the worst performance of all. The reason is the same as that for the LBA algorithm. The high blocking probability directly affects the cumulative spectrum utility obtained.

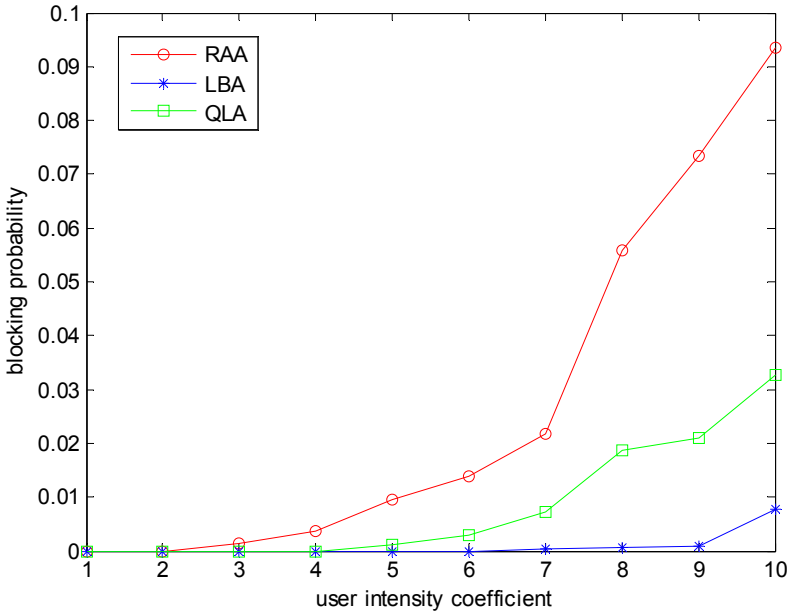

Fig.3. Comparison of blocking probability with user intensity coefficient

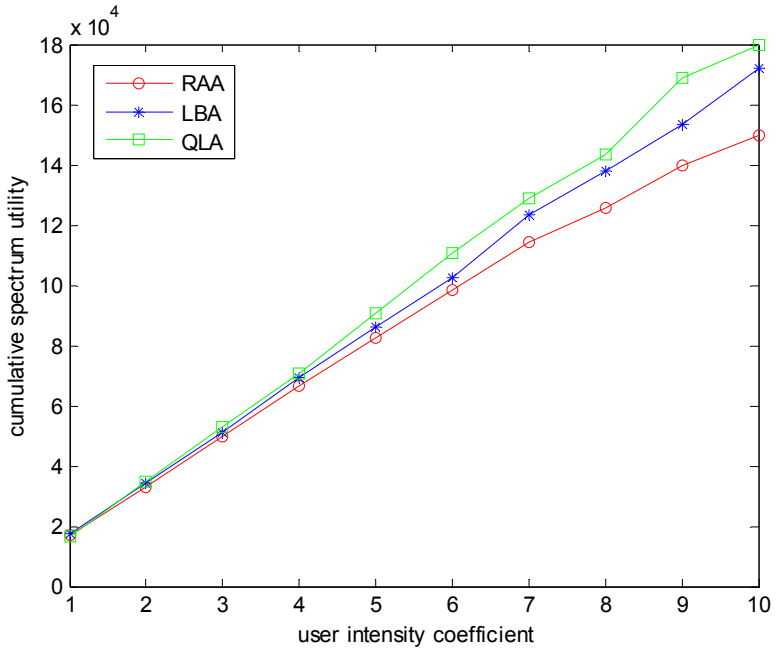

Fig.4. Comparison of cumulative spectrum utility with user intensity coefficient

Figures 5 and 6 show the blocking probability and cumulative spectrum utility of each algorithm over time. Figure 5 indicates the blocking probability convergence before and after learning. It is obvious that blocking probability based on the RAA and LBA algorithm is stable in general. Despite the QLA algorithm blindly exploring in the initial stage, it is convergent to the minimum with the learning process, which proves the effectiveness of the proposed algorithm in on-line leaning ability.

The cumulative spectral utility trending over time is given in Fig.6. Though the accumulative spectrum utility of the QLA algorithm is essentially the same with the other two in the beginning, the QLA algorithm gradually outperforms the others. Through on-line learning the system can effectively apply experience to choose a subsequent strategy to improve the performance.

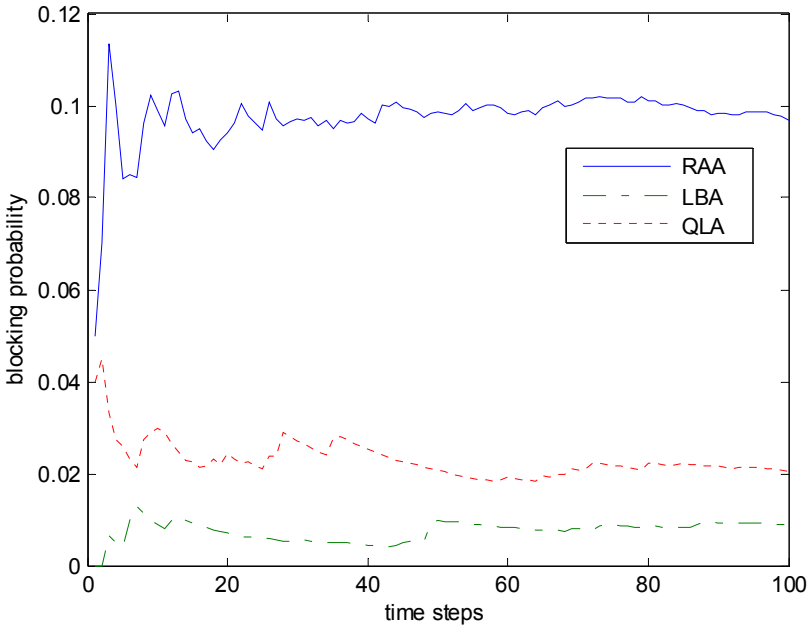

Fig.5. Comparison of blocking probability with time

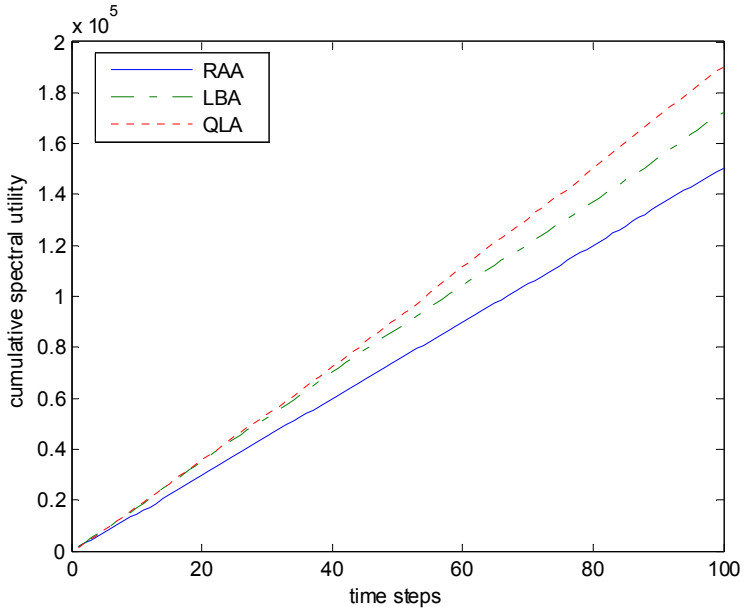

Fig.6. Comparison of cumulative spectrum utility with time

It is the algorithms shown in Figs. 7 and 8 clearly differentiate the traffic type and terminal mobility influence. Whether from the traffic type or terminal mobility view point, the RAA and LBA algorithms are unable to distinguish sessions well. The QLA algorithm allows applying resources in the optimum configuration. The proportion of voice traffic and the traffic initiated by the mobile terminal are higher than the data traffic and traffic initiated by a still terminal in UMTS. Conversely, the proportion of corresponding traffic is opposite. The QLA algorithm allows different networks to fully utilize their technical superiority, thereby improving the overall system.

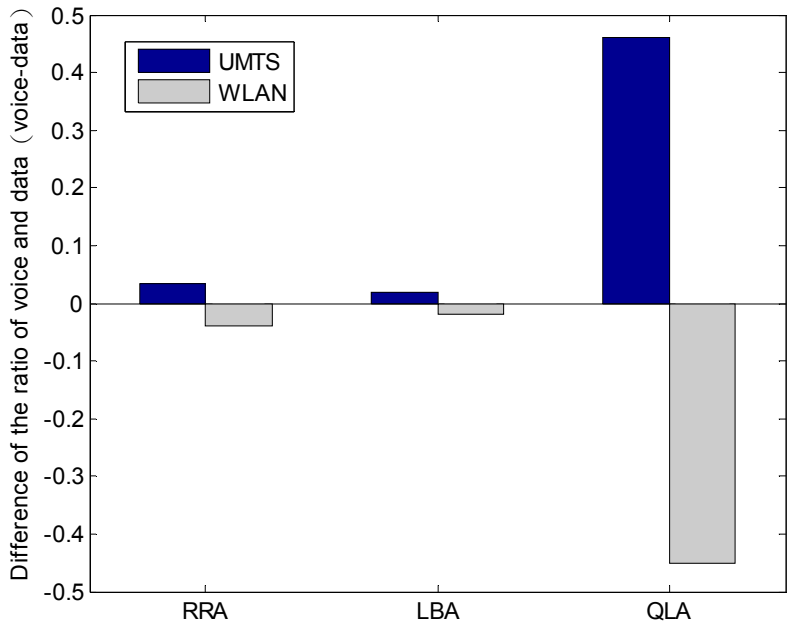

Fig.7. Difference in the voice and data ratio in UMTS and WLAN

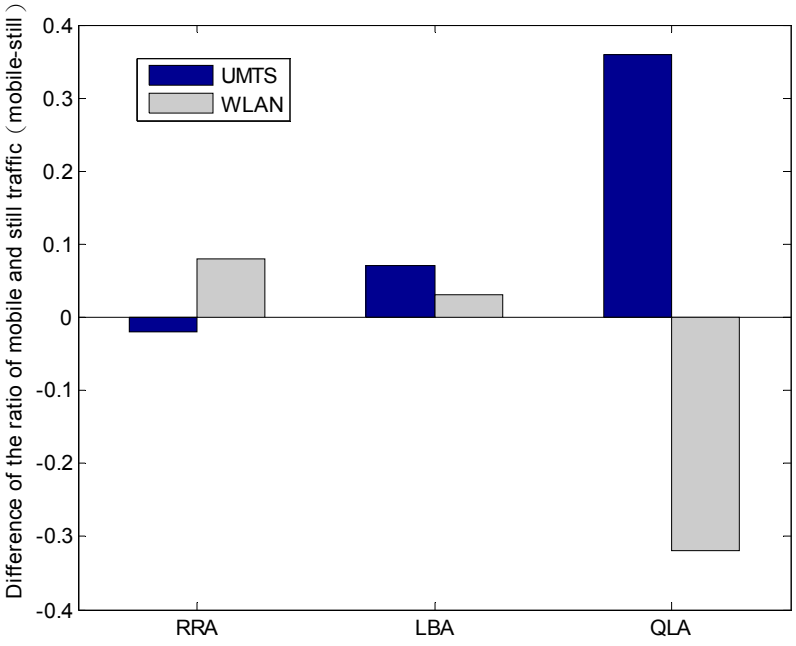

Fig.8. Difference in the mobile and still traffic ratio in UMTS and WLAN

5 Conclusion

This paper examined a dynamic selection strategy in a heterogeneous wireless network based on Q-learning. Considering the network load condition, traffic type and the terminal mobility, the JRRM controller can reasonably assigned each session to the optimum network according to the network characteristics. It was shown that user perceived performance is enhanced by learning process convergence.

Acknowledgement

The work presented in this paper was partially supported by 2011 National Natural Science Foundation of China (Grant number 61172097)， 2012 Natural Science Foundation of Fujian (Grant number 2012J01424) and by 2014 National Natural Science Foundation of China (Grant number 61371081).

References

[1] Luo, J.; Mukerjee, R.; Dillinger, M.; Mohyeldin, E.; Schulz, E., "Investigation of radio resource scheduling in WLANs coupled with 3G cellular network," *IEEE Communications Magazine*, vol.41, no.6, pp.108-115, 2003.

[2] 3GPP TR 25.881 v5.0.0. Improvement of RRM across RNS and RNS/BSS (Release5). *http://www.3gpp.org*, 2001.

[3] Qingyang Song; Jamalipour, A., "Network selection in an integrated wireless LAN and UMTS environment using mathematical modeling and computing techniques," *IEEE Wireless Communications*, vol.12, no.3, pp.42-48, 2005.

[4] Sutton, R.S.; Barto, A.G., "Reinforcement Learning: An Introduction," *IEEE Transactions on Neural Networks*, vol.9, no.5, pp.1054, 1998.

[5] L. P. Kaelbling; M. L. Littman; A. W. Moore, "Reinforcement learning: a survey," *Journal of Artificial Intelligence Research*, vol.4, no.2, pp. 237-285, 1996.

[6] Nie, Junhong; Haykin, Simon, "A Q-learning-based dynamic channel assignment technique for mobile communication systems," *IEEE Transactions on Vehicular Technology*, vol.48, no.5, pp.1676-1687, 1999

[7] Senouci S; Beylot A; Pujolle G, "Call admission control in cellular networks: A reinforcement learning solution," *International Journal of Network Management*, vol.14, no.2, pp. 89-103, 2004.

[8] Haddad, M.; Altman, Z.; Elayoubi, S.E.; Altman, E., "A Nash-Stackelberg Fuzzy Q-Learning Decision Approach in Heterogeneous Cognitive Networks," *Global Telecommunications Conference (GLOBECOM 2010)*, pp.1-6, 2010.

[9] Simsek, M.; Czylwik, A., "Decentralized Q-learning of LTE-femtocells for interference reduction in heterogeneous networks using cooperation," *2012 International ITG Workshop on Smart Antennas (WSA)*, pp.86-91, 2012.

[10] Saker, L.; Ben Jemaa, S.; Elayoubi, S-E, "Q-Learning for Joint Access Decision in Heterogeneous Networks," *Wireless Communications and Networking Conference*, pp.1-5, 2009.

[11] Tabrizi, H.; Farhadi, G.; Cioffi, J., "Dynamic handoff decision in heterogeneous wireless systems: Q-learning approach," *2012 IEEE International Conference on Communications (ICC)*, pp.3217-3222, 2012.

[12] Zhang, Y.; Feng, Z.; Zhang, P., "A Q-learning Based Autonomic Join Radio Resource Management Algorithm," *Journal of Electronics and Information Technology*, vol.33, no.3, pp.676–680, 2008.

[13] Zhao, Y.; Zhou, W.; Zhu, Q., "Q-Learning Based Heterogeneous Network Selection Algorithm, " *Recent Advances in Computer Science and Information Engineering Lecture Notes in Electrical Engineering*,vol.127, no.4, pp.471-477, 2012.

[14] G. Barto; S. J. Bradtke; S. P. Singh, "Learning to act using real-time dynamic programming," *Artificial Intelligence*, vol.72, pp.81-138, 1995.

[15] Watkins C.J.C.H; Dayan P., "Q-learning," *Machine Learning*, vol.8, no.3, pp.279-292, 1992.