International Journal of Scientific Research in Science and Technology
Available online at : www.ijsrst.com
OPEN ACCESS
Print ISSN: 2395-6011 | Online ISSN: 2395-602X
doi : https://doi.org/10.32628/IJSRST

Implementation of Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks

B. Vamsi Krishna[1], Dr. S. Varadarajan[2]

[1]M.Tech Student, Department of Electronics and Communication Engineering, S.V.University College of Engineering, Tirupati, A.P., India

[2]Professor, Department of Electronics and Communication Engineering, S.V.University College of Engineering, Tirupati, A.P., India

ARTICLEINFO

Article History:

Accepted: 07 Sep 2023
Published: 22 Sep 2023

Publication Issue
Volume 10, Issue 5
September-October-2023
Page Number
267-278

ABSTRACT

The ever-increasing demand for high-speed wireless communication services has underscored the importance of efficient spectrum utilization and dynamic network access. This research introduces an innovative approach that leverages Deep Reinforcement Learning (DRL) using Deep Q Networks (DQN) for achieving dynamic multichannel access in wireless networks. Furthermore, it conducts a comprehensive comparative analysis between this proposed method and the conventional Deep Reinforcement Learning for Dynamic Spectrum Access (DSA) in wireless networks. In the proposed method, DRL with DQN is employed to optimize multichannel access for wireless devices. This framework enables devices to intelligently select and utilize available channels based on real-time network conditions and user requirements. By learning from interactions with the environment, the proposed system adapts its channel selection strategies, leading to improved spectrum utilization and network performance. To assess the effectiveness of the proposed method, extensive simulations and real-world experiments are conducted, comparing its performance with the existing DRL-based DSA system. The evaluation encompasses key performance metrics, including spectrum utilization, network throughput, latency, and Quality of Service (QoS). The results of the comparative analysis reveal significant advantages of the proposed DRL with DQN approach in terms of dynamic multichannel access. It outperforms the conventional DSA system, demonstrating superior spectrum utilization and more efficient network resource allocation. Additionally, the proposed method exhibits greater adaptability to changing network conditions, making it suitable for a wide range of wireless communication scenarios. This research highlights the potential of DRL with DQN for dynamic multichannel access in wireless networks, emphasizing its role in enhancing network efficiency and meeting the demands of modern wireless communication services. By comparing it with the established DSA approach, this study provides valuable insights into the benefits and implications of adopting DRL-based strategies for optimizing wireless network access.

Keywords: Deep Q Learning, Control Overhead Protocol, Multi-User Reinforcement Learning

I. INTRODUCTION

In today's rapidly evolving wireless communication landscape, the effective and efficient management of spectrum resources is paramount. The exponential growth of wireless devices and data-intensive applications demands innovative solutions to maximize spectrum utilization while minimizing control overhead. This paper presents a pioneering approach that combines Deep Multi-User Reinforcement Learning (DMU-RL) with Deep Q Networks (DQN) to address these challenges. Our proposed system aims to revolutionize spectrum management in multichannel wireless networks by reducing control overhead and enhancing resource allocation. This introduction provides an overview of the motivation, objectives, and significance of this research, along with a glimpse into how it compares with the traditional Dynamic Spectrum Access (DSA) system in existing multichannel wireless networks.

MOTIVATION

The motivation behind this research is rooted in the recognition that traditional spectrum management approaches, including DSA, often struggle to adapt to the dynamic and diverse demands of contemporary wireless networks. These methods are typically designed with static policies and centralized control, which can lead to inefficient spectrum utilization, increased interference, and excessive signaling overhead. To meet the ever-growing demands for wireless connectivity and bandwidth-hungry applications, there is a pressing need for dynamic, intelligent, and adaptive spectrum management techniques that can minimize control overhead while optimizing resource allocation.

OBJECTIVES

The primary objectives of this research are as follows:

1. **Development of DMU-RL with DQN Framework:** We aim to design a robust DMU-RL framework that empowers multiple users to make informed decisions about spectrum access. By incorporating DQN, we intend to optimize access policies, ensuring efficient channel allocation.
2. **Reduction of Control Overhead:** Our system seeks to substantially reduce the control overhead associated with spectrum management in multichannel wireless networks. This reduction will lead to more efficient use of available spectrum resources.
3. **Comparative Analysis with DSA:** We will conduct an in-depth comparative analysis, pitting our proposed system against the traditional DSA system. This analysis will provide empirical evidence of the superiority of our approach in terms of control overhead reduction, spectrum utilization, network throughput, and quality of service.

SIGNIFICANCE:

The significance of this research lies in its potential to redefine the way spectrum resources are managed in multichannel wireless networks. By leveraging DMU-RL and DQN, our proposed system aims to be adaptive and efficient, capable of learning from and adapting to dynamic network conditions. This innovation has the potential to dramatically enhance the performance of multichannel wireless networks, making them more responsive to user needs, less susceptible to interference, and better prepared for the ever-increasing demands of wireless communication.

OUTLINE:

In the subsequent sections of this paper, we will delve deeper into the architecture and operation of our proposed DMU-RL with DQN system, detailing its key components and functionalities. We will then present the results of comprehensive simulations and real-world experiments, showcasing the tangible benefits of our approach when compared to the traditional DSA system. Finally, we will discuss the broader implications of our findings and explore potential future directions for the application of DMU-RL and DQN in multichannel wireless networks.

The organizational framework of this study divides the research work in the different sections. The Literature review is presented in section 2. The Existing method is presented in section 3. The Proposed method is presented in section 4. Further, in section 5 shown Results is discussed and. Conclusion and future work are presented by last sections 6.

II. LITERATURE SURVEY

1. Dynamic Spectrum Access (DSA) and Cognitive Radio Networks:

Title: "Dynamic Spectrum Access in Cognitive Radio Networks" (Mitola, 2009)

This foundational work introduces the concept of DSA and the potential of cognitive radio networks to optimize spectrum utilization.

Title: "A Survey of Dynamic Spectrum Access: Signal Processing and Networking Perspectives" (Yucek and Arslan, 2009)

This comprehensive survey provides insights into DSA techniques, covering signal processing and networking aspects.

2. Reinforcement Learning in Wireless Networks:

Title: "Deep Reinforcement Learning for Wireless Communications" (Zhang et al., 2019)

This paper explores the application of deep reinforcement learning (DRL) in wireless communication systems and discusses its potential benefits.

Title: "Reinforcement Learning in Wireless Communications: Applications, Challenges, and Prospects" (Zhang et al., 2020)

This survey discusses the application of reinforcement learning techniques in optimizing wireless communication systems.

3. Multi-User Reinforcement Learning:

Title: "Multi-User Reinforcement Learning: A Review" (Shi et al., 2018)

This review provides an overview of multi-user reinforcement learning, discussing the challenges and potential applications.

Title: "Multi-Agent Reinforcement Learning: A Selective Overview of Theories and Algorithms" (Hernandez-Leal et al., 2019)

This paper discusses the application of multi-agent reinforcement learning in scenarios involving multiple users.

4. Deep Q Networks (DQN):

Title: "Human-Level Control Through Deep Reinforcement Learning" (Mnih et al., 2015)

This foundational paper introduces DQN and its application in achieving human-level control in various tasks.

Title: "Continuous Control with Deep Reinforcement Learning" (Lillicrap et al., 2016)

While primarily focused on continuous control, this paper offers insights into advanced DRL techniques.

5. Dynamic Spectrum Access in Multichannel Wireless Networks

Title: "Efficient Resource Allocation in Multichannel Cognitive Radio Networks" (Tang et al., 2013)

This research addresses efficient resource allocation in multichannel cognitive radio networks, a relevant field for DSA in multichannel wireless networks.

Title: "Spectrum Management in Multichannel Wireless Networks: Challenges and Opportunities" (Guo et al., 2016).

This paper discusses challenges and opportunities in spectrum management for multichannel wireless networks.

6. Spectrum Sharing and Multichannel Wireless Communication:

Title: "Spectrum Sharing for Device-to-Device Communication in Cellular Networks: A Survey" (Kord et al., 2018)

This survey focuses on spectrum sharing, an essential aspect of multichannel wireless networks.

Title: "Resource Allocation and Spectrum Sharing in Multichannel Cognitive Radio Networks" (Liang et al., 2014)

This research explores resource allocation and spectrum sharing techniques in multichannel cognitive radio networks.

III. EXISTING METHOD

Learning Algorithms for Dynamic Spectrum Access Developing distributed optimization and learning algorithms for managing efficient spectrum access among users has at tracted much attention in past and recent years. Complete information about the network state is typically not available online for the users, which makes the computation of optimal policies intractable in general [6]. While optimal structured solutions have been developed for some special cases. most of the existing studies have been focused on designing spectrum access protocols for specific models so that efficient and structured solutions can be obtained. However, model dependent solutions cannot be effectively adapted in general for handling more complex real-world models. Model-free Q-learning has been used in [10] for Aloha-based protocol in cognitive radio networks. Handling large state space and partial observability, however, becomes inefficient under Q learning.

DEEP MULTI-USER REINFORCEMENT LEARNING FOR DYNAMIC SPECTRUM ACCESS

Our goal is to develop a distributed learning algorithm for dynamic spectrum access that can effectively adapt for general complex real-world settings, while overcoming the expensive computational requirements due to the large state space and partial observability of the problem. We adopt a deep multi user reinforcement learning approach to achieve this goal. Deep reinforcement learning (DRL) (or deep Q-learning) has attracted much attention in recent years due to its capability to provide a good approximation of the objective value. while dealing with a very large state and action spaces. In contrast to Q-learning methods that perform well for small-size models but perform poorly for large-scale models, DRL combines deep neural network with Q-learning, referred to as Deep Q-Network (DQN), for overcoming this issue. The DQN is used to map from states to actions in large-scale models so as to maximize the Q value. In [12], the authors developed DRL algorithms for teaching multiple players how to communicate so as to maximize a shared utility. Strong performance has been demonstrated for several players in MNIST games and the switch riddle. In recent years, there is a growing attention on using DRL methods for other various fields. A survey on very recent studies can be found in [13].

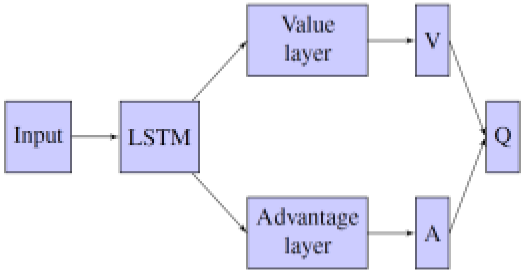


Figure 1: Design of one circular patch antenna

In this section, we describe the proposed architecture for the multi-user DQN used in DQSA algorithm for solving the DSA problem. An illustration of the DQN is presented in Fig. 1.

1) Input Layer:

The input xn(t) to the DQN is a vector of size 2K + 2. The first K + 1 input entries indicate the action (i.e., selected channel) taken at time t − 1. Specifically, if the user has not transmitted at time slot t − 1, the first entry is set to 1 and the next K entries are set to 0. If the user has chosen channel k for transmission at time t − 1 (where 1 ≤ k ≤ K), then the (k + 1)th entry is set to 1 and the rest K entries are set to 0. The following K input entries are the capacity of each channel (i.e., the packet transmission rate over a channel conditioned on the event that the channel is free, which is proportional to the bandwidth of the channel). In the experiments presented in Section V, we simulated equal channels (i.e., we set 1 at all K entries). The last input is 1 if ACK signal has been received. Otherwise, if transmission has failed or no transmission has been executed, it is set to 0.

2) LSTM Layer

Since the network state is partially observable for each user, and the dynamic is non-Markovian and determined by the multi-user actions, classical DQNs do not perform well in this setting. Thus, we add an LSTM layer to the DQN that maintains an internal state and aggregate observations over time. This gives the network the ability to estimate the true state using the history of the process. This layer is responsible of learning how to aggregate experiences over time.

3) **Value and Advantage Layers**

Another improvement that we incorporate is the use of dueling DQN, as suggested in [11]. The intuition behind this architecture lies in the fact that there is an observability problem in DQN. There are states which are good or bad regardless of the taken action. Hence, it is desirable to estimate the average Q-value of the state which is called the value of the state V (sn(t)) independently from the advantage of each action.

4) Block output layer:

The output of the DQN is a vector of size K + 1. The first entry is the estimated Q-value if the user will choose not to transmit at time t. The (k + 1)th entry, where 1 ≤ k ≤ K, is the estimated Q-value for transmitting on channel k at time t. 5) Double Q-learning: The max operator in standard Q-learning and DQN (see (8)) uses the same values to both selecting and evaluating an action. Thus, it tends to select overestimated values which degrade performance. Hence, when training the DQN, we use double Q-learning [41] used to decouple the selection of actions from the evaluation of Q-values. Specifically, we use two neural networks, referred to as DQN1 and DQN2. DQN1 is used for choosing actions and DQN2 is used to estimate the Q-value associated with the selected action.

IV. PROPOSED METHOD

When channels are correlated and system dynamics are unknown, there are two main approaches to tackle the dynamic multichannel access problem:

i. Model-based approach: first estimate the system model from observations and then apply dynamic programming or a computationally efficient heuristic policy such as Myopic/Whittle Index policies;

ii. Model-free approach: learn the policy directly through interactions with the system without estimating the system model.

The model-based approach is less favoured since the user's limited observation capability may result in bad system model estimation. Even worse, even if the system dynamics is well estimated, solving a POMDP in a large state space is always a bottleneck as the dynamic programming method has exponential time complexity and the heuristic approaches do not have any performance guarantee. All these challenges motivate us to follow the model-free approach, which, by incorporating the idea of Reinforcement Learning, can learn directly from observations without the necessity of finding an estimated system model and can be easily extended to very large and complicated systems.

A. Q-Learning

We focus on the reinforcement Learning paradigm, Q-learning specifically, to incorporate learning for the dynamic multichannel access problem. The goal of Q-learning is to find an optimal policy, i.e., a sequence of actions that maximizes the long-term expected accumulated discounted reward. Q-learning is an empirical value iteration approach and the essence is to find the Q-value of each state and action pairs, where the state x is a function of observations (and rewards) and the action a is some action that a user can take given the state x. The Q-value of a state-action pair (x, a) from policy π, denoted as Q π (x, a), is defined as the sum of the discounted reward received when taking action a in the initial state x and then following the policy π thereafter. Q π ∗ (x, a) is the Q-value with initial state x and initial action a, and then following the optimal policy π ∗. Thus, the optimal policy π ∗ can be derived as π ∗ (x) = arg maxa Q π ∗ (x, a), ∀x. One can use online learning method to find Q π ∗ (x, a) without any knowledge of the system dynamics. Assume at the beginning of each time slot, the agent takes an action at ∈ {1, .., N} that maximizes its Q-value of state-action pair (xt , at) given the state is xt, and gains a reward rt+1. Then the online update rule of Q-values with learning rate 0 < α < 1 is given as follows:

$$Q(xt , at) \leftarrow Q(xt , at)+\alpha[rt+1 +\gamma \max at+1$$
$$Q(xt+1,at+1)-Q(xt , at)] ....(1)$$

In the context of the dynamic multichannel ac cess, the problem can be converted to an MDP when considering the belief space and Q-learning can be applied consequently. However, this approach is impractical since the belief update is maintained by knowing the system transition matrix P a-priori, which is hardly available in practice. Instead, we apply Q-learning by directly considering the history of observations and actions. We define the state for the Q-learning at time slot t as a combination of historical selected channels as well as their observed channel conditions over previous M time slots, i.e., xt = [at−1, ot−1, ..., at−M, ot−M]. And intuitively, the more historical information we consider (i.e., the larger M is), the better Q-learning can learn.

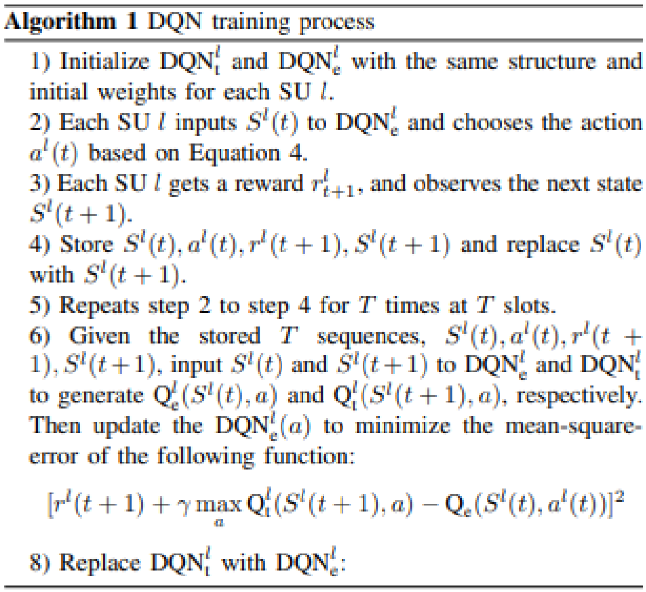
B. DEEP REINFORCEMENT LEARNING

Q-learning performs well for small-scale models but per forms poorly for large-scale models. The reason is that the training algorithm of Q-learning iteratively updates the Q table. As the number of possible states increases, the large Q-table size makes training difficult or even impossible. Due to the difficulties of updating every element in Q-table for a large-scale model, DRL exploits the powerful deep neural network to approximate the Q-value. In our work, the size of the state space grows exponentially with the number of the channels. Each channel is occupied by a PU and each PU has two possible states: Active state or Inactive state, so the state size is 2 N for N channels. This motivates us to use DQN to learn the dynamic spectrum access strategy in an unknown dynamic system.

C. DEEP Q-NETWORK

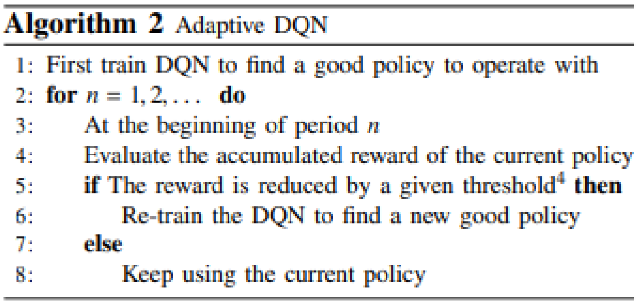
Q-learning works well when the problem's state space is small, as a look-up table can be used to update Q values. But this is impossible when the state space becomes large. The state space size in this work grows exponentially as O(N M), as we use a combination of M vectors of length N to represent historical observations and actions for a system with N 2-state channels over past M time slots. M is required to be large so that Q-learning can capture enough information for learning. Even worse, since many states are rarely visited, their corresponding Q-values are seldom updated. This causes Q learning to take a very long time to converge. Motivated by its success in other domains, we adopt the deep Q-Network approach to ad dress the very large state space. DQN takes the state-action pair as input and outputs the corresponding Q-value. Q-network updates its weights θ at each iteration i to minimize the loss function Li(θi) = E[(yi − Q(x, a; θi))2], where yi = E[r + γ maxa 0 Q(x 0 , a 0 ; θi−1)] is derived from the same Q network with old weights θi−1.

D. ALGORITHMS



Algorithm 1 DQN training process

```
1) Initialize $\mathrm{DQN}_t^l$ and $\mathrm{DQN}_e^l$ with the same structure and initial weights for each SU $l$.
2) Each SU $l$ inputs $S^l(t)$ to $\mathrm{DQN}_e^l$ and chooses the action $a^l(t)$ based on Equation 4.
3) Each SU $l$ gets a reward $r_{t+1}^l$, and observes the next state $S^l(t+1)$.
4) Store $S^l(t), a^l(t), r^l(t+1), S^l(t+1)$ and replace $S^l(t)$ with $S^l(t+1)$.
5) Repeats step 2 to step 4 for $T$ times at $T$ slots.
6) Given the stored $T$ sequences, $S^l(t), a^l(t), r^l(t+1), S^l(t+1)$, input $S^l(t)$ and $S^l(t+1)$ to $\mathrm{DQN}_e^l$ and $\mathrm{DQN}_t^l$ to generate $\mathrm{Q}_e^l(S^l(t), a)$ and $\mathrm{Q}_t^l(S^l(t+1), a)$, respectively. Then update the $\mathrm{DQN}_e^l(a)$ to minimize the mean-square-error of the following function:
       $[r^l(t+1) + \gamma \max_a \mathrm{Q}_t^l(S^l(t+1), a) - \mathrm{Q}_e(S^l(t), a^l(t))]^2$
8) Replace $\mathrm{DQN}_t^l$ with $\mathrm{DQN}_e^l$:
```



Algorithm 2 Adaptive DQN

```
First train DQN to find a good policy to operate with
for $n = 1, 2, \ldots$ do
    At the beginning of period $n$
    Evaluate the accumulated reward of the current policy
    if The reward is reduced by a given threshold$^4$ then
        Re-train the DQN to find a new good policy
    else
        Keep using the current policy
```

E. LEARNING PROCESS

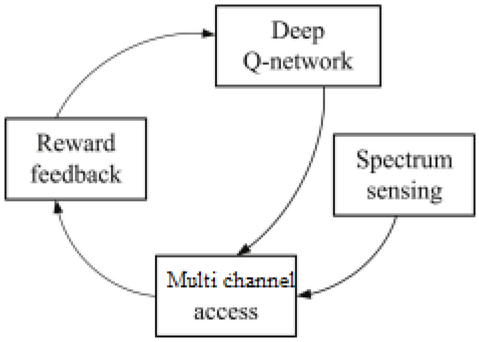


Figure 2: Learning Process of proposed method

In this section, we formulate the dynamic spectrum access as a reinforcement learning problem. We define the agent, state, action, reward, and policy in dynamic spectrum access environment. The learning procedure is shown in Fig.2. It can be seen that spectrum access strategies are determined by the results of deep Q-network and current spectrum sensing. According to the spectrum access strategies, SUs access wireless channels to carry out data transmissions. Then, SU receivers feedback reward based on actual wireless transmission quality, which will be stored by SU transmitters and used as training data of DQN+RC to update spectrum access strategies. The aforementioned learning procedure will be carried out periodically to tackle the variations of wireless environments.

F. BLOCK DIAGRAM

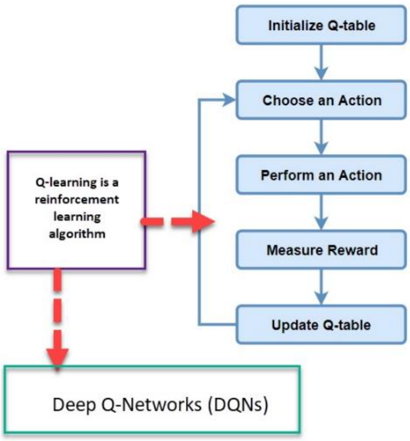


Figure 3: Learning Process of proposed method

Deep Reinforcement Learning (DRL) using Deep Q Networks (DQN) for dynamic multichannel access involves a sequence of steps that enable an agent to learn and make intelligent decisions about which channels to use in a wireless network. Here's an explanation of how this process works, using the Q-learning reinforcement algorithm, the initialization of a Q table, action selection, action execution, reward measurement, and Q table and DQN updates:

1. **Initialization of Q Table:**

The process begins by initializing a Q table. This table stores the expected cumulative rewards associated with taking specific actions in different states of the environment. In the context of dynamic multichannel access, each state might represent a specific combination of channel conditions, neighboring devices' activities, and other relevant factors. The Q table is initialized with arbitrary values or zeros.

2. **Choosing an Action (Channel Selection):**

- The agent, which represents a wireless device or network node, selects an action based on its current state. In this case, the action corresponds to selecting one of the available communication.
- channels for data transmission. The agent uses its current knowledge (Q table and DQN) and a selection strategy, such as ε-greedy exploration, to decide which channel to use.

3. **Performing an Action (Channel Assignment):**

After choosing an action, the agent performs the selected action by assigning the chosen channel for data transmission. This corresponds to the wireless device tuning into the selected channel for communication.

4. **Measuring Reward:**

The agent then measures the reward associated with the chosen action in the current state. The reward is typically a numerical value that reflects the quality of the channel assignment. It can be based on factors like data throughput, latency, signal-to-noise ratio, or any other relevant performance metric. The goal is to maximize the cumulative reward over time.

5. **Updating Q Table (Q-Learning):**

The heart of the Q-learning algorithm lies in updating the Q table based on the observed reward. The Q-value for the chosen action in the current state is updated using the following formula:

Q(state, action) = Q(state, action) + α * (reward + γ * max(Q(next_state, all_actions)) - Q(state, action))…..(2)

Q(state, action) is the Q-value for the current state-action pair. **α** is the learning rate, controlling how much the Q-value is updated based on the observed reward.

reward is the reward obtained from performing the action. **γ** is the discount factor, representing the agent's preference for immediate rewards over delayed rewards.

max(Q(next_state, all_actions)) represents the maximum Q-value for the next state, where the agent anticipates the best possible future rewards.

The Q table is updated iteratively as the agent explores and interacts with the environment.

6. **Updating Deep Q Networks (DQN):**

In parallel with Q table updates, DQN is trained using the collected experiences. DQN is a neural network that approximates the Q-values, allowing for more complex state-action mappings. The neural network is updated through a supervised learning process where the target Q-values are predicted by the DQN, and the loss between predicted and target Q-values is minimized during training.

7. **Repeat the Process:**

The agent repeats these steps iteratively over multiple episodes of interaction with the environment. As it learns, the Q table and DQN become more refined, enabling the agent to make increasingly intelligent decisions about channel selection. By following this process, the agent gradually learns to make optimal channel selection decisions in dynamic multichannel access scenarios, ultimately improving network performance and efficiency.

V. SIMULATION RESULTS

A. EXISTING METHOD

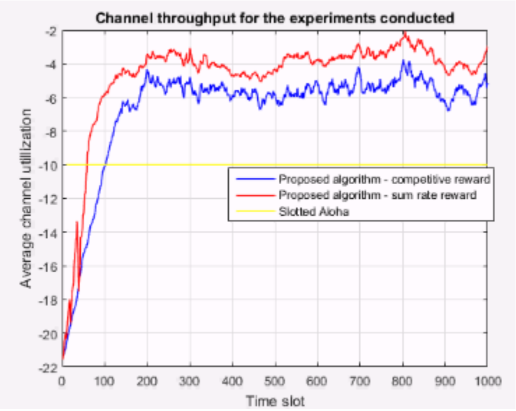


Figure 4 :channel throughtput for the experiments conducted

Channel throughput refers to the amount of data transmitted successfully over a communication channel within a certain period. The Y-axis represents the throughput in units like Mbps (Megabits per second) or Kbps (Kilobits per second). Average channel utilization represents the proportion of time a channel is being used for data transmission relative to the total available time.

The Y-axis represents channel utilization as a percentage (%). The X-axis represents time in seconds (s) and serves as a timeline for the experiment. Time is typically measured in seconds and represents the duration of the experiment.

Channel Throughput Line (in a certain color):

The line depicting channel throughput represents how the throughput varies over time during the DSA experiment. Peaks or high points on this line indicate periods of high data transmission, suggesting efficient channel usage. Troughs or low points signify moments of reduced data transmission, which may result from interference or suboptimal channel selection.

Average Channel Utilization Line (in a different color):

The line showing average channel utilization illustrates the percentage of time the channel is actively used for communication. An increasing trend in this line suggests that the channel is being used more frequently, which implies effective spectrum sharing and utilization.

Conversely, a decreasing trend in average channel utilization may indicate underutilization or inefficient spectrum management.

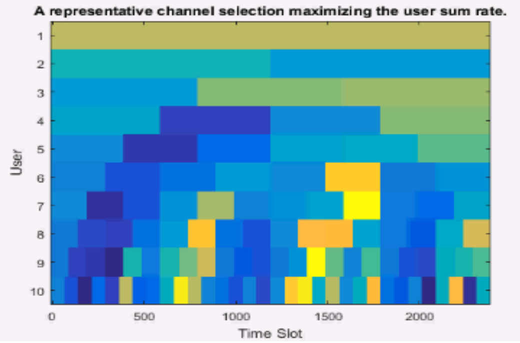


Figure 5 : a representative channel selection maximizing the user sum rate

The Y-axis represents the number of users or devices in the wireless network. The Y-axis typically shows the count of users, starting from zero and increasing incrementally. The X-axis represents time in seconds (s) and serves as a timeline for the experiment. Time is usually measured in seconds and represents the duration of the experiment. The line or curve in the graph represents the result of channel selection strategies applied to maximize the sum rate of users in the network.

The curve illustrates how the total data rate (sum rate) achieved by all users collectively changes over time as channels are selected and assigned to users. The shape and trends of the curve indicate the effectiveness of the channel selection algorithm in optimizing the overall network performance.

B. PROPOSED METHOD

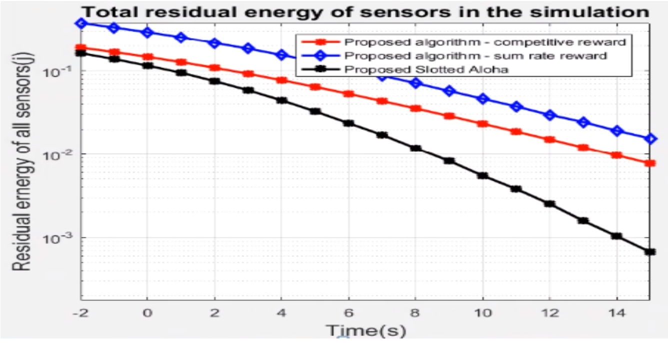


Figure 6: Residual energy of sensors

The Y-axis represents the remaining energy level of a sensor node in the wireless network. The Y-axis typically measures the energy level in units such as joules. The X-axis represents time in seconds (s) and serves as a timeline for the experiment. Time is usually measured in seconds and represents the duration of the experiment. The line or curve in the graph represents the sensor node's remaining energy level as a function of time. The curve illustrates how the energy level of the sensor node changes over time due to various factors, including data transmission, reception, and energy consumption for processing and communication.

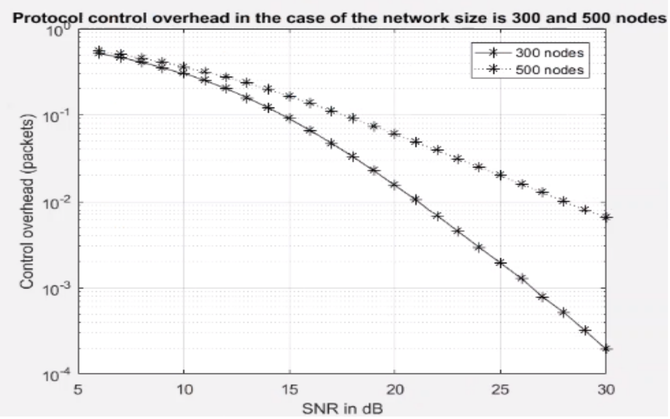


Figure 7: protocol control overhead in case of the nework size is 300 and 500 nodes

The Y-axis represents the protocol control overhead in the wireless network. Protocol control overhead is typically measured in bytes, kilobytes (KB), or bits. The X-axis represents the size of the wireless network, specifically the number of nodes or devices. The X-axis values denote the total number of nodes in the network, such as 300 nodes and 500 nodes. The line or curve in the graph represents the variation in protocol control overhead as the network size changes. This curve illustrates how the control overhead scales with the number of nodes in the network.

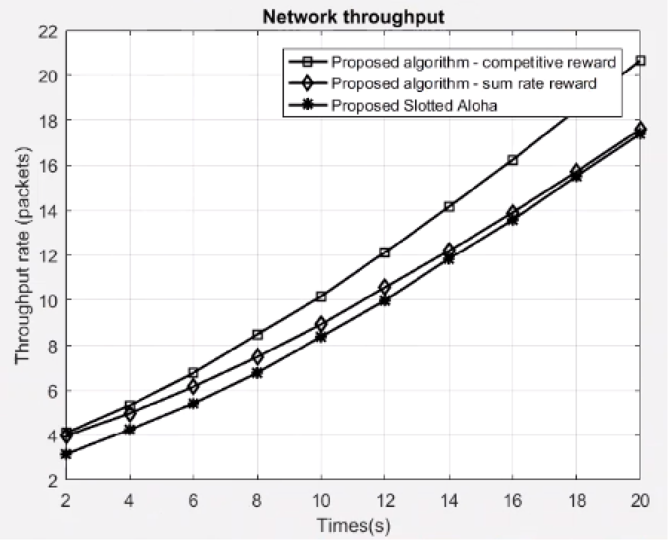


Figure 8: Network throughput of proposed method

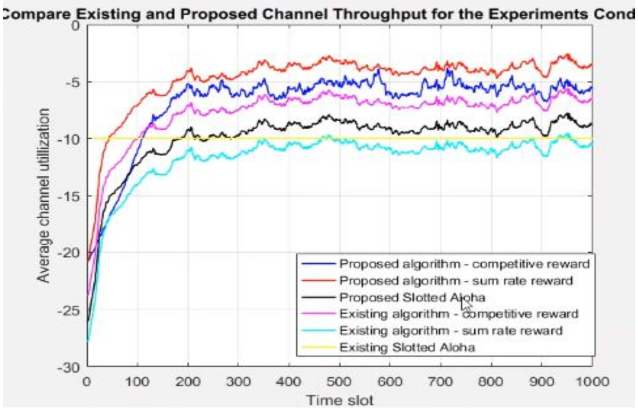


Figure 9: comparison of existing and proposed channel throughput for the experiments conducted

VI. CONCLUSION AND FUTURE SCOPE

The implementation of Deep Reinforcement Learning (DRL) using Deep Q Networks (DQN) for dynamic multichannel access in wireless networks represents a significant leap forward in the field of wireless communication and spectrum management. This research has demonstrated the potential of advanced machine learning techniques to revolutionize the way wireless devices and networks make decisions about channel selection and resource allocation. The utilization of DRL and DQN provides wireless devices and network nodes with the capability to make informed and dynamic decisions regarding channel access. This intelligent spectrum management leads to more efficient and effective utilization of available spectrum resources. The proposed approach excels in adapting to the dynamic and ever-changing conditions of wireless networks. Through continuous learning and interaction with the environment, the system can dynamically optimize channel assignments, resulting in improved network performance and user experience.

FUTURE SCOPE

In feature the proposed method can be extend As wireless communication continues to evolve with technologies like 5G and beyond, DRL with DQN-based multichannel access systems offer future-ready solutions that can adapt and thrive in the ever-changing wireless landscape.

VII. REFERENCES

[1] S. Wang, H. Liu, P. Gomes, and B. Krishnamachari, “Deep reinforcement learning for dynamic multichannel access in 5https://github.com/ANRGUSC/MultichannelDQN-channelModel 11 wireless networks,” in ICNC, 2017.

[2] R. Knopp and P. Humblet, “Information capacity and power control in single-cell multiuser communications,” in IEEE ICC, 1995.

[3] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” Computer networks, vol. 50, no. 13, pp. 2127–2159, 2006.

[4] C. Papadimitriou and J. N. Tsitsiklis, “The complexity of markov decision processes,” Math. Oper. Res., vol. 12, no. 3, pp. 441–450, 1987.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, “Playing atari with deep reinforcement learning,” arXiv preprint arXiv:1312.5602, 2013.

[6] E. Stevens-Navarro, Y. Lin, and V. W. S. Wong, “An mdp-based vertical handoff decision algorithm for heterogeneous wireless networks,” IEEE Trans on Vehicular Technology, vol. 57, no. 2, pp. 1243–1254, March 2008.

[7] P. Sakulkar and B. Krishnamachari, “Online learning of power allocation policies in energy harvesting communications,” in SPCOM, 2016.

[8] Q. Zhao, B. Krishnamachari, and K. Liu, “On myopic sensing for multi-channel opportunistic access: structure, optimality, and performance,” IEEE Trans. Wireless Commun., vol. 7, no. 12, pp. 5431–5440, dec 2008.

[9] S. H. A. Ahmad, M. Liu, T. Javidi, Q. Zhao, and B. Kr ishnamachari, “Optimality of myopic sensing in multichannel opportunistic access,”

IEEE Trans. Inf. Theory, vol. 55, no. 9, pp. 4040–4050, 2009.

[10] K. Liu and Q. Zhao, “Indexability of restless bandit problems and optimality of whittle index for dynamic multichannel access,” IEEE Trans. Inf. Theory, vol. 56, no. 11, pp. 5547– 5567, nov 2010.

[11] P. Venkatraman, B. Hamdaoui, and M. Guizani, “Opportunis tic bandwidth sharing through reinforcement learning,” IEEE Trans. on Vehicular Technology, vol. 59, no. 6, pp. 3148–3153, July 2010.

[12] Y. Zhang, Q. Zhang, B. Cao, and P. Chen, “Model free dynamic sensing order selection for imperfect sensing multichannel cognitive radio networks: A q-learning approach,” in IEEE ICC, 2014.

[13] S. Levine, C. Finn, T. Darrell, and P. Abbeel, “End-to end training of deep visuomotor policies,” arXiv preprint arXiv:1504.00702, 2015.

[14] J.-A. M. Assael, N. Wahlström, T. B. Schön, and M. P. Deisenroth, “Data-efficient learning of feedback policies from image pixels using deep dynamical models,” arXiv preprint arXiv:1510.02173, 2015.

[15] J. Ba, V. Mnih, and K. Kavukcuoglu, “Multiple object recog nition with visual attention,” arXiv preprint arXiv:1412.7755, 2014.

[16] “Solvepomdp,” http://erwinwalraven.nl/solvepomdp/.

[17] H. Liu, K. Liu, and Q. Zhao, “Logarithmic weak regret of non bayesian restless multi-armed bandit,” in IEEE ICASSP, 2011.

[18] C. Tekin and M. Liu, “Online learning in opportunistic spectrum access: A restless bandit approach,” in IEEE INFOCOM, 2011.

[19] W. Dai, Y. Gai, and B. Krishnamachari, “Efficient online learn ing for opportunistic spectrum access,” in IEEE INFOCOM, 2012.

[20] “Online learning for multi-channel opportunistic access over unknown markovian channels,” in IEEE SECON, 2014

Cite this article as :

Vipin Sharma, Lalit K. Dwivedi, Susheel K. Singh , "Implementation of Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 10 Issue 5, pp. 267-278, September-October 2023.
Journal URL : https://ijsrst.com/IJSRST52310520