Deep Reinforcement Learning for Dynamic Multichannel Access in Multi-Cognitive Radio Networks

Ao Wang, Luyong Zhang*, Dianjun Chen and Jinhua Chen
Institute of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China

*Corresponding author’s e-mail: lyzhang@bupt.edu.cn

Abstract. We propose a reinforcement learning framework based on deep recurrent learning to solve the dynamic spectrum access problem in the scenario where multiple cognitive networks coexist. In this scenario, the shared spectrum is divided into multiple channels, and the channel occupation by the primary user is modeled as a Markov model. The observation of the channel status by secondary users in this area obeys the partially observed Markov process, that is, each user can only observe the status of one channel in each time slot, and cannot obtain global information. To avoid collision in different cognitive networks, we adopt the distributed multi-agent deep reinforcement learning method. Without the interaction of different cognitive networks, based on the current partial observations, a suitable channel access policy can be obtained by a neural network. Comparing with Slotted- aloha and DQN-DSA Algorithm, the result indicates DRQN-DSA proposed in this paper performs better.

1. Introduction
Cognitive radio enables secondary user(SU) in cognitive networks to use the primary user's(PU) spectrum resources opportunistically, which makes spectrum can be used more efficiently[1][2]. In some application such as military, emergency medical, and Internet of Things, multiple cognitive radio networks(CRNs) may coexist in the same interference space. The most common case is that there is no central node between CRNs. At the same time, the CRN belonging to different application systems cannot conduct effective information interaction because lack of unified protocol. Multiple cognitive networks cannot coordinate the strategies for accessing spectrum, and it is possible for SU belonging to different cognitive networks to access the same channel which not be occupied by PU.

Deep reinforcement learning is a model-free method, and has the ability to learn to the best policy when the prior knowledge is insufficient[3]. Therefore, deep reinforcement learning technology has the ability to solve the problem of spectrum access[4]. Reference [5] used deep reinforcement learning to establish the order of SU’s access in the coexistence area of multiple CRNs, and achieved good results in fairness and throughput. However, this paper assumes that there is no PU in the environment, which cannot represent the general situation of the cognitive networks. At the same time, the literature requires that the central node need the experience from all SUs to train a unique deep reinforcement learning model, and then deploy the model on all cognitive devices. This causes a large amount of information interaction between the cognitive device and the central node. The channel status was modelled as a Markov process in[6]. They used a distributed reinforcement learning algorithm to train the process of SU accessing the channel. However, the algorithm only considers the impact of the signal-to-noise ratio(SNR), and does not consider the communication rate of SUs. The SNR has a non-linear relationship with transmission rate, which results in that the sum of the rates of CRNs is not...
good enough. There are also some research on dynamic spectrum access. In [7], it is assumed that the channels change independently, so it can be modelled as restless multi-armed bandit problem. In [8] they use deep reinforcement learning techniques in the Vehicular Networks to avoid interference. In [9], they used reinforcement learning to study spectrum access techniques in the presence of interference.

This paper proposes a dynamic spectrum access framework based on Deep Recurrent Q Network to improve Spectrum utilization. We assume that the channel state changes conform to the Markov model, and our algorithm works on every distributed device which do not need to interaction with each other.

2. System Model

We assume that multiple CRNs coexist in the same interference area. SUs access the authorized channel in Overlay mode. As shown in the figure, there are $M$ channels in this area, one primary user base station (BS) and $N$ PUs, $M \geq N$. PU can occupy the channel at any time and there is no mutual interference between PUs, that is, they will not access the same channel. There are also $M$ SUs belonging to different CRN in the region. There is no information interaction between SU, so they may access the same channel simultaneously. Both PU and SU in this area only access one channel at a time, and there is no adjacent-frequency interference by method in [10].

![Diagram of Multiple CRNs coexist model](image)

Figure 1. Multiple CRNs coexist model

Assume that the transmit power of the PU and SU are fixed values, which are represented by $P_s, P_c$, and $P_s > P_c$. The channel gain from the BS to SU and the channel gain between the SUs are $G_{s, i}, G_{s, j}$. Assuming that the $i$-th SU accesses $m$-th channel, the SINR of the $i$-th SU is

$$
SINR_{im} = \frac{G_{s, i} P_s}{\sum_{j \in D_C, j \neq i} \phi(j, m) G_{s, j} P_s + \sum_{j \in D_S} \phi(j, m) G_{B, j} P_s + N_0}
\quad \text{Subject to } \sum_{j \in D_S} \phi(j, m) \leq 1
$$

(1)

$D_C$ represents the set of PU in the area, and $D_S$ represents the set of SU in the area. In this paper, the noise is assumed to be Additive White Gaussian Noise (AWGN). $N_0$ represents Gaussian noise power. $\phi(j, m)$ is the indicator function:

$$
\phi(j, m) = \begin{cases} 
1 & \text{if user } j \text{ occupy channel } m \\
0 & \text{if user } j \text{ not occupy channel } m 
\end{cases}
$$

(2)

the data transmission rate of the $i$-th SU can be represented as:
\[ C_i = W \log_2(1 + SINR_i) \] (3)

In overlay mode, SUs are not allowed to access the same channel as the PU. If the PU accesses the channel, the SU must exit the channel to protect the rights of the PU. At the same time, because the power of PU is greater than that of ordinary SU, we can determine whether the PU is accessing the channel by:

\[
SINR_{i,m} = \frac{G_{i,s}P_{i,s}}{\sum_{j \in D_i, j \neq i} \phi(j,m)G_{j,i}P_{j} + N_0} \leq \gamma
\]

\[
G_{i,s}P_{i,s} \leq \sum_{j \in D_i, j \neq i} \phi(j,m)G_{j,i}P_{j} + N_0
\]

\[
\gamma \text{ is the signal-to-noise ratio threshold. If } SINR_{i,m} \leq \gamma, \text{ indicates that the SU accesses channel } m \text{ at the same time as the primary user, and the SU will exit this channel.}
\]

The optimization goal is to find the optimal access strategy \( \pi^* \) for SU to maximize the long-term average transmission rate, that is,

\[
\text{Maximize } \frac{1}{T} \sum_{t=1}^{T} \sum_{m=1}^{M} \phi(m)C(m)
\]

\[
\text{Subject to } \sum_{m=1}^{M} \phi(m) = 1
\] (5)

3. Deep Recurrent Reinforcement Learning Framework

In this section, we will introduce the Deep Recurrent Reinforcement Learning Framework for dynamic spectrum access in multiple CRNs.

3.1. Agent’s Observation, action and reward

Observation: In this paper, two types of channel states are considered: occupied by the PU or idle. And all M channels are correlated. At time slot t, the channel state is defined as \( X_t = \{x_{1,t}, x_{2,t}, ..., x_{M,t}\} \), \( x_{m,t} \) represents the state of the m-th channel at time t. In time slot t, the PU can only decide whether to use the channel at the beginning of the time slot. The occupation of the channel will not change until the end of the time slot. The probability of transitioning from the current state of the current channel to the next state is represented by \( p \), and the probability of maintaining the current state is \( 1-p \). SUs can only indirectly understand the channel conversion model through their own access process, and cannot accurately obtain the global observation information. In this case, the access process of Sus can be modeled as a partially observable Markov decision process (POMDP). Define \( O_t \) as the observation of the channel state in the environment by the SU at time t, then \( O_t = \{o_{1,t}, o_{2,t}, ..., o_{M,t}\} \)

\[
o_{m,t} = \phi(m)F(m)
\]

\[
F(m) = \begin{cases} 
1 & \text{if } SINR_m > \gamma \\
-1 & \text{if } SINR_m \leq \gamma
\end{cases}
\]

\( o_{m,t} \) is the observed value of the corresponding channel \( x_{m,t} \) by the SU. \( F(m) \) represents the observation of the status of the access channel by the SU. When \( F(m) = -1 \), it means that the PU occupies the channel. Otherwise, the channel is idle.

Action: The action space is the set of all channels. If the output of the neural network is i, it means that the SU selects the i-th channel to access in the current time slot.

Reward: SUs will have two choices after accessing the channel selected by the neural network, using or existing, depending on PUs. So we define reward as follow:

\[
R_i = \begin{cases} 
W \log_2(1 + SINR_i) & \text{SINR}_i > \gamma \\
0 & \text{SINR}_i \leq \gamma
\end{cases}
\]

(7)
3.2. Deep Recurrent Reinforcement Learning algorithm
The traditional DQN-DSA algorithm uses the information of the past T moments as the state to input the neural network, but T is a hyperparameter and it is difficult to adjust[11]. In this paper, we Add LSTM as a type of recurrent neural network to avoid adjusting T[12]. We use neural networks to approximate the action value function. The action value function fitted by the neural network model can be expressed as $q(s,a,\theta)$, where $\theta$ is a parameter of the neural network. The neural network updates its parameters to minimize the error loss function:

$$L(\theta) = E[(y_i - q(O_t,a,\theta))^2]\quad (8)$$

$$y_i = r + \gamma \max_a(q(O_{t+1},a,\theta))\quad (9)$$

$y_i$ represents the target action-value function, and $\gamma$ represents the discount factor. The parameter $\theta$ in the neural network will continuously change towards the gradient of the target action value function during continuous training, so that the neural network has the ability to fit the action-value function. We can describe the algorithm as bellow.

| Table 1. Deep Recurrent Reinforcement Learning for dynamic spectrum access. |
|---------------------------------------------------------------|
| Initialize neural network $\theta_i$ for all secondary users, |
| For $t = 0$ to $T$ do:                                       |
| Agent select channel $a$ to access by observation $O_t$, and get reward $r$. Then agent |
| observe next state $O_{t+1}$, Save $<O_t,a,r,O_{t+1}>$ in local history. |
| Agent Choose mini-batch history for local history to train neural network: |
| Calculate $y_i = r + \gamma \max_a(q(O_{t+1},a,\theta))$ |
| Update $\theta$ by $L(\theta) = E[(y_i - q(O_t,a,\theta))^2]$ |
| Update $O_t \leftarrow O_{t+1}$ |
| End for                                                       |

4. Experiments
The DRQN-DSA algorithm model contains one hidden layer, one LSTM layer and one advantage layer[13]. Each layer has 128 Neurons. Use Adam as optimization algorithm of the neural network. exploration probability is 0.1.

We first set the number of channels $M = 16$, the number of channels occupied by the PU is 10, and 4 SU need to occupy 4 channels. There are 16 kinds of Markov states of channels, and the channel state transition probability is 0.8. After simulation, the channel status of 50 time slots is obtained and plotted as the following figure 2.

![Figure 2. channel state change with time](image)

We compare DQN with DRQN-DSA in this situation, and we set the past moments $T=2$, the result shows in figure 3:
As the number of training episode increases, the transmission rates of both algorithms continue to increase, which indicates that agents based on both algorithms are constantly optimizing their strategies. Due to the limitation of observation ability, this makes the DQN-DSA agent unable to accurately estimate the current channel state. DRQN-DSA does not need to use the hyperparameter observation length T. Historical information is stored inside the neural network, which can better judge the position of the channel state in the Markov chain.

Next we will compare DQN-DSA algorithm and DRQN-DSA algorithm mentioned in this paper with the optimal algorithm, model-based strategy, and random strategy [6][11]. Unlike the literature [11], we extend the optimal strategy to a multi-CRNs environment. If the SU fails to access, the channel status needs to be synchronized with the PU before the next access. The channel gain conforms to Winner 2 model and the bandwidth is 1Mhz. The distance of SU’s Tx and Rx is 10m. and All SUs are randomly distributed within a 20-meter radius area. We compared all the methods in p=0.8 with different SU shown in figure 4.

The transmission rate under the optimal strategy is a two-segment polyline. The reason is that the optimal strategy not only has prior knowledge of the channel changes in the environment, but also understands the strategies of other agents to achieve the best. The model-based strategy assumes that SUs only know the channel state transition model in space, but cannot know the access strategies of other SUs, so the transmission rate rises when the number of users is small. As the number of SU increases, the collision probability between SUs increases, and the transmission rate starts to decrease. Under the random strategy, SUs access the channel randomly, making the transmission rate low and slow growth. It can be seen that the transmission rate of the DRQN-DSA algorithm is closer to the optimal strategy.

5. Conclusion
We consider the scenario where multiple cognitive networks coexist in the same area. In this chapter, in this scenario, a DRQN-DSA algorithm based on deep recurrent reinforcement learning is proposed. Simulations verify that the proposed algorithm can establish a spectrum access order in this scenario, improve the transmission rate of cognitive users in the area, and reduce the probability of collision between all the users. In the case of a small number of secondary users, the performance of the DRQN-DSA algorithm can approach the optimal strategy optimal to achieve efficient reuse of spectrum resources.

References
[1] Zhang L, Yang Y, Chen J. Cognitive Communication in Link-Layer Evaluation Based Cellular-Vehicular Networks[J]. IEEE Access, 2018, 6: 75293-75301.
[2] Zhang L, Yang Y, Li X, et al. Effective capacity in cognitive radio networks with relay and primary user emulator[J]. China Communications, 2019, 16(11): 130-145.
[3] Li Y. Deep reinforcement learning: An overview[J]. arXiv preprint arXiv:1701.07274, 2017

[4] X. Zhou, M. Sun, G. Y. Li and B. Fred Juang, "Intelligent wireless communications enabled by cognitive radio and machine learning," in China Communications, vol. 15, no. 12, pp. 16-48, Dec. 2018.

[5] O. Naparstek and K. Cohen, "Deep Multi-User Reinforcement Learning for Dynamic Spectrum Access in Multichannel Wireless Networks," GLOBECOM 2017 - 2017 IEEE Global Communications Conference, Singapore, 2017, pp. 1-7.

[6] C. Zhong, Z. Lu, M. C. Gursoy and S. Velipasalar, "A Deep Actor-Critic Reinforcement Learning Framework for Dynamic Multichannel Access," in IEEE Transactions on Cognitive Communications and Networking, vol. 5, no. 4, pp. 1125-1139, Dec. 2019.

[7] Nguyen H Q, Nguyen B T, Dong T Q, et al. Deep Q-Learning with Multiband Sensing for Dynamic Spectrum Access[C]//2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN). IEEE, 2018: 1-5.

[8] Liang L, Ye H, Li G Y. Spectrum sharing in vehicular networks based on multi-agent reinforcement learning[J]. IEEE Journal on Selected Areas in Communications, 2019, 37(10): 2282-2292.

[9] Bi Y, Wu Y, Hua C. Deep Reinforcement Learning Based Multi-User Anti-Jamming Strategy[C]//ICC 2019-2019 IEEE International Conference on Communications (ICC). IEEE, 2019: 1-6.

[10] Zhang L, Yao H, He L, et al. A novel ultra-wide band signal generation scheme based on carrier interference and dynamics suppression[J]. EURASIP Journal on Wireless Communications and Networking, 2010, 2010: 1-6.

[11] S. Wang, H. Liu, P. H. Gomes and B. Krishnamachari, "Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks," in IEEE Transactions on Cognitive Communications and Networking, vol. 4, no. 2, pp. 257-265, June 2018.

[12] Hausknecht M, Stone P. Deep recurrent q-learning for partially observable mdps[C]//2015 AAAI Fall Symposium Series. 2015.

[13] Wang Z, Schaul T, Hessel M, et al. Dueling network architectures for deep reinforcement learning[J]. arXiv preprint arXiv:1511.06581, 2015.