I.J. Wireless and Microwave Technologies, 2026, 5, 20-36
Published Online on October 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.05.02

Decentralized Meta-Reinforcement Learning with Graphical Neural Networks for Dynamic Spectrum Access in 5G IoT Environments

Jayesh Kumar Dabi*
Department of Electronics & Communication Engineering, SVCE Indore, RGPV, India
E-mail: jayeshdabi@svceindore.ac.in
ORCID iD: https://orcid.org/0009-0005-9453-5704
*Corresponding Author

Priyadarshi Ashok Dahat
Department of Electronics &Telecommunication Engineering,IET, DAVV Indore, India
E-mail: pdahat@ietdavv.edu.in
ORCID iD: https://orcid.org/0000-0002-0049-2522

Received: May 21, 2026; Revised: June 17, 2026; Accepted: July 14, 2026; Published: October 8, 2026

Abstract: 5G IoT networks which use cognitive radio technology need dynamic spectrum access (DSA) to achieve fast response times during extreme environmental changes while managing extensive network operations. The research introduces a decentralized graph-based meta-reinforcement learning framework which enables cognitive IoT devices to learn spectrum access methods through decentralized learning. The proposed method uses Model-Agnostic Meta-Learning (MAML) with Graph Neural Networks (GNNs) to enable structure-aware few-shot adaptation which requires only local observations and neighbor interactions to function. Agents require between 1 and 5 gradient steps to develop new spectrum adaptation capabilities. The proposed framework achieved 83% spectrum utilization, an average throughput of 1.6 packets/slot, Jain's fairness index of 0.92, and a collision rate of 5%, outperforming Meta-RL and GCN-RL baselines. The study demonstrates that decentralized Meta-RL with relational learning offers an effective and scalable method for managing intelligent spectrum in future wireless IoT networks.

Index Terms: Decentralized Meta-Reinforcement Learning, Dynamic Spectrum Access, Cognitive Radio, 5G IoT, Spectrum Sharing, Multi-Agent System, Adaptive Resource Management

1. Introduction

The radio frequency (RF) spectrum is under more stress than ever as the number of wireless devices and apps grows at an unprecedented rate in the 5G and Internet of Things (IoT) eras. Many networked devices such as sensors, automated control systems, and autonomous vehicles, particularly in densely populated cities and industries [1], have squeezed the available bandwidth. The diverse and dynamic requirements of emerging wireless systems pose difficulties for current static spectrum assignment strategies, which allocate particular frequency bands to services. Dynamic Spectrum Access (DSA) is a well-known method that lets unlicensed users use licensed spectrum that isn't being used much without bothering prime users. This lets secondary users use the spectrum when they need to [2].

Some people have said that Cognitive Radio (CR) technology is very important for DSA to work. Cognitive radios, also known as cognitive transceivers, can change how they send signals based on the spectrum environment they are in [3]. 5G enables the connection of cognitive radios with smart objects, thus enhancing the performance of the IoT through lower power consumption, adaptation to diverse networks, and efficient use of spectrum resources. However, maintaining DSA in dynamic and distributed systems is a challenge. Some of the problems include that the system needs to work with a lot of devices, handle traffic patterns that change, cope with different amounts of interference, and make choices in real time. These problems get worse when centralized management systems can't be used because of problems like security, latency, or communication overhead [4].

In cognitive radio-based Internet of Things settings, DSA needs to accommodate a lot of devices that all have different Quality of Service (QoS) needs because they are all competing over a small amount of spectrum resources.

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

Because there is no centralization, each device has to figure out the best access rules on its own utilizing limited information and local observations. Also, the environment isn't always stable; the availability of spectrum changes based on the movement of devices, the main activity of the user, and the changing needs of applications. In a multi-agent reinforcement learning (MARL) situation, this causes devices to depend on each other in ways that aren't obvious when they interact with each other and the environment [5].

Recent studies have used Reinforcement Learning (RL) approaches to reduce DSA. This lets Internet of Things (IoT) devices learn how to get access to spectrum through experience [6]. Using approaches like Deep Q-Networks (DQN) and Q-learning [7], [8], in static or semi-dynamic contexts has led to good results. In many environmental situations, classic RL approaches have problems with not being able to be used in other situations, not being able to adapt to new problems, and taking a long time to converge. If there is a big change in their environment, it is normal to retrain them. This process might take a lot of time and money.

A global controller should also collect state data from all devices and use centralized reinforcement learning techniques to figure out what to do next. In theory, these approaches should work efficiently, but they are impractical for IoT deployments due to issues such as high latencies, communication overhead, and single point of failure [9]. The approach that may help to overcome these issues is the Multi-Agent RL (MARL), according to which each device becomes an individual agent. However, even using these approaches, one may face the issues of credit assignment, scalability, and non-stationarity (which arises when the environment is constantly changing under the influence of other learning agents) [10]. Nevertheless, despite these enhancements, much work remains to be done in order to create a decentralized, adaptive and scalable way of spectrum access for 5G IoT networks that are extremely unpredictable and volatile. For instance, most of the existing techniques require significant re-training in order to promptly respond to the changes in spectrum conditions. Applying an RL-based DSA system to real-life networks may be challenging due to various factors such as heterogeneity of devices, varying traffic patterns and mobility of people that can contribute to dynamic changes of the spectrum environment.

Recently, a powerful idea called meta-reinforcement learning (Meta-RL) has come out to help close this gap. Meta-RL helps agents quickly adapt to new tasks by teaching them how to learn on their own by using what they already know. This makes it a great candidate for DSA, where devices have to deal with a wide range of changing spectrum circumstances [11]. Even though Meta-RL could be used in decentralized, multi-agent IoT networks, most of its present uses are in centralized settings or non-wireless areas like simulation or robotics. To make dynamic spectrum access possible in cognitive radio-based 5G IoT networks, a new decentralized meta-reinforcement learning system is suggested. Our main goal is to give all IoT devices the ability to work on their own as cognitive agents that can learn spectrum access rules that apply to a wide range of traffic situations and patterns of interference. During the meta-training phase, the agent is exposed to a wide range of spectrum situations, which helps it learn a meta-policy that includes the basic structure of these tasks. During the deployment (meta-testing) phase, there is no need for long retraining because the agent can quickly change its policy with a few local contacts in the new environment.

There is no need for a central controller because the suggested strategy depends on agents learning and adjusting on their own utilizing only information that is available locally. Because of this decentralization, responsiveness, error tolerance, and scalability all get better. The meta-learning paradigm also makes the system more flexible and speeds up convergence in contexts that aren't stationary, where regular RL methods don't work. The main value of this study is that it created a new distributed self-assembly (DSA) learning model for cognitive 5G Internet of Things (IoT) networks. It is possible to access the spectrum without any central coordination. By giving devices a lot of different DSA jobs to do, we've come up with a meta-training method that lets them learn rules that work in a lot of different situations. The major contribution of the proposed work can be summarized as follows:

1. The Meta-RL framework for DSA functions as a complete decentralized system which eliminates the need for a central parameter server.
2. The GNN-based relational encoding system has been integrated into the meta-learning process through its implementation within the meta-learning loop.
3. The system provides a few-shot adaptation mechanism which enables users to quickly adjust policies for changing spectrum environments.
4. The system supports communication-efficient local learning which enables its use in large-scale IoT environments.
5. The study assessed system performance by testing scalability with 50 to 200 devices and evaluating multiple performance metrics.

The organization of the paper is as follows: In Section II, previous studies on reinforcement learning and spectrum access in cognitive radio networks are reviewed. In Section III, the system model with the network architecture and problem formulation is provided. In Section IV, the decentralized meta-RL scheme is described. Section V contains the experiment setup and experimental results. Finally, Section VI concludes the paper and discusses the future work directions.

2. Related Work

Dynamic spectrum access (DSA) is one of the main solutions for tackling the problem of spectrum scarcity in cognitive radio networks (CRNs) amid the growth in 5G-based IoT devices. The application of AI technologies, such as reinforcement learning (RL), has allowed for remarkable advancements in automation of spectrum management. Nevertheless, existing approaches to RL still contain a number of drawbacks related to adaptability, scalability, and generalization. In order to solve the stated problems, researchers have recently introduced several new methods, including deep reinforcement learning, multi-agent systems, meta-learning, and graph-based algorithms. This part covers the review of ten relevant papers on the topic, their methodologies and limitations as the motivation for our model. Authors in [12] have used deep Q-networks (DQN) for resource allocation in vehicular-to-vehicular (V2V) communication. They have used Markov Decision Process (MDP) to model the wireless environment. It is very useful for capturing the dynamics of the vehicle network and the improvement in spectrum efficiency. The centralized structure of the method makes it inappropriate for use in large scale IoT systems since scalability and device independence are extremely important there. Moreover, the learning process had problems with generalization to multiple interference situations. Using multi-agent deep reinforcement learning (MADRL) for DSA in CRNs was an advancement from this by [13]. The decentralization was possible because each agent learned its own policy. It made the agents independent; nevertheless, they faced the problem of non-stationarity due to the simultaneous learning process and slow convergence. There is no way to generalize common structures to various situations.

A method for DSA based on actor-critic based RL under partial observability has been proposed by Zhong et al. [14], which helps the agent to learn in such scenarios. They found that the system was fair and efficient in operation, but it was very sensitive to its settings and could not adapt quickly to any new variables. The assumption made by their algorithm that the network topology remains static and does not change with time was incorrect.

The introduction of Meta-RL in C-RANs by Huang et al. [15] is essential progress made in this field. The model utilizes MAML in order to make policies adaptable to different distributions of tasks. This means that meta-learning can be advantageous, but it is not very scalable and reliable due to the centralized approach. Moreover, there were no questions regarding the issue of spectrum sharing in IoT systems where nodes are located randomly. The model proposed in reference [16] is based on the DQN algorithm and GNN in order to provide cooperative DSA in the context of changing the network. The solution uses a graph in order to demonstrate the relationships between IoT devices as nodes and edges. This helps to share policies and demonstrates the relations between the objects in space. However, the solution is not appropriate in fast-changing situations due to the centralized approach and lack of generalization at the level of tasks.

In a similar fashion, reference [17] has proposed the Graph Convolutional RL framework which deals with dynamic spectrum management in non-uniform wireless networks. The agents were capable of reasoning about their relationship through the use of structural information. It showed improved coordination and throughput. Nevertheless, the use of synchronous communication and a stable graph topology makes it unsuitable for practical deployment for IoT applications which cannot always be available. Researchers also explored the use of federated reinforcement learning (FRL) for ensuring privacy in DSA within the peripheral devices [18]. Communication of models instead of data was used by the agents. It ensured the privacy of the community while acquiring a common policy. This type of learning was greatly efficient for overcoming security issues; however, it was inefficient for IoT nodes since it had slow adaptation and high communication cost.

A promising hybrid method was proposed by the researchers to improve spectrum access in CRNs through leveraging both deep reinforcement learning and transfer learning methods [19]. By using the method they adopted, they managed to increase the speed of convergence in the new setting by making use of the experience they gained in the previous settings. The method proved to be too rigid to adapt to the changes in the connectivity structure of the network, and it required fine tuning of the source and target domains.

Researchers in reference [20] have shown how the hierarchical reinforcement learning (HRL) can be used to manage the various layers of the spectrum. The HRL makes the distinction between the short-term and long-term decision-making processes. While the former provides fairness and stability in the long run, it was not clear how it could be used in the decentralized environment, and it does not capture interactions between the agents in the form of structured data, which hinders its scalability. Finally, Tan et al. [21] have come up with a collaborative multi-agent meta-RL model for DSA. It was more effective compared to the conventional approach of MADRL; yet, it did not have the inductive bias necessary for modeling the relations between the agents in large-scale IoT systems, namely the graph representation of the agents' interaction structure.

There are three common limitations in the above literature: (1) lack of sufficient scalability of decentralized schemes because of centralized learning or communication requirements, (2) low ability to adapt to changing and heterogeneous environments because of poor convergence or inability to generalize, and (3) lack of structural awareness, as most of the reinforcement learning approaches do not take into account the relationships between the nodes of the IoT network.

In order to overcome the limitations above, a new Graph Reinforcement Learning-based Decentralized Meta-learning Approach is proposed, which consists in modeling of the IoT environment by a dynamic interaction graph. Through the use of graph neural networks for encoding the relational dependencies, the combination of the graph learning approach and meta-reinforcement learning allows each device to learn the policy initialization generalized to various

spectrum access tasks and fast on-site adaptation based on local experience.

3. System Preliminaries

3.1. Network Architecture

The cognitive radio network considered is the one that is decentralized and is used in the 5G enabled IoT ecosystem where there are many intelligent IoT devices operating within shared spectrum bands as depicted in Fig. 1.

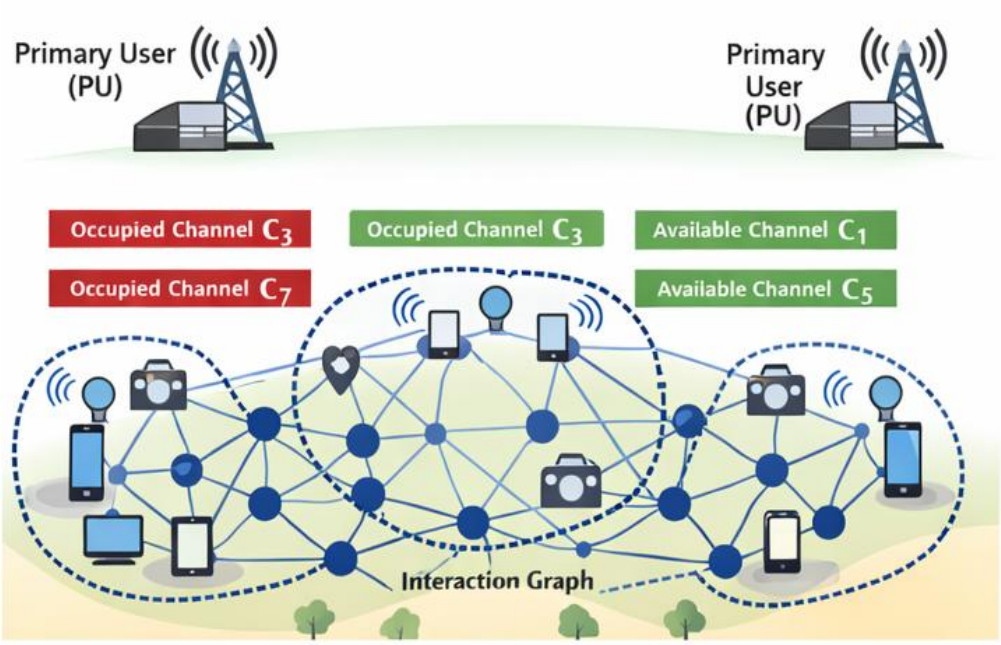


Fig. 1. Decentralized cognitive radio-based 5G IoT network architecture

The key elements of this approach include the following: [22]

- Primary Users (PUs): Primary Users are licensed and they have access to frequency bands for which they hold licenses. The usage of spectrum by the primary users is random and unpredictable.
- Secondary Users (SUs): Cognitive IoT devices (unlicensed) access the spectrum if it is free from the usage of primary users. These secondary users possess sensing, communication and learning capabilities.
- Spectrum Channels: Let there be C non-overlapping frequency channels. The spectrum channel can either be occupied or vacant based on the presence of primary users at any instant.
- Decentralized Agents: Each secondary user is assumed to be an autonomous agent and each agent has local observations and there is no centralized controller. Communication between agents is possible only among their immediate neighbors through graph-topology if needed.

The environment is dynamic, observable and non-stationary; thus, it requires more efficient dynamic and learning-based methods rather than rule-based approaches. [23]

To model inter-agent relationships and spectrum competition, we define a connectivity graph $G=(V,E)$, where each node $v_i \in V$ represents a secondary user, and an edge $e_{ij} \in E$ indicates the presence of communication or interference between agents i and j. This graph may change over time based on device mobility or dynamic channel interference patterns.

3.2. Primary User Activity and Spectrum Dynamics

The Dynamic Spectrum Access (DSA) problem is formulated as a Partially Observable Markov Decision Process (POMDP) formulation that has been generalized to include a multi-agent and meta-learning framework. Following are the key components of the model:

1. State Space (S)

Each agent i has a local view of the environment described by a local state vector $s_i^t \in S$ at time t. This includes:

- Local spectrum sensing outcomes: binary indicators of channel availability.
- Interference level on each channel from neighboring SUs.
- Recent history of successful/failed transmissions.
- PU activity estimates, if observable.

The global state S^t is not directly available to any agent due to the decentralized nature of the system.

2. Action Space (A)

Each agent selects an action $a_i^t \in A$, where:

$$A = \{0,1,\ldots,C\} \tag{1}$$

Here, $a_i^t = 0$ represents no transmission, while $a_i^t = c$ for $c \in \{1,\ldots,C\}$ indicates selecting channel c for data transmission.

3. Observation Space (O)

Due to partial observability, agent i receives an observation $o_i^t \in O$, a noisy and limited representation of its local state s_i^t. Observations may include:

- Channel sensing results.
- Local transmission success/failure.
- Estimated neighbor activity on each channel.

4. Reward Function (R)

Each agent receives a scalar reward r_i^t at each time step t, defined as:

$$r_i^t = \begin{cases} +1, & \text{successful transmission} \\ -2, & \text{collision with primary user} \\ -1, & \text{collision with secondary user} \\ -0.2, & \text{idle channel selection} \\ 0, & \text{spectrum sensing only} \end{cases} \tag{2}$$

A larger penalty is assigned to collisions involving primary users to ensure protection of licensed spectrum holders and to better reflect practical cognitive radio network requirements. The goal is to maximize the expected cumulative reward over time:

$$\max_{\pi} E\left[\sum_{t=0}^{T} \gamma^t r_i^t\right] \tag{3}$$

where $\gamma \in [0,1]$ is the discount factor, and π is the learned policy.

5. Policy (π_θ)

Each agent learns a parameterized policy π_θ that maps observations to actions. The policy is trained using Meta-Reinforcement Learning techniques to enable rapid adaptation to new interference conditions or PU patterns.

3.3. Meta-Learning Integration

To enhance generalization and adaptability, we define a task distribution $\Gamma \sim p(\Gamma)$, where each task represents a distinct spectrum environment configuration (e.g., different PU traffic, topology, SU density). The meta-learning

objective is to find an initial policy parameter θ^* that enables fast adaptation across tasks:

$$\theta^* = \arg\min_{\theta} E_{\Gamma \sim p(\Gamma)} \left[L_\Gamma \left(U\left(\theta, \Gamma\right) \right) \right] \tag{4}$$

Here, $U\left(\theta, \Gamma\right)$ denotes the updated policy parameters after a few local gradient steps on task Γ, and L_Γ is the task-specific loss (negative cumulative reward)

3.4. Graph Reinforcement Learning Perspective

To capture the dependencies between agents, we represent the environment using graph neural networks (GNNs). Each agent's local policy is updated not only based on its own observation but also on messages passed along the edges of the interaction graph *G*. [24]

At each time step:

$$h_i^{(k)} = GNN_\varphi \left(h_i^{(k-1)}, \left\{ h_j^{(k-1)} \middle| j \in N\left(i\right) \right\} \right) \tag{5}$$

where:

- $h_i^{(k)}$ is the hidden state of agent *i* after k message-passing steps,
- $N\left(i\right)$ is the set of neighboring agents,
- φ denotes the graph-based aggregation and update function.

The final policy is conditioned on this relational representation:

$$\pi_\theta \left(a_i^t \middle| o_i^t, h_i^{(k)} \right) \tag{6}$$

This formulation enables each agent to learn both task-level adaptability and structure-aware cooperation, facilitating robust and decentralized decision-making in highly dynamic IoT networks.

4. Proposed Framework for Decentralized Meta-Reinforcement Learning based Dynamic Spectrum Allocation

This section presents the proposed Decentralized Meta-Reinforcement Learning Framework for enabling intelligent, adaptive, and efficient dynamic spectrum access (DSA) in cognitive radio-enabled 5G IoT environments. The framework is designed to operate under decentralized conditions, where cognitive IoT devices learn and adapt spectrum access policies independently while incorporating relational knowledge through a graph-based message passing mechanism. The key innovation lies in the integration of Meta-Reinforcement Learning (Meta-RL) with Graph Neural Networks (GNNs) to enable fast adaptation and cooperation among agents in dynamic, multi-agent environments.

4.1. Overview of the Framework

The proposed overall methodology has two major stages as depicted in Fig. 2:

- Meta-training stage: A collection of cognitive agents (SUs) is trained using a wide distribution of spectrum access tasks. Every task involves a specific spectrum environment defined in terms of PU traffic, SU density, interference network, and noise. The purpose is to discover an initialization of meta-policy that generalizes over the task distribution.
- Meta-testing (deployment) stage: The deployment happens in a live environment where the agents are placed in a new and unseen spectrum environment. With the help of the initialized meta-policy, each agent learns from its experience and quickly adapts itself to the environment.

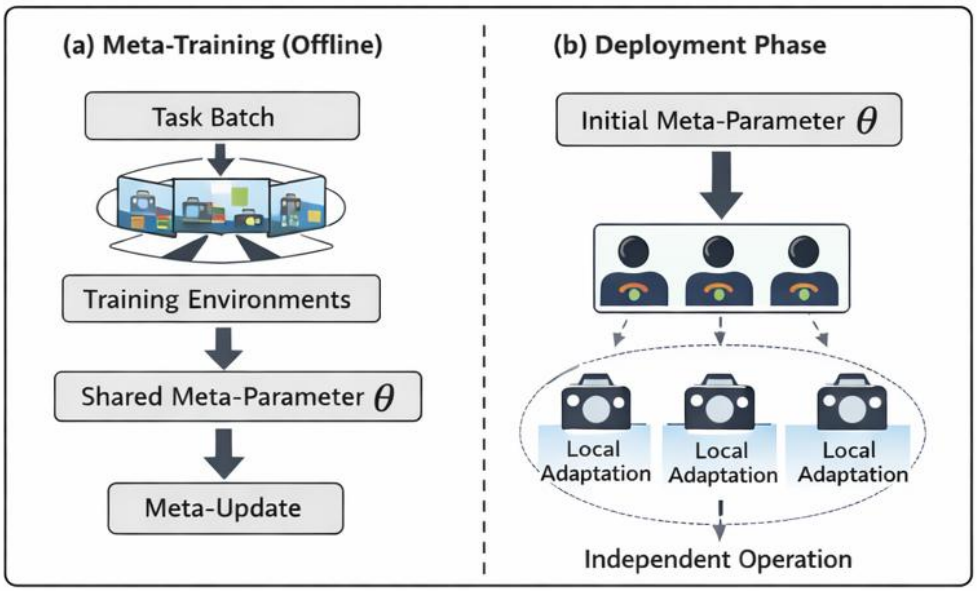


Fig. 2. Training and Deployment phases

As illustrated in the figure above, the overall architecture consists of:

- Decision and observation by agents individually,
- Propagation of information using graphs for structure learning and coordination,
- Meta-learning using policy gradients for quick adaptation.

4.2. Environment Task Distribution for Meta-Learning

Let $\Gamma_i \sim p(\Gamma)$ denote a task sampled from a task distribution $p(\Gamma)$. Each task Γ_i defines a unique dynamic spectrum environment characterized by:

- Channel availability patterns due to primary user (PU) activity,
- Number of active secondary users (SUs),
- Device mobility patterns and interference topology,
- Channel noise and fading parameters.

During meta-training, the goal is to expose the agents to a diverse set of such tasks to learn a generalizable policy initialization that can be fine-tuned quickly when faced with new tasks during deployment.

4.3. Graph-Based Relational Representation Learning

For cognitive radio-based decentralized 5G IoT systems, spectrum access decisions by individual SUs are by nature interdependent because of shared channels, mutual interference, and varying network topologies. Traditional methods in reinforcement learning do not take into account such dependencies among the agents and hence do not exploit them, which results in inefficient spectrum usage and collisions. To overcome this limitation, GNNs are introduced into the proposed model to help in representing the network in terms of structure and cooperate in spectrum access.

The cognitive radio network is modeled as a dynamic graph

$$G_t = (V_t, E_t) \tag{7}$$

where V_t denotes the set of active secondary users at time t, and E_t represents the communication or interference relationships among neighboring users. Each node $v_i \in V_t$ corresponds to an autonomous cognitive radio agent, while an edge $(i, j) \in E_t$ indicates that agents i and j can potentially interfere with each other or exchange local information.

Each agent maintains a feature vector that captures its local network state:

$$x_i = [s_i, I_i, H_i, P_i] \tag{8}$$

where

- s_i represents local spectrum sensing observations,
- I_i denotes measured interference levels,
- H_i contains recent transmission history,
- P_i represents estimated primary user activity.

The objective of the graph representation learning module is to transform these local observations into a richer latent representation that incorporates neighborhood information and interference awareness.

1) Message Passing Mechanism

To capture interactions among neighboring agents, message passing is performed over the graph. During each graph convolution iteration, an agent receives information from its neighboring nodes and aggregates it to update its hidden representation.

For agent i, the message received from neighboring agent j at iteration k is defined as

$$m_{ij}^{(k)} = h_{msg}\left(h_i^{(k)}, h_j^{(k)}\right) \tag{9}$$

where

$$h_{msg}\left(\cdot\right) = MLP\left(\left[h_i^{(k)} \middle\| h_j^{(k)}\right]\right) \tag{10}$$

and

- $MLP\left(\cdot\right)$ denotes a two-layer multilayer perceptron,
- $\|$ represents vector concatenation,
- $h_i^{(k)}$ is the hidden representation of agent i at iteration k.

The messages from all neighboring nodes are aggregated as

$$M_i^{(k)} = \frac{1}{\left|N\left(i\right)\right|}\sum_{j\in N(i)} m_{ij}^{(k)} \tag{11}$$

where $N\left(i\right)$ denotes the set of neighboring agents of node *i*.

2) Node Representation Update

After neighborhood aggregation, each agent updates its hidden representation using

$$h_i^{(k+1)} = \sigma\left(Wh_i^{(k)} + M_i^{(k)}\right) \tag{12}$$

where

- W is a trainable weight matrix,
- $\sigma\left(\cdot\right)$ is the ReLU activation function.

The hidden representation is initialized as

$$h_i^{(0)} = x_i \tag{13}$$

and the message-passing process is repeated for *K* iterations.

In our proposed design, a two-layer GNN with hidden layer size 64 is used. Such architecture ensures that agents can gather information from neighborhoods up to two hops away while ensuring computational efficiency for real-world use cases.

3) Relational Policy Representation

The final graph embedding for agent *i* is obtained as

$$z_i = h_i^{(K)} \tag{14}$$

This embedding captures both local spectrum observations and the structural characteristics of neighboring users. The graph embedding is concatenated with the local observation vector and supplied to the policy network:

$$\hat{o}_i = \left[o_i \| z_i \right] \tag{15}$$

The policy network then generates the probability distribution over available spectrum access actions:

$$\pi_\theta \left(a_i | o_i, z_i \right) \tag{16}$$

where $a_i \in \{0,1, \ldots, C'\}$ and C denotes the number of available channels.

4) Integration with Meta-Learning

The module responsible for learning the graph representation is very important within the meta-learning framework presented in the paper. In the process of meta-learning, agents are faced with a number of spectrum access tasks, each one having various characteristics: user density, mobility, interference levels, and activity of the primary users. GNN finds invariant relational features which are shared among all the mentioned environments.

This implies that the meta-learning mechanism doesn't rely only on its observations but works on graph representations containing information on the dynamics of the environment and relations between agents. This results in better generalization ability of the learned policy initialization across the different spectrum environments. Upon facing a new task during the deployment stage, agents need just a couple of gradient updates to learn to work properly since the graph embeddings already contain structural information about the local spectrum environment. Thus, the combination of meta-reinforcement learning and graph neural networks makes it possible to achieve rapid adaptation and coordination in the environment. Relational representation learning and meta-learning can be used as a way to enable rapid and efficient spectrum access in highly heterogeneous 5G IoT networks.

4.4. Meta-Reinforcement Learning (MAML-based Optimization)

Each agent consists of the following components:

1. **Observation Encoder**: Processes local state and environment feedback, including spectrum sensing results, interference levels, and recent transmission outcomes.
2. **Graph Encoder (GNN Module)**: Performs message passing and neighborhood aggregation over the local subgraph centered at each agent. Formally, the hidden state update at time t and message-passing iteration k is:

$$h_i^{(k)} = \sigma\left(W_1 h_i^{(k-1)} + \sum_{j \in N(i)} \varphi\left(h_i^{(k-1)}, h_j^{(k-1)} \right) \right) \tag{17}$$

where:

- $h_i^{(k)}$ is the hidden embedding of agent *i* after *k* iterations,
- φ is the message function (e.g., MLP or attention),
- σ is a non-linear activation (e.g., ReLU),
- W_1 is a learnable weight matrix.

3. **Policy Network**: Maps the concatenated encoded observation and graph embedding to an action **probability** distribution $\pi_\theta\left(a_i \middle| o_i, h_i\right)$, representing the probability of selecting a given channel or remaining idle.
4. **Value Network (for Actor-Critic Models):** Estimates the expected return (value function) based on current observation and graph state.

4.5. Meta-Reinforcement Learning (MAML-based Optimization)

We adopt a Model-Agnostic Meta-Learning (MAML) approach for optimizing the shared initial policy parameters θ. The idea is to learn a policy initialization that can be rapidly fine-tuned using a small number of gradient steps with respect to a task-specific loss.

Meta-Training Procedure:

1. Sample a batch of tasks $\Gamma_i \sim p(\Gamma)$,
2. For each task Γ_i, perform **inner-loop updates** using RL (e.g., policy gradient):

$$\theta_i' = \theta - \alpha \nabla_\theta L_{\Gamma_i}\left(\pi_\theta\right) \tag{18}$$

3. Compute **meta-loss** using the updated parameters θ_i' :

$$L_{meta} = \sum_i L_{\Gamma_i}\left(\pi_{\theta_i'}\right) \tag{19}$$

4. Update meta-policy parameters θ using the outer-loop gradient:

$$\theta \leftarrow \theta - \beta \nabla_\theta L_{meta} \tag{20}$$

where α,β are learning rates for inner and outer loops respectively.

Meta-Testing Procedure:

At deployment, each agent receives θ and performs rapid local updates using observations from the new environment Γ_{new} to obtain an adapted policy θ_{new}.

4.6. Decentralized Learning and Adaptation

Unlike centralized RL approaches, each agent:

- Performs **local training and updates** using only its own rewards and observations,
- Communicates with neighbors via the GNN to gain relational awareness,
- Executes policies **asynchronously**, making the system robust to delays, failures, and partial observability.

This architecture ensures **scalability**, **robustness**, and **autonomy** in dense and heterogeneous 5G IoT networks.

4.7. Loss Function and Optimization

The overall loss function for each agent *i* in task Γ includes:

1. **Policy Loss** (e.g., for REINFORCE or PPO):

$$L_{policy}^i = -E_{\pi_\theta}\left[\log \pi_\theta\left(a_i \middle| o_i\right) \cdot \hat{A}_i\right] \tag{21}$$

where $\hat{A}_i$ is the advantage estimate.

2. **Value Loss** (for actor-critic):

$$L_{value}^{i} = \left(V_{\theta}\left(o_i\right) - \hat{R}_i\right)^2 \tag{22}$$

3. **Entropy Loss** (for exploration):

$$L_{entropy}^{i} = -E_{\pi_{\theta}}\left[\pi_{\theta}\left(a_i \middle| o_i\right)\log \pi_{\theta}\left(a_i \middle| o_i\right)\right] \tag{23}$$

The final optimization objective per agent is:

$$L_i = L_{policy}^{i} + \lambda_1 L_{value}^{i} + \lambda_2 L_{entropy}^{i} \tag{24}$$

where λ_1, λ_2 are weighting coefficients.

4.8. Computational Complexity Analysis

For N agents and E graph edges:

$$O\left(Ed\right) \tag{25}$$

for graph message passing.
Meta-learning complexity:

$$O\left(KNd\right) \tag{26}$$

where:

- K = inner-loop updates
- d = hidden dimension

Total complexity:

$$O\left(Ed + KNd\right) \tag{27}$$

Algorithm 1: Decentralized Meta-RL with GNN for Dynamic Spectrum Access

Input:

- Task distribution $p\left(\Gamma\right)$ over dynamic spectrum environments
- Learning rates: α (inner loop), β (meta-update)
- Meta-iterations M, gradient steps K, agents N
- Initial policy parameters θ

Output:

- Meta-learned policy parameters θ^*
- Rapidly adaptable decentralized policy for new spectrum tasks

---

Phase 1: Meta-Training (Offline Stage)
1: Initialize shared policy parameters θ
2: for meta-iteration m=1 to M **do**
3: Sample a batch of spectrum tasks $B = \left\{\Gamma_i \sim p\left(\Gamma\right)\right\}$
4: for each task $\Gamma_i \in B$ **do**
5: Construct dynamic interaction graph $G_i = \left(V, E\right)$ for agents in Γ_i
6: Initialize task-specific parameters $\theta_i \leftarrow \theta$
7: for gradient step k=1 to K **do**

8: for each agent $j \in V_i$ **:**

- Observe local state o_j^t
- Perform **message passing** using GNN over $N(j)$

$$h_j^{(k)} = GNN_\phi\left(o_j^t, \left\{o_n^t \middle| n \in N(j)\right\}\right)$$

- Compute action $a_j^t \sim \pi_\theta\left(a \middle| o_j^t, h_j^{(k)}\right)$
- Receive reward r_j^t, store trajectory τ_j.

9: Compute task-specific loss $L_{\Gamma_i}(\theta_i)$

$$L_{\Gamma_i} = \sum_j \left(L_{policy}^j + \lambda_1 L_{value}^j + \lambda_2 L_{entropy}^j\right)$$

10: Update task parameters using policy gradient:

$$\theta_i' = \theta_i - \alpha \nabla_\theta L_{\Gamma_i}$$

11: end for

12: end for

13: Compute meta-objective across all tasks:

$$L_{meta}(\theta) = \sum_{\Gamma \in B} L_{\Gamma_i}(\theta_i')$$

14: Update shared parameters using meta-gradient:

$$\theta \leftarrow \theta - \beta \nabla_\theta L_{meta}$$

15: end for

Phase 2: Meta-Testing (Deployment Stage)

16: Given new spectrum task $\Gamma_{new} \sim p_{test}(\Gamma)$

17: Construct interaction graph $G_{new}=(V,E)$

18: Initialize agent policies with $\theta_{new} \leftarrow \theta$

19: for each agent $j \in V$ **do in parallel**

20: Observe local state o_j^t, sense PU/SU activity

21: Perform message passing over $N(j)$ to get h_j

22: Execute action $a_j^t \sim \pi_{\theta_{new}}\left(a \middle| o_j^t, h_j\right)$

23: Receive reward r_j^t, store trajectory τ_j

24: Optionally fine-tune θ_{new} using few gradient steps:

$$\theta_{new} \leftarrow \theta_{new} - \alpha \nabla_{\theta_{new}} L_{\Gamma_{new}}$$

25: Repeat for each decision epoch

26: end for

---

5. Results and Discussion

An experimental setup was designed to comprehensively evaluate the proposed framework's adaptability, scalability, and efficiency. The combination of meta-learning and GNN-based policy networks provides a powerful foundation for decentralized, intelligent, and robust spectrum management in cognitive radio-based 5G IoT environments. The simulation setting assumes the existence of a 500 m × 500 m geographical area with an uncertain number of SUs and PUs. The quantity of SUs varies from 50 to 200 in different experimental conditions to check the scaling of the proposed approach. SUs were considered decentralized intelligent agents with restricted abilities of sensing and communications. They interacted via 10 orthogonal frequencies used by 5 to 20 intermittent PUs. The PUs were simulated using a two-state ON-OFF Markov chain model, which emulates the variability of the licensed spectrum. Mobility of the SUs was assumed to follow a random walk with velocities varying from 0 to 5 meters per second to investigate the effectiveness of the framework in dynamic networks.

The agents operate with a limited radius of communication equal to 100 meters; therefore, GNN message passing was restricted to neighborhood areas only. The decision on channel selection of each agent depends on the results of the

local observation, received messages from neighbors, and policy learned in the meta-learning process. The transmission power was constant for all agents and set to 20 dBm. In order to learn the meta-policy, the tasks for the spectral environment were created in varying conditions, such as the density of PU activity (sparse or dense), the density of SUs, noise (AWGN, Rayleigh fading), and mobility. Such variety was necessary to allow the meta-learning algorithm to generalize its knowledge on the basis of various environments. Every task corresponded to some real situation that could arise within the work of the network.

The primary reinforcement learning technique employed by the learning system was a combination of Proximal Policy Optimization (PPO) and Model-Agnostic Meta-Learning (MAML). Every agent employed the actor-critic architecture, in which the critic estimated value functions and the actor chose how to allocate actions. During deployment, agents were able to swiftly modify their local policies with only a few gradient steps because of the meta-learning process's discovery of the optimal approach to establish policy parameters θ for every job. The observations were encoded using a two-layer multilayer perceptron (MLP), and messages were sent between graphs using a two-hop Graph Neural Network (GNN). The structure of their immediate surroundings became clearer to the agents as a result.

Each training event lasted for 100 time steps, and the meta-training process continued for 500 iterations. The learning rate for the inner loop was set to 0.001, while the meta-update step used a slower rate of 0.0005. The discount factor was fixed at 0.95 to ensure that both short-term and long-term rewards were considered. Exploration was promoted through entropy regularization, with a coefficient of 0.01. The hyperparameters for the respective proposed model are as given in Table 1:

Table 1. Hyperparameters settings

Parameter	Value
GNN Type	Graph Attention Network (GAT)
Number of GNN Layers	2
Hidden Dimension	64
Activation Function	ReLU
Aggregation Method	Attention-weighted Sum
Observation Encoder Layers	64, 32
Policy Network Layers	64, 32
PPO Clip Parameter	0.2
Inner Learning Rate (α)	0.001
Meta Learning Rate (β)	0.0005
Discount Factor (γ)	0.95
Batch Size	32
Meta Iterations	500

The effectiveness of the proposed framework was further investigated through an ablation study. As shown in Table 2, four different configurations were evaluated to quantify the individual contributions of Graph Neural Networks (GNNs) and Meta-Reinforcement Learning (Meta-RL).

Table 2. Ablation study

Method	Spectrum Utilization	Throughput	Collision Rate
PPO	69±1.8	1.21±0.06	14±1.2
PPO+GNN	74±1.4	1.35±0.05	10±0.8
Meta-RL	76±1.3	1.40±0.04	8±0.7
Proposed	83±1.1	1.60±0.03	5±0.5

The basic PPO framework yielded a spectrum utilization of 69%, reflecting poor ability to adapt to varying spectrum conditions. The integration of GNN-based relational learning led to an increase in spectrum utilization to 74% and a lowering of the collision ratio from 14% to 10%. Such results reveal the significance of utilizing neighborhood relations and interference between the secondary users. Application of Meta-RL without the use of graph-based representations increased the spectrum utilization ratio to 76%, proving the efficiency of meta-learning in achieving fast adaptation to the unknown spectrum environment.

The proposed Meta-RL + GNN framework achieved the best performance across all evaluation metrics, attaining 83% spectrum utilization, 1.60 packets/slot throughput, and only 5% collision rate. Compared with Meta-RL alone, the inclusion of graph-based relational learning yielded an additional 7% improvement in spectrum utilization and a 37.5% reduction in collision rate. These results confirm that Meta-Learning and Graph Neural Networks provide complementary benefits. Meta-learning enables fast adaptation across diverse spectrum access tasks, whereas graph-based relational

learning captures interference dependencies and facilitates cooperative decision-making. Their integration results in a synergistic improvement that significantly enhances spectrum efficiency, throughput, and collision avoidance in decentralized cognitive radio-based 5G IoT networks.

We compared the efficacy of the suggested decentralized Meta-RL methodology using a number of cutting-edge spectrum access schemes as baseline models as shown in Table 3. These included a meta-reinforcement learning model without any structural graph information, a GCN-RL model that used graph convolution but lacked meta-learning, a Multi-Agent DQN (MADQN) with learners that operated independently, and a centralized Deep Q-Network (DQN) that could see everything. As a lower bound reference, a non-learning random access method was also used. Performance was assessed using a variety of quantitative metrics. Spectrum usage efficiency, which is calculated by dividing the number of successful SU broadcasts by the total number of available channel-time slots, was the first significant metric. This metric examines the network's ability to utilize available spectrum without encountering PUs. Another crucial metric was the average throughput per SU. It is a useful tool for tracking the network's quality of service since it calculates the quantity of data that each agent successfully sends in a specified period of time. To determine how equally each agent has access to the spectrum, we used Jain's fairness index. A higher fairness index indicates equitable resource distribution, which is essential for maintaining excellent service quality in congested IoT environments. Another important metric was the collision rate, which demonstrated the frequency of transmission failures due to SU-SU or SU-PU interference. When there are fewer crashes, more thoughtful and well-coordinated decisions are made. In addition to these key performance metrics, we also examined the learning algorithm's rate of convergence. The number of training sessions required to bring the agent's performance within 95% of its final plateau value was the definition of this. Faster convergence is preferable in dynamic environments where adaptation must occur faster.

Table 3. Performance Comparison of the Proposed technique

Metric	Proposed Meta-RL (GNN)	Meta-RL (No GNN)	GCN-RL
Spectrum Utilization	**83.0 ± 1.1**	76.0 ± 1.3	74.0 ± 1.5
Avg. Throughput	**1.60 ± 0.03**	1.40 ± 0.04	1.35 ± 0.05
Jain’s Fairness	**0.92 ± 0.01**	0.88 ± 0.02	0.89 ± 0.02
Collision Rate	**5.0 ± 0.5**	8.0 ± 0.7	10.0 ± 0.8

Regarding the efficiency of spectrum use, the proposed Meta-RL with GNN has shown the highest efficiency level of 83% which is higher than Meta-RL without GNN (76%) and GCN-RL (74%). This means that although meta-learning helps agents adapt and learn to generalize tasks in dynamic environments, the use of graph-based relational modeling can still help them in better optimizing channel selection through interference avoidance using local neighborhood context. The same is evident with regard to average throughput where the proposed framework has shown a packet/slot rate of 1.6 which is higher than 1.4 and 1.35 packets/slot of Meta-RL (No GNN) and GCN-RL, respectively. This shows the efficiency of combining task adaptation from meta-learning and spatial coordination from GNN in enabling the agents to make better spectrum access decisions based on time-varying spectrum environment. With regard to the fairness metric, the Jain's Index value of the proposed approach was always greater than 0.92 as compared to 0.88 and 0.89 in the case of the other two approaches. This means that the use of graph-based message passing has helped the agents make better decisions in distributing the opportunity for spectrum access instead of adopting a selfish approach. This means that GNNs enable the agents to predict interference hazards with greater ease, whereas meta-learning allows the agents to adapt to new forms of interference easily. In conclusion, the model demonstrates that the cognitive 5G IoT systems can have efficient, decentralized spectrum management through the use of Meta-RL and GNNs. The above is the perfect balance of efficiency, equity, and reliability.

6. Conclusion

The concept of a decentralized Meta-RL method that would help cognitive radio-enabled 5G IoT networks with dynamic spectrum access needs was examined in this paper. Our model consists of combining the capabilities of Graph Neural Networks and Model-Agnostic Meta-Learning in order to allow IoT nodes to adapt their spectrum access policies rapidly and efficiently without relying on centralized mechanisms in the case of dynamic and interfered environments. Extensive simulations have proven that our model is superior to the current state-of-the-art Meta-RL and GNN architectures when evaluated through the lenses of spectrum utilization, throughput, fairness, and collision avoidance. The meta-learning part allows for quick adaptation of the policies within tasks while the GNN-based message passing part enables cooperation between agents in the same environment. These results confirm that smart and scalable spectrum management should be structure-aware and flexible in dense IoT environments. This method proves to have great potential for actual application in the next generation of IoT networks, but it still needs to be validated in real-world testbeds.

All the Declarations and Statements

Author Contributions Statement
Jayesh Kumar Dabi – Conceptualization, Methodology, Software Implementation, Model Training, Validation, and Performance Evaluation and Drafted the initial manuscript.
Priyadarshi Ashok Dahat – Writing – Review and Editing, and Project Management: Reviewed and edited the manuscript, ensured clarity and coherence, and helped coordinate project milestones and deadlines.

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement
The authors declare no conflicts of interest.

Funding Declaration
The authors received no financial support for the research, authorship, and/or publication of this article.

Data Availability Statement
No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Ethical Declarations
No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Acknowledgments
The authors would like to state that there are no acknowledgments for this work.

Declaration of Generative AI in Scholarly Writing
During the preparation of this manuscript, the authors used generative AI solely to improve the language and readability of the text. The authors reviewed and edited the output as necessary and take full responsibility for the content of this publication.

Abbreviations
The following abbreviations are used in this manuscript:

DSA - Dynamic Spectrum Access
MAML - Model-Agnostic Meta-Learning
GNN - Graph Neural Network
RL - Reinforcement Learning
RF - Radio Frequency
QoS - Quality of Service
DQN - Deep Q-Network
IoT - Internet of Things
Meta-RL - meta-reinforcement learning
V2V - vehicular-to-vehicular
PUs - Primary Users
SUs - Secondary Users
POMDP - Partially Observable Markov Decision Process
PPO - Proximal Policy Optimization
MLP - multilayer perceptron
MADQN - Multi-Agent DQN

Appendix A/B/C…, with appendix title
Not applicable.

References

[1] Cisco. (2020). *Cisco Annual Internet Report (2018–2023) White Paper*. Cisco Systems Inc.
[2] Qing Zhao., & Brian M. Sadler. (2007). A survey of dynamic spectrum access: Signal processing, networking, and regulatory policy. *IEEE Signal Processing Magazine*. DOI: https://doi.org/10.1109/MSP.2007.361604.

[3] Simon Haykin. (2005). Cognitive radio: Brain-empowered wireless communications. *IEEE Journal on Selected Areas in Communications*. DOI: https://doi.org/10.1109/JSAC.2004.839380.
[4] Ian F. Akyildiz., W.-Y. Lee., M. C. Vuran., & S. Mohanty. (2006). NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey. *Computer Networks*. DOI: https://doi.org/10.1016/j.comnet.2006.05.001.
[5] H. Ye., G. Y. Li., & B.-H. Juang. (2019). Deep reinforcement learning based resource allocation for V2V communications. *IEEE Transactions on Vehicular Technology*. DOI: https://doi.org/10.1109/TVT.2019.2897134.
[6] W. Bai., G. Zheng., W. Xia., Y. Mu., & Y. Xue. (2025). Multi-user opportunistic spectrum access for cognitive radio networks based on multi-head self-attention and multi-agent deep reinforcement learning. *Sensors*. DOI: https://doi.org/10.3390/s25072025.
[7] L. Chen., Z. Wang., X. Zhao., X. Shen., & W. He. (2024). A dynamic spectrum access algorithm based on deep reinforcement learning with novel multi-vehicle reward functions in cognitive vehicular networks. *Telecommunication Systems*. DOI: https://doi.org/10.1007/s11235-024-01188-5.
[8] O. Naparstek., & K. Cohen. (2019). Deep multi-user reinforcement learning for distributed dynamic spectrum access. *IEEE Transactions on Wireless Communications.* DOI: https://doi.org/10.1109/TWC.2018.2879433.
[9] Y. Zhou., F. Zhou., Y. Wu., R. Q. Hu., & Y. Wang. (2020). Subcarrier assignment schemes based on Q-learning in wideband cognitive radio networks. *IEEE Transactions on Vehicular Technology*. DOI: https://doi.org/10.1109/TVT.2019.2953809.
[10] J. Foerster., G. Farquhar., T. Afouras., N. Nardelli., & S. Whiteson. (2018). Counterfactual multi-agent policy gradients. *Proceedings of the AAAI Conference on Artificial Intelligence*. DOI: https://doi.org/10.1609/AAAI.V32I1.11794.
[11] X. Liu, J. Wu, S. Chen, "A context-based meta-reinforcement learning approach to efficient hyperparameter optimization," Neurocomputing, vol. 478, pp. 89-103, Mar. 2022, doi: 10.1016/j.neucom.2021.12.086.
[12] G. Chai., W. Wu., Q. Yang., R. Liu., & F. R. Yu. (2022). Learning-based resource allocation for ultra-reliable V2X networks with partial CSI. *IEEE Transactions on Communications*. DOI: https://doi.org/10.1109/TCOMM.2022.3199018.
[13] S. Wang., H. Liu., P. H. Gomes., & B. Krishnamachari. (2018). Deep reinforcement learning for dynamic multichannel access in wireless networks. *IEEE Transactions on Cognitive Communications and Networking*. DOI: https://doi.org/10.1109/TCCN.2018.2809722.
[14] C. Zhong., Z. Lu., M. C. Gursoy., & S. Velipasalar. (2018). Actor-critic deep reinforcement learning for dynamic multichannel access. *Proceedings of the IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. DOI: https://doi.org/10.1109/GlobalSIP.2018.8646405.
[15] J. Huang., J. Wan., B. Lv., Q. Ye., & Y. Chen. (2023). Joint computation offloading and resource allocation for edge-cloud collaboration in internet of vehicles via deep reinforcement learning. *IEEE Systems Journal. DOI: https://doi.org/10.1109/JSYST.2023.3249217.*
[16] M. Marwani., & G. Kaddoum. (2024). Graph neural networks approach for joint wireless power control and spectrum allocation. *IEEE Transactions on Machine Learning in Communications and Networking. DOI: https://doi.org/10.1109/TMLCN.2024.3408723.*
[17] H. Song., L. Liu., J. Ashdown., & Y. Yi. (2021). A deep reinforcement learning framework for spectrum management in dynamic spectrum access. *IEEE Internet of Things Journal*. DOI: https://doi.org/10.1109/JIOT.2021.3052691.
[18] H.-H. Chang., Y. Song., T. T. Doan., & L. Liu. (2023). Federated multi-agent deep reinforcement learning (Fed-MADRL) for dynamic spectrum access. *IEEE Transactions on Wireless Communications*. DOI: https://doi.org/10.1109/TWC.2022.3233436.
[19] Hainan Qi., Wenjie Ma., & Bingsong Zhao. (2025). Federated heterogeneous multi-agent deep reinforcement learning-based attack resilience scheduling for heterogeneous multi-integrated energy system. *Energy Reports*. DOI: https://doi.org/10.1016/j.egyr.2025.05.085.
[20] S. Zhang., K.-Y. Lam., B. Shen., L. Wang., & F. Li. (2023). Dynamic spectrum access for internet of things with hierarchical federated deep reinforcement learning. Ad Hoc Networks. DOI: https://doi.org/10.1016/j.adhoc.2023.103257.
[21] X. Tan., L. Zhou., H. Wang., Y. Sun., H. Zhao., B.-C. Seet., J. Wei., & V. C. M. Leung. (2022). Cooperative multi-agent reinforcement learning based distributed dynamic spectrum access in cognitive radio networks. *IEEE Internet of Things Journal. DOI: https://doi.org/10.1109/JIOT.2022.3168296.*
[22] H. Zhang., N. Liu., X. Chu., K. Long., A.-H. Aghvami., & V. C. M. Leung. (2017). Network slicing based 5G and future mobile networks: Mobility, resource management, and challenges. *IEEE Communications Magazine*. DOI: https://doi.org/10.1109/MCOM.2017.1600940.
[23] A. Al-Fuqaha., M. Guizani., M. Mohammadi., M. Aledhari., & M. Ayyash. (2015). Internet of things: A survey on enabling technologies, protocols, and applications. *IEEE Communications Surveys & Tutorials*. DOI: https://doi.org/10.1109/COMST.2015.2444095.
[24] Y. Zeng., J. Xu., & R. Zhang. (2019). Energy minimization for wireless communication with rotary-wing UAV. *IEEE Transactions on Wireless Communications*. DOI: https://doi.org/10.1109/TWC.2019.2902559.

Authors' Profiles



Jayesh Kumar Dabi received his B.E. degree in Electronics Engineering in 2003 from Jawaharlal Institute of Technology. He obtained his M.Tech degree in Digital Communication in 2013 from Acropolis Institute of Technology and Research. He is currently working as an Assistant Professor in the Department of Electronics and Communication Engineering at Swami Vivekanand College of Engineering. His research interests include wireless communication, digital communication, and cognitive radio networks. He has been actively involved in teaching, research, and academic activities related to modern communication systems.



Priyadarshi Ashok Dahat is currently serving as an Associate Professor in the Department of Electronics and Telecommunication Engineering at Institute of Engineering and Technology Devi Ahilya University. He completed his Bachelor of Engineering in Electronics and Communication Engineering during 1995–1999 from Malaviya National Institute of Technology Jaipur. He obtained his M.E. degree in Electronics Engineering with specialization in Digital Communication in 2008 from Institute of Engineering and Technology Devi Ahilya University. Dr. Dahat earned his Ph.D. degree in Wireless Communication from Indian Institute of Technology Kharagpur during 2012–2017. His research interests are in the field of wireless communication and include OFDMA-based cellular networks, ad hoc networks, heterogeneous networks, and future wireless networks. He has been actively engaged in teaching, research, and academic guidance in advanced wireless communication systems

How to cite this paper

Jayesh Kumar Dabi, Priyadarshi Ashok Dahat, "Decentralized Meta-Reinforcement Learning with Graphical Neural Networks for Dynamic Spectrum Access in 5G IoT Environments", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.5, pp. 20-36, 2026. DOI:10.5815/ijwmt.2026.05.02