Reinforcement Learning Based Seamless Handover Algorithm in SDN-Enabled WLAN

Jianjun Lei[1](✉), Xin Liu[1], Ying Wang[1], Xunwei Zhao[2], and Ping Gai[2]

[1] School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
leijj@cqupt.edu.cn

[2] State Grid Information and Telecommunication Group CO., LTD., Beijing 102211, China
{zhaoxunwei,gaiping}@sgitg.sgcc.com.cn

Abstract. In Wireless Local Area Network (WLAN), Station (STA) performs handover according to received signal strength indication (RSSI), which results in long handover delay and inability during selecting the best Access Point (AP). This paper proposes T-DQN (enhanced Throughput based on Deep Q-Network), a novel intelligent handover algorithm based on deep reinforcement learning (DRL), which is modeled as a Markov decision process (MDP) and enables network to dynamically handover according to current network states. Meanwhile, the T-DQN not only considers the STA throughput, but also takes into account system throughput and fairness of Basic Service Set (BSS) during the handover process. Moreover, we employ the software defined network (SDN) to perform a centralized control and seamless handover of network. Simulation results demonstrate that the T-DQN can effectively improve STA throughput and system throughput, and also achieve the lower packet loss rate, which outperforms some traditional handover schemes.

Keywords: WLAN · SDN · Reinforcement learning · Handover

1 Introduction

WLAN has been widely used because of its high data-rate and convenience [1]. In traditional handover scheme, When RSSI between STA and AP is lower than threshold, handover is triggered [2]. And it takes a lot of time for STA to complete the process of associating with target AP. In recent years, the emergence of software-defined networking (SDN) technology is promoting network architecture. SDN decouples control and data layers of network, and controller provides programmable interfaces that administrators can control and manage network through it [3, 4].

In this paper, we propose an intelligent seamless handover algorithm based on DRL. Firstly, we design a network architecture based on software defined wireless network (SDWN). Wherein, the STA is associated with a unique logic AP (LAP), and therefore the soft handover of STA is operated only by removing and adding LAP on AP. Secondly, we propose T-DQN handover algorithm, which can choose the trigger time of handover

Z. Liu et al. (Eds.): WASA 2021, LNCS 12939, pp. 318–326, 2021.
https://doi.org/10.1007/978-3-030-86137-7_35

and the better target AP according to current network condition. Finally, we design a sophisticated reward mechanism for our proposed T-DQN algorithm that considers not only the throughput of STA itself, but also the system throughput and the fairness of BSS that the STA is associating. The simulation results verify that our proposed algorithm improves greatly the network performance in terms of throughput and packet loss rate.

2 Related Work

In traditional WLAN handover process, the communication between STA and AP will be interrupted for a few seconds [5]. In [6], the authors show that communication interruption is between 13 ms and 30 ms during handover process. In IEEE 802.11 protocol, a complete handover process is divided into three parts: scanning, authentication, and reassociation. Some documents suggest enhancing scanning phase to reduce the total handover delay. In [7], APs are equipped with dual interfaces, one of interfaces broadcasts beacon frames to scan channels through channel used by AP. Although this method can reduce scanning delay, AP's two interfaces will cause additional expenses. In [8], the authors use the neighbor graph model, which is used to store working channels of neighbor APs and construct neighbor graph of APs, but when neighbor APs are distributed in all channels, this method does not decrease scanning delay. In [9], the algorithm equips a redundant AP for each AP, and the redundant AP collects channel states for whole system, Although the mechanism reduces the number of STA scanning channels, however, redundant APs cause waste of resources. In [10], the authors propose a scheme that uses multiple physical APs to provide multiple transmission paths for a STA. Although it improves network performance, also faces very complex flow scheduling problem. In [11], the authors propose an architecture that combines SDN, deep neural network, recurrent neural network, and convolutional neural network, and present a handover algorithm based on Q-learning, although this algorithm improves throughput of mobile STA, it does not consider system throughput.

3 Proposed Algorithm

3.1 System Model

This section mainly introduces network scenarios. We consider an environment, which contains N STAs and M APs. All STAs continuously send data to APs. Since we focus on handover problem under the same frequency deployment scenario, all APs are located on same channel. In addition, our algorithm is based on the SDWN architecture, so the controller controls and monitors all STAs and APs, as shown in Fig. 1. The controller obtains the RSSI of all BSSs through OpenFlow protocol. If the RSSI of some BSSs is too low to be measured, we will set their value as a lower value than the minimum measured value.

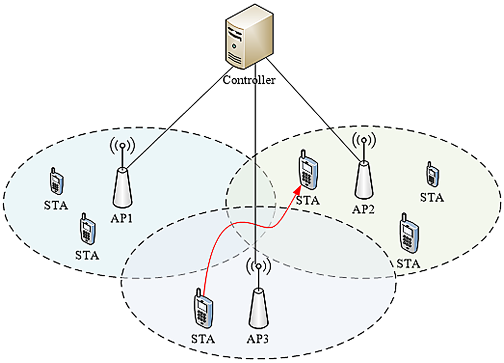


Fig. 1. Network scenario

3.2 System Architecture and Modules

In order to overcome limitations and challenges of handover algorithms in WLAN, we propose a SDWN-based network architecture, as shown in Fig. 2. This architecture separates control layer from data layer, allowing the controller to logically concentrate the control network to achieve network programmability. The DQN architecture includes three layers: physical layer, control layer and application layer.

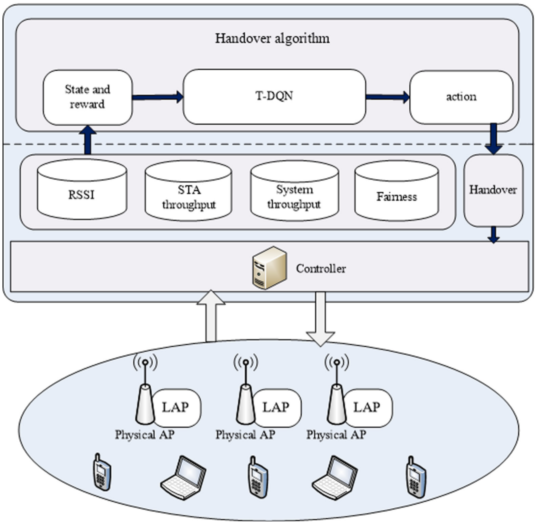


Fig. 2. DQN architecture

Data layer is mainly responsible for following functions: 1) Using LAP technology to achieve seamless handover; 2) forming network topology and data transmission between nodes; 3) transmitting real-time network information to control layer through southbound interface. Controller layer is mainly responsible for following functions: 1) Obtaining underlying network states information through OpenFlow protocol; 2) assigning LAP to STA; 3) Controlling handover of STA. When the STA accesses the network, the controller will detect whether the STA connects a LAP, and will assign a unique LAP for the STA if it still does not. In addition, the controller controls handover of STA by removing or adding LAP on AP, so the entire handover process is transparent to STA. Application layer is mainly responsible for following functions: 1) Obtaining information through northbound interface as the input to algorithm; 2) passing the input of algorithm through a fully connected neural network to estimate the next action; 3) outputting the command to control handover of the entire network.

3.3 Handover Algorithm

MDP Model. The construction of MDP is based on agent and environment, and its elements include state, action and reward. Wherein, the current network system is used as environment and controller is used as agent, so controller is the decision maker and the learner in reinforcement learning system.

The state space contains all network states and is defined as:

$$S = \{s_1, s_2, s_3, \ldots, s_n, \ldots\} \tag{1}$$

The state of the STA collected by the controller from the network at time t is defined as:

$$s_t = \{RSSI_1, RSSI_2, \ldots, RSSI_{M-1}, RSSI_M\} \tag{2}$$

Where M is the number of APs. $RSSI_1$ represents RSSI between STA and AP_1.

In addition, the agent obtains different rewards by taking actions, and we set action space as A:

$$A = \{a_1, a_2, \ldots, a_{M-1}, a_M\} \tag{3}$$

The action space has M dimensions, so M is the number of APs, a_i means that STA handover to AP_i. If the algorithm selects AP that STA is associated with, the STA does not handover; if unavailable AP is selected, the environment will give a negative feedback to the controller.

The reward at time t is defined as:

$$r_t = w_1\left(\sum_{j=1}^{M} T_j^t - \sum_{j=1}^{M} T_j^{t-1}\right) + w_2 T_{k,j}^t (1 - \sigma) \tag{4}$$

w_1 and w_2 are weights, and $w_1 + w_2 = 1.T_j^t$ represents the throughput of AP_j at time t, and $\sum_{j=1}^{M} T_j^t$ is the system throughput, T_j^t is defined as:

$$T_j^t = \sum_{i=1}^{N} T_{i,j}^t \tag{5}$$

N is the number of STAs associated with AP_j, $T_{i,j}^t$ is the throughput from STA$_i$ to AP$_j$. $\sum_{i=1}^{N} T_{i,j}^t$ represents the sum of the throughput of all STAs associated with AP$_j$, which is also the throughput of BSS_j. STA_k is observation STA. σ is the throughput fairness of BSS_j:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}\left(T_{i,j}^{t} - \overline{T}_{j}^{t}\right)^{2}}{N}} \tag{6}$$

At time t, STA_i is sending data to AP_j, $T_{i,j}^{t}$ represents the throughput of STA_i. $\overline{T}_{j}^{t}$ is the average throughput of AP_j, $\overline{T}_{j}^{t}$ be calculated as:

$$\overline{T}_{j}^{t} = \frac{1}{N}\sum_{i=1}^{N} T_{i,j}^{t} \tag{7}$$

T-DQN. In T-DQN, the input of algorithm is RSSI between STA and AP, and the output is association between STA and target AP. The algorithm initializes experience pool D, its maximum capacity is B, and then preprocesses environment, and inputs state to the evaluation network and the evaluation network returns all Q values in state. In this algorithm, the action is selected according to greedy strategy, that is, a random action with probability ε is selected, and vice versa, action with the highest Q value is selected. After selecting an action, the controller executes action, and updates states and receives rewards. Next, the algorithm stores experience in experience pool, then randomly extracts experience from experience pool, calculates loss function, and then performs gradient descent to minimize loss. Finally, for every step, the evaluation network passes the parameters to the target network until the entire training finishes.

Algorithm 1: T-DQN algorithm

```
Inputs: State information, $s_t$
Output: Association with target AP, $a_t$
Initialize experience pool $D$ to capacity $B$
Initialize the scenario parameters $M$, $A$
Initialize the statistic parameters $Z$, $\varepsilon_0$, $\gamma$, $\alpha$, $\beta$, $few\_steps$
Initialize evaluation network $Q$ with random parameters $\theta$
Initialize the target network $\hat{Q}$ with $\hat{\theta} = \theta$
Read state $s_t$
$\varepsilon = \varepsilon_0$
for $t = 1$ to $Z$ do
    Input $s_t$ to evaluation network and output the Q-values of actions
    Randomly select actions $a_t$ from $A$ with ε-greedy strategy
    $\varepsilon = \varepsilon * \beta$
    Execute action $a_t$, handover to target AP
    Obtain rewards $r_t$ by calculating real-time throughput of STAs
    Next state $s_{t+1}$
    Store transition $(s_t, a_t, r_t, s_{t+1})$ in $D$
    Sample minibatch of transition $(s_j, a_j, r_j, s_{j+1})$ from $D$
    Perform gradient descent s on $(r_j + \gamma \max \hat{Q}\left(s_{j+1}, a_{j+1}; \theta\right) - Q\left(s_j, a_j; \theta\right))^2, a_{j+1} \in A$
    $s_t = s_{t+1}$
    Every $few\_steps$ step, update target network with $\hat{\theta} = \theta$
end for
```

4 Performance Evaluation

4.1 Simulation Settings

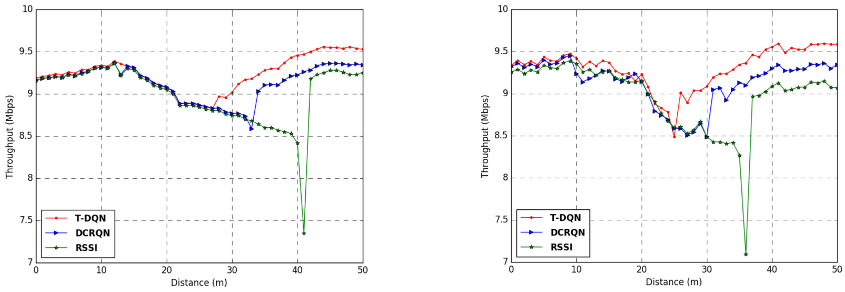
In this section, we use Mininet-WiFi to evaluate the performance of T-DQN and compare with a traditional handover algorithm based on RSSI and DCRQN [11] algorithm based on DQN. WLAN is deployed in a space with an area of 200 m * 200 m. All STAs are connected to network during simulation and continuously send data of different priorities to AP. At first, one of STAs is in moving state as observation station, and the other STAs and APs are in static state. Observation station is advancing at a constant speed from a fixed starting point at a speed of 0.5 m/s. The parameters used in the simulation are shown in Table 1.

Table 1. Simulation parameters

Parameters	Value	Parameters	Value
STA Number	13	Frequency band	2.4 GHz
AP Number	9	Protocol	IEEE 802.11g
CCA Threshold	−62 dBm	Channel	1
Propagation Loss Model	Log distance	Receiving gain	5 dBm
Transmitting power	21 dBm	Transmitting gain	5 dBm

4.2 Simulation Results

STA Throughput. In order to verify the throughput performance of T-DQN algorithm, we let the STA governed by UDP and TCP protocols to transmit data respectively. Figure 3(a) and Fig. 3(b) show throughput changes of mobile STA using TCP and UDP streams to send data.



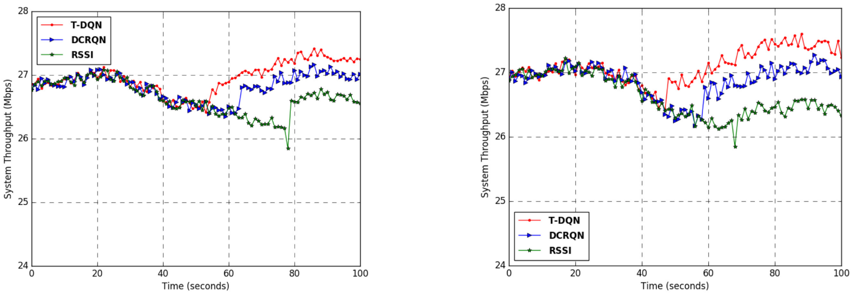
(a) throughput of STA using TCP streams

(b) throughput of STA using UDP streams

Fig. 3. STA throughput performance of the three algorithms under different streams

When mobile STA uses TCP to send data, T-DQN, DCRQN and RSSI handover at 27 m, 33 m, and 41 m. When mobile STA uses UDP to send data, algorithms T-DQN, DCRQN, and RSSI handover at 25 m, 30 m, and 36 m. Because UDP is unreliable transmission, when UDP transmission is used, throughput of STA fluctuates greatly per second. Because RSSI is traditional handover scheme, change scheme needs that STA disconnect previous connection before handover to new AP, so handover delay is longer. It can be observed that no matter whether UDP or TCP stream is used to transmit data, T-DQN handover earlier than DCRQN and RSSI. After handover, the AP can provide higher bandwidth.

System Throughput. The Fig. 4(a) and Fig. 4(b) show changes in the system throughput of TCP and UDP streams. At the beginning of simulation, there was no handover, the system throughput of the three algorithms changes similarly, but as observation STA moved, the link between STA and current connection gradually slackened, resulting in a slight decrease in system throughput. T-DQN, DCRQN and RSSI algorithms handover in sequence because T-DQN selects APs with the better performance for associating and triggers handover in time, which makes reasonable use of network resources and improves the system throughput.



(a) system throughput of STAs using TCP (b) system throughput of STAs using UDP

Fig. 4. System throughput of the three algorithms under different streams

Packet Loss Rate. Figure 5 shows change in packet loss rate of mobile STA. In RSSI algorithm, STA will disconnect from current AP during handover, and then associates with the other target AP. The entire process is controlled by STA, so handover delay is very long and results in a packet loss rate as high as 83%. DCRQN and T-DQN adopt SDWN-based network architecture, and controller is responsible for performing STA handover, and can perform seamless handover.

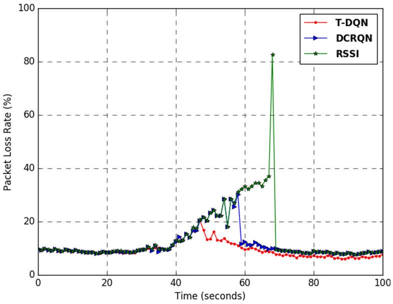


Fig. 5. Packet loss rate of mobile STA

5 Conclusion

In this paper, we propose a seamless handover algorithm for WLAN based on SDN and DQL. We present a network architecture that can utilize the controller as the agent and considers the system throughput, STA throughput and fairness of BSS as the reward of DQL. Meanwhile, the agent can interact dynamically with the environment when the STA's handover occurs, and thus an optimal policy capable of maximizing the long-term returns is eventually obtained. Finally, the simulation results by Mininet-WiFi platform demonstrate that our proposed algorithm can significantly improve the performance of handover in terms of the throughput and the packet loss rate.

References

1. Coronado, E., Gomez, B., Riggio, R.: A network slicing solution for flexible resource allocation in SDN-BASED WLANs. In: 2020 IEEE Wireless Communications and Networking Conference Workshops, WCNCW, pp. 1–2. IEEE, Seoul (2020)
2. Lei, J., Tao, J., Yang, S.: Joint AP association and bandwidth allocation optimization algorithm in high-dense WLANs. Future Internet **10**(8), 73–87 (2018)
3. Ahmad, S., Mir, A.H.: Scalability, consistency, reliability and security in SDN controllers: a survey of diverse SDN controllers. J. Netw. Syst. Manage. **29**(1), 1–59 (2020). https://doi.org/10.1007/s10922-020-09575-4
4. Lembke, J., Ravi, S., Eugster, P., et al.: RoSCo: robust updates for software-defined networks. IEEE J. Sel. Areas Commun. **38**(7), 1352–1365 (2020)
5. Feirer, S., Sauter, T.: Seamless Handover in Industrial WLAN Using IEEE 802.11k. In: 2017 IEEE 26th International Symposium on Industrial Electronics, ISIE, pp. 1234–1239. IEEE, Edinburgh (2017)
6. Fernández, Z., Seijo, Ó., Mendicute, M., et al.: Analysis and evaluation of a wired/wireless hybrid architecture for distributed control systems with mobility requirements. IEEE Access **7**, 95915–95931 (2019)
7. Jeong, J.P., Park, Y.D., Suh, Y.J.: An efficient channel scanning scheme with dual-interfaces for seamless handoff in IEEE 802.11 WLANs. IEEE Commun. Lett. **22**(1), 169–172 (2018)
8. Ma, H., Zhou, Z., Chen, X.: Leveraging the power of prediction: predictive service placement for latency-sensitive mobile edge computing. IEEE Trans. Wireless Commun. **19**(10), 6454–6468 (2020)

9. Sun, Y., Liu, Y., Jiang, L.: A scheme of seamless handover in wireless communications based on sentinel mechanism. In: 2019 15th International Wireless Communications and Mobile Computing Conference, IWCMC, pp. 1725–1730. IEEE, Tangier (2019)
10. Guimaraes, R.S., García, V.M., Mello, R.C., et al.: An SDN-NFV orchestration for reliable and low-latency mobility in off-the-shelf WiFi. In: ICC 2020–2020 IEEE International Conference on Communications, ICC, pp. 1–6. IEEE, Dublin (2020)
11. Han, Z., Lei, T., Lu, Z., et al.: Artificial intelligence based handoff management for dense WLANs: a deep reinforcement learning approach. IEEE Access **7**, 31688–31701 (2019)