

Learning-based Spectrum Sensing in Cognitive Radio Networks via Approximate POMDPs

Bharath Keshavamurthy, Nicolò Michelusi *Senior Member, IEEE*

Abstract

In this paper, a novel spectrum sensing and access strategy is proposed, wherein a cognitive radio learns a time-frequency correlation model defining the occupancy behavior of incumbents via the Baum-Welch algorithm, and concurrently devises an optimal spectrum sensing and access strategy that exploits this learned correlation model, under spectrum sensing constraints. The optimal strategy is optimized via an approximate point-based value iteration method, to facilitate control of the trade-off between secondary network throughput and incumbent interference, along with fragmentation and Hamming distance state filters to alleviate its computational complexity. Numerical results demonstrate improvements of 60% over correlation-based clustering, 25% over Neyman-Pearson Detection, 6% over Viterbi, and 7% over adaptive deep Q-network. The proposed solution is extended to a distributed multi-agent settings with neighbor discovery and channel access rank allocation, which improves throughput by 43% over cooperative Temporal Difference SARSA, 84% over cooperative greedy distributed learning, and 324% over non-cooperative learning via g-statistics and ACKs. This multi-agent scheme is implemented on the DARPA Spectrum Collaboration Challenge (SC2) platform, demonstrating superior performance over competitors in a real-world TDWR-UNII WLAN scenario emulation, and its implementation feasibility is demonstrated on an ad-hoc distributed wireless platform of ESP32 radios, exhibiting 96% channel access success probability.

Index Terms

Hidden Markov Model, Cognitive Radio, Spectrum Sensing, POMDP

Part of this research has been submitted to IEEE ICC 2021 [1].

This research has been funded in part by NSF under grant CNS-1642982.

B. Keshavamurthy is with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN.

N. Michelusi is with the School of Electrical, Computer and Energy Engineering, Arizona State University, AZ.

Email: bkeshava@purdue.edu, nicolo.michelusi@asu.edu

I. INTRODUCTION

Cognitive radios have been touted as instrumental in solving resource-allocation problems in resource-constrained radio environments. Their adaptive computational intelligence facilitates the dynamic allocation of scarce network resources, particularly the spectrum. With the advent of fifth-generation (5G) cellular technologies [2], [3], a multitudinous array of devices have been or will be brought into the wireless communication ecosystem, resulting in an enormous strain on the available spectrum resources. Dynamic Spectrum Access, the key defining feature of cognitive radio networks, is being widely studied as a solution to the problem of spectrum scarcity, in both military and consumer spheres: cognitive radios intelligently access portions of the spectrum unused by the sparse and infrequent transmissions of licensed users in the network, in order to deliver their own network flows, while adhering to non-interference compliance requirements laid down by bureaucratic agencies such as the Federal Communications Commission [4], [5].

In order to intelligently access the spectrum white-spaces, the cognitive radio, referred to as a Secondary User (SU), needs to solve for a channel sensing and subsequent access policy based on noisy observations of the occupancy behavior of the licensed users or incumbents in the network, referred to as Primary Users (PUs). Yet, critical design limitations, driven by energy efficiency requirements or constraints on sensing times [6], prevent the SU from sensing simultaneously all the channels in the discretized spectrum of interest. Under these constraints, the SU can only sense a small fraction of all the available channels, and access those deemed idle, as studied in [6]–[13]. Nevertheless, this approach is quite conservative, since it does not allow the SU to access the large pool of channels that have not been sensed. Yet, incumbent occupancy may exhibit correlation across both time and frequency, as demonstrated in [14] and visualized in Fig. 2. Exploiting this time-frequency correlation structure may significantly improve white-space detection, thus enabling SUs to predict the state of those channels that have not been directly sensed, and unlocking additional opportunities for SU spectrum access.

Departing from these works, in this paper we propose to learn these time-frequency correlation statistics in PU occupancy behavior via a parametric model, and to concurrently utilize these learned statistics to solve for an optimal sensing and access policy using approximate point-based value iteration. While [15] leverages time-frequency correlation and allows the SU to access channels that have not been sensed, it employs pre-loaded databases to estimate the correlation statistics: an unrealistic approach in non-stationary settings, which requires instead concurrent

learning and sensing-access strategy optimization, as we do in our paper. We also extend our single-agent system model to distributed and centralized multi-agent deployment settings, with neighbor discovery and channel access rank allocation schemes over the control channel, to not only study the implementation feasibility of our POMDP framework, but to also illustrate the performance disparities between collaborative and opportunistic (non-cooperative or competitive) access.

Related Work: The work [8] details a solution for spectrum sensing and access employing TD-SARSA with linear value function approximation. However, this work fails to capitalize on the correlated occupancy behavior of the PUs across frequencies. Additionally, the authors fail to provide a mechanism to manage the trade-off between secondary network throughput and incumbent interference, which we do. Frequency correlation is studied in [15] under a noiseless setting. In our work, we employ a Hidden Markov Model (HMM) system-level framework in which the true occupancy states are hidden behind noisy observations at the SU’s spectrum sensor. Under the HMM framework, the work [9] develops a Viterbi algorithm for white-space detection, using prior knowledge of the transition model and a non-adaptive sensing strategy. In contrast, in our proposed solution, SUs learn the transition model with the Baum-Welch algorithm, and leverage this knowledge to concurrently optimize spectrum sensing and access under sensing limitations via approximate point-based value iteration. Alternatively, the proposals outlined in [9] and [15] determine the time-frequency incumbent occupancy correlation structure offline using pre-loaded databases, which is inefficient in non-stationary settings.

Furthermore, works in the state-of-the-art develop spectrum sensing and access algorithms with assumptions that either the occupancy behavior of incumbents is correlated over time but independent across frequency [8], [9], or independent across both time and frequency [6], [7], [10]–[13], which is not only impractical but also imprudent because critical information aiding the accurate detection of white-spaces can be gleaned by exploiting the correlation in their time-frequency occupancy behavior. Prudently, in this paper, we exploit both frequency and temporal correlation. Not dissimilar to the system model adhered to in our work, the adaptive DQN framework in [16] considers both time and frequency correlation in incumbent occupancies, governed by an unknown Markov model, but fails to provide a mechanism to regulate the trade-off between cognitive radio throughput and incumbent interference. In contrast, our framework does provide such a mechanism and exhibits superior performance, as illustrated in Sec. IV.

Analyzing the state-of-the-art in the distributed cognitive radio networks domain, we find

both collaborative as well as opportunistic schemes for channel access, namely: [8] describes a TD-SARSA framework with Linear Function Approximation, while [17] details a collaborative scheme (greedy learning under pre-allocation) as well as an opportunistic scheme (g-statistics with ACKs). However, [8] fails to leverage the correlation in incumbent occupancy behavior across frequencies, and estimates spectrum occupancy directly from observations via energy detection, hence neglecting the underlying probabilistic observation model. The multi-agent frameworks in [17] are based on assumptions of independence in incumbent occupancy behavior across both time and frequency, which as mentioned earlier, is impractical and imprudent. Finally, to the best of our knowledge, spectrum access and sensing algorithms have not been implemented in real-world settings. In contrast, we implement our solution on the DARPA SC2 emulation test-bed and on a custom-built ESP32 network.

Contributions: In a nutshell, the contributions of this paper are as follows:

- We develop a Partially Observable Markov Decision Process (POMDP) formulation for spectrum sensing and access in a radio environment with a single cognitive radio node and multiple licensed users exhibiting Markovian correlation in their occupancy behavior across both time and frequency, under sensing limitations;
- We develop an online parameter estimation algorithm to learn the incumbents' occupancy correlation model via the Baum-Welch algorithm;
- Concurrently, we leverage these learned statistics in a randomized point-based value iteration algorithm known as PERSEUS, to devise the optimal spectrum sensing and access policy; additionally, we alleviate the computational complexity associated with PERSEUS by introducing fragmentation heuristics and belief update simplification tactics (via Hamming distance state filters);
- Next, we extend this single-agent formulation to distributed multi-agent deployment settings, with neighbor discovery (RSSI-based thresholding) and channel access rank allocation (quorum-based preferential ballot voting), and demonstrate enhanced performance over both collaborative and opportunistic distributed multi-agent state-of-the-art;
- In order to evaluate the performance of our POMDP policy in centralized multi-agent settings, we retrofit it into our BAM! Wireless radio [18], designed specifically for the DARPA Spectrum Collaboration Challenge (SC2) [19], [20], emulate its operations during the Active Incumbent scenario (TDWR-UNII WLAN) [21], and prove superior performance over heuristics that perform channel and bandwidth allocation via weighted PSD + CIL

heuristics [22]–[26]; and in distributed multi-agent settings, we evaluate the implementation feasibility of our distributed POMDP optimal policy by testing it on an ad-hoc wireless platform of ESP32 radios [27], [28].

Specifically, highlighting the features of our framework against black-box ML/DL models, we find that our online estimation of the MDP transition model (time-frequency correlation statistics) scales well to non-stationary settings, unlike pre-loaded static training datasets. Unlike non-adaptive strategies like the Viterbi algorithm [9] which employ a fixed channel sensing set throughout its period of operation, our solution adapts the channel sensing set in accordance to the estimated occupancy transitions and reward/penalty feedback. Next, contrasting our solution against model-based [8] and model-free RL models [16], we achieve superior performance owing to more accurate estimation of the MDP transition model parameters and better approximations based on the time-frequency correlation in incumbent occupancy behavior, namely, fragmentation (frequency correlation) and Hamming distance state filters (temporal correlation).

The rest of this paper is organized as follows: Sec. II details the system model; Sec. III describes our algorithmic solutions; Sec. IV presents numerical evaluations for the single-agent case; Sec. V elucidates an extension of our solution to a distributed multi-agent setup, followed by implementations in a centralized multi-agent settings on DARPA SC2, and in a decentralized ad-hoc platform of ESP32 radios; finally Sec. VI provides concluding remarks.

II. SYSTEM MODEL

A. Signal Model

We consider a primary network of J incumbents/licensed users referred to as Primary Users (PUs) and a secondary network of \tilde{J} cognitive radio nodes referred to as Secondary Users (SUs), trying to exploit portions of the spectrum left unused by these PUs, as illustrated in Fig. 1. In the following, we focus on the single-agent case ($\tilde{J} = 1$); we will discuss the multi-agent scenario in Sec. V. The spectrum of interest is discretized into K channels of equal bandwidth W . The discretized wide-band signal received at the SU’s spectrum sensor in time-slot i at carrier frequency k can be expressed in the frequency domain as

$$Y_k(i) = \sum_{j=1}^J H_{j,k}(i) X_{j,k}(i) + V_k(i), \quad (1)$$

where $X_{j,k}(i)$ represents the frequency domain signal of PU $j \in \{1, 2, \dots, J\}$ in channel $k \in \{1, 2, \dots, K\}$, with $X_{j,k}(i)=0$ if PU j is not transmitting over channel k in time-slot i ; $H_{j,k}(i)$

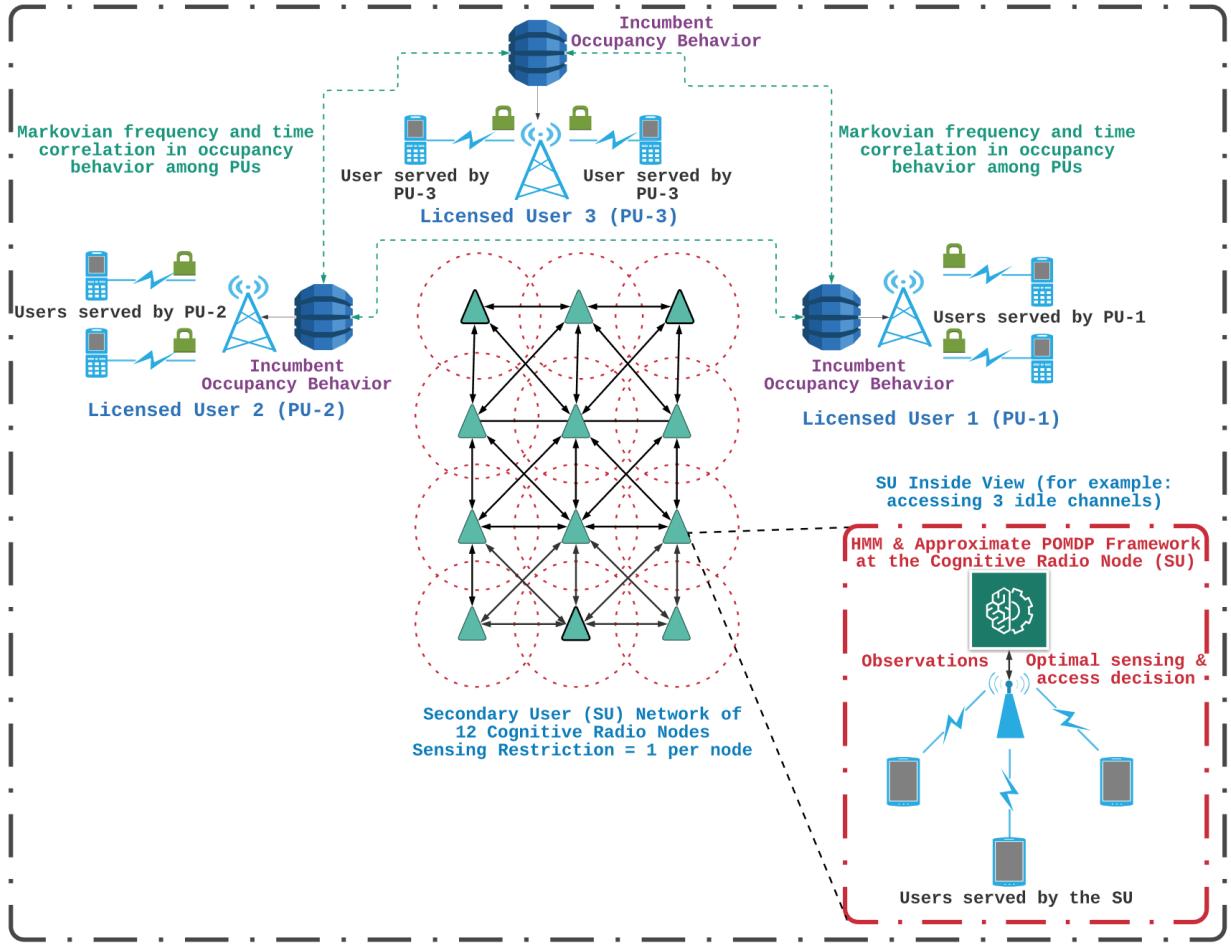


Fig. 1. The radio ecosystem under analysis: An exemplification of the system model detailed in Sec. II-A with $J=3$ and $\tilde{J}=12$: we first study deployment scenarios with $\tilde{J}=1$ before extending our analysis to multi-agent settings

denotes the frequency domain channel between the SU and PU j ; and $V_k(i) \sim \mathcal{CN}(0, \sigma_V^2)$ constitutes the zero-mean circularly symmetric additive complex Gaussian noise with variance σ_V^2 , i.i.d across time and frequency, and independent of the channel H and the PU signal X . Assuming an Orthogonal Frequency Division Multiple Access (OFDMA) strategy among the PUs, and letting $X_k(i) \triangleq X_{j_{k,i},k}(i)$ and $H_k(i) \triangleq H_{j_{k,i},k}(i)$, where subscript $j_{k,i}$ denotes the index of the PU that occupies channel k in time-slot i , we can rewrite (1) as

$$Y_k(i) = H_k(i)X_k(i) + V_k(i), \quad (2)$$

where $X_k(i)=0$ if channel k is idle in time-slot i . We model the frequency domain channel as Rayleigh fading with variance σ_H^2 , $H_k(i) \sim \mathcal{CN}(0, \sigma_H^2)$, i.i.d across time and frequency.

B. Occupancy Correlation Structure

The frequency domain signal of the PU occupying channel k in time-slot i is modeled as

$$X_k(i) = \sqrt{P_T} B_k(i) S_k(i), \quad (3)$$

where P_T denotes the transmission power of the occupant PU; $B_k(i)$ represents the binary channel occupancy variable, with $B_k(i)=1$ if channel k is occupied by a PU in time-slot i , and $B_k(i)=0$ otherwise; $S_k(i)$ is the transmitted symbol, i.i.d across time and frequency, modeled from a certain constellation. Then, $H_k(i)X_k(i)=\sqrt{P_T}B_k(i)H_k(i)S_k(i)$. Herein, we approximate $H_k(i)S_k(i)$ as a zero-mean complex Gaussian random variable with variance $\sigma_H^2 \mathbb{E}[|S_k|^2]$. We denote the spectrum occupancy state in time-slot i as

$$\vec{B}(i) = [B_1(i), B_2(i), B_3(i), \dots, B_K(i)]^\top \in \{0, 1\}^K. \quad (4)$$

We assume that spectrum occupancy is correlated in time and frequency because PUs typically occupy a set of adjacent channels (frequency correlation), repeating similar motifs in behavior over an extended period of time (temporal correlation) [14], [29], [30]. To capture temporal correlation, we model the evolution of $\vec{B}(i)$ over time as a Markov process

$$\mathbb{P}(\vec{B}(i+1)|\vec{B}(j), \forall j \leq i) = \mathbb{P}(\vec{B}(i+1)|\vec{B}(i)). \quad (5)$$

In addition, to model frequency correlation, we further decompose $\mathbb{P}(\vec{B}(i+1)|\vec{B}(i))$ as

$$\mathbb{P}(\vec{B}(i+1)|\vec{B}(i)) = \mathbb{P}(B_1(i+1)|B_1(i)) \prod_{k=2}^K \mathbb{P}(B_k(i+1)|B_{k-1}(i+1), B_k(i)). \quad (6)$$

In other words: the occupancy of frequency band k in time-slot $i+1$ depends on the occupancy of the adjacent frequency band $k-1$ in the same time-slot $i+1$, and that of the same frequency band k in the previous time-slot i , as illustrated in Fig. 2. If the frequency correlation direction is changed, i.e., the occupancy of channel $k+1$ influences the occupancy of channel k , $k \in \{1, 2, \dots, K-1\}$ (bottom-up correlation), our model and subsequent analyses still hold. We parameterize this two-chain Markovian correlation structure with the parameter vector $\vec{\theta} = [\vec{p} \ \vec{q}]^\top$, subsequently estimated with the Baum-Welch in Sec. III-A, where

$$\begin{aligned} \vec{p} &= [p_{uv} = \mathbb{P}(B_k(i+1) = 1 | B_{k-1}(i+1) = u, B_k(i) = v) : u, v \in \{0, 1\}]^\top, \text{ and} \\ \vec{q} &= [q_w = \mathbb{P}(B_1(i+1) = 1 | B_1(i) = w) : w \in \{0, 1\}]^\top. \end{aligned} \quad (7)$$

To experimentally validate the proposed parameterized time-frequency correlated model, we

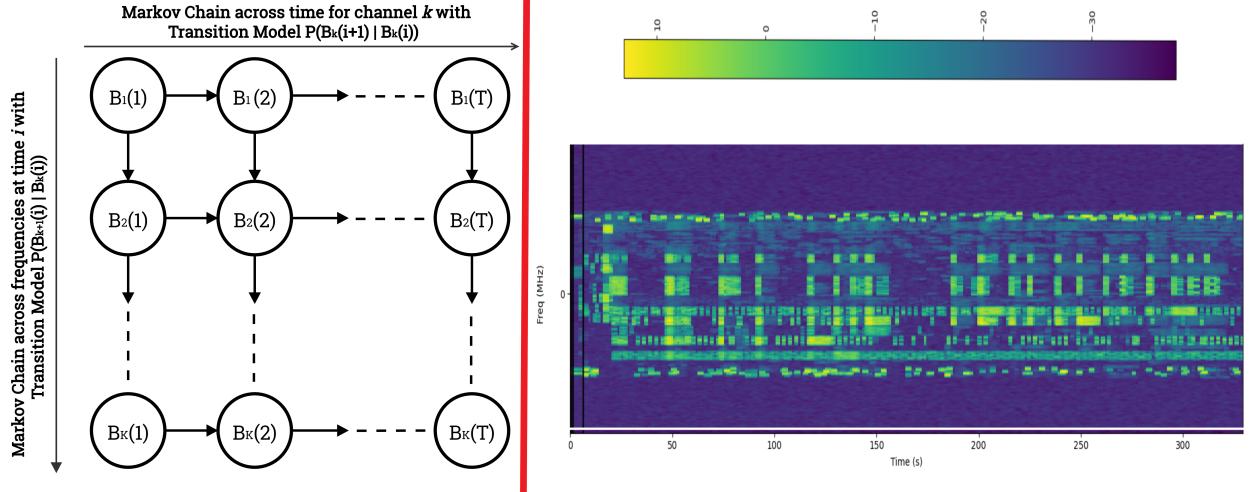


Fig. 2. The visualization of the incumbent occupancy time-frequency correlation structure as two dependent Markov chains: one across time and the other across frequencies (L) | The combined PSD plot of the occupancy behavior of incumbents and competitors during the DARPA SC2 Active Incumbent scenario emulation (R)

evaluated the Bayesian Information Criterion (BIC) metric, defined as

$$BIC = \gamma \ln \nu - 2 \ln \mathbb{P}(\mathbf{B} | \hat{\theta}^*), \quad (8)$$

based on a dataset constituted of PSD measurements of the incumbent and the competitors in the DARPA SC2 Active Incumbent scenario emulated on the Colosseum [19]–[21], [31], depicted in Fig. 2 (R). In (8), ν is the sample size, γ is the number of model parameters, \mathbf{B} is the time-frequency binary occupancy matrix from the dataset, and $\hat{\theta}^*$ is the parameterized model fit to the dataset using the Baum-Welch algorithm detailed in Sec. III-A. We used a 70-30 training-test split to evaluate the BIC metric, i.e., the occupancy data collected during the first 70% of the 330 seconds of scenario emulation is used to estimate the model parameters, while the remaining 30% is employed to evaluate the BIC metric. We found that our proposed time-frequency correlation model yields a BIC of 72.061, the best compared to other state-of-the-art models: time-frequency independence models (98.868) [7], [10]–[13], only temporal correlation models (95.257) [8], and only frequency correlation models (77.300). This evaluation reveals that exploiting frequency correlation is more important than time correlation, but exploiting *both* time and frequency correlation provides a better fit to the dataset.

C. Channel Sensing Model

Equipped with a spectrum sensor, the SU detects white-spaces and accesses them to deliver its network flows. Due to concerns about energy-efficiency and sensing/data aggregation times

[6], the SU can sense a maximum of κ spectrum bands in a time-slot, with $1 \leq \kappa \leq K$. Let $\mathcal{K}_i \subseteq \{1, 2, \dots, K\}$ be the set of channels sensed by the SU at time i , with $|\mathcal{K}_i| \leq \kappa$. This selection is dictated by a sensing policy, as defined in Sec. III-B. After sensing the channels listed in \mathcal{K}_i , the obtained observation vector is $\vec{Y}(i) = [Y_k(i)]_{k \in \mathcal{K}_i}$, with $Y_k(i)$ given in (2).

Owing to (2), the i.i.d. assumptions of the noise $V_k(i)$, the transmitted symbols $S_k(i)$, and the channels $H_k(i)$ across frequency, as discussed in Sec. II-A, the probability density function (pdf) of $\vec{Y}(i)$ conditional on $\vec{B}(i)$ and the sensing set \mathcal{K}_i is given by

$$f(\vec{Y}(i)|\vec{B}(i), \mathcal{K}_i) = \prod_{k=1}^K f(Y_k(i)|B_k(i)), \text{ where } Y_k(i)|B_k(i) \sim \mathcal{CN}(0, \sigma_H^2 P_T B_k(i) + \sigma_V^2). \quad (9)$$

D. POMDP Formulation

POMDPs model the repeated, sequential interactions of an agent tasked with maximizing its reward, with a stochastic environment, in which the agent has only access to noisy observations of the state. Our POMDP formulation, depicted in Fig. 3 and represented by the 5-tuple $(\mathcal{B}, \mathcal{A}, \mathcal{Y}, \mathbf{A}, \mathbf{M})$, features the state space of the underlying MDP, denoted by $\mathcal{B} \equiv \{0, 1\}^K$, which is given by all possible realizations of the occupancy vector \vec{B} ; the action space of the SU, denoted by \mathcal{A} , which is described by all possible combinations in which $1 \leq \kappa \leq K$ channels are chosen to be sensed in a time-slot (discussed in Sec. II-C); the observation space, denoted by \mathcal{Y} , which is discussed in Sec. II-A; the transition model of the underlying MDP, denoted by \mathbf{A} , which is discussed in Sec. II-B; and the observation model (also known as the emission model), denoted by \mathbf{Y} , and described by (9). The POMDP process flow, relevant to our discussions in this section, is illustrated in Fig. 3.

Prior to gathering the occupancy information in time-slot i , based on the measurements obtained by the SU's spectrum sensor up to, but not including, time-slot i , the POMDP state is described by the prior belief, denoted by β_i , representing the probability distribution of the underlying MDP state $\vec{B}(i)$ given the history. Given β_i , the SU chooses a sensing action according to a sensing policy $\mathcal{K}_i = \pi(\beta_i) \in \mathcal{A}$, senses the frequency bands corresponding to the channel indices in the set \mathcal{K}_i (Sec. II-C), observes $[Y_k(i)]_{k \in \mathcal{K}_i} \in \mathcal{Y}$, and updates its belief of the underlying MDP state $\vec{B}(i)$ to obtain the posterior belief

$$\begin{aligned} \hat{\beta}_i(\vec{B}') &= \mathbb{P}(\vec{B}(i) = \vec{B}' | \beta_i, \mathcal{K}_i, [Y_k(i)]_{k \in \mathcal{K}_i}) \\ &= \frac{\mathbb{P}([Y_k(i)]_{k \in \mathcal{K}_i} | \vec{B}', \mathcal{K}_i) \beta_i(\vec{B}')}{\sum_{\vec{B}'' \in \{0,1\}^K} \mathbb{P}([Y_k(i)]_{k \in \mathcal{K}_i} | \vec{B}'', \mathcal{K}_i) \beta_i(\vec{B}'')}. \end{aligned} \quad (10)$$

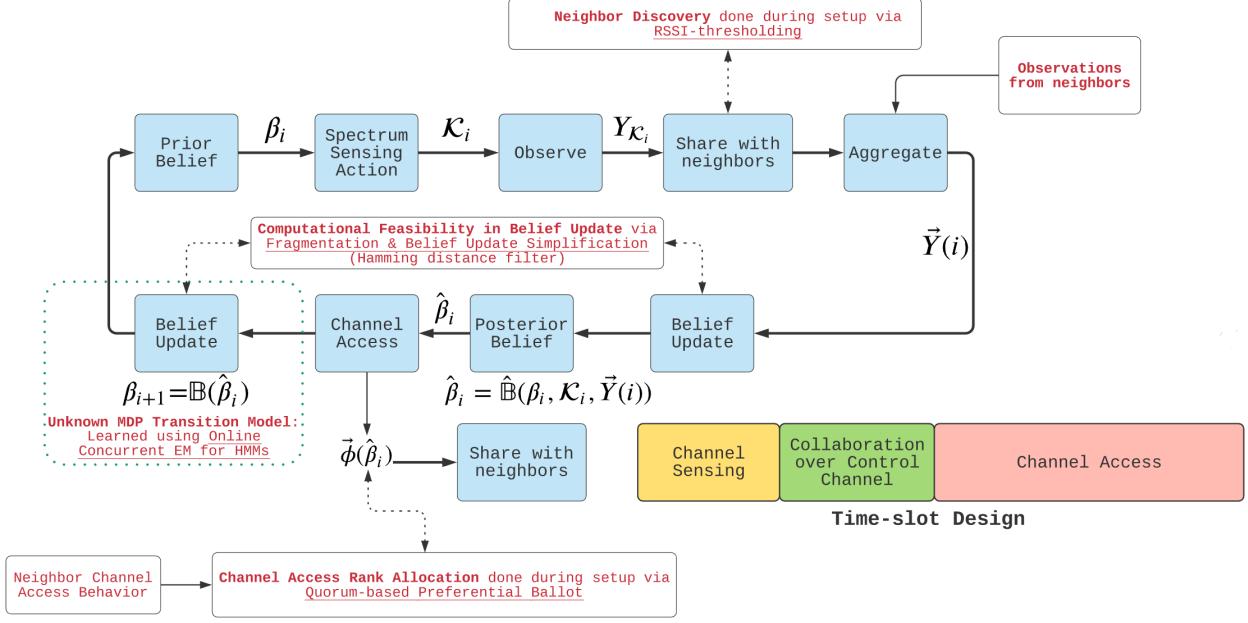


Fig. 3. The POMDP process flow as discussed in Sec. II-D, with neighbor discovery, channel access rank allocation, and time-slot design being relevant design points discussed in our multi-agent deployment analysis (Sec. V)

Given $\hat{\beta}_i$ the SU then performs channel access decisions $\vec{\phi}(i) \in \{0, 1\}^K$, where $\vec{\phi}_k(i) = 1$ if the SU accesses the k th spectrum band, and $\vec{\phi}_k(i) = 0$ otherwise. To determine $\vec{\phi}(i)$, we define the expected reward metric

$$R(\vec{\phi}(i), \hat{\beta}_i) = \sum_{k=1}^K (1 - \hat{\beta}_{i,k})\phi_k(i) - \lambda\hat{\beta}_{i,k}\phi_k(i). \quad (11)$$

Note that, if the SU uses the k th spectrum band ($\phi_k(i) = 1$), it accrues a reward if the spectrum band is truly idle (with posterior probability $1 - \hat{\beta}_{i,k}$), and a penalty λ if it is occupied (with posterior probability $\hat{\beta}_{i,k}$); it accrues no reward for not using a channel. Hence, this reward metric captures both the number of truly idle channels (correctly estimated idle) accessed by the cognitive radio, accounting for the throughput maximization aspect of our objective, as well as the number of truly occupied channels (incorrectly estimated idle) accessed by it, accounting for the incumbent interference minimization aspect of our objective, where λ regulates such trade-off. The optimal access decision is $\vec{\phi}^*(i) = \arg \max R(\vec{\phi}(i), \hat{\beta}_i)$, which can be solved in closed form as

$$\phi_k^*(i) = \begin{cases} 1, & \hat{\beta}_i < \frac{1}{1+\lambda} \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

yielding the optimal reward

$$R^*(\hat{\beta}_i) = \max_{\vec{\phi} \in \{0,1\}^K} R(\vec{\phi}(i), \hat{\beta}_i) = \sum_{k=1}^K \max\{1 - (1 + \lambda)\hat{\beta}_{i,k}, 0\}.$$

In other words, if the agent is confident that the channel is idle ($\hat{\beta}_i < \frac{1}{1+\lambda}$), then the SU accesses it; otherwise, it remains idle.

Ensuing the determination of the reward for its access decision from the radio environment, the SU computes the prior belief for the next time-slot $i + 1$ as

$$\beta_{i+1}(\vec{B}'') = \sum_{\vec{B}'} \mathbb{P}(\vec{B}(i+1) = \vec{B}'' | \vec{B}(i) = \vec{B}') \hat{\beta}(\vec{B}'). \quad (13)$$

Let

$$\hat{\beta}_i = \hat{\mathbb{B}}(\beta_i, \mathcal{K}_i, \vec{Y}(i)), \quad \beta_{i+1} = \mathbb{B}(\hat{\beta}_i) \quad (14)$$

denote the functions that map the prior belief β_i to the posterior belief $\hat{\beta}_i$ in time-slot i , and the posterior belief $\hat{\beta}_i$ to the next prior belief β_{i+1} in time-slot $i + 1$. The objective of the SU is to determine the optimal spectrum sensing policy (based on which the access decisions are made in the corresponding time-slots) to maximize its infinite-horizon discounted reward, i.e.,

$$\pi^* = \arg \max_{\pi} V^\pi(\beta), \quad (15)$$

where

$$V^\pi(\beta) = \mathbb{E}_\pi \left[\sum_{i=1}^{\infty} \gamma^i R^*(\hat{\beta}_i) \middle| \beta_0 = \beta \right], \quad (16)$$

where $0 < \gamma < 1$ is the discount factor, $\beta_0 = \beta$ is the initial belief, and $\hat{\beta}_i$ is the posterior belief induced by the policy $\mathcal{K}_i = \pi(\beta_i)$ and the observation vector $[Y_k(i)]_{k \in \mathcal{K}_i}$ via $\hat{\beta}_i = \hat{\mathbb{B}}(\beta_i, \mathcal{K}_i = \pi(\beta_i), [Y_k(i)]_{k \in \mathcal{K}_i})$. The optimal value function $V^*(\beta)$ can be shown to be solution of the Bellman's optimality equation $V^* = \mathcal{H}(V^*)$ [32], where \mathcal{H} is the Bellman's operator, defined as $V_{t+1} = \mathcal{H}(V_t)$ with

$$V_{t+1}(\beta) = \max_{\mathcal{A}} \sum_{\vec{B} \in \mathcal{B}} \beta(\vec{B}) \mathbb{E}_{[Y_k]_{k \in \mathcal{K}} | \vec{B}, \mathcal{K}} \left[R^*(\hat{\mathbb{B}}(\beta, \mathcal{K}, [Y_k]_{k \in \mathcal{K}})) + \gamma V_t(\mathbb{B}(\hat{\mathbb{B}}(\beta, \mathcal{K}, [Y_k]_{k \in \mathcal{K}}))) \right], \quad \forall \beta. \quad (17)$$

The optimal value function V^* can be determined via the value iteration algorithm $V_{t+1} = \mathcal{H}(V_t)$, which converges to V^* as $t \rightarrow \infty$ [32]. However, this direct approach results in complications associated with the lack of prior knowledge about the incumbent occupancy time-frequency correlation structure that defines the transition model of the underlying MDP, and the computational infeasibility of the approach: as the number of channels in the discretized spectrum of

interest increases, the number of states of the underlying MDP scales exponentially, resulting in a high-dimensional belief space, which makes the approach intractable. To address these two challenges, we propose the following solutions:

- We incorporate an HMM EM estimator, i.e., the Baum-Welch algorithm, to learn the time-frequency occupancy correlation structure while concurrently solving for the optimal sensing and access policy. This is developed in Sec. III-A.
- We incorporate a low-complexity approximate value iteration algorithm known as PERSEUS [33], with fragmentation (into independent subsets of highly-correlated channels) and belief update simplification heuristics (Hamming distance state filters). This is developed in Sec. III-B.

III. PROPOSED SOLUTION: THE ALGORITHMS

Practical MAC layer implementations of cognitive radios involve solving for the optimal sensing and access policy, without having any prior information about the time-frequency correlation structure underlying the occupancy behavior of the incumbents in the network. [24], [34]. As discussed earlier, this correlation structure may be leveraged to improve white-space detection, hence utilization. In this section, we propose a parameter estimator algorithm that learns this correlation structure over time. In Sec. III-B, we then use this knowledge in an approximate point-based value iteration framework based on PERSEUS [33] to determine the optimal sensing and access policy. Crucially, the parameter estimation and PERSEUS algorithms are executed concurrently, which is especially vital in non-stationary settings.

A. Occupancy Correlation Structure Estimation

Let τ refer to the learning period of the parameter estimation algorithm: this can be equal to the entire duration of the SU's interaction with the radio environment while solving for the optimal policy, implying concurrent model learning, or it can be equal to an initial learning period that has been set aside exclusively for the SU to estimate the underlying MDP's transition model, after which the PERSEUS algorithm is initiated, employing these final estimated (converged) transition probabilities. Defining $\mathbf{B} = [\vec{B}(i)]_{i=1}^{\tau}$ as the unknown sequence of states and $\mathbf{Y} = [\vec{Y}(i)]_{i=1}^{\tau}$ as the sequence of observations made at the SU's spectrum sensor from $i=1$ to $i=\tau$, we formulate the

Maximum Likelihood Estimation (MLE) problem to estimate the vector $\vec{\theta}$ that parameterizes the PU occupancy time-frequency correlation structure (detailed in Sec. II-B) as

$$\vec{\theta}^* = \arg \max_{\vec{\theta}} \log \left(\sum_{\mathbf{B}} \mathbb{P}(\mathbf{B}, \mathbf{Y} | \vec{\theta}) \right). \quad (18)$$

Solving this MLE formulation using the Baum-Welch algorithm, an Expectation-Maximization algorithm for HMMs [35], the E-step constitutes

$$Q(\vec{\theta} | \vec{\theta}^{(t)}) = \mathbb{E}_{\mathbf{B} | \mathbf{Y}, \vec{\theta}^{(t)}} \left[\log (\mathbb{P}(\mathbf{B}, \mathbf{Y} | \vec{\theta}^{(t)})) \right], \quad (19)$$

which can be computed using the Forward-Backward algorithm [35]; and the M-step constitutes

$$\vec{\theta}^{(t+1)} = \arg \max_{\vec{\theta}} Q(\vec{\theta} | \vec{\theta}^{(t)}), \quad (20)$$

which involves the re-estimation of $\vec{\theta}$ by employing the statistics $Q(\vec{\theta} | \vec{\theta}^{(t)})$ obtained from the Forward-Backward algorithm [BK: Equations?].

B. The PERSEUS Algorithm

In our proposed solution, we solve for the optimal spectrum sensing (and access, based on reward maximization detailed in Sec. II-D) policy, in parallel with the parameter estimation algorithm, employing its published iterative transition model estimates, until both the EM algorithm and the POMDP policy solver algorithms converge.

As alluded to in Sec. II-D, in order to solve the computational infeasibility precipitated by the exponential increase in the number of states of the underlying MDP, induced by an increase in the number of frequency bands in the discretized spectrum of interest, we employ approximate POMDP value iteration methods to ensure that the formulations and the algorithms scale well to a large number of relevant channels in the radio environment in which the SU operates. We choose the PERSEUS algorithm [33] to solve for the optimal policy, primarily motivated by the following: the exact value iteration strategies proposed in [32], namely the Exhaustive Enumeration algorithm and the Witness algorithm are untenable for large belief spaces, because these techniques involve performing the backup procedure, i.e., determining the optimal action (or hyperplane in a Piece-Wise Linear Convex (PWLC) context) for every belief point in the belief space; and a fellow contemporary approximate value iteration algorithm known as the Point-Based Value Iteration (PBVI) algorithm proposed in [36], although involves performing the backup operation over a reduced set of beliefs known as the "reachable beliefs," unlike

the strategies in [36], is computationally expensive due to the task of computing the distances between all the belief points in the set of reachable beliefs in addition to the subsequent backup operation on all these belief points. The PERSEUS algorithm, on the other hand, does not involve performing the backup operation for every point in the belief space, unlike the Exhaustive Enumeration and Witness algorithms detailed in [32]; and unlike the PBVI algorithm [36] does not involve computing the distances between all the belief points in the set of reachable beliefs, and furthermore, does not involve performing backups on all the reachable belief points, instead, PERSEUS involves "backing-up" only on a subset of this set of reachable beliefs, while ensuring that the computed solution is effective for all the points in the reachable belief set.

PERSEUS, a randomized, point-based, approximate POMDP value iteration algorithm, involves an initial phase of exploration, wherein the set of "reachable-beliefs," denoted by $\tilde{\mathcal{B}}$, is determined by allowing the SU to randomly interact with the radio environment. As referenced earlier, one simplifying (or approximating) feature of PERSEUS is to improve the value of all the belief points in the set $\tilde{\mathcal{B}}$, by computing the value of only a subset of these belief points, which are chosen iteratively at random. For finite-horizon POMDP formulations, the optimal value function V^* described by (17), can be approximated by a Piece-Wise Linear Convex (PWLC) function [33], in other words, the value function at iteration t is parameterized by a set of hyperplanes, denoted by $\{\vec{\alpha}_t^u\}_{u \in \{1, 2, \dots, |\tilde{\mathcal{B}}|\}}$, wherein each hyperplane represents a region of the belief space for which the action corresponding to this hyperplane, denoted by \mathcal{K}_t^u , is the maximizer. Ergo, the value function of belief β in a given iteration t is approximated as

$$V_t(\beta) \approx \beta \max_{u \in \{1, 2, \dots, |\tilde{\mathcal{B}}|\}} \cdot \vec{\alpha}_t^u, \quad (21)$$

where

$$\beta \cdot \vec{\alpha} = \sum_{\tilde{\mathcal{B}}} \beta(\tilde{\mathcal{B}}) \vec{\alpha}(\tilde{\mathcal{B}}) \quad (22)$$

denotes inner product. The approximately optimal spectrum sensing action is the one associated with the minimizing hyperplane α^* , and denoted as $\mathcal{K}_t^{u^*}$.

We define a set of unimproved belief points, denoted as $\tilde{\mathcal{U}}$, which initially corresponds to the set of reachable beliefs $\tilde{\mathcal{B}}$ obtained by the random exploration procedure detailed earlier. Pick a belief β_u from this set $\tilde{\mathcal{U}}$, and perform the backup operation on this chosen belief point, which as discussed earlier, involves associating a new hyperplane and its corresponding spectrum sensing

action with this belief β_u . In iteration $t+1$, defining \mathcal{K}_{t+1}^u as the action associated with hyperplane $\vec{\alpha}_{t+1}^u$, corresponding to belief $\beta_u \in \tilde{\mathcal{U}}$, we can describe the backup procedure mathematically as

$$\begin{aligned}\vec{\alpha}_{t+1} &= \xi_{\mathcal{K}_{t+1}^u}^u, \\ \mathcal{K}_{t+1}^u &= \arg \max_{\mathcal{K} \in \mathcal{A}} \beta_u \cdot \xi_{\mathcal{K}}^u,\end{aligned}\tag{23}$$

where $\xi_{\mathcal{K}}^u$ is the hyperplane corresponding to the one-step look-ahead under action $\mathcal{K} \in \mathcal{A}$ and belief β_u , i.e.,

$$\xi_{\mathcal{K}}^u(\vec{B}) = \mathbb{E}_{\vec{Y}|\vec{B}, \mathcal{K}} \left[R(\vec{B}, \hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y})) + \gamma \sum_{\vec{B}'} \mathbb{P}(\vec{B}(i+1) = \vec{B}' | \vec{B}(i) = \vec{B}) \xi_{\mathcal{K}, \vec{Y}}^u(\vec{B}') \right], \tag{24}$$

where $\xi_{\mathcal{K}, \vec{Y}}^u$ refers to the hyperplane corresponding to the future value function computed from the previous set of hyperplanes from the new belief $\hat{\mathbb{B}}(\hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y}))$ obtained from β_u by executing action \mathcal{K} and observing \vec{Y} , as

$$\xi_{\mathcal{K}, \vec{Y}}^u = \arg \max_{\vec{\alpha}_t^{u'}, u' \in \{1, 2, \dots, |\mathcal{B}|\}} \hat{\mathbb{B}}(\hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y})) \cdot \vec{\alpha}_t^{u'}. \tag{25}$$

After determining the hyperplane $\vec{\alpha}_{t+1}^u$ associated with this chosen belief point β_u using the backup procedure detailed above, we now know that $V_{t+1}(\beta_u) = \beta_u \cdot \vec{\alpha}_{t+1}^u$ is its approximate value function. The most crucial aspect of PERSEUS that approximates the optimization of a randomly chosen belief point to the entire set $\tilde{\mathcal{U}}$ is as follows: if the approximate value function for belief point $\beta_u \in \tilde{\mathcal{U}}$ is improved by the aforementioned backup iteration, i.e., $V_{t+1}(\beta_u) \geq V_t(\beta_u)$, the belief point β_u is removed from the set $\tilde{\mathcal{U}}$, and now, we check if this hyperplane $\vec{\alpha}_{t+1}^u$ improves the approximate value functions of the other beliefs in $\tilde{\mathcal{U}}$, i.e., $\forall \beta' \in \tilde{\mathcal{U}} - \{\beta_u\}$, if $\beta' \cdot \vec{\alpha}_{t+1}^u \geq V_t(\beta')$, this new hyperplane generates an improved approximate value function, and these respective belief points for which this hyperplane improves their approximate value functions, are removed from the set $\tilde{\mathcal{U}}$. In other words,

$$\begin{aligned}\tilde{\mathcal{U}} &\leftarrow \tilde{\mathcal{U}} - \{\beta_u\}, \text{ if } \beta_u \cdot \vec{\alpha}_{t+1}^u \geq V_t(\beta_u), \text{ and subsequently} \\ \tilde{\mathcal{U}} &\leftarrow \tilde{\mathcal{U}} - \{\beta' \in \tilde{\mathcal{U}} : \beta' \cdot \vec{\alpha}_{t+1}^u \geq V_t(\beta')\}.\end{aligned}\tag{26}$$

On the other hand, if this hyperplane $\vec{\alpha}_{t+1}^u$ worsens the approximate value function of β_u , i.e., $\beta_u \cdot \vec{\alpha}_{t+1}^u < V_t(\beta_u)$, the old hyperplane and its associated sensing action persist for β_u , mathematically, $\vec{\alpha}_{t+1}^u := \vec{\alpha}_t^u$ and $\mathcal{K}_{t+1}^u := \mathcal{K}_t^u$; but we still check for improvements with respect to the other belief points in $\tilde{\mathcal{U}}$, and remove all those belief points $\beta' \in \tilde{\mathcal{U}}$ for which $\beta' \cdot \vec{\alpha}_{t+1}^u \geq V_t(\beta')$.

In general, if a hyperplane determined from the backup procedure improves a belief point in the set of unimproved belief points $\tilde{\mathcal{U}}$, this news hyperplane (and its associated sensing action)

becomes the relevant hyperplane (and the relevant sensing action) for this belief point, and the belief point will be removed from the set of unimproved belief points $\tilde{\mathcal{U}}$. These sequence of operations (random choice from $\tilde{\mathcal{U}} \rightarrow$ backup \rightarrow check for improvement and removal) are performed iteratively until the set $\tilde{\mathcal{U}}$ is empty: this constitutes a single PERSEUS iteration. These PERSEUS iterations are executed until the specified value iteration termination condition is satisfied, i.e.,

$$|V_{t+1}(\beta) - V_t(\beta)| < \epsilon, \quad \forall \beta \in \tilde{\mathcal{B}}, \quad (27)$$

where $\epsilon > 0$ (a very small value), is the value iteration difference threshold.

The PERSEUS algorithm, although is an approximate POMDP method which eliminates the computational overhead associated with the exhaustive belief space and reachable space optimization techniques [32], [36] by approximating the optimization of a randomly chosen belief point to the entire set of unimproved, reachable belief points, still possesses computational intractability challenges because it involves iterations over all possible combinations of the occupancy state vector, i.e., $\vec{B} \in \{0, 1\}^K$, the computational cost scales exponentially with the number of states in the underlying MDP, which is induced by the number of channels K in the discretized spectrum of interest. In order to solve this computational tractability problem, we introduce two simplifying heuristics into the PERSEUS algorithm. Firstly, we avoid iterating over all possible occupancy states by considering only those state transitions that involve a Hamming distance of $d \in \{1, 2, \dots, K\}$ between two consecutive state vectors, $\vec{B}(i)$ and $\vec{B}(i + 1)$, this is practical because the temporal dynamics of the occupancy of the radio environment, dictated by the behavior of the PUs in the network, are typically slower than the processing dynamics of the POMDP agent, i.e., the SU. Secondly, we fragment the discretized spectrum into smaller, independent sets of correlated channels (for example, an 18 channel radio environment with 3 PUs is fragmented into 3 independent fragments, each comprising 6 channels correlated by the occupancy behavior of the corresponding PU); run PERSEUS on these fragments concurrently by employing multi-threading capabilities in software frameworks; and finally, combine the results from each of these fragmented, parallel runs to get a full picture about the performance of our POMDP agent: this is practical because in a radio environment with multiple PUs, each PU is typically restricted to a portion (a set of adjacent frequency bands) of the spectrum, either by design or by bureaucracy.

IV. NUMERICAL EVALUATIONS

Sticking with the single-agent deployment setting, our simulations evaluate the operational capabilities of the proposed POMDP framework and compare it against the state-of-the-art. The simulated radio environment constitutes $J=3$ incumbents, i.e., PUs, accessing a 2.88 MHz spectrum, discretized into $K=18$ channels, each having a bandwidth of $W=160$ kHz, and a cognitive radio node ($\tilde{J}=1$ SU) trying to intelligently access spectrum holes to deliver its network flows while limiting incumbent interference, as illustrated in Fig. 1. The 3 PUs access these 18 channels according to a time-frequency Markovian correlation structure parameterized by

$$\vec{\theta} = \begin{bmatrix} \vec{p} & \vec{q} \end{bmatrix}$$

as described in Sec. II-B, where

$$\vec{p} = \begin{bmatrix} p_{00} = 0.1 & p_{01} = 0.3 & p_{10} = 0.3 & p_{11} = 0.7 \end{bmatrix},$$

and

$$\vec{q} = \begin{bmatrix} q_0 = 0.3 & q_1 = 0.8 \end{bmatrix}.$$

Regarding the channel sensing limitations induced by a need to minimize the amount of time and energy spent sensing the spectrum [6], we model our simulation framework on $\kappa=6$, i.e., in any given time-slot i , the SU can sense a maximum of 6 channels out of the 18 in the discretized spectrum of interest. Regarding the expected Signal to Interference Noise Ratios (SINR) at the PUs and the SU, subject to fading, and conditioned on the PU and SU access decisions, we model our simulation framework based off the following numbers:

$\text{SINR}_{\text{SU}}(k, i)=0$, if the SU does not access channel k in time-slot i ,

$\text{SINR}_{\text{SU}}(k, i)=11$ dB, if the SU accesses a truly idle channel k in time-slot i ,

$\text{SINR}_{\text{SU}}(k, i)=-6$ dB, if SU accesses an incumbent-occupied channel k in slot i ,

$\text{SINR}_{\text{PU}_j}(k, i)=0$, if the PU j does not access channel k in time-slot i ,

$\text{SINR}_{\text{PU}_j}(k, i)=17$ dB, if PU j occupies channel k in slot i without SU interference,

$\text{SINR}_{\text{PU}_j}(k, i)=6$ dB, if PU j occupies channel k in slot i with SU interference.

As described in Sec. II, the only objective of the SU is to maximize its throughput subject to a constraint on the amount of interference its transmissions can cause to incumbents in the network. To this end, assuming an always back-logged SU, i.e., the SU always has network

flows to deliver, the optimal POMDP sensing policy dictates which channels should be sensed by the SU's spectrum sensor in a given time-slot, based off the learned correlated occupancy dynamics of the PUs, in order to obtain an optimal picture about the occupancy of the channels in this time-slot, and then access all the channels deemed to be idle by the reward maximization procedure detailed in Sec. II-D. The average throughput attained by the SU over T time-slots is given by

$$C^{\text{SU}} = \frac{1}{T} \sum_{i=1}^T \sum_{k=1}^K R_{\text{SU}} \mathcal{I} \left\{ \text{SINR}_{\text{SU}}(k, i) \geq 2^{\frac{R_{\text{SU}}}{W}} - 1 \right\}, \quad (28)$$

where $R_{\text{SU}}=0.6$ Mbps is the transmission rate of the SU on each channel, and \mathcal{I} is an indicator variable; and the throughput attained by the PUs in the network over the same T time-slots, normalized over time and the number of transmissions (normalization is necessary here because of the temporally intermittent transmissions of the PUs, the PUs are not always back-logged, unlike the SU) is given by

$$C^{\text{PUs}} = \frac{\sum_{i=1}^T \sum_{k=1}^K R_{\text{PU}} B_k(i) \mathcal{J} \left\{ \text{SINR}_{\text{PU}}(k, i) \geq 2^{\frac{R_{\text{PU}}}{W}} - 1 \right\}}{\sum_{i=1}^T \sum_{k=1}^K B_k(i)}, \quad (29)$$

where $R_{\text{PU}}=0.9$ Mbps is the transmission rate of the PUs on each channel, \mathcal{J} is an indicator variable, $\text{SINR}_{\text{PU}}(k, i)=\text{SINR}_{\text{PU}_j}(k, i)$, $j \in \{1, 2, \dots, J\}$ being the index of the PU occupying channel k in time-slot i , and $B_k(i)=1$ if channel k is occupied by an incumbent in time-slot i (note here that PUs do not interfere with each other because of OFDMA, i.e., due to the clearly laid out administrative guidelines about licensed frequency use for incumbents, so only one PU accesses a frequency band and that band would have been specifically licensed for that PU).

In the simulated radio environment described above, as illustrated in Fig. 4, we compare the performance of the proposed framework with the following state-of-the-art solutions detailed in current literature, in terms of the secondary network throughput achieved vis-à-vis incumbent interference:

- MEM with GC-CCE and MPE [15]: Minimum Entropy Merging (MEM) with Greedy Clustering based Channel Correlation Estimation (GC-CCE) and Markov Process Estimation (MPE), Correlation Threshold $\rho_{th}=0.7$, Number of clusters $T=6$, i.e., a channel sensing restriction of 6—our solution offers a 104% improvement over this strategy;
- MEM with MEI-CCE and MPE [15]: Minimum Entropy Merging (MEM) with Minimum Entropy Increment Clustering based Channel Correlation Estimation (MEI-CCE) and Markov Process Estimation (MPE), Correlation Threshold $\rho_{th}=0.7$, Number of clusters

$T=6$, i.e., a channel sensing restriction of 6--our solution achieves 38% better performance over this strategy;

- Imperfect HMM-MAP State Estimation [9]: The Viterbi algorithm, assuming a priori knowledge of the time-frequency Markovian correlation structure in incumbent occupancy behavior, with a channel sensing restriction of 6--our solution attains a 6% boost over this strategy;
- Neyman-Pearson Detection [13]: A Neyman-Pearson Detector, assuming independence across channels and across time, with no channel sensing restrictions, an AND fusion rule across 300 samplings, and threshold determination via a false alarm probability of 30%--our solution offers a 25% enhancement over this strategy;
- HMM EM + Fragmented PERSEUS with Belief Update Simplification: The proposed framework in practical settings – A Fragmented PERSEUS algorithm with Belief-Update Simplification (Hamming distance state filters), with no prior occupancy behavior correlation model information, instead, this model is learned over time, concurrently with the optimal channel sensing & access policy solver;
- Prior Perfect Model Knowledge + Fragmented PERSEUS with Belief Update Simplification: The proposed framework in ideal settings – A Fragmented PERSEUS algorithm with Belief-Update Simplification (Hamming distance state filters), with prior occupancy behavior correlation model information--the proposed HMM EM + Fragmented PERSEUS with Hamming State Filters exhibits 3.75% worse performance than this strategy, i.e., knowing the model beforehand offers a meagre 3.75% boost in performance compared to the proposed online concurrent model estimation and optimal policy solver strategy – a testament to the accuracy of our estimator;
- Temporal Difference Learning via SARSA with Linear Function Approximation [8]: TD-SARSA with Linear Function Approximation (LFA) in single-agent deployment settings, with a sensing restriction of 6, a belief update heuristic constant $\lambda=0.9$, a discount factor of $\gamma=0.9$, a fixed exploration factor $\epsilon=0.01$, and a raw false alarm probability of $p_{fa,1}=5\%$ --our solution exhibits a 3% superior performance over this strategy;
- Greedy Learning under Pre-Allocation [17]: Greedy Learning in single-agent deployment settings, with a channel sensing restriction of 6, and a time-varying exploration factor $\epsilon=\min(\frac{\beta}{i}, 1)$, where $\beta>\max(20, \frac{4}{\Delta_{\min}^2})$, with Δ_{\min} referring to the smallest Kullback-Liebler distance between a pair of channels--our solution offers a 10% enhancement over this

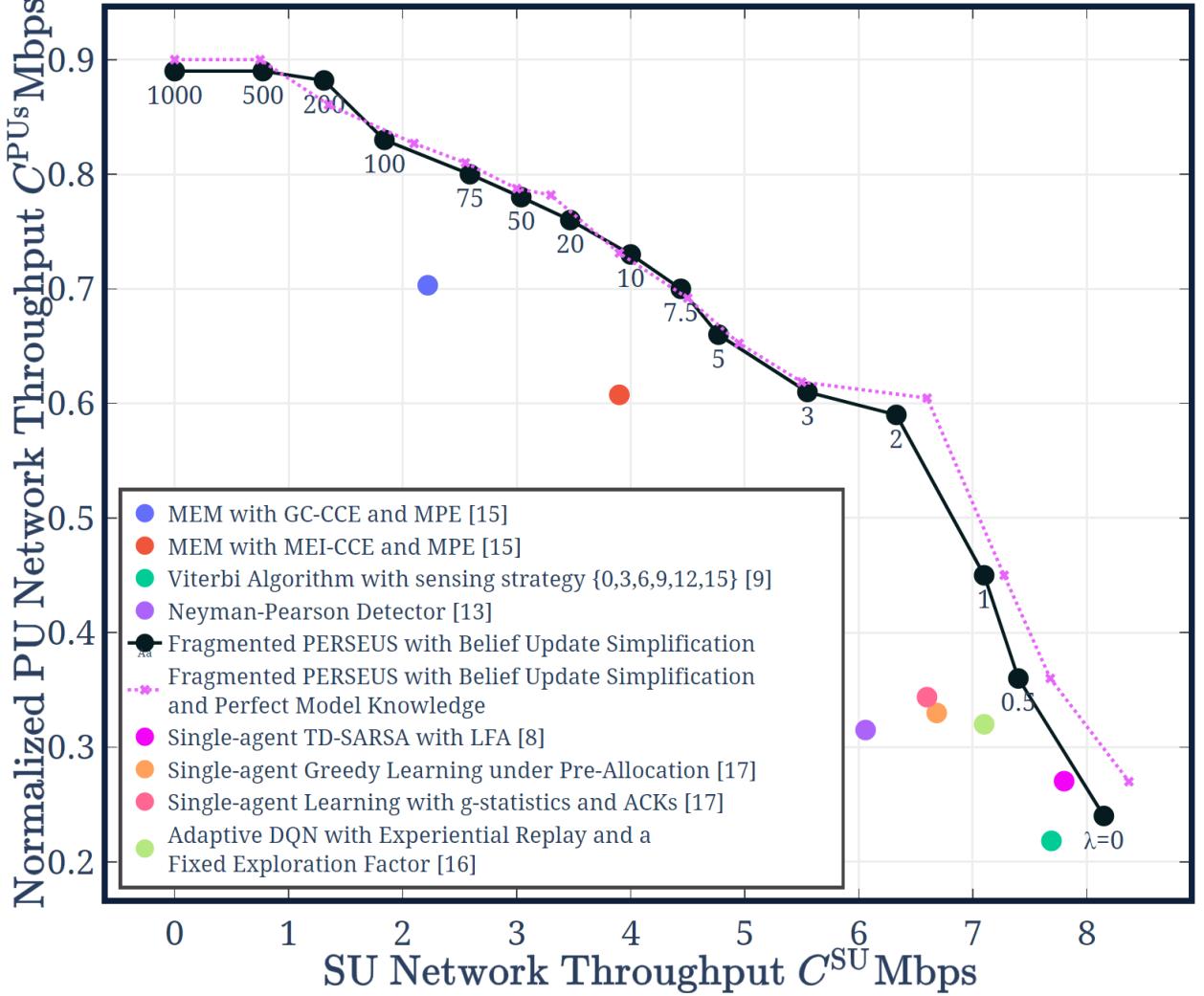


Fig. 4. The evaluation of SU and PU network throughputs for different values of λ , along with comparisons with the state-of-the-art

strategy;

- g-statistics [17]: Learning with g-statistics and ACKs in single-agent deployment settings, with a channel sensing restriction of 6--our solution achieves a 15% boost in performance over this strategy;
- Adaptive Deep Q-Networks [16]: An adaptive Deep Q-Network (DQN) with Experiential Replay (Memory Size $C=10^6$), 2048 input neurons, 4096 neurons with ReLU activation functions in each of the 2 hidden layers of the Neural Network, a Mean-Squared Error cost function with an Adam Optimizer, a Fixed Exploration Factor $\epsilon=0.1$, a Learning Rate of $\alpha=10^{-4}$, a Batch Size of $W=32$, and a sensing restriction of 6--our solution offers a 9% improvement over this strategy.

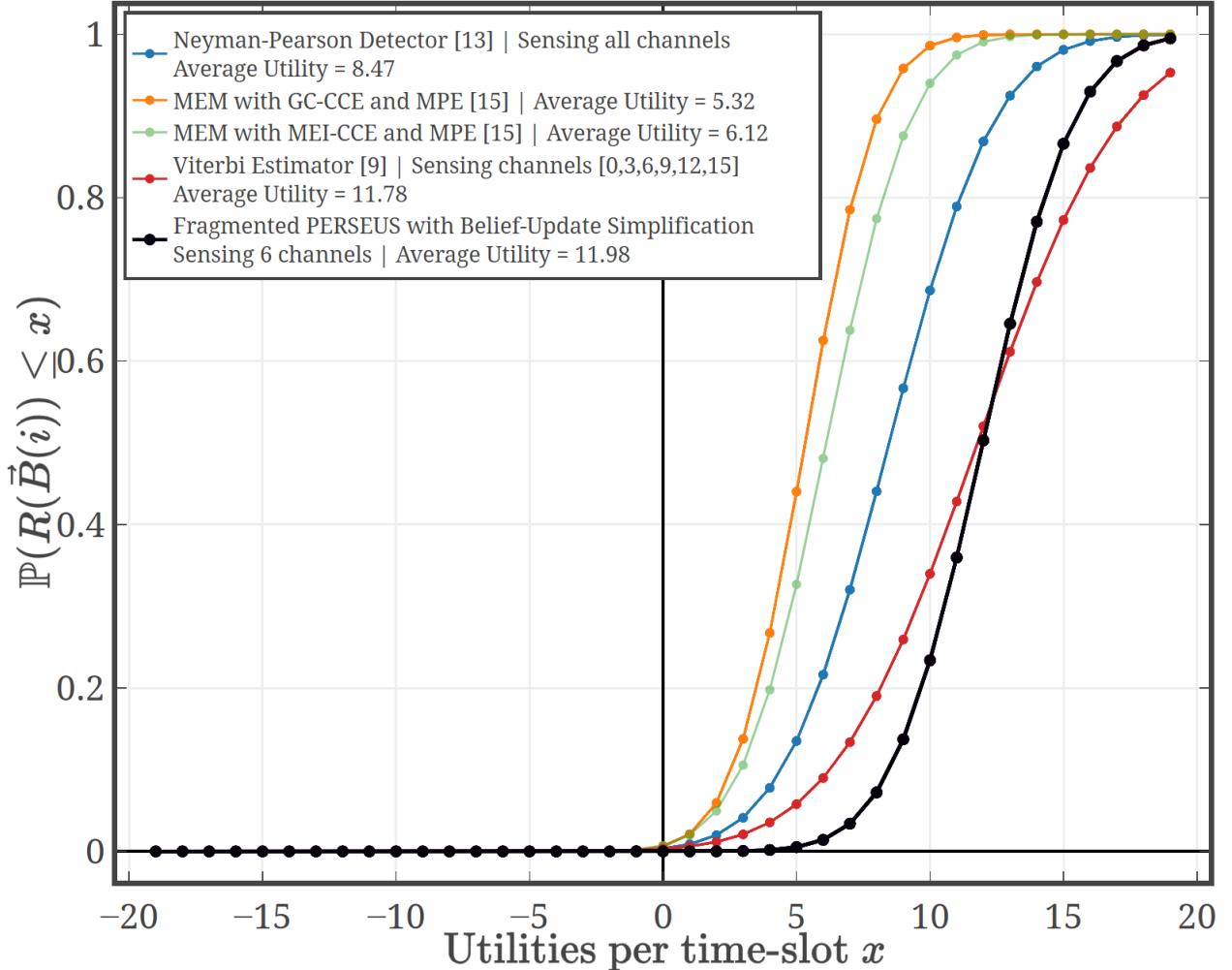


Fig. 5. The evaluation of the proposed solution, from an average utility per time-slot perspective, against a medley of approaches in the state-of-the-art: $\mathbb{P}(R(\vec{B}(i)) \leq x)$ versus utility per time-slot x

Analyzing the performance of our POMDP solution from a different perspective, we find that, as illustrated in Fig. 5, our framework obtains an average utility, i.e., $R(\vec{\phi}(i), \hat{\beta}_i)$ described in Sec. II-D, of 11.98 per time-step i , 125% higher than that achieved by the MEM with GC-CCE and MPE algorithm from [15], 96% higher than that achieved by the MEM with MEI-CCE and MPE algorithm from [15], and 42% more than that attained by the Neyman-Pearson Detector detailed above [13]. Furthermore, in order to understand how our framework performs against a standard HMM state estimation solution like the Viterbi algorithm described earlier [9], especially, one with a priori transition model information and one that senses a maximum of 6 channels per time-slot (channel sensing restriction of 6), we compare our solution with this Viterbi agent, and find that the average utility per time-slot obtained by the Viterbi agent ($=11.78$) is 2% lower than ours ($=11.98$).

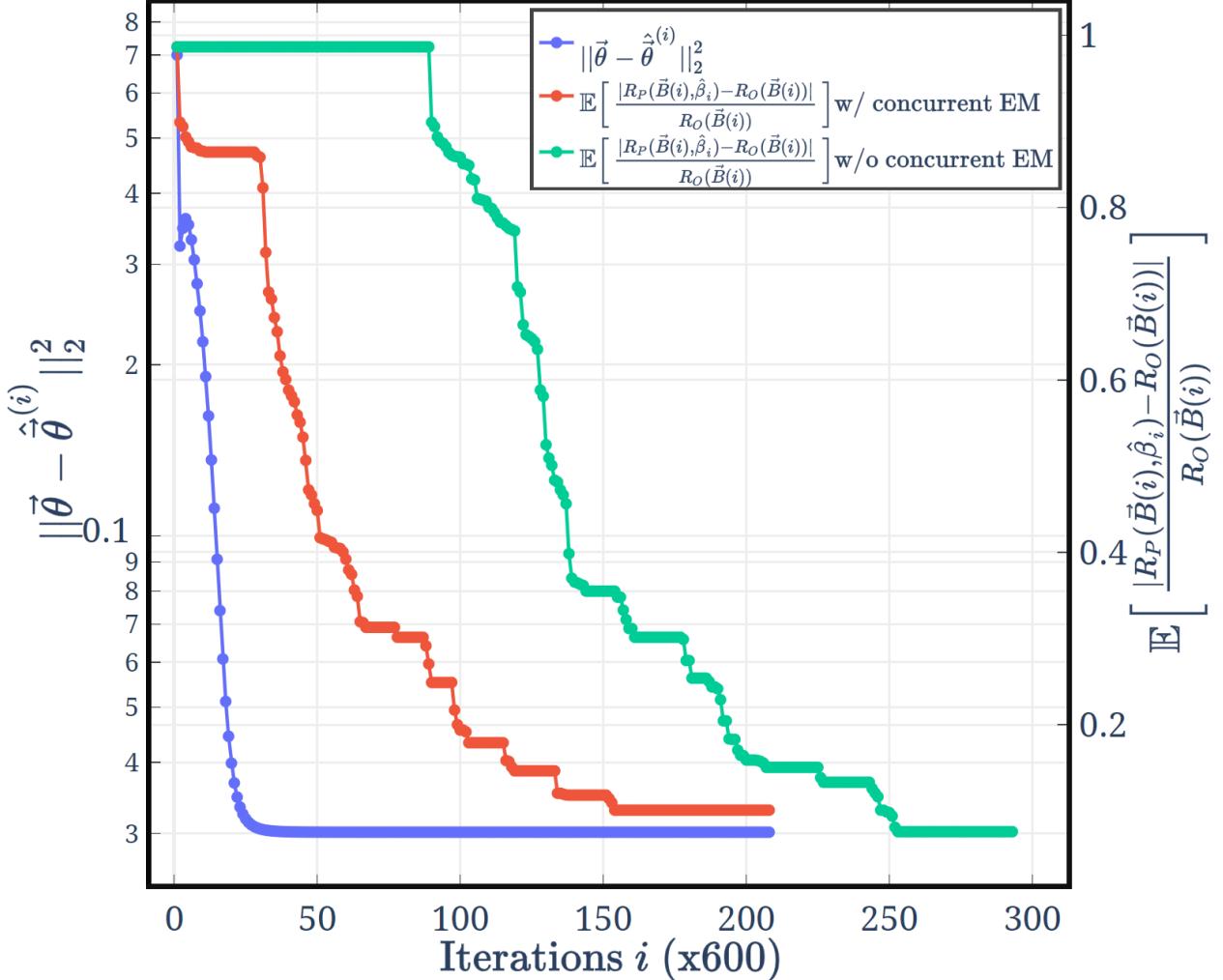


Fig. 6. The convergence of the MSE of the HMM EM algorithm to estimate $\vec{\theta}$, and the convergence of the loss of the fragmented PERSEUS algorithm with belief update simplification

Specifically discussing the performance of the parameter estimation algorithm, i.e., the HMM EM algorithm (Baum-Welch), we find that, with initial estimates of 0.5, i.e., $p_{uv}=0.5, \forall u, v \in \{0, 1\}$ and $q_w=0.5, w \in \{0, 1\}$, the estimator converges to the true parameter vector $\vec{\theta}$ with an error/delta of $\eta=10^{-8}$ in 45,000 iterations: this corresponds to an observation and estimation period of 135 s, considering a typical time-slot duration of 3 ms. We illustrate this convergence via the Mean Square Error (MSE) plot depicted in Fig. 6, in which the MSE in iteration t given by,

$$\|\vec{\theta} - \hat{\vec{\theta}}^{(t)}\|_2^2 = \sum_{\theta \in \vec{\theta}} \mathbb{E}[(\theta - \hat{\theta}^{(t)})^2] \quad (30)$$

is decreased iteratively, as the estimation process goes through the E-step and the M-step in each iteration t until $\mathbb{E}[\theta - \hat{\theta}^{(t)}] \leq 10^{-8}, \forall \theta \in \vec{\theta}$.

On the same time-scale as the parameter estimation algorithm, focusing on the loss convergence of the PERSEUS algorithm with a discount factor of $\gamma=0.9$ and a termination threshold of $\epsilon=10^{-5}$, wherein we define the expected loss as the difference between the utility obtained by the proposed PERSEUS framework, denoted by $R_P(\vec{B}(i))$ (discussed in Sec. II-D), and that obtained by an Oracle, which knows the exact occupancy behavior of the incumbents in the network, denoted by $R_O(\vec{B}(i))$, we find that, as depicted in Fig. 6, the loss convergence of PERSEUS is relatively slower while the parameter estimator is learning the transition model; as opposed to after the convergence of the parameter estimator, when we see a more consistent gradient towards the optimality. Also, note the normalized sub-optimality gap of 0.05, i.e., the average normalized difference between the utility obtained by our optimal POMDP policy (post-convergence) and the utility obtained by the Oracle (which knows the exact incumbent occupancy behavior) is 0.05. Moreover, Fig. 6 depicts the computational time difference between running the parameter estimator and the PERSEUS algorithm concurrently via the iterative publisher-subscriber architecture, as opposed to initiating the PERSEUS run after the convergence of the parameter estimator: we cut down the time to completion of our HMM-POMDP framework by half by employing the former approach as opposed to the latter, without worsening the sub-optimal gap significantly.

Finally, inspecting Fig. 4 in a new light, we see that our POMDP agent limits channel access when the penalty (λ) is high, leading to lower SU throughput and lower PU interference, and conversely, follows a more lenient channel access strategy when the penalty is low, resulting in higher SU throughput and higher PU interference. Generally speaking, Fig. 4 depicts a trend of increasing SU throughput and increasing incumbent interference, as the penalty for missed detections, i.e., λ is lowered. Therefore, our framework provides a crucial practical tool in cognitive radio MAC design: the ability to tune the trade-off between the throughput obtained by the cognitive radio and the interference caused by it to incumbent transmissions in the network.

V. MULTI-AGENT DEPLOYMENT MODEL: AN EXTENSION TO THE SINGLE-AGENT SETTING

A. *Distributed Multi-Agent Spectrum Sensing and Access*

In this section, we evaluate the performance of the proposed framework: HMM EM + Fragmented PERSEUS with Belief Update Simplification, in distributed multi-agent deployment settings. Operating under the same signal and observation models as in Sec. II, consider a network of 3 PUs operating in an 18-channel radio environment, with their occupancy behaviors in this

discretized spectrum of interest governed by Markovian time-frequency correlation structure ($\vec{\theta}$), and 12 SUs intelligently trying to access white-spaces in the spectrum (cooperatively [8] or opportunistically [17]), with an added restriction of being able to sense only 1 channel per SU per time-slot, as illustrated in Fig. 1.

The POMDP model described in Sec. II-D has been adapted to this multi-agent deployment setting by incorporating neighbor discovery, channel access rank allocation, and data aggregation algorithms into the original POMDP process flow, as depicted in Fig. 3. Designating the band-edges as the control channel, for neighbor discovery, each cognitive radio node broadcasts its control frames (with a frame header and node identifier) over the control channel, and upon receiving control messages from all its surrounding nodes, each cognitive radio node checks if the expected RSSI of the radio signals corresponding to a certain node is above a threshold RSSI_{th} : if yes, adds that node's identifier to its list of neighbors. With a similar control channel strategy for channel access rank allocation, we employ a quorum-based preferential ballot voting scheme to determine the order in which the "estimated-idle" channels are accessed by the SUs in the network. This procedure kicks in only after a quorum has been achieved, i.e., the number of neighbors identified by an SU should be equal to or exceed a node-specific pre-defined number. Over the control channel, each cognitive radio exchanges a ranked list of its neighbors in the decreasing order of their respective RSSIs, with itself being on the list at position-1 (ties are broken via uniform random choice). Upon receiving an "RSSI-ranked" list from one of its neighbors, each cognitive radio node assigns points to each ranked position, with higher ranks getting larger point values, and re-broadcasts an "aggregated-ranked" list of neighbors (with itself being on the list) with the ranking based on the point-values aggregated across all the ranked lists received from its neighbors (ties are broken via uniform random choice). If the "aggregated-ranked" lists received from its neighbors matches the one at the SU, and this is true for a pre-specified consecutive period of time, a consensus has been reached, the channel access order is determined by this "harmonized-aggregated-ranked" list. If the "aggregated-ranked" lists received from its neighbors differ from the one at the SU, then the SU repeats the re-ranking of these list members based on their new aggregated point-values and broadcasts the new "aggregated-ranked" list to its neighbors over the control channel. This repetitive process continues until a consensus is reached.

Analyzing the performance of the proposed framework (HMM EM + Fragmented PERSEUS with Belief-Update Simplification) against other distributed multi-agent schemes in the state-

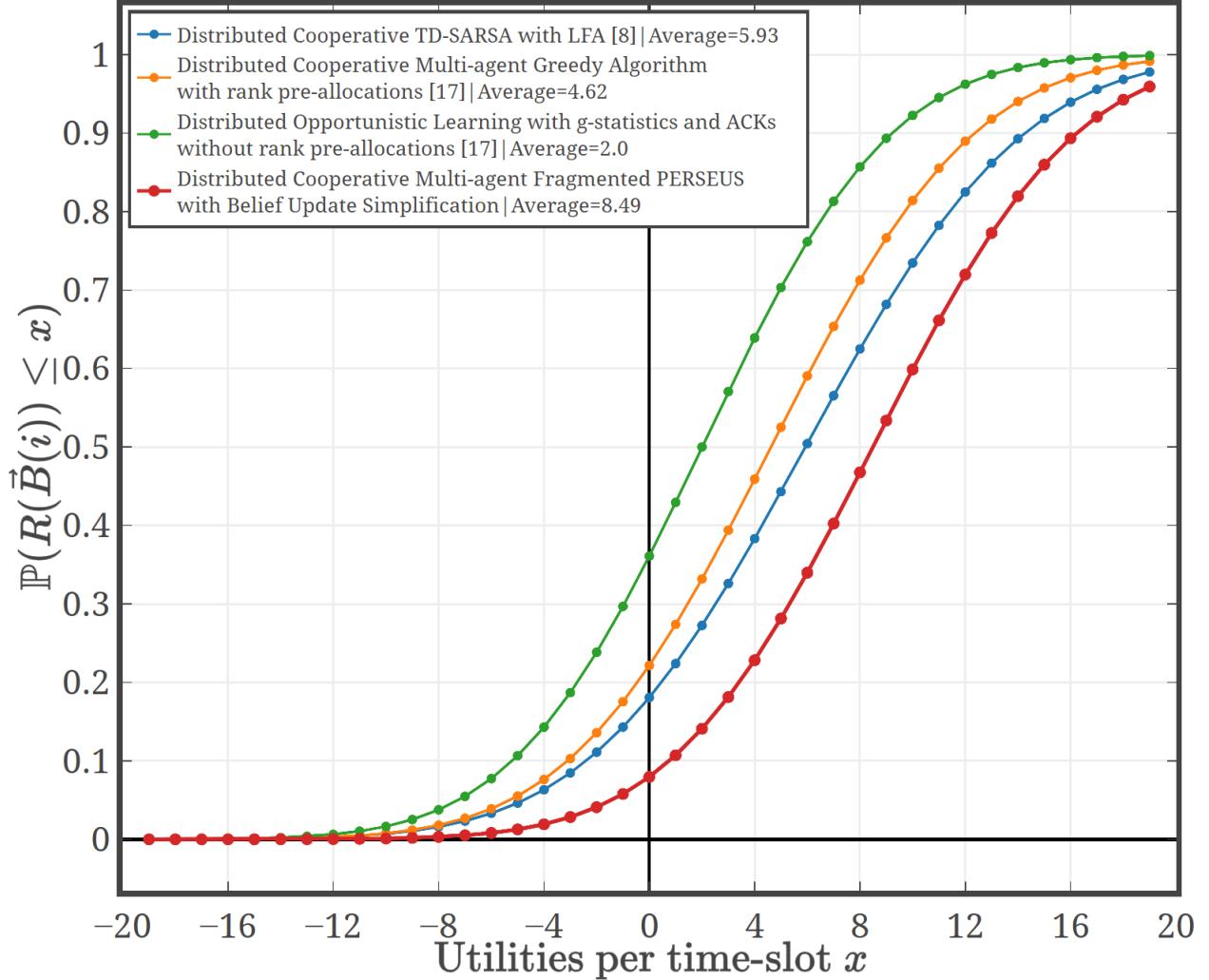


Fig. 7. An evaluation of the performance (average utility per time-slot) of the proposed framework in a distributed multi-agent deployment setting, against other distributed cooperative/opportunistic multi-agent channel sensing & access frameworks in the state-of-the-art: $\mathbb{P}(R(\vec{\beta}(i)) \leq x)$ versus utility per time-slot x

of-the-art, as shown in Fig. 7, we find that our framework, in terms of the average utility $R(\vec{\phi}(i), \hat{\beta}(i))$ obtained per time-slot, out-performs the distributed, cooperative, ϵ -greedy TD-SARSA with Linear Function Approximation framework from [8] by 43%; out-performs the distributed, cooperative, time-decaying ϵ -greedy algorithm with channel access rank pre-allocations from [17] by 84%; and out-performs the distributed, opportunistic, g-statistics algorithm with ACKs (without channel access rank pre-allocations) from [17] by 324%.

B. Centralized Multi-Agent Spectrum Sensing and Access: SC2 Active Incumbent Emulation

In order to evaluate the performance of the proposed framework (HMM EM + Fragmented PERSEUS with Belief Update Simplification) in real-world settings, we retrofit it into the

MAC layer (channel & bandwidth allocation) of our BAM! Wireless radio [18], and analyze its operational capabilities in the DARPA SC2 Active Incumbent scenario [21] emulated on the Colosseum [20], [31]. The DARPA SC2 Active Incumbent scenario consists of a Terminal Doppler Weather Radar (TDWR) system functioning as the PU, and 5 competitor networks (ours included), each constituting 2 UNII WLANs: 2 Access Points (APs) and 4 STAs (STAs) per AP, serving as the SUs, in a 10 MHz radio environment (995 MHz to 1005 MHz), for 330 seconds of emulation on the Colosseum [21].

During the Active Incumbent scenario emulation, every competitor network receives network flows from the Colosseum which need to be delivered to the appropriate destination nodes within the network, while satisfying the imposed QoS mandates per flow (for example: max_latency, min_throughput, file_transfer_deadline, etc.). If the QoS mandates imposed on a particular network flow have been satisfied for a pre-specified period of time (referred to as "Measurement Periods" (MPs)), then the Individual Mandates (IMs) associated with the flow are said to have been met. With this concept of IMs in mind, we can define the points achieved or the "score" of a participant network corresponding to a certain time-slot i as $\sum_{v \in \mathcal{V}_i} p_v$, where \mathcal{V}_i denotes the set of IMs achieved by a participant network in time-slot i . The scenario also incorporates ensemble performance thresholds, i.e., all the participant networks should meet the scoring threshold of 8 [21]: if a participant network fails to meet this threshold, all the participant networks get the lowest score, i.e., the score corresponding to that achieved by this under-performing network, else, if all the participant networks in the emulation achieve scores that exceed the threshold, their scores are incremented beyond this threshold commensurate with the IMs achieved by them in that time-slot.

After having understood the scoring mechanism involved in the DARPA SC2, we can now evaluate the performance of the proposed framework retrofitted into our standard BAM! Wireless radio [18] against other radios designed by our peers who also participated in this competition, in addition to a performance comparison with the weighted PSD + CIL [22] channel & bandwidth allocation scheme employed, as a standard out-of-the-box protocol, in our traditional BAM! Wireless network. Leveraging the aggregated PSD measurements obtained at the gateway node of our BAM! Wireless network, as shown in Fig. 2 (R), we evaluate the scores of the proposed framework retrofitted into our standard BAM! Wireless radios against our traditional channel & bandwidth allocation scheme (titled "Standard BAM! Wireless Radio [Purdue]), and against the designs of our peers (identified by their collaboration network registered IP address [22],

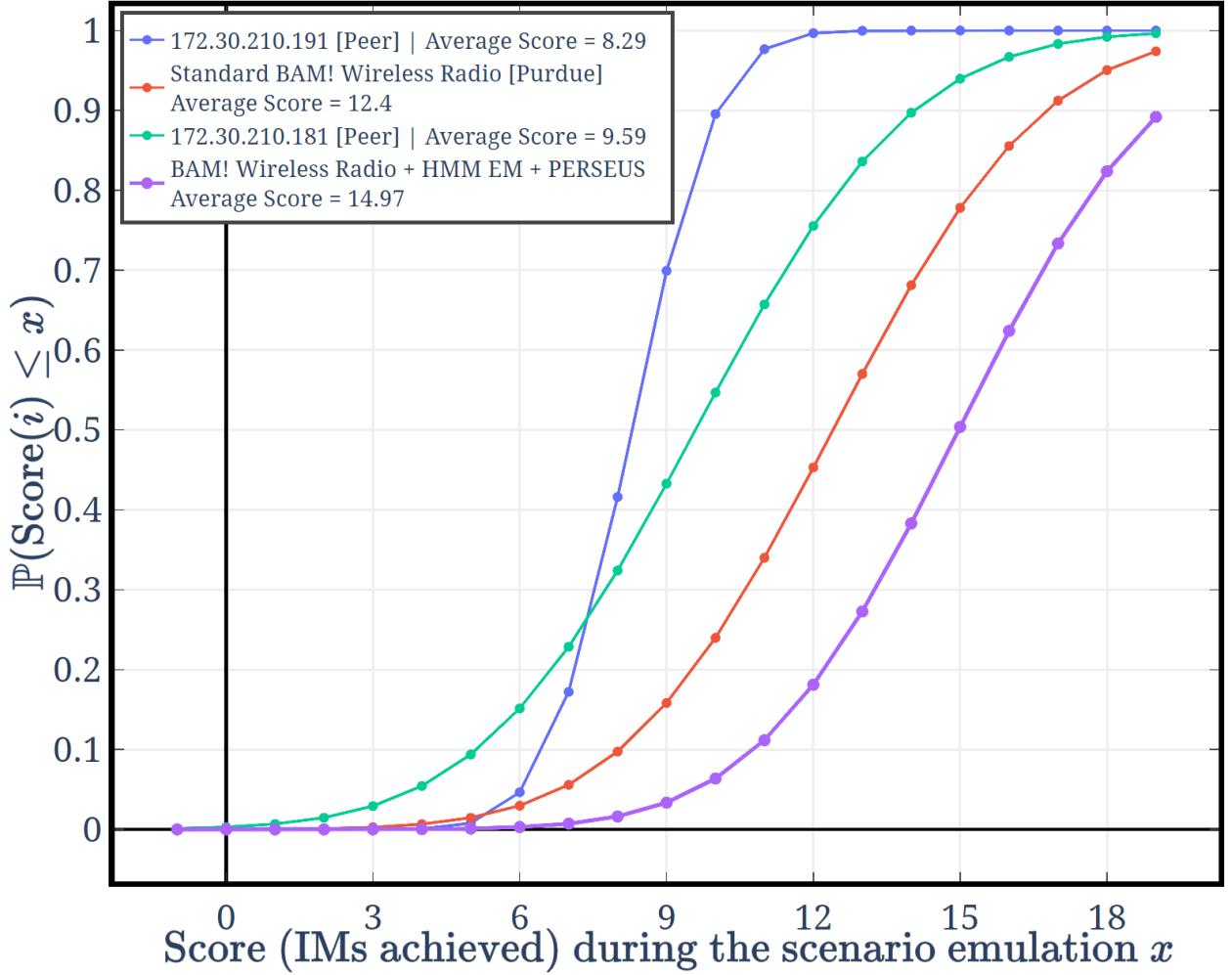


Fig. 8. An evaluation of the performance (scores/Individual Mandates (IMs) achieved) of our solution by retrofitting the proposed POMDP framework into our BAM! Wireless cognitive radio network design, with respect to an emulation of the DARPA SC2 Active Incumbent scenario, against other competitor network radio designs: $\mathbb{P}(\text{Score} \leq x)$ versus the scores achieved during the course of this emulation x

"172.30.210.191 [Peer]" and "172.30.210.181 [Peer]"): in terms of the average score achieved per time-slot, we deduce from Fig. 8 that the proposed framework ("BAM! Wireless Radio + HMM EM + Fragmented PERSEUS with Belief Update Simplification") out-performs our traditional channel & bandwidth allocation scheme (a simple weighted PSD + CIL heuristic) by 21%; provides a 56% better performance than one of our peers, identified by "172.30.210.181"; and attains an 81% boost in performance over another one of our peers, identified by "172.30.210.191".

C. Feasibility Analysis of the Distributed Multi-Agent POMDP Optimal Policy on ESP32 Radios

We employ 8 ESP32 radios [28], with each one embedded in a GCTronic e-puck2 robot [27], categorized into a network of 3 PUs (and their 3 corresponding sinks) occupying 6 channels in the

discretized spectrum of interest according to a Markovian time-frequency correlation structure (described by (6)), and 2 independent SUs, with each having the capability of sensing only one channel at a time, intelligently trying to exploit the white-spaces in the spectrum. The detailed methodology of this implementation is provided below:

- Considering a network with $J=3$ PUs and one SU (work split over 2 ESP32 radios due to design limitations) with a channel sensing restriction of $\kappa=2$ out of $K=6$ channels in the discretized spectrum of interest, and assuming a linear AWGN observation model, with a Rayleigh channel fading model (discussed in Sec. II-A), we simulate the occupancy behavior of the PUs according to a Markovian time-frequency correlation structure parameterized by $\vec{\theta}=[\vec{p}, \vec{q}]^\top$, where $\vec{p}=[p_{00}=0.1, p_{01}=0.3, p_{10}=0.3, p_{11}=0.7]^\top$ and $\vec{q}=[q_0=0.3, q_1=0.8]^\top$; and solve for the optimal spectrum sensing and access policy using PERSEUS, embedded with a concurrent parameter estimation algorithm learning the parameter vector $\vec{\theta}$, by mimicking the observational capabilities of the actual ESP32 radios. Note this step is performed on a PC.
- The simulated PU occupancy behavior, Markovian correlated according to (6) and parameterized by $\vec{\theta}$, and the time-slot specific optimal channel access decisions (derived off of the POMDP optimal sensing policy and the simulated PU occupancy behavior), are stored in databases (for export onto the ESP32 network).
- Peer-to-Peer communication links are established between a PU ESP32 radio and its sink, using the 3 ESP32 radios designated as PUs. In other words, 3 wireless communication links are established: one for each ESP32 PU pair (a source and a sink), over WiFi (2.4 GHz) and using a channel according to the occupancy information detailed in the exported PU occupancy database, in time-slot i .
- Note here that in this ESP32 PU network implementation, in time-slot i , while establishing a wireless communication link between a ESP32 PU $j \in \{1, 2, 3\}$ and its respective sink $i \in \{1, 2, 3\}$ s.t. i is the designated sink for PU j , i.e., while forming link l_{ij} over channel $k_{l_{ij}}=k \in \{1, 2, \dots, 6\}$ (as determined by the exported PU occupancy database which contains simulated PU occupancy behavior according to the Markovian time-frequency correlation structure described above) such that $k_{l_{ij}} \neq k_{l_{i',j'}}, \forall i, i' \in \{1, 2, 3\}, j, j' \in \{1, 2, 3\}$, PU j serves as an Access Point (AP) accepting transmission requests from PU i , which is designated as a STATION (STA). In the next synchronized time-slot $i + 1$, this link l_{ij} moves to channel

$k' \in \{1, 2, \dots, 6\}$, as detailed in the exported PU occupancy database. This same procedure takes place for the other two incumbent communication links in every time-slot until the end of the implementation evaluation period.

- Although the PC-based POMDP solver employs an SU which can access 2 channels at a time in order to deliver its flows (see the access part of the POMDP formulation in Sec. II-D), we employ 2 ESP32 SU radios in the network (serving as one), with the channel access work synchronously and evenly split between the two, due to the actual physical design limitations of the ESP32 radio that it can only access one channel at a time, forcing us to be creative: split the optimal 2 channel access decision in time-slot i , as determined by the time-slot specific optimal POMDP channel access database, into a 1 channel access action at each ESP32 SU radio. Next, based on whether the channel access at the 2 ESP32 SU radios was successful, we compute the success rate.

The channel access success rate metric defined as

$$\text{Channel Access Success Probability} = \frac{\sum_{j=1}^2 \mathcal{I} \left\{ B_{k_{SU_j}}(i) = 0 \right\}}{2}, \quad (31)$$

where \mathcal{I} corresponding to $\mathcal{I} \left\{ B_{k_{SU_j}}(i) = 0 \right\}$ is an indicator variable whose value is 1 if the channel accessed by the ESP32 SU $j \in \{1, 2\}$ in time-slot i is not occupied by an incumbent PU ESP32 radio, and $B_{k_{SU_j}} \in \{0, 1\}$ is the occupancy variable of the channel accessed by the ESP32 SU j in time-slot i , is evaluated per time-slot i , and the resultant channel access success probability is 95.75%.

VI. CONCLUSION

In this paper, we formulate the optimal spectrum sensing and access problem as an approximate POMDP, which leverages learning of the spectrum occupancy correlation model of the incumbents via the Baum-Welch algorithm. Through system simulations, we demonstrate the advantages of exploiting the correlation structure--as opposed to Neyman-Pearson Detector which assumes independence--and of adapting the spectrum sensing decision to optimize the performance--as opposed to Viterbi, which uses a fixed sensing strategy. We also demonstrate the feasibility of a concurrent learning and decision-making framework, as opposed to state-of-the-art correlation-coefficient based clustering algorithms, which rely on pre-loaded datasets for determining the correlation in the incumbent occupancy behavior. Our framework enables a critical feature in practical scenarios: the ability of the SU to regulate the interference caused to

PUs, by adjusting a penalty parameter. Also, extending our single-agent model to multi-agent settings, we demonstrate superior performance over the state-of-the-art, in both centralized and distributed deployment settings (collaborative and opportunistic access).

REFERENCES

- [1] B. Keshavamurthy and N. Michelusi, “Learning-based Cognitive Radio Access via Randomized Point-Based Approximate POMDPs,” 2020, Under review at IEEE ICC 2021.
- [2] Ericsson, “5G use cases—Explore how 5G will revolutionize 5 key industries including: TV and media; manufacturing; healthcare; telecommunications; and transportation and infrastructure.” *Ericsson*, 2019. [Online]. Available: <https://www.ericsson.com/en/5g/use-cases>
- [3] D. Goldin, “Keep 5G Safe From Chinese Domination,” *The Wall Street Journal*, 2020. [Online]. Available: <https://www.wsj.com/articles/keep-5g-safe-from-chinese-domination-11580342112>
- [4] D. Fitzgerald, “Wireless Companies Share the Airwaves,” *The Wall Street Journal*, 2019. [Online]. Available: <https://www.wsj.com/articles/wireless-companies-share-the-airwaves-11577538001>
- [5] H. W. Jenkins Jr., “How Government Can Get Brave About Spectrum,” *The Wall Street Journal*, 2019. [Online]. Available: <https://www.wsj.com/articles/how-government-can-get-brave-about-spectrum-11560552078>
- [6] S. Maleki, S. P. Chepuri, and G. Leus, “Energy and throughput efficient strategies for cooperative spectrum sensing in cognitive radios,” in *2011 IEEE 12th International Workshop on Signal Processing Advances in Wireless Communications*, June 2011, pp. 71–75.
- [7] K. Cohen, Q. Zhao, and A. Scaglione, “Restless Multi-Armed Bandits under time-varying activation constraints for dynamic spectrum access,” in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Nov 2014, pp. 1575–1578.
- [8] J. Lundén, S. R. Kulkarni, V. Koivunen, and H. V. Poor, “Multiagent Reinforcement Learning Based Spectrum Sensing Policies for Cognitive Radio Networks,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 5, pp. 858–868, Oct 2013.
- [9] C. Park, S. Kim, S. Lim, and M. Song, “HMM Based Channel Status Predictor for Cognitive Radio,” in *2007 Asia-Pacific Microwave Conference*, Dec 2007, pp. 1–4.
- [10] L. Ferrari, Q. Zhao, and A. Scaglione, “Utility Maximizing Sequential Sensing Over a Finite Horizon,” *IEEE Transactions on Signal Processing*, vol. 65, no. 13, pp. 3430–3445, July 2017.
- [11] N. Michelusi and U. Mitra, “Cross-Layer Estimation and Control for Cognitive Radio: Exploiting Sparse Network Dynamics,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 1, no. 1, pp. 128–145, March 2015.
- [12] N. Michelusi, M. Nokleby, U. Mitra, and R. Calderbank, “Multi-Scale Spectrum Sensing in Dense Multi-Cell Cognitive Networks,” *IEEE Transactions on Communications*, vol. 67, no. 4, pp. 2673–2688, April 2019.
- [13] S. Mosleh, A. A. Tadaion, and M. Derakhtian, “Performance analysis of the Neyman-Pearson fusion center for spectrum sensing in a Cognitive Radio network,” in *IEEE EUROCON 2009*, May 2009, pp. 1420–1425.
- [14] S. Yin, D. Chen, Q. Zhang, M. Liu, and S. Li, “Mining Spectrum Usage Data: A Large-Scale Spectrum Measurement Study,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 6, pp. 1033–1046, June 2012.
- [15] M. Gao, X. Yan, Y. Zhang, C. Liu, Y. Zhang, and Z. Feng, “Fast Spectrum Sensing: A Combination of Channel Correlation and Markov Model,” in *2014 IEEE Military Communications Conference*, Oct 2014, pp. 405–410.
- [16] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, “Deep Reinforcement Learning for Dynamic Multichannel Access in Wireless Networks,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 4, no. 2, pp. 257–265, 2018.

- [17] A. Anandkumar, N. Michael, and A. Tang, “Opportunistic Spectrum Access with Multiple Users: Learning under Competition,” in *Proceedings of the 29th Conference on Information Communications*, ser. INFOCOM’10. IEEE Press, 2010, p. 803–811.
- [18] DARPA, “Purdue University BAM! Wireless Radio,” *Defense Advanced Research Projects Agency (DARPA) Spectrum Collaboration Challenge (SC2)*, 2019. [Online]. Available: <https://archive.darpa.mil/sc2/news/spectrum-collaboration-challenge-awards-four-teams-with-half-prizes>
- [19] M. Rosker, “Spectrum Collaboration Challenge (SC2),” *Defense Advanced Research Projects Agency (DARPA) Spectrum Collaboration Challenge (SC2)*, 2018. [Online]. Available: <https://www.darpa.mil/program/spectrum-collaboration-challenge>
- [20] DARPA, “Scenarios Summary List,” *Defense Advanced Research Projects Agency (DARPA) Spectrum Collaboration Challenge (SC2)*, 2019. [Online]. Available: <https://sc2colosseum.freshdesk.com/support/solutions/articles/22000236679--scenarios-summary-list-phase-3->
- [21] DARPA, “Active Incumbent Scenario Specifications,” *DARPA Spectrum Collaboration Challenge (SC2)*, 2019. [Online]. Available: <https://sc2colosseum.freshdesk.com/support/solutions/articles/22000239489-active-incumbent->
- [22] DARPA SC2 GitLab CIL Schematics, “CIL Specifications,” *DARPA Spectrum Collaboration Challenge (SC2)*, 2019. [Online]. Available: <https://gitlab.com/darpa-sc2-phase3/CIL>
- [23] F. A. P. d. Figueiredo, D. Stojadinovic, P. Maddala, R. Mennes, I. Jabandžić, X. Jiao, and I. Moerman, “SCATTER PHY: A Physical Layer for the DARPA Spectrum Collaboration Challenge,” in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.
- [24] R. J. Baxley and R. S. Thompson, “Team Zylinium DARPA Spectrum Collaboration Challenge Radio Design and Implementation,” in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.
- [25] D. Stojadinovic, F. A. P. de Figueiredo, P. Maddala, I. Seskar, and W. Trappe, “SC2 CIL: Evaluating the Spectrum Voxel Announcement Benefits,” in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.
- [26] S. Giannoulis, C. Donato, R. Mennes, F. A. P. de Figueiredo, I. Jabandžić, Y. De Bock, M. Camelo, J. Struye, P. Maddala, M. Mehari, A. Shahid, D. Stojadinovic, M. Claeys, F. Mahfoudhi, W. Liu, I. Seskar, S. Latre, and I. Moerman, “Dynamic and Collaborative Spectrum Sharing: The SCATTER Approach,” in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.
- [27] GCTronic, “Epuck2 Specifications and General Wiki,” *GCTronic e-puck2 online wiki*, 2020. [Online]. Available: <https://www.gctronic.com/doc/index.php/e-puck2>
- [28] Espressif Systems (Shanghai) Co. Ltd., “Espressif ESP32: A Different IoT Power and Performance,” *Espressif*, 2019. [Online]. Available: <https://www.espressif.com/en/products/hardware/esp32/overview>
- [29] R. I. C. Chiang, G. B. Rowe, and K. W. Sowerby, “A Quantitative Analysis of Spectral Occupancy Measurements for Cognitive Radio,” in *2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring*, April 2007, pp. 3016–3020.
- [30] M. A. McHenry, P. A. Tenhula, D. McCloskey, D. A. Roberson, and C. S. Hood, “Chicago Spectrum Occupancy Measurements & Analysis and a Long-term Studies Proposal,” in *Proceedings of the First International Workshop on Technology and Policy for Accessing Spectrum*, ser. TAPAS ’06. New York, NY, USA: ACM, 2006. [Online]. Available: <http://doi.acm.org/10.1145/1234388.1234389>
- [31] DARPA, “Radio Command and Control API,” *Defense Advanced Research Projects Agency (DARPA) Spectrum Collaboration Challenge (SC2)*, 2018. [Online]. Available: <https://sc2colosseum.freshdesk.com/support/solutions/articles/22000220460-radio-command-and-control-c2-api>

- [32] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, “Planning and Acting in Partially Observable Stochastic Domains,” *Artif. Intell.*, vol. 101, no. 1–2, p. 99–134, May 1998.
- [33] M. T. J. Spaan and N. A. Vlassis, “Perseus: Randomized Point-based Value Iteration for POMDPs,” *CoRR*, vol. abs/1109.2145, 2011. [Online]. Available: <http://arxiv.org/abs/1109.2145>
- [34] S. Giannoulis, C. Donato, R. Mennes, F. A. P. de Figueiredo, I. Jabandžić, Y. De Bock, M. Camelo, J. Struye, P. Maddala, M. Mehari, A. Shahid, D. Stojadinovic, M. Claeys, F. Mahfoudhi, W. Liu, I. Seskar, S. Latre, and I. Moerman, “Dynamic and Collaborative Spectrum Sharing: The SCATTER Approach,” in *2019 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, 2019, pp. 1–6.
- [35] W. Turin, “MAP decoding using the EM algorithm,” in *1999 IEEE 49th Vehicular Technology Conference (Cat. No.99CH36363)*, vol. 3, May 1999, pp. 1866–1870 vol.3.
- [36] J. Pineau, G. Gordon, and S. Thrun, “Point-Based Value Iteration: An Anytime Algorithm for POMDPs,” in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, ser. IJCAI’03. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, p. 1025–1030.