Received 31 May 2024; accepted 14 June 2024. Date of publication 19 June 2024; date of current version 9 July 2024.

Digital Object Identifier 10.1109/OJCOMS.2024.3416902

Partially Cooperative RL for Hybrid Action CRNs With Imperfect CSI

SADIA KHAF[1] (Student Member, IEEE), GEORGES KADDOUM[1] (Senior Member, IEEE), AND JOAO VICTOR DE CARVALHO EVANGELISTA[2] (Member, IEEE)

[1]Department of Electrical Engineering, École de technologie supérieure, Montreal, QC H3C 1K3, Canada
[2]Baseband Systems, Ericsson Canada, Mississauga, ON L4W 5E3, Canada

CORRESPONDING AUTHOR: S. KHAF (e-mail: sadiakhaf@ieee.org)

This work was supported in part by the Fonds de recherche du Québec-Nature et technologies (FRQNT) Scholarship under Grant 305094, and in part by the Canada Research Chair Program Tier-II titled "Towards a Novel and Intelligent Framework for the Next Generations of IoT Networks."

ABSTRACT Cognitive radio networks (CRNs) mitigate spectrum scarcity by leveraging the holes in the licensed spectrum to enable Internet of Things (IoT) devices to opportunistically access the spectrum. However, IoT devices need to sense the spectrum before they can access it, which is an energy-intensive process and hinders the practical implementation of opportunistic spectrum access for energy-constrained IoT devices. In this context, reinforcement learning-based algorithms that encourage cooperation among IoT devices to eliminate the need for constant sensing are promising candidates for practical CRN implementation. As exciting as the application of reinforcement learning to CRNs is, benchmarking the performance of different algorithms is a huge challenge due to a lack of standardized comparison metrics, especially for hybrid action spaces that comprise both discrete and continuous actions. We propose a hybrid discrete-continuous space deep reinforcement learning algorithm that maximizes the energy efficiency of CRNs by optimizing sensing, cooperation, and transmission by IoT devices. We also analyze the algorithm's performance by setting the theoretical upper bound for throughput and find that it reaches 99.4% of the theoretical upper bound, while its discrete action-space version reaches 96% and other baseline algorithms range between 70% and 86%.

INDEX TERMS Smart spectrum sensing, cognitive radio Internet of Things (CR-IoT), channel utilization, energy efficiency, hybrid action space.

I. INTRODUCTION

THE EXPONENTIAL growth of the Internet of Things (IoT), which is expected to reach 41.6 billion devices and generate 79.4 zettabytes (ZB) of data in 2025, has resulted in increased demand for bandwidth. Cognitive radio networks (CRNs) have emerged as a promising solution to address the limited spectrum issue [1]. CRNs enable unlicensed IoT devices, which are termed secondary users (SUs), to opportunistically access the underutilized frequency bands that are licensed to primary users (PUs). However, the effective coexistence of heterogeneous devices in CRNs poses significant challenges for optimal spectrum utilization and network performance [2].

One primary condition for SUs to access the licensed spectrum is to avoid causing interference to PUs. Spectrum sensing, which is a mechanism that enables SUs to access the licensed spectrum without interfering with PU traffic, detects spectrum holes—free slots where PUs are not transmitting—but is energy-intensive [3]. The lack of energy-efficient sensing and transmission mechanisms is a dominant barrier to practical CRN implementation.

Reinforcement learning (RL)-based sensing and scheduling algorithms offer promise for addressing the energy constraints of CRNs [4], [5], [6]. Unlike heuristic or game-theoretic methods, RL-based algorithms adapt to PU traffic patterns and eliminate the need for constant sensing by learning in real time and facilitating SU cooperation to conserve energy [3]. Nevertheless, applying RL in CRNs and maintaining scalability becomes complicated when there are a variety of devices.

Hybrid action spaces, which integrate discrete and continuous actions, pose another challenge to RL-based CRNs. Sophisticated algorithms are required to manage the different actions and handle the heterogeneity effectively [7].

Furthermore, the limitations of low-powered IoT devices hinder the gathering of perfect channel state information (CSI), which complicates decision-making with imperfect CSI (ICSI) for single-agent RL [8], [9]. Single-agent sensing and decision making also suffer from spatial false alarms (SFA) due to active PUs outside the sensing region and require either adjusting the decision threshold based on aggregate interference [10] or reliance on spatio-temporal sensing [11], [12].

Cooperative RL presents promise for complex decision-making in multi-agent systems such as CRNs. However, current approaches often rely on centralized structures, which poses scalability challenges as networks grow in complexity and have a greater number of agents.

Developing decentralized cooperative RL algorithms is an open challenge to enable scalable and efficient CRN operations. These algorithms must facilitate cooperative learning without having overwhelming computational or memory requirements and address challenges like hybrid action spaces and ICSI. Successful development could optimize spectrum allocation, alleviate the curse of dimensionality, and enhance spectral and energy efficiency in CRNs.

This paper is organized as follows: The next section provides a comprehensive review of the related literature and highlights existing methodologies and approaches in the field. Section III presents the system model and problem formulation. Section IV presents the proposed algorithms in detail. Section V describes the experimental setup and discusses the simulation results obtained. Finally, the paper concludes with a summary of our findings and avenues for future research in Section VI.

II. RELATED WORKS

Research in the domain of CRNs has evolved significantly in recent years and aims to tackle the multifaceted challenges associated with spectrum access, interference mitigation, and network optimization. Spectrum sensing techniques play a pivotal role in identifying spectrum opportunities for SUs while avoiding causing interference to PUs [13]. Traditional methods like energy detection, matched filtering, and cyclostationary feature detection have been extensively investigated for spectrum hole identification. However, SFA and the energy-intensive nature of these approaches remains a pressing concern, particularly for resource-constrained IoT devices operating in CRNs [10], [14], [15], [16]. In response to this challenge, recent research endeavors have increasingly focused on harnessing machine learning approaches to enhance spectrum sensing efficiency, adaptability, and energy conservation in CRNs [4], [5], [6].

Efforts to mitigate the time-intensiveness of spectrum sensing have led to various approaches. Some strategies involve trade-offs between sensing accuracy and transmission opportunities, while others advocate periodic sensing to reduce overall sensing time [16], [17]. Passive listening schemes have also been explored to maximize transmission opportunities while minimizing the interference caused to PUs and SU energy consumption [5]. Despite these attempts, passive listening still consumes relatively a lot of energy [18], which poses challenges for implementation in low-powered IoT devices. Moreover, cooperative spectrum sensing (CSS) has been shown to exhibit more accurate sensing and lower collision rates compared to non-cooperative mechanisms. CSS is more resilient to malicious attacks like sensing data falsification (SDF) attacks, which further emphasizes the importance of minimizing SU energy consumption in CSS approaches.

Real spectrum data analysis indicates there are certain patterns in PU traffic that SUs can use to their advantage [19]. However, exploiting these patterns requires prior knowledge of the statistical characteristics of the spectrum data [20], which are seldom available to SUs [18]. RL algorithms are gaining traction [5], [6], [21] because they are able to adapt in real-time without the need for prior statistical knowledge of spectrum data [20]. However, their adaptability in CRNs when the CSI is delayed, noisy, falsified, or unavailable remains an open research area [22], [23].

The coexistence of discrete and continuous action spaces in CRNs poses challenges for conventional RL algorithms. Recent research has addressed this challenge by exploring novel algorithmic architectures that are capable of effectively handling continuous action spaces [24], [25], [26]. Furthermore, due to the limitations of low-powered IoT devices, gathering accurate CSI becomes challenging. Consequently, adapting RL-based decision-making with ICSI remains a crucial focus area for improving RL performance in CRNs [22], [23], [27].

To address these challenges, we propose the intelligent multi-user participatory algorithm for cooperative transmission (IMPACT). IMPACT leverages a deep RL-based approach in a decentralized cooperative framework to optimize the SUs' actions in order to maximize the CRN's energy efficiency. The algorithm, which employs a deep deterministic policy gradient (DDPG) structure with four deep neural networks, can handle hybrid action spaces and is benchmarked against exhaustive search and other algorithms. Participatory spectrum sensing and results sharing enable partial cooperation among agents to conserve energy and improve sensing accuracy while avoiding causing interference to PUs [4]. The proposed coalition scheme enhances the SUs' learning with ICSI and mitigates SFA and SDF attacks for granular optimization of the sensing, cooperation, and transmission processes.

A. NOVELTY AND CONTRIBUTION

IMPACT is novel in a number of ways and makes several contributions:

- *Handling Hybrid Action Space:* It efficiently manages both discrete and continuous action sets, which enables

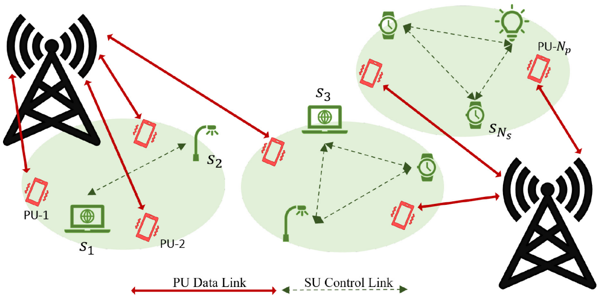

FIGURE 1. System Model.

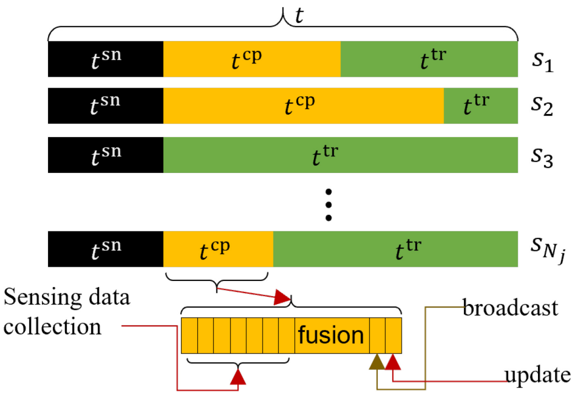

FIGURE 2. Time slot model.

effective decision-making in heterogeneous CRN environments.

- *Addressing ICSI:* It incorporates mechanisms to adapt RL-based decision-making strategies with ICSI, which mitigates the impact of delayed, inaccurate, falsified, or unavailable channel information in CRN environments.
- *Facilitating Decentralized Cooperation:* It introduces a decentralized cooperative learning framework that enables agents in CRNs to cooperate in select small groups and make informed decisions without relying on centralized structures, which helps to improve scalability and efficiency.
- *Enhanced Scalability:* It leverages decentralized cooperation to reduce the dependency on extensive shared resources and enable efficient operation in large-scale CRN scenarios.
- *Benchmarking Against Various Methods:* It is benchmarked against exhaustive search, bandit-based approaches, random selection, and a variant termed IMPACT-D, which discretizes the action space using two actor-critic deep Q-networks (DQNs). This comprehensive evaluation makes it possible to assess IMPACT's performance against that of various baseline methods that are commonly used in CRNs.

Therefore, the proposed algorithm has the potential to significantly improve the energy efficiency and performance of CRNs, and we hope that it will inspire further research on applying RL to CRNs.

III. SYSTEM MODEL

We consider a cellular system consisting of N_p PUs surrounded by N_s SUs, as shown in Figure 1. Each SU s_i can independently decide the portion of time slot t it spends in sensing and cooperating with other SUs to improve its sensing accuracy. The SUs have limited power and can sense only a limited number of channels in a given time slot. SUs communicate over common control channel for handshake, neighbour discovery, channel access negotiation, topology change, and collaboration [28], and opportunistically access the spectrum holes in licensed PU channels after sensing and/or collaboration. Moreover, both active sensing and passive listening to cooperate with others are energy-consuming processes, and the SUs' objective is to minimize the amount of energy spent on learning over time.

A. TIME SLOTS

Each time slot t can consist of a sensing t^{sn}, a cooperation t^{cp}, and a transmission t^{tr} period, as shown in Figure 2. SUs that do not participate in sensing or cooperation can conserve time and energy and use the entire time slot for transmission if they so choose; however, they then run a greater risk of causing harmful interference to PUs by transmitting when the PU channel is busy. Therefore, SUs' objective is to maximize their throughput and energy efficiency while staying within the PU interference thresholds.

Participation in sensing is a binary decision, with a minimum amount of time allocated for sensing. If an SU decides to sense a channel, it will receive the PU's signal power for that channel and infer the channel status by thresholding [29]. We consider both the perfect CSI scenario in which PU signal detection is assumed to be accurate and the ICSI scenario in which the PU's detected signal power is noisy, which can lead to the inaccurate inference of channel status $\{idle, busy\}$. Other forms of ICSI explored include delayed channel state and wrong channel state.

B. TRANSMISSION PATTERNS (TXPATTERNS)

TxPatterns represent different channel usage behaviors in a CRN. They offer channel utilization models to facilitate the evaluation of CRN protocols and algorithms. Exploring different TxPatterns is particularly important in CR, since SUs with simpler learning algorithms might be able to learn well from the environment with periodic channel utilization, but may suffer in more challenging scenarios. A variety of TxPatterns representing channel utilization by humans (bursty), sensor networks (periodic), medical equipment (frequently busy, random), streetlights (frequently idle), public transport systems (periodic, prolonged bursty) etc. is important in order to analyze the learning capabilities of SUs. Figure 3 shows a brief snapshot of a detected PU signal and its corresponding inferred channel status with different types of TxPatterns. The inferred channel

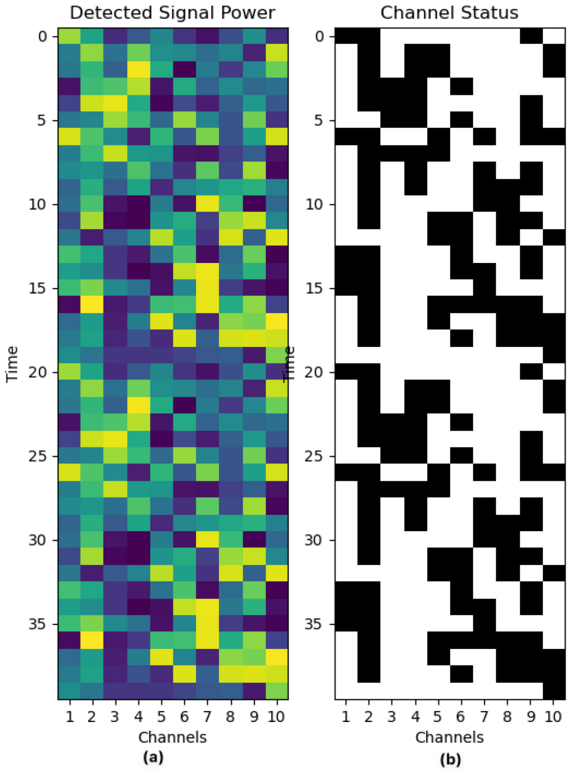

FIGURE 3. Detected PU Signal (a) and Inferred Channel Status (b) with corresponding channel types: bursty (Channels 6, 7, 8), frequently busy (Channels 2), frequently idle (Channels 1, 10), long bursty (Channels 3, 4), periodic (Channels 5), and random (Channels 9).

status shown is assuming perfect CSI. With ICSI, the SUs need to adapt their decisions to base them not only on the current time slot's CSI but also on their previous experience with the environment. The amalgamation of these patterns forms a comprehensive and realistic representation of channel usages, which is crucial for running simulations with the proposed algorithms and testing them on realistic scenarios.

C. TRAINING AND COOPERATION

An SU can decide not to sense if it is confident enough about the channel status from its previous interactions with the environment after a certain amount of training, or if it decides to be a passive listener and rely on other SUs to sense the channels and transmit their results in the cooperation time slot. The SUs that participate in cooperation transmit only their inferred channel statuses. Additionally, the SUs can choose to optimize their collaboration time by trading off accuracy for more transmission time. The cooperation time is divided into mini-slots for gathering sensing results from other SUs, applying a local data fusion rule to determine the channel status, and deciding whether or not to transmit. The majority rule is used as the local data fusion rule as it has been shown to be effective at improving sensing accuracy and mitigating SDF attacks [4].

D. SU TRANSMISSION

Each SU can access one or more of the M PU channels in the set of channels $\mathcal{C} = \{c_i, \ldots, c_M\}$ and transmit with transmit power p^{s_i} to achieve a desired throughput. Channel selection is a discrete decision variable, whereas transmit power is a continuous variable that the SU aims to optimize in order to maximize its throughput without causing harmful interference to PUs. Together, the decisions about sensing participation, cooperation time, channel selection, and transmit power optimization create a hybrid discrete-continuous action space.

E. THE ENVIRONMENT MODEL

The CR-IoT system is modeled as a partially observable Markov decision process (POMDP), as established in previous works [5], [6], [18], which enables the SUs to determine the state of a channel through sensing and cooperation. The observation o_t, action a_t, reward r_t tuple is represented as

$$\mathcal{X} = \{o_t, a_t, r_t\}.$$

At each time slot, each SU chooses an action $\mathbf{a}_t \in \mathcal{A}$, performs the action, and receives a reward $r_t \in R$.

1) OBSERVATION SPACE

The complete state space consists of the PU signal powers in each of the M channels at each time slot $\{t, p_{t,c_1}, p_{t,c_2}, \ldots, p_{t,c_M}\}$. The SUs do not have access to the complete state space. Instead, they observe a part of it $o_t = \{t, p_{t,c_j}, c_j \subseteq \mathcal{C}\}$ depending on which channels they decide to sense in a given time slot. If they decide not to sense or participate in cooperation, they may either use o_{t-n}, where $t-n$ is the last time slot in which they sensed the channel in question or approximate o_t from their previous interactions with the environment.

a) Perfect and Imperfect CSI In practical systems, SUs don't always have access to perfect CSI due to SU receiver accuracy limitations, environmental conditions, channel conditions, and the presence of malicious agents in the environment [4], and this can impact their decisions. It is, therefore, crucial to model a system that allows for ICSI and ensures that SUs adapt their decisions to avoid causing interference to PUs. The system model considered includes the following forms of ICSI:

- Delayed CSI: SUs receive CSI in a delayed manner, i.e., instead of receiving o_t, they receive o_{t-n}.
- Noisy CSI: SUs receive a noisy PU signal power, i.e., instead of receiving $o_t = \{t, p_{t,c_j}, c_j \subseteq \mathcal{C}\}$, they receive $o_t = \{t, p_{t,c_j} + p_{n_i} \sim \mathcal{N}(\mu_{\text{noise}}, \sigma^2_{\text{noise}}).,c_j \subseteq \mathcal{C}\}$, where p_{n_i} is the noise signal power with $\mu_{\text{noise}} = 0$, $\sigma^2_{\text{noise}} = 1$.
- Falsified CSI: Malicious agents in the system create sensing data falsification attacks during the cooperation period as outlined in [4].
- Unavailable CSI: SUs do not receive any discernible channel information despite participating in sensing,

TABLE 1. Table of notations.

$\mathbb{1}$	interference indicator	$\mathcal{F}$	update frequency	P_{od}	overall probability of detection
α	soft update parameter	f_s	sampling frequency	P_{of}	overall probability of false alarm
$\mathcal{A}$	action space	γ	discount factor	p	transmit power
$\mathcal{A}_d$	discrete action space	γ_{snr}	average received SNR	p_θ	power threshold
a	action	h	channel gain	ψ	cooperation ratio
$\mathbb{B}$	binary space	I	interference power	Q	estimated Q-value
$\mathcal{B}$	mini-batch sampled from memory	κ	channel sensing indicator	$\mathbb{R}$	continuous space
B	channel bandwidth	L	loss	r	reward
$\mathcal{C}$	set of channels	μ_{noise}	noise mean	ρ	throughput
c_j	channel j	$\mathbb{N}$	discrete space	$\mathcal{S}$	set of SUs
δ	number of discretization intervals	N_{ep}	number of episodes	σ^2_{noise}	noise variance
∇	gradient	N_o	Gaussian noise	s_i	SU i
$\mathcal{D}$	agent memory	N_p	number of PUs	T	horizon
d	terminal state done flag	N_s	number of SUs	t	time slot
E	energy consumption	o	observation	$\theta(.)$	$ActorNet$ output
$\phi(.)$	$CriticNet$ output	o'	next observation	ζ	channel allocation indicator

e.g., they cannot infer a channel status $\{idle, busy\}$ from the received signal and detection threshold [29].

2) ACTION SPACE

The action space $\mathcal{A}$ is defined as the Cartesian product of all possible combinations of actions an SU can choose from:

$$\mathcal{A} = \prod_{s_i=1}^{N_s}\left(\prod_{c_j=1}^{N_p}(\mathbb{B}_\kappa \times \mathbb{R}_\psi \times \mathbb{B}_\zeta \times \mathbb{R}_p)\right), \qquad (1)$$

where $\mathbb{B}$ represents a binary space and $\mathbb{R}$ represents a continuous real space. Note that the notation $(.)_t^{s_i,c_j}$ will be used to represent variable/parameter $(.)$ for SU s_i and channel c_j at time t. Omitting the superscript s_i indicates that $(.)$ is a vector containing values for all s_i, and omitting c_j indicates a constant value of $(.)$ for all c_j.

An action $a_t^{s_i,c_j} \in \mathcal{A}$ represents the action of SU s_i for channel c_j at time t and is defined as:

$$a_t^{s_i,c_j} = \left(\kappa_t^{s_i,c_j}, \psi_t^{s_i}, \zeta_t^{s_i,c_j}, p_t^{s_i}\right), \qquad (2)$$

where

- $\kappa_t^{s_i,c_j} \in \mathbb{B}_\kappa$ represents s_i's sensing decision for channel c_j at time t.
- $\psi_t^{s_i} \in \mathbb{R}_\psi$ represents the fraction of cooperation time t^{cp} that s_i decides to spend cooperating in time slot t. $\psi_t^{s_i}$ varies between $\psi_{\min}$ and $\psi_{\max}$.
- $\zeta_t^{s_i,c_j} \in \mathbb{B}_\zeta$ represents s_i's transmission decision for channel c_j at time t.
- $p_t^{s_i} \in \mathbb{R}_p$ represents s_i's transmit power at time t, which is restricted to the range $[0, p_{\max}^{s_i}]$, where $p_{\max}^{s_i}$ represents s_i's maximum transmit power.

Similarly, $a_t^{s_i}$ represents s_i's action for all channels at time t and is defined as:

$$a_t^{s_i} = \left(\kappa_t^{s_i,c_j}\ \forall c_j \in \mathcal{C}, \psi_t^{s_i}, \zeta_t^{s_i,c_j}\ \forall c_j \in \mathcal{C}, p_t^{s_i}\right). \qquad (3)$$

Note that $\mathbb{B}_\kappa$ and $\mathbb{B}_\zeta$ are binary spaces representing sensing and transmission decisions. And $\mathbb{R}_\psi$ and $\mathbb{R}_p$ are continuous spaces representing cooperation time and transmit power. Hence, action space $\mathcal{A}$ is a hybrid discrete-continuous space. A discretized version of $\mathcal{A}$, $\mathcal{A}_d$ will be referenced in parts of this paper where a discretization function f maps $\mathcal{A}$ to $\mathcal{A}_d$ using the discretization factors δ_t and δ_p such that:

- $f_\psi : \mathbb{R}_\psi \rightarrow \mathbb{N}^{\delta_\psi}$
- $f_p : \mathbb{R}_p \rightarrow \mathbb{N}^{\delta_p}$

The cooperative CRN system's goal is to maximize the overall system's energy efficiency while minimizing interference to PUs by choosing the optimal combination of aforementioned actions.

3) REWARD

Each SU receives a reward from the environment for its actions based on the success or failure of transmission, the amount of energy it consumes, and its achieved throughput. Since an SU must not cause harmful interference to the PU channel it is accessing, such interference is penalized in the reward function. We take all these factors into account and define the reward $r_t^{s_i}$ for SU s_i at time t as follows:

$$r_t^{s_i} = \frac{1}{E_t^{s_i}}\sum_{c_j=1}^{M}\mathbb{1}_{sp,t}^{s_i,c_j}\rho_t^{s_i,c_j}, \qquad (4)$$

where $\mathbb{1}_{sp,t}^{s_i,c_j}$ is the SU-to-PU collision indicator, $\rho_t^{s_i,c_j}$ is s_i's throughput from channel c_j at time t, and $E_t^{s_i}$ is s_i's energy consumption at time t. We define $E_t^{s_i}$ as:

$$E_t^{s_i} = \psi_t^{s_i}E_{cp}^{s_i} + \sum_{c_j=1}^{M}\kappa_t^{s_i,c_j}E_{sn}^{s_i} + \zeta_t^{s_i,c_j}p_t^{s_i}t^{s_i,tr}, \qquad (5)$$

where $t^{s_i,tr}$ is s_i's transmission time, and $E_{sn}^{s_i}$ and $E_{cp}^{s_i}$ are s_i's sensing, and cooperation energy consumption per slot, respectively. We define the collision indicator $\mathbb{1}^{s_i,c_j}_{sp,t}$ as:

$$\mathbb{1}^{s_i,c_j}_{sp,t} = \begin{cases} -1 & \text{if } \zeta_t^{s_i,c_j} p_t^{s_i} \geq P_{th} \ \& \ p_t^{c_j} \geq P_{th}^{ch} \\ 0 & \text{otherwise} \end{cases}, \tag{6}$$

where P_{th} is the power threshold to prevent interference with PU traffic, and P_{th}^{ch} is the power threshold for channel being busy.

SU s_i's throughput, which is given by Shannon's capacity theorem, is determined by:

$$\rho_t^{s_i}\left(a_t^{s_i}, o_t\right) = \sum_{c_j=1}^{M} B_{c_j} \log_2\left(1 + \frac{|h|^2 p_t^{s_i}}{N_0 B_{c_j} + \mathbb{1}^{c_j}_{ss,t} I}\right), \tag{7}$$

where $\mathbb{1}^{c_j}_{ss,t}$ is the SU-to-SU interference indicator, I is the interference power from other SUs, B_{c_j} is the channel bandwidth, N_o is the Gaussian noise power, h is the channel gain, and $p_t^{s_i}$ is s_i's transmission power. I follows Gaussian distribution with mean -80 dbm and variance $1e-9$, and h follows a Rayleigh distribution. We define the interference indicator $\mathbb{1}^{c_j}_{ss,t}$ as:

$$\mathbb{1}^{c_j}_{ss,t} = \sum_{s_i \in \mathcal{S}} \zeta_t^{s_i,c_j} \ \forall c_j \in \mathcal{C}. \tag{8}$$

As the sensing time increases, the probability of detection increases and the probability of a false alarm decreases [30], [31] as follows:

$$\begin{aligned} P_{od} &= 1 - (1 - P_d)^M \\ &= 1 - \left(1 - Q\left[\left(\frac{\epsilon}{\sigma_s^2} - \gamma_{snr} - 1\right)\sqrt{\frac{t^{sn} f_s}{2\gamma_{snr} + 1}}\right]\right)^M, \end{aligned} \tag{9}$$

where $Q[\,.\,]$ is the Gaussian Q-function, f_s is the sampling frequency, and ϵ is the energy detection threshold. Moreover, t^{sn} denotes time spent sensing, and γ_{snr} is the average signal-to-noise ratio received from the PU at the SU antenna. This relationship is used to model SUs' probability of detecting PU's presence when they participate in sensing under perfect CSI conditions. With delayed, noisy, falsified, or unavailable CSI, they receive o_t as described in Section III-E1a).

The overall objective of cooperative CRN is to maximize the SU energy efficiency while limiting the SU-to-PU and SU-to-SU interference and meeting SU throughput constraints, which can be formulated as follows:

$$\begin{aligned} \text{P1:} \quad & \max_{\kappa_t^{s_i,c_j}, \psi_t, \zeta_t^{s_i,c_j}, p_t} \sum_{s_i \in \mathcal{S}} \left(r_t^{s_i}\left(\kappa_t^{s_i,c_j}, \psi_t^{s_i}, \zeta_t^{s_i,c_j}, p_t^{s_i}\right)\right) \\ & C1 \ \sum_{c_j=1}^{M} |h|^2 p_t^{s_i} \leq I_{pu}, \ \ \forall s_i \in \mathcal{S}, \ \ \forall c_j \in \mathcal{C} \\ & C2 \ \sum_{s_i \in \mathcal{S}} \zeta_t^{s_i,c_j} \leq I_{su} \ \forall c_j \in \mathcal{C}, \\ & C3 \ \sum_{c_j=1}^{M} \rho_t^{s_i}\left(a_t^{s_i}, o_t\right) \geq \rho_{min}^{s_i}, \ \ \forall s_i \in \mathcal{S} \end{aligned} \tag{10}$$

where $C1$ constrains the SU-to-PU interference to remain under the PU interference threshold I_{pu}, $C2$ constrains the SU-to-SU interference to remain under the SU interference threshold I_{su}, and $C3$ constrains each SU's achieved throughput to be above the minimum required SU throughput $\rho_{min}^{s_i}$. Additionally, we assume that the SUs have a full buffer, i.e., they always have some data to transmit.

The problem in (10) is extremely hard to solve. It is noted that if we rewrite the argument in the summation of the objective function, which is given in (7), as:

$$\begin{aligned} \rho_t^{s_i} = \sum_{c_j=1}^{M} B_{c_j} \log_2 & \left(|h|^2 p_t^{s_i} + N_0 B_{c_j} + \mathbb{1}^{c_j}_{ss,t} I\right) \\ & - \log_2\left(N_0 B_{c_j} + \mathbb{1}^{c_j}_{ss,t} I\right), \end{aligned} \tag{11}$$

we see that (11) consists of the maximization of a double summation of differences of concave functions with integer arguments and integer constraints, and therefore a non-convex integer programming problem, which is non-trivial to solve with conventional optimization methods.

F. THE DIMENSIONALITY OF THE ACTION SPACE

The dimensionality of the action space in the system model considered is expansive, and accounts for multiple decision-making facets. With N_p PUs (channels) and N_s SUs, the SUs' individual decisions regarding channel sensing, cooperation time (discretized into δ_t bins), channel selection for transmission, and transmit power (discretized into δ_p bins) result in a substantial action space. The total number of possible action combinations is given by:

$$\begin{aligned} N_{act} &= \left(N_s \times 2^{N_p}\right) \times \left(\delta_t^{N_s}\right) \times \left(N_s \times 2^{N_p}\right) \times \left(\delta_p^{N_s}\right) \\ &= N_s^2 \times 2^{2N_p} \times \delta_t^{N_s} \times \delta_p^{N_s} \end{aligned}$$

For example, when $N_s = 2$, $N_p = 5$, $\delta_t = 10$, and $\delta_p = 10$, the resulting action space encompasses $40,960,000$ potential combinations, making methods like exhaustive search and Q-learning impractical and thus motivate using the DRL algorithms with a lower complexity.

IV. THE PROPOSED ALGORITHMS

The POMDP can maximize the SU's throughput and energy efficiency in many ways, one of which is exhaustive search, where each SU searches through all possible actions for each channel to maximize its throughput. However, the complexity of exhaustive search $O(N_s^2 2^{2N_p} \delta_t^{N_s} \delta_p^{N_s})$ makes it far from practical. We propose the intelligent multi-user participatory algorithm for cooperative transmission (IMPACT) that uses deep RL to maximize the SU's throughput while minimizing energy consumption. For the sake of comparison, we also propose a less complex algorithm called discretized IMPACT (IMPACT-D) that uses half as many neural networks as IMPACT and a discretized action space $\mathcal{A}_d$ that can still achieve performance higher than simpler algorithms such as greedy bandits.

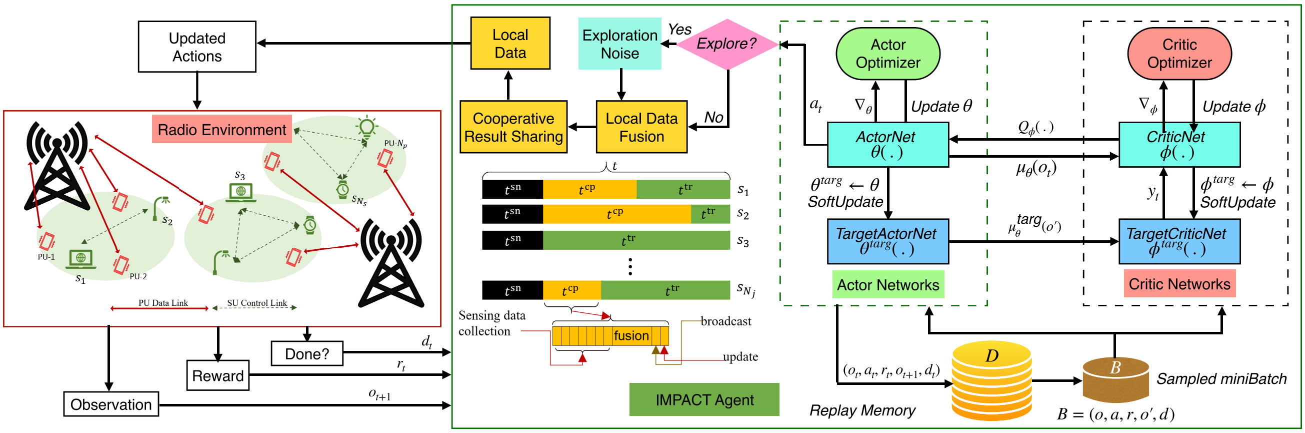

FIGURE 4. Flowchart of the proposed IMPACT approach.

A. IMPACT

IMPACT is shown as a flowchart in Figure 4, and its main steps are highlighted in Algorithm 1. It uses a DDPG architecture with four identical neural networks, called *ActorNet* $\theta(.)$, *CriticNet* $\phi(.)$, *TargetActorNet* $\theta^{\text{targ}}(.)$ and *TargetCriticNet*$\phi^{\text{targ}}(.)$. The algorithm's advantage lies in its ability to use *ActorNet* to learn a deterministic policy that directly maps observations to continuous actions. During training, the network learns to approximate the optimal deterministic policy by adjusting its weights through gradient descent, so the network's aim is to keep adjusting the mapping of input o_t to output a_t through training until it learns the optimal mapping.

At the beginning of each time slot t all SUs choose which channels, if any, they are going to sense. Afterwards, each cooperative SU shares its channel inferences with the other SUs, and each passive listener gets channel inferences from the other cooperative SUs. During the cooperation slots, the SUs perform local data fusion on the channel inferences to improve their sensing accuracy and mitigate SDF attacks. The SUs choose their transmission power and which channels they want to use for transmission based on the coalitions' data fusion rules [4]. All SUs then perform the actions by transmitting with the selected $p_t^{s_i}$ over the selected channels, and achieve a certain throughput $\rho_t^{s_i}(o_t, a_t)$, which, together with any interference they create, the constraints mentioned in (10), and feedback they receive from the environment, is used to determine their reward. They also receive the next observation o' and terminal state done flag d from the environment for the channels they chose to sense. This information (o_t, a_t, r_t, o', d) is stored in each agent's memory $\mathcal{D}$ for future reference.

CriticNet is the network responsible for approximating the action value and evaluates the quality of *ActorNet*. *ActorNet* uses o_t as its input and a_t as the output. *CriticNet* uses (o_t, a_t) as its input and the estimated action value $Q_\phi(o_t, a_t)$ as its output, which is used to measure the quality of training. *CriticNet*'s objective is to maximize this value in order to maximize *ActorNet*'s quality.

The inputs and outputs to networks are normalized. The *ActorNet* uses an input layer with 3 neurons, 2 hidden layers with 64 neurons and ReLU activation, and an output layer with 3 neurons and *tanh* activation. The *CriticNet* uses input layer with 6 neurons, 2 hidden layers with 64 neurons and ReLU activation, and output layer with 1 neuron and linear activation. *TargetActorNet* and *TargetCriticNet* are copies of *ActorNet* and *CriticNet*, respectively, that have the same input and output structure but are used in a soft update manner to stabilize the latter.

During training, a random batch of transitions $\mathcal{B} = (o, a, r, o', d)$ is sampled from memory $\mathcal{D}$ and used to compute the target y as

$$y\big(r, o', d\big) = r + \gamma(1-d)Q_\phi^{\text{targ}}\Big(o', \mu_\theta^{\text{targ}}\big(o'\big)\Big), \qquad (12)$$

where $Q_\phi^{\text{targ}}(.)$ represents the output of *TargetCriticNet* and $\mu_\theta^{\text{targ}}(.)$ represents the output of *TargetActorNet*. Gradient descent is performed to update *CriticNet* as:

$$\nabla_\phi \frac{1}{\mathcal{B}} \sum_{\mathcal{B}} \big(Q_\phi(o, a) - y\big(r, o', d\big)\big)^2, \qquad (13)$$

where $\mathcal{B}$ is a mini-batch of samples drawn from memory $\mathcal{D}$ without replacement. The mini-batch size is 64, which is selected through hyper-parameter turning. The actor network is updated using gradient ascent as:

$$\nabla_\theta \frac{1}{\mathcal{B}} \sum_{\mathcal{B}} Q_\phi(o, \mu_\theta(o)), \qquad (14)$$

where $\mu_\theta(.)$ represents the output of *ActorNet*. *TargetActorNet*'s and *TargetCriticNet*'s weights are updated at each step in a soft update manner as:

$$\phi^{\text{targ}} \leftarrow \alpha\phi^{\text{targ}} + (1-\alpha)\phi \qquad (15)$$

$$\theta^{\text{targ}} \leftarrow \alpha\theta^{\text{targ}} + (1-\alpha)\theta. \qquad (16)$$

Algorithm 1: Algorithm: IMPACT

```
Input: ActorNet $\theta$ and CriticNet $\phi$ parameters, $\alpha$; Initialize replay memory $\mathcal{D}$;
Initialize TargetActorNet $\theta^{targ}$ and TargetCriticNet $\phi^{targ}$ as: $\theta^{targ} \leftarrow \theta$,   $\phi^{targ} \leftarrow \phi$;
for $episode = 1$ to $N_{ep}$ do
    Reset Environment, observe $o_t$;
    for $t = 1$ to $T$ do
        Choose $a_t = \mu_\theta(o_t) + \psi \sim \mathcal{N} + p \sim \mathcal{N}$ ;
        Cooperative agents transmit inferred channel status ;
        Every participatory agent performs local data fusion to update $a_t$ using majority rule on inferred channel status;
        Execute action $a_t$, obtain reward $r_t$, the next observation $o'$, and terminal flag done $d$ ;
        Store tuple $(o_t, a_t, r_t, o', d)$ in replay memory $\mathcal{D}$;
        Sample mini-batch of transitions $\mathcal{B} = (o, a, r, o', d)$ from $\mathcal{D}$;
        Compute $y = r + \gamma(1-d)Q_\phi^{\text{targ}}(o', \mu_\theta^{\text{targ}}(o'))$ ;
        Update CriticNet by one step of gradient descent as
            $$\nabla_\phi \frac{1}{\mathcal{B}} \sum_{\mathcal{B}} \left(Q_\phi(o,a) - y(r,o',d)\right)^2.$$
        Update the ActorNet by one step of gradient ascent as
            $$\nabla_\theta \frac{1}{\mathcal{B}} \sum_{\mathcal{B}} Q_\phi(o, \mu_\theta(o)),$$
        Update target networks using a soft update as
            $\phi^{\text{targ}} \leftarrow \alpha\phi^{\text{targ}} + (1-\alpha)\phi$
            $\theta^{\text{targ}} \leftarrow \alpha\theta^{\text{targ}} + (1-\alpha)\theta$
```

To facilitate the exploration of different actions in a continuous action space, we add uncorrelated Gaussian noise to the continuous parts of the actions as:

$$\psi_t^{s_i} \leftarrow \psi_t^{s_i} + \psi_n \sim \mathcal{N}\left(\mu_{\text{noise}}, \sigma_{\text{noise}}^2\right) \tag{17}$$

$$p_t^{s_i} \leftarrow p_t^{s_i} + p_n \sim \mathcal{N}\left(\mu_{\text{noise}}, \sigma_{\text{noise}}^2\right). \tag{18}$$

The model is labeled as the current best model in the beginning of training after at least $\mathcal{W}_n$ episodes, and is updated periodically as the training progresses whenever the average throughput and energy efficiency of the model over $\mathcal{W}_n$ recent episodes exceeds the current best performance. This approach prevents unnecessary storage. The complexity of a training pass of IMPACT is $O(\mathcal{B}.(2f_\theta + 2f_\phi))$, where f_θ and f_ϕ represent the number of neurons in *ActorNet* and *TargetNet*, respectively. Complexity of decision making by a trained network is $O(f_\theta)$.

B. IMPACT-D

IMPACT is a powerful algorithm that can deal with hybrid action spaces. To analyze its performance in comparison to that of a closely matched discretized algorithm we propose IMPACT-D, which uses the discretized action space $\mathcal{A}_d$ instead of $\mathcal{A}$ and tries to find the optimal actions while utilizing half as many networks as IMPACT. IMPACT-D uses the DQN architecture and is shown as a flowchart in Figure 5. It utilizes two identical neural networks, called *PolicyNet* and *TargetNet*, to approximate the Q-values of the SU's actions. *PolicyNet* serves to estimate the Q-value $Q(o_t, a_t; \theta)$ for all (o_t, a_t) pairs, where θ denotes *PolicyNet*'s parameters. *TargetNet* is identical to *PolicyNet* except for the fact that it is updated only every $\mathcal{F}$ steps. Using *PolicyNet* and *TargetNet* prevents network oscillations and instability so that the network is not chasing a constantly moving target. Unlike IMPACT, IMPACT-D cannot work with continuous or hybrid action spaces and does not strive to learn a deterministic policy. Instead, its [IMPACT-D's] only objective is to find the (discrete) actions that maximize the estimated action values.

At each time step t each SU observes o_t and chooses its action a_t using an epsilon-greedy policy. The value of epsilon decreases exponentially as training progresses. After the SUs interact with the environment, they receive a reward r_t and the tuple (o_t, a_t, r_t, o', d) is stored in memory $\mathcal{D}$. Each episode ends in a terminal state, called terminal o_t, after a fixed number of steps. After the SUs reach the terminal state, the environment is reset and a new episode begins. A mini-batch is sampled from the memory in each episode, and the Q-values are calculated as:

$$y_j = \begin{cases} r_j & \text{for terminal } o_t \\ r_j + \gamma \max_{a'} \hat{Q}(o', a'; \theta) & \text{for non-terminal } o_t \end{cases} \tag{19}$$

Afterward, *PolicyNet* is updated using the following loss function:

$$L(x, y) = \frac{1}{n} \sum_i z_i, \tag{20}$$

$$z_i = \begin{cases} 0.5(x_i - y_i)^2 & \text{if} |x_i - y_i| < 1 \\ |x_i - y_i| - 0.5 & \text{otherwise,} \end{cases} \tag{21}$$

where (21) defines *SmoothL1Loss*, which has been proven to be less sensitive to outliers than *MSELoss*.

Finally, gradient descent is used to minimize the loss function and update *PolicyNet*'s weights, and training continues until a certain level of accuracy is achieved or the maximum number of episodes has passed.

The SUs explore the discrete action space using an epsilon-greedy policy during training. They also periodically store the trained model and update it whenever the model performance over the recent $\mathcal{W}_n$ episodes exceeds the previous best performance. IMPACT-D's main steps are highlighted in Algorithm 2. IMPACT-D's training and testing complexity is $O(\mathcal{B}.(f_\theta + f_\phi))$ and $O(f_\theta)$, respectively.

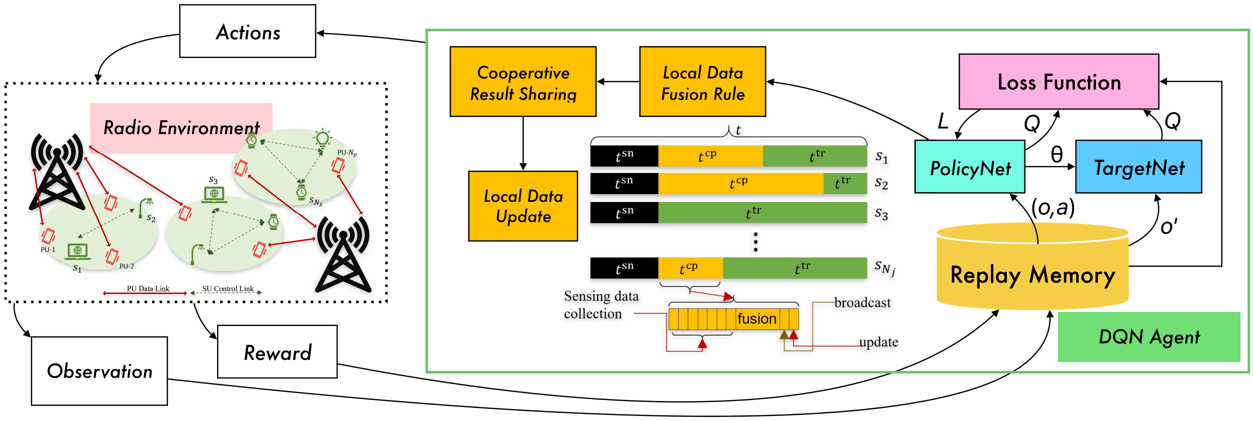

FIGURE 5. Flowchart of the proposed IMPACT-D approach.

```
Algorithm 2: IMPACT-D
Input: $\delta_\psi, \delta_p, \alpha, \gamma, N_{ep}, \mathcal{F}$;
Create $\mathcal{A}_d$ from $\mathcal{A}$ such that $f{:}\mathcal{A} \rightarrow \mathcal{A}_d$ as:
    $f_\psi{:}\mathbb{R}_\psi \rightarrow \mathbb{N}^{\delta_\psi}$, and $f_p{:}\mathbb{R}_p \rightarrow \mathbb{N}^{\delta_p}$ ;
Initialize replay memory $\mathcal{D}$;
Initialize PolicyNet with random weights $\theta$ ;
Initialize TargetNet with weights $\theta^{\text{targ}} = \theta$ ;
for $episode = 1$ to $N_{ep}$ do
    Reset Environment;
    for $t = 1$ to $T$ do
        Choose $a_t$ with an epsilon-greedy policy as
            $a_t = argmaxPolicyNet(o_t, a_t; \theta)$;
        Cooperative agents transmit their inferences,
            passive listeners get the inferred results from
            other;
        Perform local data fusion, decide on $\zeta, p$ ;
        Execute action $a_t$, obtain reward $r_t$ and the
            next observation $o'$ ;
        Store tuple $(o_t, a_t, r_t, o', d)$ in replay memory
            $\mathcal{D}$;
        Sample mini-batch of transitions $(o, a, r, o', d)$
            from memory $\mathcal{D}$;
        Set $y_j = r_j + \gamma \max_{a'} \hat{Q}(o_{j+1}, a'; \theta^{\text{targ}})$ ;
        Update the PolicyNet by minimizing the loss
            according to equations 20 and 21;
    Every $\mathcal{F}$ perform a hard update as $\theta^{\text{targ}} \leftarrow \theta$
```

V. PERFORMANCE EVALUATION

In this section, we present simulation results and compare the two proposed algorithms' performance with that of random agents (RAs), greedy bandit agents, and exhaustive search agents (ESAs). The greedy bandit algorithm is an iterative algorithm that solves multi-agent CRNs as multi-armed bandit problems in which the agents always select the arm with the highest estimated reward at each step.

Each SU's maximum sensing capacity is assumed to be 10, which is a realistic assumption for most low-powered IoT devices. We evaluate the algorithms with $N_p = [2, 3, 5, 7, 10]$ and $N_s = [2, 3, 5, 7, 10]$. The simulations were carried out for $1,000$ episodes, with $T = 20$ steps per episode. We use $E_{sn} = 0.1$ mJ $- 5$ mJ and $E_{cp} = 0.05$ mJ $- 1.5$ mJ for different SUs based on [25], [32], [33].

Figure 6 shows the training performance of the various agents considered with different types of CSI. IMPACT outperforms all the other agents with perfect CSI and achieves a near-optimal reward when compared with that of the ESAs, as shown. IMPACT-D doesn't perform optimally due to the discretization of the action space limiting the actions the agents can take. A comparison with different values of δ_ψ and δ_p is not provided for the sake of brevity, but IMPACT-D's performance degrades as $\mathcal{A}_d$ becomes smaller. Both IMPACT and IMPACT-D learn to adapt their actions with noisy CSI and closely match the performance achieved under perfect CSI, although IMPACT is more affected by the ICSI than IMPACT-D is. IMPACT using more neural networks and its continuous parts being more sensitive to noisy CSI. All the agents' training performance is degraded more under delayed and falsified CSI conditions, with falsified having a greater impact. This is expected, and the impact of said data falsification on cooperative agents and the proposed solutions can be found in our previous work [4]. When CSI is unavailable, IMPACT-D seems to outperform IMPACT in training at least. We suspect this is also due to IMPACT-D having fewer neural networks involved and its discrete-only decision-making being less complex, but further research that is beyond the scope of this paper is needed to draw a more definitive conclusion. The greedy bandit agents' performance is severely limited due to the size of the action space and the fact that the bandits easily get stuck in the local optimums. As expected, the RAs perform the worst and are not affected by any variation in CSI.

Figure 7 shows the training and testing throughput of various agents under different traffic patterns explored in this work. Agents that are trained on a periodic traffic pattern outperform the other agents during training but generalize relatively poorly when they are tested on all types of traffic patterns, whereas the agents that are trained in a hybrid

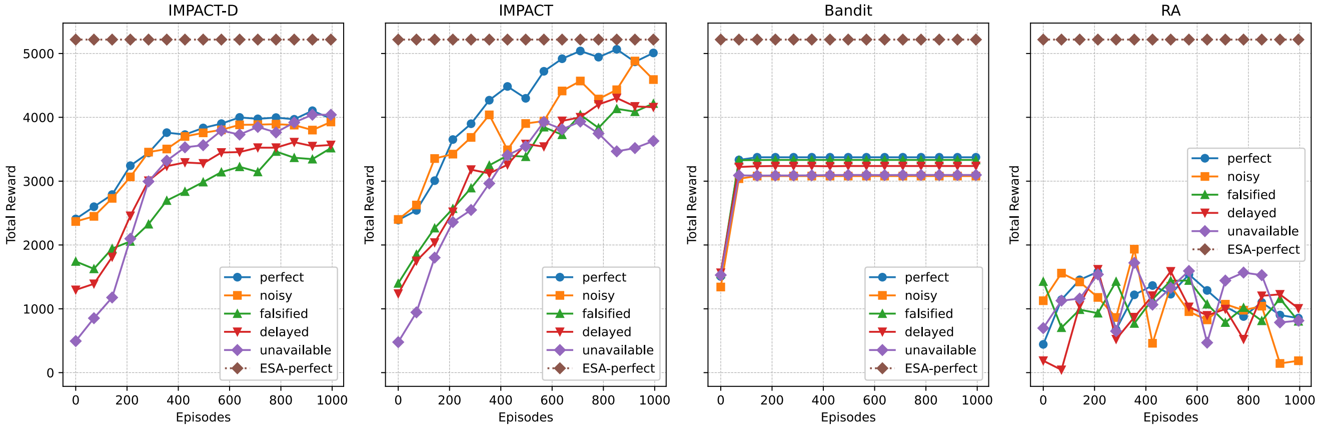

FIGURE 6. Training performance of different agents under different CSI conditions.

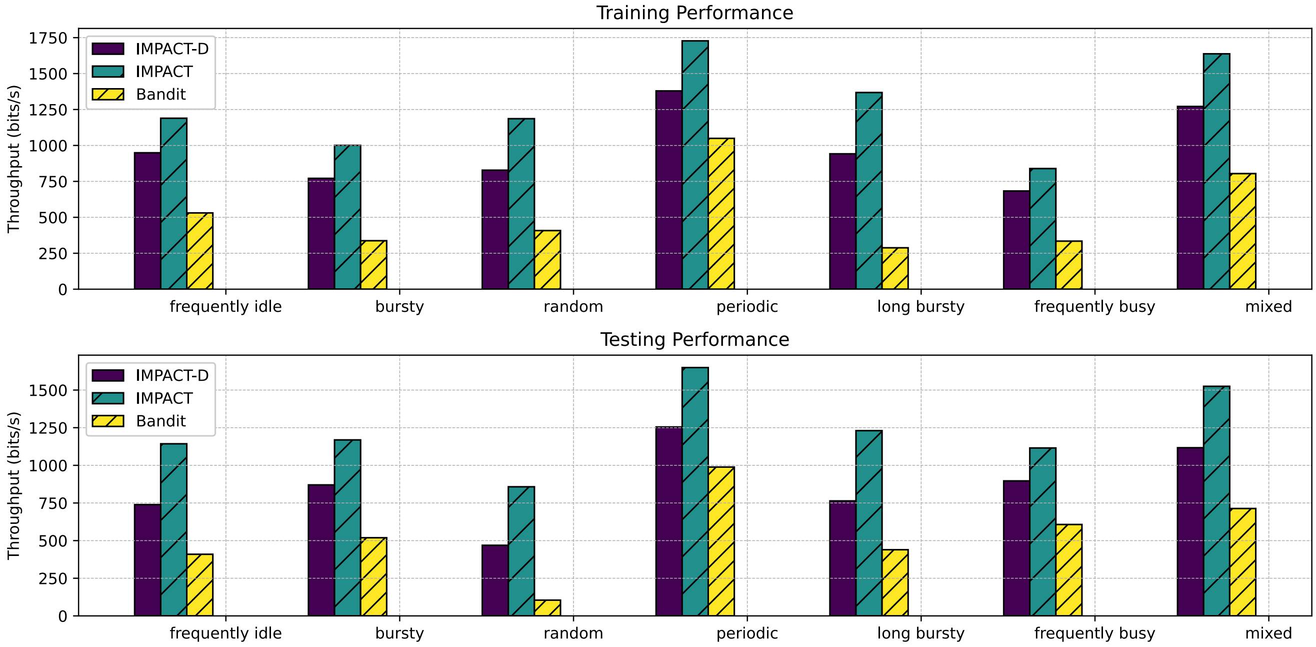

FIGURE 7. Average throughput (bits/s) under different traffic patterns explored for training and testing.

environment comprising multiple traffic patterns generalize far better than the other agents. As expected, random traffic patterns do not provide any meaningful information for agents to learn. The simpler greedy bandit agents perform relatively well in a periodic traffic pattern but lose to others by a significant margin for all the other traffic types.

Figure 8 shows the energy efficiency of SUs as the number of SUs per coalition varies from 2 to 10. All the SUs in a coalition sense and transmit over the same subset of channels. The number of channels is set at $N_p = [2, 3, 5, 7, 10]$ from one subplot to the next going from left to right. When $N_s = 2$ there is no significant difference in average agent's energy efficiency as the number of SUs participating in sensing and cooperation increases. When $N_s = [3, 5, 7]$, both IMPACT and IMPACT-D benefit from having more SUs involved in sensing and cooperation, and receive overall higher efficiency. IMPACT-D starts running into scalability challenges at $N_s = 10$, where effectively exploring the action space becomes a challenge. IMPACT, on the other hand, remains scalable and continues to benefit from having more SUs participating in sensing and cooperation as the number of channels grows. The non-cooperative algorithm (NoCoop) has a considerably lower energy efficiency due to frequent sensing, SFA and SDF attacks, and demonstrates the benefits of cooperation.

Figure 9 shows the distribution of the average throughput and energy consumption of the approaches considered for [R1C4]$N_p = N_s = 10$. IMPACT achieves near-optimal throughput while keeping its energy consumption low. IMPACT-D achieves less throughput and consumes slightly less energy than IMPACT. IMPACT's energy consumption is expected to be lower in smaller environments where N_p

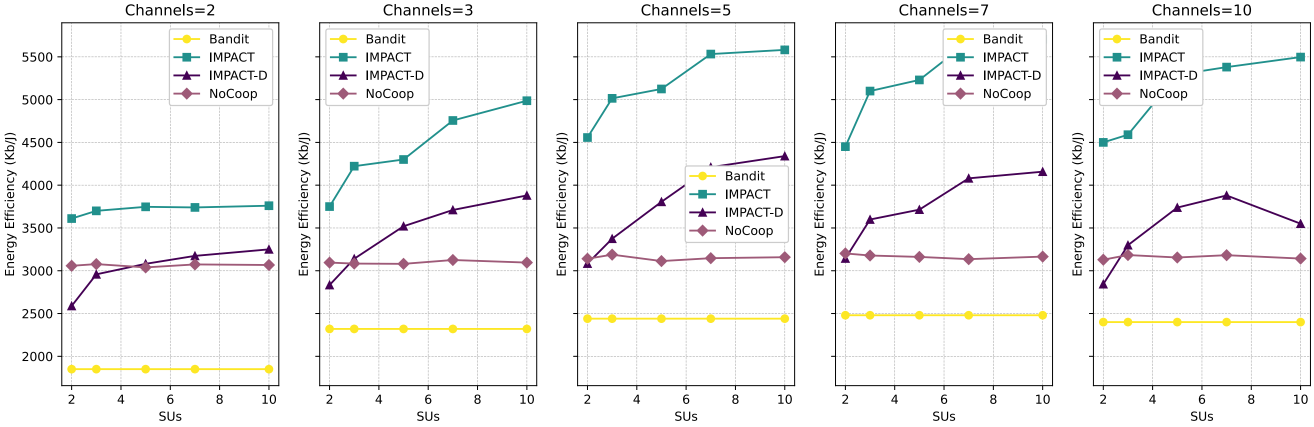

FIGURE 8. Average energy efficiency (Kb/J) for different number of SUs and channels for different agents.

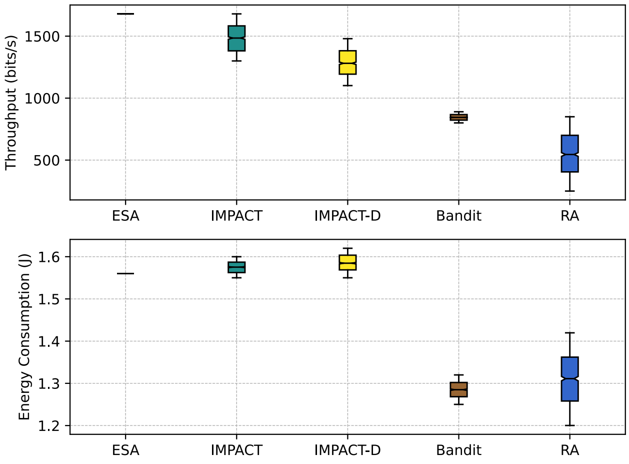

FIGURE 9. Average throughput (bit/s) and energy consumption (J) for different algorithms.

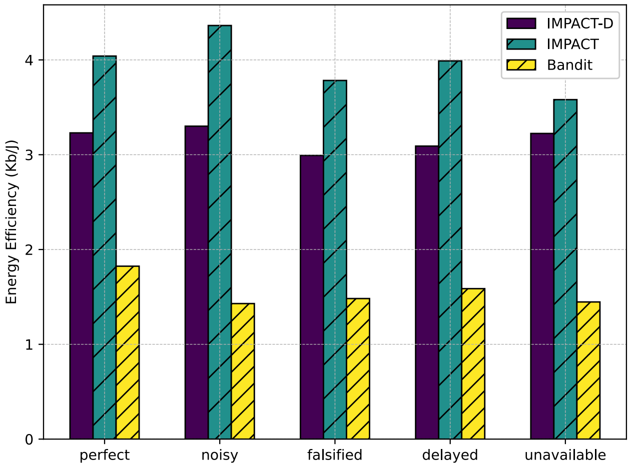

FIGURE 10. Energy efficiency of different agents various CSI conditions.

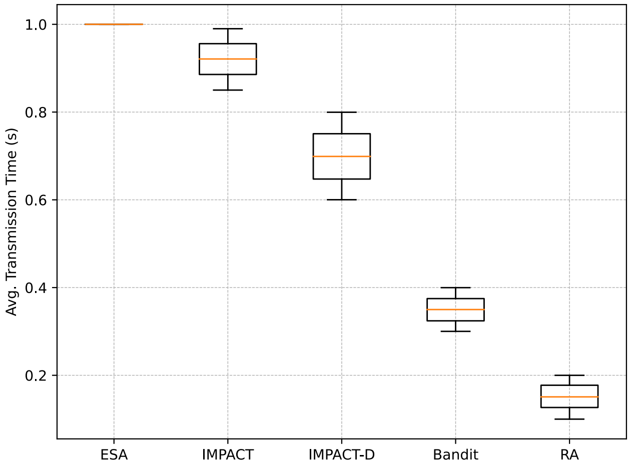

FIGURE 11. Average transmission time (s) for different agents.

and N_s are relatively small. The greedy bandit agents did not consume a lot of energy or achieve a high throughput and were stuck in local optimal solutions, and the RAs perform the worst of all. For the ESAs, it is important to note that the amount of energy consumed when searching through all the actions is not taken into account here; the plot shows only the energy consumed for the optimal actions. It can be observed from this figure that IMPACT achieves near-optimal performance on both fronts.

Figure 10 generalizes the energy efficiency of the various agents considered under different CSI conditions. The agents that are trained under noisy CSI conditions outperform all the other agents when tested in all types of CSI conditions. The CSI conditions have a greater impact on IMPACT than IMPACT-D, whereas the greedy bandit agents generalize poorly due to the simplistic nature of their algorithm. IMPACT's heightened sensitivity to training conditions is due to its underlying policy gradient methods' inherent sensitivity to environmental condition, and serves as an advantage when it is trained under noisy CSI conditions. As expected, training under falsified CSI conditions results in worse generalization than any other type of CSI conditions due to the highly irregular and non-deterministic nature of this falsification.

Figure 11 shows different agents' distributions of average transmission time. It is important to note that IMPACT and IMPACT-D cannot and should not use the full time slot for transmission because they are always relying on cooperative mechanisms to achieve high throughput without always having to sense. When that is taken into account, both IMPACT and IMPACT-D manage to utilize most of each time slot for transmission and use only a small portion of time overall for sensing and cooperation.

VI. CONCLUSION

In this paper, we proposed two cooperative spectrum sensing and transmission algorithms, IMPACT and IMPACT-D, that maximize the energy efficiency of CRNs. The former offers the ability to incorporate hybrid action spaces comprising discrete and continuous actions, achieves higher performance in challenging traffic conditions, and generalizes better overall under different CSI conditions, while the latter offers a low-complexity alternative for relatively simpler environments. Both algorithms outperform both the greedy bandit agents and the random agents, and approach the performance upper bound of the exhaustive search agents. The fact that these algorithms can achieve the same throughput and energy efficiency as exhaustive search is a testament to their performance, since the computational complexity of exhaustive search makes it impractical for implementation on low-powered IoT devices. These hybrid and scalable algorithms are monumental for the rapid growth of distributed heterogeneous IoT networks. Future works may consider more granular optimization of the cooperation process by optimizing participation based on IoT device characteristics. Other possible directions include networks comprising both terrestrial and non-terrestrial components, on-the-fly network reconfiguration, along with exploring more algorithms, e.g., proximal policy optimization (PPO), advantage actor critic (A2C), trust region policy optimization (TRPO), and so forth.

REFERENCES

[1] A. A. Khan, M. H. Rehmani, and A. Rachedi, "When cognitive radio meets the Internet of Things?" in *Proc. Int. Wireless Commun. Mobile Comput. Conf.*, pp. 469–474, 2016.

[2] W. Zhang, Y. Yang, and C. K. Yeo, "Cluster-based cooperative spectrum sensing assignment strategy for heterogeneous cognitive radio network," *IEEE Trans. Veh. Technol.*, vol. 64, no. 6, pp. 2637–2647, Jun. 2015.

[3] H. Zhang, N. Yang, W. Huangfu, K. Long, and V. C. Leung, "Power control based on deep reinforcement learning for spectrum sharing," *IEEE Trans. Wireless Commun.*, vol. 19, no. 6, pp. 4209–4219, Jun. 2020.

[4] S. Khaf, M. T. Alkhodary, and G. Kaddoum, "Partially cooperative scalable spectrum sensing in cognitive radio networks under SDF attacks," *IEEE Internet Things J.*, vol. 9, no. 11, pp. 8901–8912, Jun. 2022.

[5] R. Sarikhani and F. Keynia, "Cooperative spectrum sensing meets machine learning: Deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 24, no. 7, pp. 1459–1462, Jul. 2020.

[6] X. Liu, C. Sun, M. Zhou, C. Wu, B. Peng, and P. Li, "Reinforcement learning-based multislot double-threshold spectrum sensing with Bayesian fusion for industrial big spectrum data," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3391–3400, May 2021.

[7] X. Li, L. Dong, L. Xue, and C. Sun, "Hybrid reinforcement learning for optimal control of non-linear switching system," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 11, pp. 9161–9170, Nov. 2023.

[8] A. Kaur, K. Kumar, A. Prakash, and R. Tripathi, "Imperfect CSI-based resource management in cognitive IoT networks: A deep recurrent reinforcement learning framework," *IEEE Trans. Cogn. Commun. Netw.*, vol. 9, no. 5, pp. 1271–1281, Oct. 2023.

[9] A. Kaur and K. Kumar, "Imperfect CSI based intelligent dynamic spectrum management using cooperative reinforcement learning framework in cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 21, no. 5, pp. 1672–1683, May 2022.

[10] A. Furtado, L. Irio, R. Oliveira, L. Bernardo, and R. Dinis, "Spectrum sensing performance in cognitive radio networks with multiple primary users," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1564–1574, Mar. 2016.

[11] Z. Zhang, X. Wen, H. Xu, and L. Yuan, "Sensing nodes selective fusion scheme of spectrum sensing in spectrum-heterogeneous cognitive wireless sensor networks," *IEEE Sensors J.*, vol. 18, no. 1, pp. 436–445, Jan. 2018.

[12] W. Khalid and H. Yu, "Spatial–temporal sensing and utilization in full duplex spectrum-heterogeneous cognitive radio networks for the Internet of Things," *Sensors*, vol. 19, no. 6, p. 1441, 2019. [Online]. Available: https://www.mdpi.com/1424-8220/19/6/1441

[13] F. Hussain, S. A. Hassan, R. Hussain, and E. Hossain, "Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1251–1275, 2nd Quart., 2020.

[14] S. K. Sharma, T. E. Bogale, S. Chatzinotas, B. Ottersten, L. B. Le, and X. Wang, "Cognitive radio techniques under practical imperfections: A survey," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 4, pp. 1858–1884, 4th Quart., 2015.

[15] A. Ali and W. Hamouda, "Advances on spectrum sensing for cognitive radio networks: Theory and applications," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 2, pp. 1277–1304, 2nd Quart., 2017.

[16] M. Sun et al., "Adaptive sensing schedule for dynamic spectrum sharing in time-varying channel," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5520–5524, Jun. 2018.

[17] H. Zhang, Y. Nie, J. Cheng, V. C. Leung, and A. Nallanathan, "Sensing time optimization and power control for energy efficient cognitive small cell with imperfect hybrid spectrum sensing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 2, pp. 730–743, Feb. 2017.

[18] N. Mastronarde and M. Van Der Schaar, "Fast reinforcement learning for energy-efficient wireless communication," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6262–6266, Dec. 2011.

[19] H. Sun et al., "A cost-efficient skipping based spectrum sensing scheme via reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 71, no. 2, pp. 2220–2224, Feb. 2022.

[20] H. Li et al., "Utility-based cooperative spectrum sensing scheduling in cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 645–655, Jan. 2017.

[21] Y. Alghorani, G. Kaddoum, S. Muhaidat, and S. Pierre, "On the approximate analysis of energy detection over n rayleigh fading channels through cooperative spectrum sensing," *IEEE Wireless Commun. Lett.*, vol. 4, no. 4, pp. 413–416, Aug. 2015.

[22] H. Khoshkbari and G. Kaddoum, "Deep recurrent reinforcement learning for partially observable user association in a vertical heterogenous network," *IEEE Commun. Lett.*, vol. 27, no. 12, pp. 3235–3239, Dec. 2023.

[23] H. Khoshkbari, S. Sharifi, and G. Kaddoum, "User association in a VHetNet with delayed CSI: A deep reinforcement learning approach," *IEEE Commun. Lett.*, vol. 27, no. 8, pp. 2257–2261, Aug. 2023.

[24] R. Koike, R. Ariizumi, and F. Matsuno, "Simultaneous optimization of discrete and continuous parameters defining a robot morphology and controller," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, May 24, 2023, doi: 10.1109/TNNLS.2023.3272068.

[25] S. Guo and X. Zhao, "Deep reinforcement learning optimal transmission algorithm for cognitive Internet of Things with RF energy harvesting," *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 2, pp. 1216–1227, Jun. 2022.

[26] Y. Li, W. Zhang, C.-X. Wang, J. Sun, and Y. Liu, "Deep reinforcement learning for dynamic spectrum sensing and aggregation in multichannel wireless networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 2, pp. 464–475, Jun. 2020.

[27] X. Tan et al., “Cooperative multi-agent reinforcement-learning-based distributed dynamic spectrum access in cognitive radio networks,” *IEEE Internet Things J.*, vol. 9, no. 19, pp. 19477–19488, Oct. 2022.

[28] B. F. Lo, “A survey of common control channel design in cognitive radio networks,” *Phys. Commun.*, vol. 4, no. 1, pp. 26–39, 2011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1874490710000406

[29] K. Koçkaya and I. Develi, “Spectrum sensing in cognitive radio networks: Threshold optimization and analysis,” *EURASIP J. Wireless Commun. Netw.*, vol. 2020, no. 1, p. 255, 2020.

[30] A. Ahmed, D. Mishra, G. Prasad, and K. L. Baishnab, “Cognitive radio timing protocol for interference-constrained throughput maximization,” *IEEE Trans. Cogn. Commun. Netw.*, vol. 8, no. 2, pp. 989–1004, Jun. 2022.

[31] J. Lorincz, I. Ramljak, and D. Begušić, “Analysis of the impact of detection threshold adjustments and noise uncertainty on energy detection performance in MIMO-OFDM cognitive radio systems,” *Sensors*, vol. 22, no. 2, p. 631, Jan. 2022.

[32] M. Lauridsen, R. Krigslund, M. Rohr, and G. Madueno, “An empirical NB-IoT power consumption model for battery lifetime estimation,” in *Proc. IEEE 87th Veh. Technol. Conf.*, 2018, pp. 1–5.

[33] H. Davies. “Making energy harvesting work for edge IoT devices,” Embedded.com. 2021. [Online]. Available: https://www.embedded.com/making-energy-harvesting-work-for-edge-iot-devices/?nab=0

SADIA KHAF (Student Member, IEEE) received the B.E. degree in electrical engineering from the National University of Sciences and Technology, School of Electrical Engineering and Computer Science, Pakistan, in 2015, and the M.S. degree in electrical and electronics engineering from Bilkent University, Türkiye, in 2018. She is currently pursuing the Ph.D. degree with the École de technologie supérieure (ÉTS), Canada. From 2015 to 2018, she was a Research Assistant with IONOLAB, Türkiye. From 2018 to 2020, she was with the Faculty of Electrical Engineering, Ghulam Ishaq Khan Institute of Engineering Sciences and Technology, Pakistan, as a Lecturer. Her research interests include cognitive radio networks, Internet of Things, radio resource management, and machine learning. She received several research excellence awards and grants, including the Fonds de recherche du Québec, Nature et technologies Doctoral Fellowship, the P.E.O. International Peace Scholarship, ÉTS Bourses D’implication aux Supérieurs, IEEE Canada Foundation’s Women in Engineering Prize, and the ÉTS Palmarès Féminin pluriel Award.

GEORGES KADDOUM (Senior Member, IEEE) received the bachelor’s degree in electrical engineering from the École Nationale Supérieure de Techniques Avancées, Brest, France, the M.S. degree in telecommunications and signal processing (circuits, systems, and signal processing) from the Université de Bretagne Occidentale and Telecom Bretagne, Brest, in 2005, and the Ph.D. degree (Highest Hons.) in signal processing and telecommunications from the National Institute of Applied Sciences, University of Toulouse, Toulouse, France, in 2009. He is currently a Professor, the Research Director of the Resilient Machine Learning Institute, and the Tier 2 Canada Research Chair of the École de Technologie Supérieure (ÉTS), Université du Québec, Montreal, Canada. He has published over more than 300 journals, conference papers, two chapters in books, and has eight pending patents. His recent research interests include wireless communication networks, tactical communications, resource allocations, and network security. He received the Best Papers Awards at the 2014 IEEE International Conference on Wireless and Mobile Computing, Networking, Communications, the 2017 IEEE International Symposium on Personal Indoor and Mobile Radio Communications, and the 2023 IEEE International Wireless Communications and Mobile Computing Conference. Moreover, he received the IEEE Transactions on Communications Exemplary Reviewer Award in 2015, 2017, and 2019. In addition, he received the Research Excellence Award of the Université du Québec in 2018. In 2019, he received the Research Excellence Award from ÉTS in recognition of his outstanding research outcomes. He also won the 2022 IEEE Technical Committee on Scalable Computing Award for Excellence (Middle Career Researcher). Finally, he has received the prestigious 2023 MITACS Award for Exceptional Leadership. He served as an Associate Editor for IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY and IEEE COMMUNICATIONS LETTERS. He is currently serving as an Area Editor for the IEEE TRANSACTIONS ON MACHINE LEARNING IN COMMUNICATIONS AND NETWORKING and an Editor for the IEEE TRANSACTIONS ON COMMUNICATIONS.

JOAO VICTOR DE CARVALHO EVANGELISTA (Member, IEEE) was born in Recife, Brazil, in 1992. He received the B.S. and M.S. degrees in electrical engineering from the Universidade Federal de Pernambuco, Recife, in 2015 and 2016, respectively, and the Ph.D. degree in electrical engineering from the École de Technologie Supérieure, Université du Québec, Montreal, Canada, in 2021. He received the Mitacs Globalink Fellowship in 2016. He is currently a 5G System Developer with Ericsson Canada. His current research interests include machine learning applied to wireless communications, machine-to-machine communications, and the stochastic geometric modeling of wireless networks.