

# Generalization of Deep Reinforcement Learning for Jammer-Resilient Frequency and Power Allocation

Swatantra Kifle, Jithin Jagannath, Zackary Kane, Noor Biswas, Prem Sagar Vasanth Kumar, Anu Jagannath

**Abstract**—We tackle the problem of joint frequency and power allocation while emphasizing the generalization capability of a deep reinforcement learning model. Most of the existing methods solve reinforcement learning-based wireless problems for a specific pre-determined wireless network scenario. The performance of a trained agent tends to be very specific to the network and deteriorates when used in a different network operating scenario (e.g., different in size, neighborhood, and mobility, among others). We demonstrate our approach to enhance training to enable a higher generalization capability during inference of the deployed model in a distributed multi-agent setting in a hostile jamming environment. With all these, we show the improved training and inference performance of the proposed methods when tested on previously unseen simulated wireless networks of different sizes and architectures. More importantly, to prove practical impact, the end-to-end solution was implemented on the embedded software-defined radio and validated using over-the-air evaluation.

**Index Terms**—Deep reinforcement learning, wireless network, power control, frequency selection, software-defined radio.

## I. INTRODUCTION AND BACKGROUND

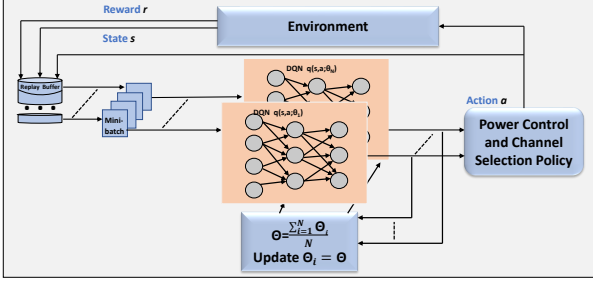
The dramatic increase in the number of connected wireless devices with the demand for higher data rates has demanded increasingly efficient use of wireless resources. Modeling next-generation of complex wireless systems and the dynamic allocation of scarce resources demand data-driven methodologies such as powerful function approximation used by deep reinforcement learning (DRL). In the last decade, several classes of machine learning algorithms were developed that have expedited research and development in different domains, including the wireless networks [1]–[6]. DRL has been studied in the wireless community, especially for the problem of dynamic resource management [7]–[12]. Power control and frequency (channel) selection have been studied in both independent and joint settings. Authors in [11] developed a centralized deep Q-Network (DQN)-based algorithm for downlink power control. The work in [12] developed a deep distributed approach for multi-agent reinforcement learning (MARL) for power control to maximize the network weighted sum rate where each agent (transmitter) exchanges its instantaneous observation with its nearby transmitter. The work in [13] assumes Device-to-Device (D2D) networks and addresses the problem of joint power control and frequency allocation. In the problem setup, a macro base station (MBS) provides signaling for the synchronization of D2D pairs and assists in allocating pilots. In [14], a DRL-based joint power control and channel selection

solution is developed in a cellular network setup. In contrast, [15] considers distributed multiple transmitter-receiver pairs (Tx-Rx pairs) setup making power control decisions but lacks the joint frequency selection considered in this work.

In most cases, these works are limited to simulation and do not generalize. We define the generalization capability of an agent as the ability of the trained agent to provide robust performance at inference when deployed in different (unseen during training) operating environment parameters such as network size, neighborhood, mobility, and jamming profile. While these are not a comprehensive list of operating environment parameters, we expect them to be defined by the operator or use case. The idea of generalization does not extend to the system parameters of the radio (that usually represent actions), such as the number of available channels and the power level, i.e., these parameters are assumed to be prespecified and continue to be the same during the deployment. None of these works address the generalization capability of agent performance during inference in networks that have mismatches to the training environment. In real-world scenarios, the wireless network used for training and testing will most likely differ, resulting in performance degradation. So, trying to incorporate all the specifics of radio frequency (RF) environments and operating scenarios will lead to explosions of the solution space. This results in an exponential increase in sample complexity, demanding very high computational resources. We address these problems by proposing a novel training approach to the multi-agent problem in real-world wireless networks. To demonstrate the effectiveness of our approach, we consider a problem of joint power control and channel selection among the wireless nodes in an active jamming environment in a low probability of intercept and detection (LPI/D) networks due to its relevance to a tactical wireless network. To the best of our knowledge, there is no previous work that has designed and deployed DRL for joint power and channel selection in a hostile jamming environment on actual Software defined radio (SDR). We use direct-sequence code-division multiple access (DS-CDMA) to implement a robust LPI/D physical layer due to its advantages, such as easy frequency management and low peak-to-average power ratio (PAPR), among others. We empirically show the convergence of the proposed training approach. However, analysis of theoretical convergence in multi-agent RL problems with model parameter averaging when the global state space is partially observable in each agent is still an open problem in the literature [16], [17].

**Contribution:** In this work, (i) we develop a new training approach to enhance the generalization of multi-agent DRL problems in wireless networks. We demonstrate the effectiveness of the solution by addressing the problem of joint power control and frequency selection in an active jamming

Swatantra Kifle, Jithin Jagannath, Zackary Kane, Noor Biswas, Prem Sagar Vasanth Kumar, and Anu Jagannath are with the Marconi-Rosenblatt AI/ML Innovation Laboratory, ANDRO Computational Solutions, LLC, 7980 Turin Rd, Rome, NY 13440, USA. Emails: {skifle, jjagannath, zkane, nbiswas, pkumar, ajagannath}@androcs.com



**Fig. 1:** Multi-agent Deep Reinforcement Learning for the joint power control and channel selection

environment on actual radio hardware for the first time in literature (to the best of our knowledge). (iii) We provide extensive results of the proposed training method when the trained agent is deployed in simulated and actual wireless networks of different sizes and architectures. (iii) Furthermore, the solution is validated using over-the-air (OTA) evaluation when the DRL agents are deployed at the edge on embedded SDRs. (iv) Finally, we provide a video demonstration of the solution operating in real-time in real-world outdoor deployment.

## II. PROBLEM FORMULATION AND PROPOSED SOLUTION

Consider a wireless network consisting of several transceiver pairs using DS CDMA protocol. Assume that there are  $2K$  wireless transceivers, each with spreading sequence matrix  $\mathbf{S} \in \mathbb{C}^{L \times 2K}$ , where  $L$  is the length of the spreading code. This forms  $K$  transmit-receive pairs that can transmit at different power levels,  $p_1 \dots, p_{n_p}$ , and multiple frequencies  $f_1, \dots, f_{n_f}$ , where  $n_p$  and  $n_f$  are the numbers of power level and available frequencies, respectively. All agents learn policy to choose power level and channel by maximizing the same discounted sum of rewards. We assume each agent has partial observability, i.e., it cannot observe the entire underlying Markov state. Hence, we are interested in designing a reward function such that agents prefer to transmit at the lowest power and use available channels uniformly when maximized. We focus on the maximization of rewards through some form of centralized training. In this setup, each agent can share their learned model for aggregation.

We consider a problem of joint power control and channel selection in an active jamming environment through the lens of generalization. In this setup, each DRL agent optimizes its DQN parameters,  $\theta$ , by the sampling data of minibatch size  $B$  from the replay buffer  $\mathcal{R}$  to optimize its policy using gradient descent algorithm. Further, agents in a network collaborate with other nodes to improve the generalization capacity, i.e., improve robustness in inference when deployed in different network operating scenarios.

1) *States*: Each Tx-Rx pair collects the local information and a few of the neighborhood information, which define the state of each node. The state of node  $i$  at the time slot  $t$  can be expressed as  $S_i^t = \{\mathbb{D}_i^t, B_i^t, I_c^t, S_i^t, SINR^t\}$ , where  $\mathbb{D}_i^t = \{d_{ij}^t | j = 1, 2, \dots, K\}$  is the set of distances to the neighboring receivers,  $B_i^t$  is the number of packets in the buffer,  $I_c^t$  is the interference caused by transmission from

node  $i$  to the neighboring nodes, and  $S_i^t$  is the spectrum sensed by the transmitter node  $i$ . Note that the SINR is measured at the receiver and is communicated to the transmitter node  $i$  through an ACK message. If the transmission from node  $i$  is not received, it will not receive an ACK message, and the value for  $S_i^t$  is set to be  $-1$ . Here, each agent maximizes its reward based on its state, which is a multi-agent reinforcement learning setup.

2) *Actions*: Let  $p_1, \dots, p_{n_p}$  be the power levels that agents can choose from and  $f_1, \dots, f_{n_f}$  be the number of frequencies that each agent can choose from. Then, the action of node  $i$  at time slot  $t$  is  $\mathcal{A}_i^t = \{(p^i, f^i) | (p^i, f^i) \in \mathcal{A}\}$ . Let  $\mathcal{P}$  be the set of all available power levels, and  $\mathcal{F}$  be the set of all available frequencies, then the action space is  $\mathcal{A} = \{(p_i, f_i) | p_i \in \mathcal{P} \text{ and } f_i \in \mathcal{F}\}$ .

3) *Rewards*: All the transmitters are required to transmit successfully while causing minimum interference to the neighboring node. Hence, it is desirable that each transmitter transmits successfully using the lowest power and choosing the available channels uniformly to minimize interference among nodes. To enforce all these constraints, we propose the following reward function for the given problem as

$$\begin{aligned} \mathcal{R} &= -C1 - C2, \quad \text{successful transmission,} \\ &= -C3, \quad \text{failure of the transmission,} \end{aligned} \quad (1)$$

### Algorithm 1 Generalized Training for DRL

---

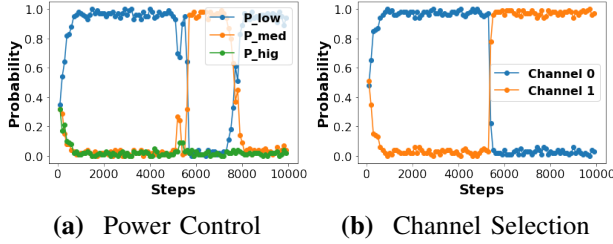
```

1: Initialize: DQN network parameters  $\theta_i^0 = \theta^0, \forall i \in [1 : K]$ , learning
   rate  $\alpha$ , Experience replay buffer  $\mathcal{R}$ , and mini-batch  $B$ 
2: for Each Network configuration do
3:   for each episode  $= 1, \dots, N$  do
4:     Observe an initial system state  $s_t$ 
5:     for each time step  $t = 0, \dots, T$  do
6:       Select action  $a_t$  at random with probability  $\epsilon$ 
7:       Otherwise select  $a_t = \arg \max_{a_t \in \mathcal{A}} q(s_t, a_t; \theta_t)$ 
8:       Execute action  $a_t$ , receive reward  $r_t$  and state  $s_{t+1}$ 
9:       Store the experience  $e_t = (s_t, a_t, r_t, s_{t+1})$ 
10:    end for
11:    Update DQN parameter  $\theta$  using gradient-descent update
12:    Model Aggregation:  $\bar{\theta} = \frac{1}{K} \sum_{i=1}^K \theta_i$ , and update  $\theta_i = \bar{\theta}, \forall i$ 
13:  end for
14:  Use Model Aggregation:  $\bar{\theta} = \frac{1}{K} \sum_{i=1}^K \theta_i$ , and update  $\theta_i = \bar{\theta}, \forall i$ .
15: end for
16: Return

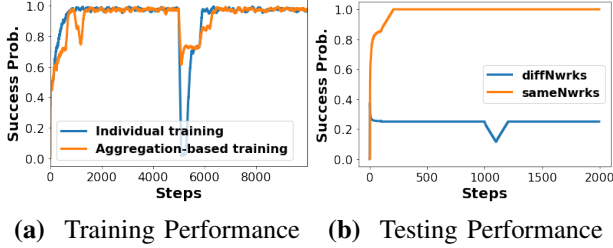
```

---

where  $C1$  refers to the cost of using power by an agent,  $C2$  refers to the cost of switching frequencies, and  $C3$  refers to the cost of failed transmission. The cost could be functions such as normalized SINR and normalized interference or constants. We choose  $C1$  as 0.05, 5, and 10 for using preset transmission power of  $P_{low}$  dBm,  $P_{med}$  dBm, and  $P_{high}$  dBm, respectively. In this work, we consider two frequencies available to each agent. To promote uniform distribution of choice of frequencies among agents, we create preference of channel uniformly among agents. We choose  $C2$  as 0.05 and 2 for transmitting at the same channel and at a different channel. Note that the reward for transmitting with  $P_{low}$  is higher than  $P_{med}$ , and the reward for transmitting with  $P_{med}$  is greater than  $P_{high}$  at any channel. Therefore, the choice of reward function makes



**Fig. 2:** Probabilities of Agent 1 as a function of steps (a) Power Control Probabilities (b) Channel Selection Probabilities.



**Fig. 3:** Performance of Node-level Aggregation-based approach vs. individual training approach in (a) training and (b) testing (when deployed in a different network), respectively.

agents prefer transmitting at a different channel to transmitting at higher power. Finally, we choose  $C3$  as 20. These values are hyper-parameters and are chosen after numerous experiments. Let  $r_t$  be the actual reward value obtained by an agent by selecting action  $a_t \in \mathcal{A}$  when the agent is in the state  $s_t \in \mathcal{S}$ .

4) *Training Strategies:* The training should focus on the generalization of performance over wireless networks of different sizes and operational scenarios. Next, we enlist steps in the proposed training strategy,

**States Uniformity:** For this consideration, we have fixed the size of the state vector. The size of input for DQN for all the networks of different sizes will be the same and hence is the first step towards the generalization. In the state vector, the dimension of  $\mathbb{D}$  depends on the size of the network. Instead, we fix the number  $K$  and choose the  $K$  most significant distances, i.e., the distances of the closest neighboring receivers where the interference caused by the agent is the most significant.

**Node-level Aggregation:** The RF environment that each agent experience is limited. If the agents are trained individually, the policy learned by each agent is limited to the data seen by each agent. When deployed, any of these agents will have worse performance as the training and testing scenario is going to be different. The problem is solved by node-level model aggregation. In the algorithm, we first train agents individually for a certain number of steps. We then aggregate the models and repeat the training. When the training converges, all the agents learn the policy corresponding to the RF scenario that all agents face in the network.

**Network-level Aggregation:** Further, we should emphasize that creating all possible RF scenarios within a network is difficult. First, the network grows very large, and second, creating individual RF scenarios across each node in a bigger network is difficult compared to a small network. We fur-

ther extend the ideas of model aggregation across nodes to aggregation over networks. Here, we design several smaller networks, where each network has specific network scenarios. Next, we train the agents of each network until convergence individually and save the DQN parameters. We aggregate all these DQN parameters from each network scenario and use the average DQN as the starting network in each node. Starting with the averaged model, we iterate over each network with model aggregation in between until convergence in all network scenarios. This approach ensures each agent learns a policy that can accommodate all wireless environments seen by agents in all networks.

**Introducing Randomness:** It is a common approach in DRL to train and test the learned agent in the same environment. However, wireless networks tend to be different from the training network during deployment. While network-level aggregation helps to address some of the variations in the wireless environments, introducing stochasticity in the environment itself, such as by using a random walk-based mobility model, improves the generalization capacity of the policy of DRL agents to small variations in wireless networks.

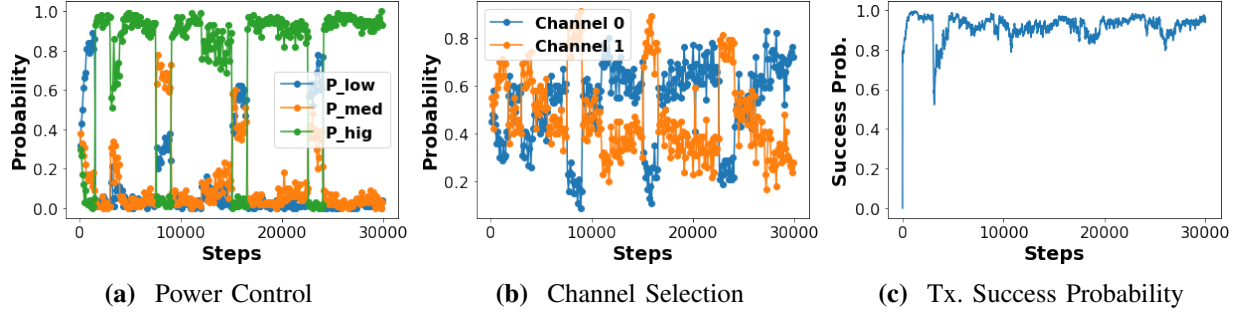
The generalized training algorithm is summarized in 1.

### III. EXPERIMENT RESULTS

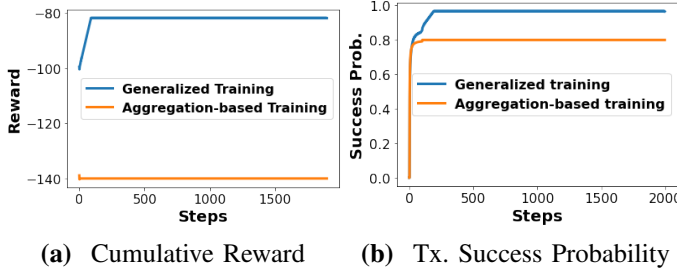
For the experiments, we use MR-iNet Gym [18] that uses ns3-gym [19]. This includes our custom DS-CDMA module for ns-3 to simulate a distributed LPI/D wireless network controlled by DRL running in an OpenAI-Gym. All training networks were developed using this framework. For each network trained and tested, we had jammers disrupting the transmission. The jammer was switched on at the same time when the Tx-Rx nodes were active. Further, for every training experiment, the jammer changed jammed frequency during the course of the training. Each agent uses the same fully connected network as the Q-network, which has hidden layers with 128 and 256 nodes, respectively. We use ReLU as the non-linear function in the hidden layers nodes and linear activation in the output layer nodes. Adam optimizer is used for the evaluation of stochastic gradient descent with a learning rate of 0.0001.

#### A. Simulation-based Performance

In the first experiment, we compare the training performance of the node-level aggregation-based training approach with individual training. Each Tx-Rx pair can communicate using three power levels in two different channels. Jammers disrupt communication between radios by changing their jamming frequency in the middle of the training. In Fig. 2, we plot Agent-1's power and channel selection probabilities, where we can see agents deciding to choose minimum power and avoid jamming frequency to get successful transmission. In Fig. 3, we can see that while the node-level aggregation-based approach learns almost as good as the individual training approach, the node-level aggregation-based approach has better performance when there is a change in network dynamics, i.e., jamming frequency. However, from Fig. 3, we can see that the test performance of the node-level aggregation-based approach



**Fig. 4:** Training performance of the proposed generalized approach as a function of steps: (a) Power Control Probabilities, (b) Channel Selection Probabilities, and (c) Transmission Success Probability with aggregation among networks, respectively.



**Fig. 5:** Performance comparison of agents when tested on an unseen network of size 10 using aggregation-based training and with proposed training approach: (a) Cumulative Reward; and (b) Network Transmission Success Probability.

Nwk	Parameters	PA	Rd	Gd
Nwk 1	PDR	89.88	45.33	75.71
	SE	3.17	0.48	3.01
Nwk 2	PDR	91.23	48.88	72.45
	SE	3.15	0.51	3.02
Nwk 3	PDR	90.63	47.69	76.13
	SE	3.15	0.48	3.01

**TABLE I:** PDR and SE of the proposed algorithms (PA), Gd, Rd in three different network scenarios.

are shown in Fig. 5. From the results, we can see from the inference that the proposed method outperforms the node-level aggregation-based approach in both cumulative rewards and network transmission success probability, demonstrating the proposed approach's *effectiveness in an unseen network*.

#### B. Over-the-air Hardware Experiments and Demonstration

In the final experiment, we conduct an OTA experiment where we deploy the inference engine obtained through the generalized training approach in the MR-iNet Gym environment. The DRL inference engine is now compressed using TensorRT, which not only reduces the model size of the inference engine but also reduces its processing time. In this experiment, we take two pairs of SDRs acting as two Tx-Rx pairs in an active jamming environment. The jammer changes the transmission frequency at random every 10 seconds to disrupt the communication between the Tx-Rx pairs.

In this setup, the Tx-Rx pairs are set to transmit within ISM band (S-Band). The transmitters are set to transmit at three different power levels. We transmit video packets over the SDRs, measure packet delivery rate (PDR) and spectral efficiency (SE), and compare the performance of the proposed algorithm to two different algorithms. The first algorithm selects each of the available frequencies and power options available to the transmitter at random. We refer to this approach as *Random* (Rd), which is similar to the baseline used in [14]. This serves as the lower baseline for the proposed algorithm. The second algorithm scans the available channel and transmits it on the channel with the least interference with the maximum power, which is referred to as *Greedy* (Gd). A single topology of the in-lab hardware testbed is shown in Fig. 6. The results in Table I shows that the PDR and SE of the proposed algorithm are consistently better than the other approaches in all networks. The proposed algorithm makes better power control and channel selection decisions than other algorithms



**Fig. 6:** In-lab Testbed

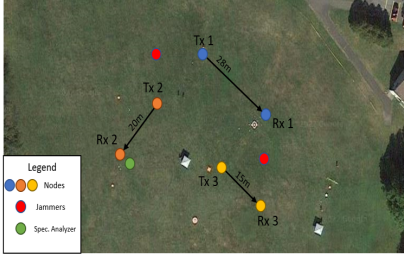
in an unseen network deteriorates, which motivates the need for the next part of our approach.

In the second experiment, we consider the proposed training approach, which also included network-level aggregation. In this setup, we consider five networks so that each agent learns a policy for different wireless scenarios. Agents are trained separately for each network, and the DQN model is saved. We then aggregate all these saved models and retrain agents with node-level aggregation for each network sequentially until we exhaust all networks. We repeat this training until convergence. Next, we plot the training results of this experiment in Fig. 4. We can see that the agent is changing power and frequencies with time depending on which network is being switched on. We should note that the network transmission success probability is approaching unity with increasing training steps. This suggests that each node is converging to a policy that can address all possible scenarios encountered by all agents in different networks. Next, we used this trained agent in an *unseen network* of size 10. The results of these tests



Jammer	Parameters	PA	Rd	Gd
CJ	PDR	97.81	54.71	88.22
	SE	3.47	0.85	3.15
RHJ	PDR	97.79	48.88	72.45
	SE	3.46	0.84	3.14
ChJ	PDR	97.89	47.69	76.13
	SE	3.47	0.84	3.14

**TABLE II:** PDR and SE of PA, Gd, and Rd in three different jamming scenarios: a) constant jammer (CJ), b) Random Hopping Jammer (RHJ), and c) Chirp Jammer (ChJ)



**Fig. 7:** Outdoor Test Scenario: Three pairs with two jammers such that most packets are transmitted successfully in all the radios. In addition, all the networks considered for OTA evaluation *were different from the networks used for training*, which shows that the inference engines obtained through the proposed training approach in the ns3-gym environment generalize over the unseen wireless networks.

Next, we compare the performance of the proposed algorithm in three different jamming environments; (i) Constant Jammer (CJ) is a Band-limited Gaussian jammer, which jams one of the channels that radios use for communication, (ii) Random Hopping Gaussian Jammer (RHJ), which randomly hops between the available channels. (iii) Chirp Jammer (ChJ), which sweeps across all frequencies in the signal bandwidth. We use a chirp period of  $\frac{1}{4}$  and a chirp rate of 4.8 MHz/sec. The Chirp jammer also hops between the available channels. The results of this comparison are listed in Table II. From the results, we can see that the DRL agent avoids the jammer in all three scenarios and hence has a very high PDR in all three jamming scenarios. Further, SE is also comparable in all three scenarios.

Finally, we deployed the DRL engine on three pairs of radios with two jammers in an outdoor real-world environment. The deployment setup is shown in Fig. 7 with the demonstration video of the real-time operation of the DRL inference engine available in [20] with a voice-over description.

#### IV. CONCLUSION

We investigated the problem of joint power control and frequency selection among multiple Tx-Rx pairs in a hostile jamming environment with the goal of hardware deployment. Striving for the real-world operating scenarios of the trained agent in unseen networks, we presented our generalized algorithmic approach to training agents for DRL problems in wireless networks. Unlike prior works that presented results on a specific network architecture, the proposed approach showed improved training and testing performance across seen and unseen networks for the problem of joint power control and frequency selection in both simulated and actual wireless

networks. Finally, we demonstrated the video streaming operation of the agent in an outdoor hostile jamming environment to prove the impact of the proposed solutions. We hope the proposed approach can accelerate the deployment of DRLs for real-world wireless networks.

#### REFERENCES

- [1] A. Jagannath, J. Jagannath, and T. Melodia, "Redefining wireless communication for 6g: Signal processing meets deep learning with deep unfolding," *IEEE Transactions on Artificial Intelligence*, vol. 2, 2021.
- [2] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li, "Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach," *IEEE access*, vol. 6, 2018.
- [3] Y. Yu, T. Wang, and S. C. Liew, "Deep-reinforcement learning multiple access for heterogeneous wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 6, pp. 1277–1290, 2019.
- [4] J. Jagannath, N. Polosky, A. Jagannath, F. Restuccia, and T. Melodia, "Neural networks for signal intelligence: Theory and practice," *Machine Learning for Future Wireless Communications*, pp. 243–264, 2020.
- [5] O. Naparstek and K. Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access," *IEEE transactions on wireless communications*, vol. 18, no. 1, pp. 310–323, 2018.
- [6] J. Jagannath, N. Polosky, A. Jagannath, F. Restuccia, and T. Melodia, "Machine learning for wireless communications in the internet of things: A comprehensive survey," *Ad Hoc Networks*, vol. 93, p. 101913, 2019.
- [7] M. Maaz, P. Mary, and M. H  lard, "Energy minimization in har-q relay-assisted networks with delay-limited users," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 8, pp. 6887–6898, 2017.
- [8] A. Anandkumar, N. Michael, A. K. Tang, and A. Swami, "Distributed algorithms for learning and cognitive medium access with logarithmic regret," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 4, pp. 731–745, 2011.
- [9] Y. S. Nasir and D. Guo, "Deep actor-critic learning for distributed power control in wireless mobile networks," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2020, pp. 398–402.
- [10] N. Modi, P. Mary, and C. Moy, "Transfer restless multi-armed bandit policy for energy-efficient heterogeneous cellular network," *EURASIP Journal on Advances in Signal Processing*, vol. 2019, no. 1, 2019.
- [11] K. I. Ahmed and E. Hossain, "A deep q-learning method for downlink power allocation in multi-cell networks," *arXiv preprint arXiv:1904.13032*, 2019.
- [12] Y. S. Nasir and D. Guo, "Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, 2019.
- [13] J. Tan, Y.-C. Liang, L. Zhang, and G. Feng, "Deep reinforcement learning for joint channel selection and power control in d2d networks," *IEEE Transactions on Wireless Communications*, vol. 20, no. 2, 2020.
- [14] Y. S. Nasir and D. Guo, "Deep reinforcement learning for joint spectrum and power allocation in cellular networks," in *2021 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2021, pp. 1–6.
- [15] J. Jagannath, K. Hamedani, C. Farquhar, K. Ramezani, and A. Jagannath, "Mr-inet gym: Framework for edge deployment of deep reinforcement learning on embedded software defined radio," in *Proc. of ACM Wrkp. on Wireless Security and Machine Learning*, 2022.
- [16] K. Zhang, Z. Yang, and T. Ba  ar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," *Handbook of reinforcement learning and control*, pp. 321–384, 2021.
- [17] J. Qi, Q. Zhou, L. Lei, and K. Zheng, "Federated reinforcement learning: Techniques, applications, and open challenges," *arXiv preprint arXiv:2108.11887*, 2021.
- [18] C. Farquhar, S. K  fle, K. Hamedani, A. Jagannath, and J. Jagannath, "Marconi-Rosenblatt Framework for Intelligent Networks (MR-iNet Gym): For Rapid Design and Implementation of Distributed Multi-agent Reinforcement Learning Solutions for Wireless Networks," *Computer Networks*, vol. 222, p. 109489, 2023.
- [19] P. Gawlowicz and A. Zubow, "Ns-3 meets openai gym: The playground for machine learning in networking research," in *Proc. of Intl ACM Conf. on Modeling, Analysis and Sim. of Wireless and Mobile Systems*, 2019.
- [20] "Outdoor Demonstration of Deep Reinforcement Learning for Jammer-Resilient Frequency and Power Allocation," <https://www.youtube.com/watch?v=2xk3YsmtY2g>, 2023.