Q-Learning-Based Spatial Reuse Method Considering Throughput Fairness by Negative Reward for High Throughput

Mirai Takematsu[1(✉)], Shota Sakai[1], Masashi Kunibe[1], and Hiroshi Shigeno[2]

[1] Graduate School of Science and Technology, Keio University, Yokohama, Kanagawa 223-8522, Japan
{takematsu,sakai,kunibe}@mos.ics.keio.ac.jp
[2] Keio University, Yokohama, Kanagawa 223-8522, Japan
shigeno@mos.ics.keio.ac.jp

Abstract. In this paper, we propose a Q-learning-based spatial reuse method considering throughput fairness in Wireless LANs (WLANs). In Spatial Reuse (SR) methods, wireless nodes try to use wireless resources efficiently by controlling both the Transmission Power (TP) and Carrier Sense Threshold (CST). When wireless nodes are densely deployed, the SR methods have difficulty to achieve both the high aggregate throughput and throughput fairness because the mutual interference among the wireless nodes becomes severe. The proposed method removes the difficulty by utilizing Q-learning where wireless nodes can learn the adequate CST and TP by themselves. The proposed method motivates nodes to use wireless resources actively by rewards, while it suppresses nodes with high throughput using the resources by negative rewards. As a result, the wireless resources are distributed among nodes with low throughput, and the proposed method achieves both the high aggregate throughput and throughput fairness. Simulation results show that the proposed method improves the aggregate throughput with keeping throughput fairness.

Keywords: Dense Wireless LAN · Spatial reuse · Q-learning

1 Introduction

With the increase of wireless nodes such as smartphones and Access Points (APs), the efficient wireless resource utilization has been paid attention in Wireless Local Area Networks (WLANs). In WLANs, the wireless nodes send packets avoiding packet collisions for utilizing the wireless resources efficiently. The simultaneous packet transmissions on the same channel will be results in packet collisions and the involved packets will be lost. The IEEE 802.11 standard implements Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) to avoid the packet collisions [1]. In CSMA/CA, nodes perform carrier sense before

T. Hara and H. Yamaguchi (Eds.): MobiQuitous 2021, LNICST 419, pp. 207–219, 2022.
https://doi.org/10.1007/978-3-030-94822-1_12

transmitting packets to detect transmissions from the other nodes. Specifically, the nodes detect the transmission if the received signal power exceeds Carrier Sense Threshold (CST). However, when the nodes are densely deployed in WLANs, they have difficulty to utilize wireless resources efficiently even by CSMA/CA. Specifically, they suffer from the following two problems in dense environments: the hidden node problem and the exposed node problem [2]. In the hidden node problem, packets of different nodes frequently collide at receiving nodes because they cannot recognize with each other by carrier sense due to the high CST of them or low Transmission Power (TP). In the exposed node problem, nodes rarely send packets because they perform carrier sense excessively due to the low CST of them or high TP of others.

To mitigate the hidden node problem and exposed node problem, spatial reuse methods have been paid attention as the methods allow nodes to conduct adequate carrier sense using Dynamic Sensitivity Control (DSC) and Transmit Power Control (TPC). Specifically, DSC allows nodes to change their CST, and TPC allows nodes to change their TP. On the other hand, the spatial reuse methods have difficulty to conduct adequate DSC and TPC in dense environments because the mutual interference among nodes becomes complex. To solve the difficulty, the spatial reuse methods using Machine Learning (ML), such as Neural Network (NN) and Q-learning, have been paid attention. By utilizing ML, the nodes can learn adequate DSC and TPC autonomously without a lot of knowledge of the complex interference among them. In NN-based methods, nodes learn their controls by training data, and the methods have to create the data based on the positions of nodes previously [3,4]. In contrast, the nodes learn their controls only by rewards in Q-learning-based methods. Therefore, Q-learning-based methods are more flexible than NN-based methods for learning DSC and TPC.

One of the challenges of Q-learning-based methods is to determine rewards for both the high aggregate throughput and throughput fairness. F. Wilhelmi et al. introduced the throughput of each node as the reward to achieve the high aggregate throughput [5]. Since the nodes try to obtain high rewards, the aggregate throughput of them also becomes high. However, the reward has a possibility to allocate wireless resources to nodes unfairly for the high aggregate throughput. To achieve throughput fairness, F. Wilhelmi et al. also introduced the minimum throughput of nodes in networks as the reward [5]. Since the nodes try to obtain high rewards, the minimum throughput in the network also increases. As a result, the difference between the minimum throughput and the maximum throughput in the network decreases, and throughput fairness is achieved. On the other hands, the reward has a possibility to suppress the aggregate throughput because it is determined based on the minimum throughput.

In this paper, we propose a Q-learning-based spatial reuse method considering throughput fairness for high throughput. The goal of the proposed method is that wireless nodes learn the adequate CST and TP for both the high aggregate throughput and throughput fairness by rewards of Q-learning. To achieve the high aggregate throughput, we utilize the throughput of nodes as rewards of

them. As a result, they seek the CST and TP that improve the throughput of them to obtain high rewards. In contrast, to prevent some nodes from monopolizing wireless resources, we give the negative rewards to the nodes with the high throughput. As a result, they learn to suppress wireless resources by DSC and TPC. The wireless resources of the nodes with the high throughput are distributed among the nodes with the low throughput, and they finally learn CST and TP that achieve throughput fairness among them.

The contribution of this paper is as follows:

- To achieve both the high aggregate throughput and throughput fairness, we propose the Q-learning-based spatial reuse method using the negative reward for the high throughput.
- Simulation results show that the proposed method improves the aggregate throughput with keeping throughput fairness in the comparison with the previous methods.

The remainder of this paper is organized as follows: Sect. 2 explains related work. Section 3 presents the proposed method. Section 4 shows the evaluation results. Section 5 concludes this paper.

2 Related Work

For achieving the efficient wireless resource utilization, spatial reuse methods have been paid attention. The challenging task of the spatial reuse methods is to achieve both the high aggregate throughput and throughput fairness by using DSC and TPC in dense environments. Since DSC and TPC of nodes influence on the those of the other nodes, it becomes difficult for them to conduct DSC and TPC considering the mutual influence on the others in dense environments [5]. To remove the difficulty from DSC and TPC, ML-based spatial reuse methods, such as NN-based methods [3,4] and Q-learning-based methods, have been paid attention. In ML, nodes can learn the optimal CST and TP autonomously without much knowledge of the mutual influence. Specifically in NN-based methods, the nodes learn their controls based on training data. On the other hands, the data depends on the positions of nodes. In contrast, the nodes can learn their controls only by rewards for the controls in Q-learning methods. Therefore, Q-learning-based-methods can easily be implemented rather than NN-based-methods.

First, we explain details of Q-learning in Sect. 2.1. Then, we explain Q-learning-based spatial reuse methods in Sect. 2.2. Finally, we explain the motivation of this paper in Sect. 2.3.

2.1 Q-Learning

Q-learning [6] is one of RL methods, and the purpose of Q-learning is that agents learn optimal actions by updating expected rewards in Q-tables. Q-tables are composed of expected rewards, states, and actions. Specifically, the expected rewards are allocated to pairs of states and actions in Q-tables. At each time

step, agents select actions for states from their Q-tables. Then, Q-learning gives rewards for the actions. By using the rewards, the agents update the expected rewards for the actions and states. Specifically, when agent i gets reward $r_{i,t}$ for action a_t and state s_t at time t, it updates Q-table $\hat{Q}(s_t, a_t)$ for the action and state as follows:

$$\hat{Q}(s_t, a_t) \leftarrow (1 - \alpha_t)\hat{Q}(s_t, a_t) + \alpha_t(r_{i,t} + \gamma(\max_{a'} \hat{Q}(s_{t+1}, a'))), \tag{1}$$

where α_t and $\max_{a'} \hat{Q}(s_{t+1}, a')$ denote a learning rate at time t and a maximum expected reward for next state s_{t+1}, respectively, and γ is a discount factor parameter. On the other hands, Eq. (1) cannot be applied with the case that agents cannot observe their states completely. Especially, WLANs belong to the case because wireless nodes cannot observe all the other nodes. To apply Q-learning with the case, stateless Q-learning was proposed [7]. The stateless Q-learning can be applied with the case that there is one state and one optimal action for the state. Q-tables are composed of the expected rewards and the actions. Specifically, when agent i gets reward $r_{i,t}$ for action a_t at time t, it updates Q-table $\hat{Q}(a_t)$ for the action as follows:

$$\hat{Q}(a_t) \leftarrow (1 - \alpha_t)\hat{Q}(a_t) + \alpha_t(r_{i,t} + \gamma(\max_{a'} \hat{Q}(a'))). \tag{2}$$

In Q-learning, agents have the possibility to learn suboptimal actions when they continuously select actions with the highest expected rewards among all the actions in Q-tables at every time step. To decrease the possibility, ε-greedy strategy was proposed [8]. In ε-greedy strategy, the agents select their actions randomly from their Q-tables with probability ε. In contrast, with probability $1 - \varepsilon$, they select the actions with the highest expected rewards among all the actions in Q-tables.

2.2 Q-Learning-Based Spatial Reuse Methods

The goal of Q-learning-based spatial reuse methods is to define adequate rewards for achieving both the high aggregate throughput and throughput fairness. By defining the rewards, nodes can learn adequate CST and TP by themselves.

F. Wilhelmi et al. introduced the throughput of nodes as the selfish reward for achieving the high aggregate throughput [5]. Specifically, reward $r_{i,t}$ for node i at time t is expressed as:

$$r_{i,t} = \frac{\Gamma_{i,t}}{\Gamma_i^*}, \tag{3}$$

where $\Gamma_{i,t}$ and Γ_i^* denote the throughput of node i at time t and the maximum achievable throughput of node i, respectively. The reward motivates node i to learn CST and TP that improve $\Gamma_{i,t}$. As a result, the aggregate throughput is also improved by the reward. Furthermore, to achieve throughput fairness, F. Wilhelmi et al. proposed environment-aware reward considering the minimum

throughput in networks [5]. Specifically, reward $r_{\mathscr{O},t}$ for the nodes in network $\mathscr{O}$ at time t is expressed as:

$$r_{\mathscr{O},t} = \frac{\min_{i \in \mathscr{O}} \Gamma_{i,t}}{\Gamma^*_{\mathscr{O},t}}, \tag{4}$$

where $\min_{i \in \mathscr{O}} \Gamma_{i,t}$ and $\Gamma^*_{\mathscr{O},t}$ denote the minimum throughput and the maximum throughput in network $\mathscr{O}$ at time t, respectively. The reward motivates the nodes to learn CST and TP that improve the minimum throughput in the network. As a result, the difference between the minimum throughput and maximum throughput gradually decreases, and throughput fairness is achieved.

2.3 Motivation

Although the selfish reward-based method improves the aggregate throughput, the method does not consider throughput fairness. As a result, the method allocates wireless resources to nodes unfairly. Although the environment-aware reward-based method can allocate the wireless resources to the nodes fairly, the method has the limitation to improve the aggregate throughput because the reward is determined based on the minimum throughput in networks. Motivated by the above, we argue that there is a room to improve the aggregate throughput with keeping throughput fairness among the nodes.

3 Proposal

In this paper, we propose the Q-learning-based spatial reuse method considering throughput fairness by negative reward for high throughput. In the proposed method, wireless nodes learn the adequate CST and TP for the high aggregate throughput and throughput fairness by rewards of Q-learning. The main idea of the proposed method is that it distributes wireless resources of nodes with the high throughput among nodes with the low throughput. To distribute the wireless resources, the proposed method gives negative rewards to the nodes with the high throughput. As a result, the nodes with the high throughput learn to suppress the wireless resource utilization by DSC and TPC. In contrast, the proposed method gives rewards to the nodes with low throughput for motivating them to use the wireless resources actively. As a result, the nodes with the low throughput learn to obtain the wireless resources from the nodes with the high throughput by TPC and DSC, and they achieve both the high aggregate throughput and throughput fairness.

First, we explain the system model of the proposed method in Sect. 3.1. Then, we show the flowchart of the proposed method in Sect. 3.2. Finally, we explain the algorithm of the proposed method in Sect. 3.3.

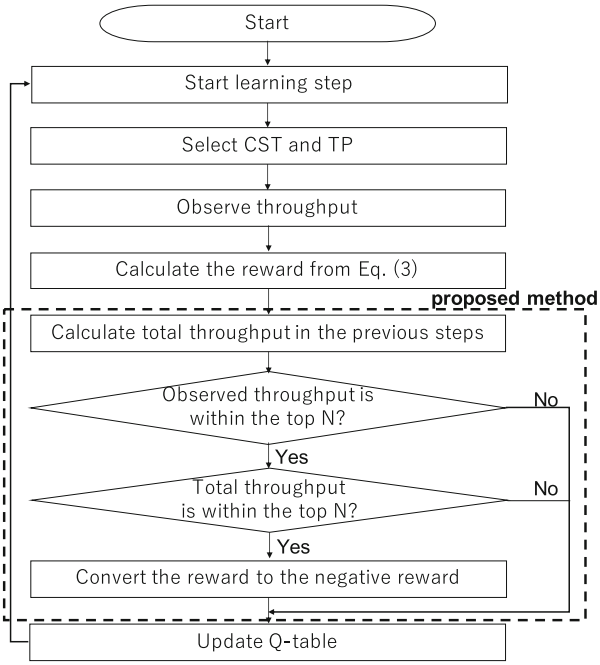


Fig. 1. Flowchart of the proposed method

3.1 System Model

We assume that wireless nodes are composed of wireless Stations (STAs) and APs. The Q-learning-based methods including the proposed method are implemented in the STAs. We assume that the STAs can estimate the throughput of all the STAs and APs by observing channels. The STAs conduct carrier sense using DSC and TPC and send packets to neighbor nodes. The APs only receive the packets. At each time step, STAs estimate the throughput of the others and calculate their rewards based on the throughput. After the calculation, the STAs determine CST and TP based on their rewards.

3.2 Flowchart of the Proposed Method

Figure 1 shows the flowchart of the proposed method. The proposed method is implemented in STAs. First, the STAs select the CST and TP based on their Q-tables. After that, they observe the throughput of them. By using the throughput, they calculate their rewards based on Eq. (3). In the proposed method, if the throughput of them is higher than that of other STAs, the proposed method converts their rewards to negative rewards. Specifically, the proposed method gives the negative reward to STA i if throughuput $\Gamma_{i,t}$ of STA i at time t satisfies with the following two conditions:

Algorithm 1. Algorithm for Determing Action of STA i

Input: $\mathcal{A}$: set of possible action, $|A| = K$
Initialize : timestep $t = 0$, $\varepsilon_0 = 1.0$, $\hat{Q}(a_k) = 0$, $a_k \in \mathcal{A}$
while true **do**
Action selection

$$a_t = \begin{cases} \arg\max_{1,\ldots,K} \hat{Q}(a_k) & \text{with probabikity } 1-\varepsilon \\ \text{Randomly select from } \hat{Q} & \text{with probability } \varepsilon. \end{cases}$$

Reward calculation

$$r_{i,t} = \begin{cases} -\frac{\Gamma_{i,t}}{\Gamma_i^*} & \text{if throughput } \Gamma_{i,t} \text{ satisfies conditions (5) and (6),} \\ \frac{\Gamma_{i,t}}{\Gamma_i^*} & \text{otherwise.} \end{cases}$$

Q-table updation
$\hat{Q}(a_t) \leftarrow (1-\alpha_t)\hat{Q}(a_t) + \alpha_t(r_{i,t} + \gamma(\max_{a'} \hat{Q}(a')))$
$\varepsilon_t \leftarrow \frac{\varepsilon_0}{\sqrt{t}}, t \leftarrow t+1$
end while

$$|\{j \mid \Gamma_{i,t} \leq \Gamma_{j,t}, 1 \leq j \leq n\}| \leq N, \tag{5}$$

$$|\{j \mid \Gamma_{i,sum} \leq \Gamma_{j,sum}, 1 \leq j \leq n\}| \leq N, \quad \Gamma_{i,sum} = \sum_{t=1}^{T} \Gamma_{i,t}, \tag{6}$$

where n and T denote the number of the STAs in the network and a current timestep, respectively. Equation (5) means that STA i is in the top N STAs with the highest throughput among all the STAs. Equation (6) means that STA i is in the top N STAs with the highest total throughput among all the STAs. If STAs satisfy with conditions (5) and (6), they convert their rewards by multiplying the rewards by -1. Finally, they update their Q-tables based on Eq. (2).

3.3 Algorithm for Determing CST and TP

In the proposed method, STAs select the CST and TP with high rewards from Q-tables and update their Q-tables with rewards. The algorithm of the proposed method is shown in Algorithm 1. The algorithm is composed of action selection, reward calculation, and Q-table updation.

In the action selection, STAs select their actions based on their Q-tables. The actions are the pairs of CST and TP. The STAs select their actions from their Q-tables with ε-greedy strategy [8]. Specifically, with probability ε, they randomly select their actions from their Q-tables. In contrast, with probability $1-\varepsilon$, they select their actions with the highest rewards among all the actions in their Q-tables. The proposed algorithm decreases the value of ε as time passes because the STAs do not need to explore for adequate actions as Q-learning progresses. In the reward calculation, they calculate their rewards using the throughput of them. Specifically, throughput $\Gamma_{i,t}$ of STA i at time t is calculated as:

$$\Gamma_{i,t} = \frac{\beta^i_{(t,t-T_{observe})}}{T_{observe}}, \tag{7}$$

Table 1. Simulation parameters

Wi-Fi standard	IEEE802.11ac
Frequency band (GHz)	5
Channel number	38
Channel bandwidth (MHz)	20
MCS	7
Propagation loss	Residential path loss model [9]
Fading/Shadowing	None
Mobility model	Static
Traffic model	CBR
Traffic load	Full buffer (uplink)
Max Aggregation	64
RTS/CTS	None
Antenna gain (dBi)	0
Noise figure (dBm)	7
TP (dBm)	AP: 20, STA: {3, 5, 8, 11, 14, 17, 20, 23}
CST (dBm)	AP: −76, STA: {−82, −79, −76, −73, −70, −67, −64, −62}
Simulation time (s)	10000
Timestep (s)	0.5
ε_0	1.0
α_t	0.5
γ	0.9
$T_{observe}$ (s)	0.5

where $T_{observe}$ and $\beta^i_{(t,t-T_{observe})}$ denote the observation time for the throughput and the number of successfully received bits of the STA i between $t - T_{observe}$ and t, respectively. We assume that the STAs can observe the throughput of the other STAs by observing channels. By using the throughput, they calculate their rewards based on Eq. (3). If the throughput of them satisfies with the conditions (5) and (6), they convert their rewards by multiplying the rewards by −1. Finally, in the Q-table updation, they update their Q-tables by their rewards based on Eq. (2).

4 Evaluation

We assume the scenario as one floor of 10 × 2 apartments. The size of one apartment is 10 m × 10 m × 3 m. We deployed one STA and one AP to each apartment, and the STAs and APs were randomly placed in the apartments. The height of all the STAs and APs was fixed to 1.5 m. In this scenario, the STAs created packets based on the Wi-Fi standard at regular interval, and the

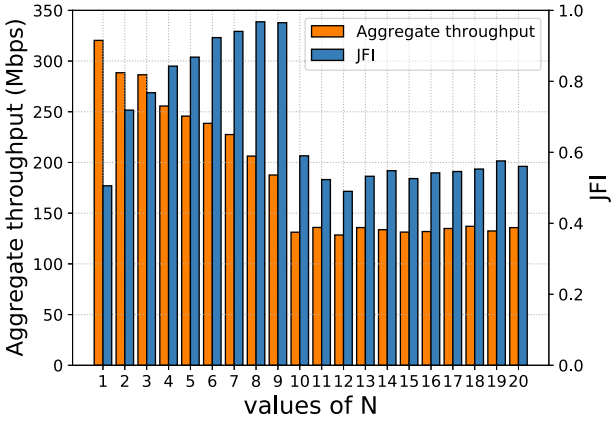


Fig. 2. Aggregate throughput and JFI with changing N in the proposed method.

APs only received the packets. We used ns-3.30.1 as a network simulator [10]. The propagation loss was calculated using the residential path model [9]. The simulation parameters are shown in Table 1.

We compare the proposed reward (Proposal) with the selfish reward (Selfish) and environment-aware reward (Env) in terms of the aggregate throughput and throughput fairness in Sect. 4.2. The selfish reward and environment-aware reward are calculated by Eqs. (3) and (4). To evaluate throughput fairness, we utilized Jain's Fairness Index (JFI) [11] that is calculated as:

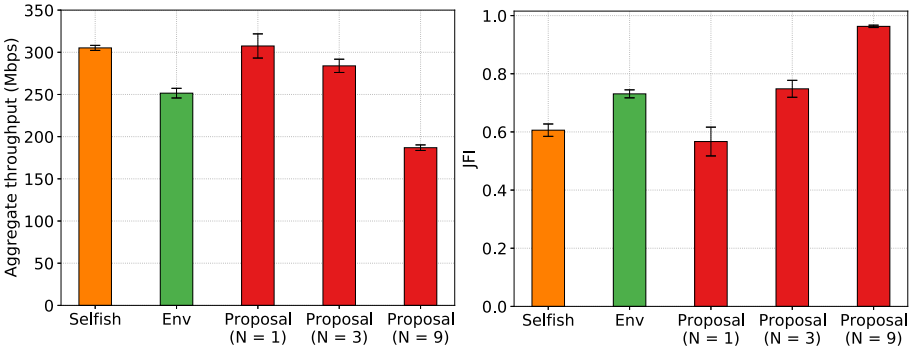
$$\mathcal{J}(x_1, x_2, ..., x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}, \tag{8}$$

where x_i and n denote the experienced throughput by STA i and the total number of STAs, respectively.

First, we evaluate the influence of the variable N for negative rewards in Sect. 4.1. Then, we compare the proposed reward with the selfish reward and environment-aware reward in terms of the aggregate throughput and throughput fairness in Sect. 4.2.

4.1 Influence of the Variable N for Negative Rewards

Figure 2 shows the aggregate throughput and JFI with changing N in the proposed method. As shown in Fig. 2, the value of JFI increases as the value of N increases from 1 to 9. This is because the throughputs of N STAs are distributed among the other STAs with the low throughput. As the value of N increases, the more the STAs with the low throughput can obtain the throughput from STAs with the high throughput. As a result, the proposed method improves throughput fairness among all the STAs by increasing the value of N. In contrast with JFI, the aggregate throughput decreases as the value of N increases from 1 to



(a) Aggregate throughput (b) JFI

Fig. 3. Comparison among the proposed reward, selfish reward, and environment-aware reward in terms of the aggregate throughput and JFI.

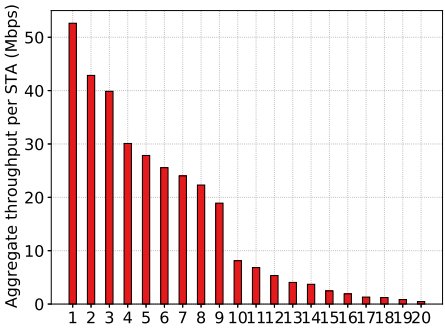
9. This is because the proposed method gives the negative rewards to the N STAs for their high throughput. As a result, the aggregate throughput decreases as the number of N increases. These results indicate that there is the trade-off between the aggregate throughput and throughput fairness.

When the value of N is between 10 and 20, the values of the aggregate throughput and JFI become low. This is because the penalty excessively suppresses the throughput of the STAs. As a result, the aggregate throughput decreases. Moreover, in the case where N is more than half of the learning STAs, the STAs with the low throughput are suppressed as much as the STAs with the high throughput. As a result, throughput fairness decreases.

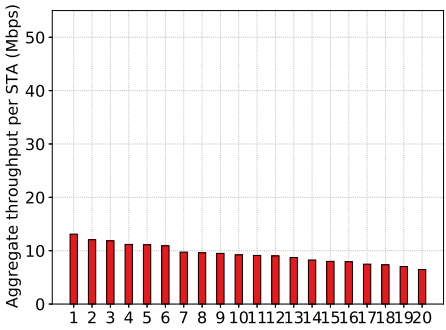
In Fig. 2, when the values of N are 1, 3, and 9, the proposed method achieves the highest aggregate throughput, balance between the aggregate throughput and throughput fairness, and high throughput fairness, respectively. Therefore, we use 1, 3, and 9 as the value of N to compare the proposed method with the other methods in the next subsection.

4.2 Comparison with Previous Methods

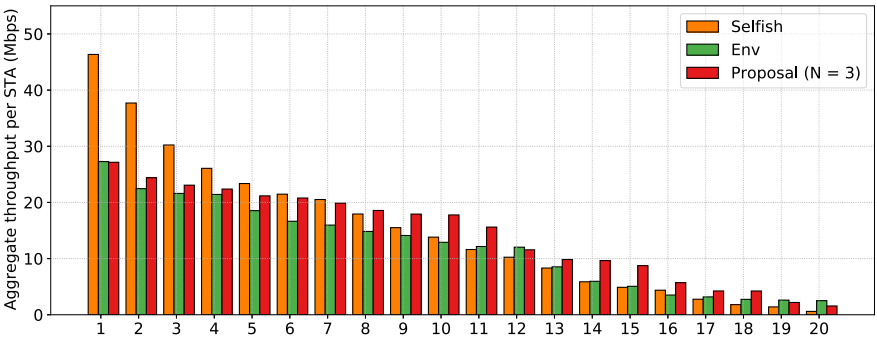
Figure 3 shows the comparison among the proposed reward, selfish reward, and environment-aware reward in terms of the aggregate throughput and JFI. As shown in Fig. 3, the proposed method with $N = 3$ improves the aggregate throughput and JFI by 12.7% and 2.3% compared with the environment-aware reward-based method, respectively. Furthermore, in comparison with the selfish reward-based method, the proposed method with $N = 3$ improves JFI by 23.4% with slightly decreasing the aggregate throughput. These results indicate that the proposed method can achieve both the high aggregate throughput and throughput fairness effectively compared with the other methods by setting the adequate value to N. When the value of N is 9, the proposed method improves JFI by 31.7% compared with the environment-aware reward-based method. In



(a) The proposed reward with $N = 1$



(b) The proposed reward with $N = 9$



(c) Comparison among the proposed reward with $N = 3$, selfish reward, and environment-aware reward in terms of the throughput of each STA

Fig. 4. Aggregate throughput of each STA.

the proposed method, a lot of STAs receive negative rewards as the value of N increases. Therefore, throughput fairness increases as the value of N increases. In the environment-aware reward-based method, the minimum throughput in the network is given to all the STAs. On the other hands, all the STAs do not influence on the minimum throughput. Therefore, the reward contributes to throughput fairness to some extent.

To analyze the aggregate throughput and throughput fairness in details, we evaluated the aggregate throughput of each STA in Fig. 4. As shown in Fig. 4-(a), the STAs 1–9 get almost all the aggregate throughput in the network. This result indicates that the proposed method with $N = 1$ has the tendency to allocate wireless resources to STAs unfairly for achieving the high aggregate throughput in the network. The similar tendency can be seen in the selfish reward-based method in Fig. 4-(c). In contrast with these methods, as shown in Fig. 4-(b), all the STAs get the aggregate throughput fairly, and the aggregate throughput of them is low, however. This is because the mutual interference among the STAs becomes severe as the wireless resources are allocated to them fairly. As a result, the STAs suppress the other STAs to get the wireless resources.

As shown in Fig. 4-(c), the proposed method with $N = 3$ improves the aggregate throughput of almost all the STAs compared with the environment-aware reward-based method. By giving positive and negative rewards to the STAs with the low throughput and those with the high throughput respectively, the proposed method improves the aggregate throughput of the STAs with keeping throughput fairness. In contrast, the environment-aware reward-based method gives the minimum throughput in the network to the STAs for achieving throughput fairness. As a result, the method has the limitation to improve the aggregate throughput of STAs.

5 Conclusions

In this paper, we proposed the Q-learning-based spatial reuse method considering throughput fairness by negative reward for high throughput. The proposed method motivates nodes with the low throughput to use wireless resources by the rewards, while it suppresses the nodes with the high throughput to use the wireless resources by negative rewards. In simulation results, we have confirmed that the proposed method with $N = 3$ improved throughput fairness and the aggregate throughput by 12.7% and 2.3% compared with environment-aware reward-based method, respectively. Furthermore, the proposed method with $N = 3$ improved throughput fairness by 23.4% compared with the selfish reward-based method with keeping the aggregate throughput.

References

1. IEEE Standard for Information technology—telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE STD 802.11-2016 (Revision of IEEE STD 802.11-2012), pp. 1–3534 (2016)
2. Nishide, K., Kubo, H., Shinkuma, R., Takahashi, T.: Detecting hidden and exposed terminal problems in densely deployed wireless networks. IEEE Trans. Wireless Commun. **11**(11), 3841–3849 (2012)
3. Jamil, I., Cariou, L., Hélard, J.-F.: Novel learning-based spatial reuse optimization in dense WLAN deployments. EURASIP J. Wirel. Commun. Netw. **2016**(1), 1–19 (2016). https://doi.org/10.1186/s13638-016-0632-2
4. Ak, E., Canberk, B.: FSC: two-scale AI-driven fair sensitivity control for 802.11ax networks. In: GLOBECOM 2020–2020 IEEE Global Communications Conference, pp. 1–6 (2020)
5. Wilhelmi, F., Barrachina-Muñoz, S., Bellalta, B., Cano, C., Jonsson, A., Neu, G.: Potential and pitfalls of multi-armed bandits for decentralized spatial reuse in WLANs. J. Netw. Comput. Appl. **127**, 26–42 (2019)
6. Watkins, C., Dayan, P.: Q-learning. Mach. Learn. **8**, 279–292 (1992)
7. Morozs, N., Clarke, T., Grace, D.: Cognitive spectrum management in dynamic cellular environments: a case-based Q-learning approach. Eng. Appl. Artif. Intell. **55**, 239–249 (2016)

8. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. A Bradford Book. The MIT Press, Cambridge (2018)
9. TGax simulation scenarios. https://mentor.ieee.org/802.11/dcn/14/11-14-0980-16-00ax-simulation-scenarios.docx. Accessed 19 July 2021
10. ns-3 (online). https://www.nsnam.org/. Accessed 19 July 2021
11. Jain, R., Chiu, D., Hawe, W.: A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. CoRR cs.NI/9809099 (1998)