AUTHOR QUERIES

AUTHOR PLEASE ANSWER ALL QUERIES

PLEASE NOTE: We cannot accept new source files as corrections for your paper. If possible, please annotate the PDF proof we have sent you with your corrections and upload it via the Author Gateway. Alternatively, you may send us your corrections in list format. You may also upload revised graphics via the Author Gateway.

Carefully check the page proofs (and coordinate with all authors); additional changes or updates WILL NOT be accepted after the article is published online/print in its final form. Please check author names and affiliations, funding, as well as the overall article for any errors prior to sending in your author proof corrections. Your article has been peer reviewed, accepted as final, and sent in to IEEE. No text changes have been made to the main part of the article as dictated by the editorial level of service for your publication.

AQ1: According to our records, Bharath Keshavamurthy is listed as a Graduate Student Member, IEEE. However, the files provided list them as a Student Member, IEEE. Please verify. Yes, Bharath Keshavamurthy is an IEEE Graduate Student Member (#95181635 | 2019 - present).

AQ2: Please confirm or add details for any funding or financial support for the research of this article. Confirmed: The funding details in this PDF are accurate.

AQ3: Note that if you require corrections/changes to tables or figures, you must supply the revised files, as these items are not edited for you. No changes are needed to the table and to the figures.

AQ4: Please provide the page range for Reference [34].

pp. 1-es


```
@inproceedings{mchenry2006chicago,
  title={Chicago spectrum occupancy measurements \& analysis and a long-term studies proposal},
  author={McHenry, Mark A and Tenhula, Peter A and McCloskey, Dan and Roberson, Dennis A and Hood, Cynthia S},
  booktitle={Proceedings of the first international workshop on Technology and policy for accessing spectrum},
  pages={1--es},
  year={2006}
}
```

Learning-Based Spectrum Sensing and Access in Cognitive Radios via Approximate POMDPs

AQ1 Bharath Keshavamurthy, *Graduate Student Member, IEEE*, and Nicolò Michelusi, *Senior Member, IEEE*

***Abstract*—A novel LEarning-based Spectrum Sensing and Access (LESSA) framework is proposed, wherein a cognitive radio (CR) learns a time-frequency correlation model underlying spectrum occupancy of licensed users (LUs) in a radio ecosystem; concurrently, it devises an approximately optimal spectrum sensing and access policy under sensing constraints. A Baum-Welch algorithm is proposed to learn a parametric Markov transition model of LUs' spectrum occupancy based on noisy spectrum measurements. Spectrum sensing and access are cast as a Partially-Observable Markov Decision Process, approximately optimized via randomized point-based value iteration. Fragmentation, Hamming-distance state filters and Monte-Carlo methods are proposed to alleviate the inherent computational complexity, and a weighted reward metric to regulate the trade-off between CR's throughput and interference to the LUs. Numerical evaluations demonstrate that LESSA performs within 5% of a genie-aided upper bound with foreknowledge of LUs' spectrum occupancy, and outperforms state-of-the-art algorithms across the entire trade-off region: 71% over correlation-based clustering, 26% over Neyman-Pearson-based spectrum sensing, 6% over the Viterbi algorithm, and 9% over adaptive Deep Q-Network. LESSA is then extended to a distributed Multi-Agent setting (MA-LESSA), by proposing novel neighbor discovery and channel access rank allocation. MA-LESSA improves CRs' throughputs by 43% over cooperative TD-SARSA, 84% over cooperative greedy distributed learning, and 3× over non-cooperative learning via g-statistics and ACKs. Finally, MA-LESSA is implemented on the DARPA SC2 platform, manifesting superior performance over competitors in a real-world TDWR-UNII WLAN emulation; its implementation feasibility is further validated on an ad-hoc distributed wireless testbed of ESP32 radios, exhibiting 96% success probability.**

***Index Terms*—Hidden Markov Model, cognitive radio, spectrum sensing, POMDP.**

I. Introduction

COGNITIVE radios (CRs) 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 cellular technologies [15], a multitudinous array of devices 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 CR networks, is being widely studied as a solution to the problem of spectrum scarcity, in both military and consumer spheres: CRs intelligently access portions of the spectrum unused by the sparse and infrequent transmissions of incumbents or licensed users (LUs), in order to deliver their own network flows, while adhering to interference compliance requirements. In order to intelligently access the spectrum white-spaces, CRs need to solve for a spectrum sensing and subsequent access policy based on noisy observations of the occupancy behavior of LUs. Yet, critical design limitations, driven by energy efficiency requirements or constraints on sensing times [14], prevent CRs from sensing simultaneously the entire spectrum of interest. Under these constraints, CRs can only sense a small portion of the spectrum to determine access opportunities, as studied in [6]–[10], [12]–[14]. However, this approach is quite conservative, since it does not allow CRs to access the large pool of subcarriers that have not been sensed.

LUs' occupancy may exhibit correlation across both time and frequency, as demonstrated in [16] and visualized in Fig. 2(a). Exploiting this time-frequency correlation structure may significantly improve white-space detection, by enabling CRs to predict the occupancy state of those subcarriers that have not been directly sensed, and may unlock additional spectrum access opportunities. In this paper, focusing first on deployments with a single CR, we propose LESSA, a LEarning-based Spectrum Sensing and Access framework that leverages these time-frequency correlations: a CR learns a parametric time-frequency correlation model underlying the occupancy behavior of LUs in the radio ecosystem; concurrently, it exploits this learned model to construct an approximately optimal sensing and access policy using a Partially Observable Markov Decision Process (POMDP) formulation, solved via randomized Point-Based Value Iteration (PBVI). We propose fragmentation techniques, Hamming distance state filters and Monte-Carlo methods to alleviate the inherent computational complexity, and we introduce a weighted reward metric to regulate the trade-off between CR's throughput and interference to the LUs, unlike the state-of-the-art. Our numerical evaluations demonstrate that LESSA improves spectrum occupancy across the entire trade-off region, over state-of-the-art algorithms including correlation-based clustering [3], Neyman-Pearson-based spectrum sensing [13], the Viterbi algorithm [9], and adaptive Deep Q-Network (DQN) [2].

AQ2 Manuscript received January 15, 2021; revised June 18, 2021; accepted November 10, 2021. Funded in part by NSF under grants CNS-1642982 and CNS-2129015. The associate editor coordinating the review of this article and approving it for publication was F. Gao. Part of this research appeared in the proceedings of IEEE ICC 2021 [1]. *(Corresponding author: Bharath Keshavamurthy.)*

The authors are with the School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ 85287 USA (e-mail: bkeshav1@asu.edu; nicolo.michelusi@asu.edu).

Digital Object Identifier 10.1109/TCCN.2021.3129802

TABLE I
COMPARISON OF OUR FRAMEWORK WITH RELEVANT SCHEMES IN THE STATE-OF-THE-ART AQ3

Paper	Adaptive sensing	Sensing constraints	Training mode: Model/Policy	LUs' occupancy correlation model	Observation model	CR-LU trade-off regulation	CRs' deployment
This work	Yes	Yes	Online/Online	Time-Freq., parametric	Noisy	Yes	Single- & Multi-
[2]	Yes	Yes	Online/Online	Time-Freq., Model-free	Noise ignored	No	Single-
[3]	Yes	Yes	Offline/Offline	Time-Freq., parametric	Noiseless	No	Single-
[4]	No	No	Offline/Offline	Time-Freq., Model-free	Noiseless	No	Single-
[5]	No	Yes	Offline/Offline	Time-Freq., Model-free	Noisy	No	Multi-
[6]	Yes	Yes	Online/Online	Time only	Noisy	No	Multi-
[7]	Yes	Yes	Known/Online	Time only	Noisy	No	Single-
[8]	Yes	Yes	Online/Online	Time only	Noisy	No	Multi-
[9]	No	No	Offline/Offline	Time only	Noisy	No	Single-
[10]	Yes	Yes	-/Online	Independence	Noisy	No	Single-
[11]	Yes	Yes	-/Online	Independence	Noisy	No	Multi-
[12]	No	Yes	-/Online	Independence	Noisy	No	Multi-
[13]	No	No	-/-	Independence	Noisy	No	Multi-
[14]	No	Yes	-/-	Independence	Noisy	No	Multi-

Furthermore, we extend LESSA to Multi-Agent (MA-LESSA) deployments, enveloping radio environments with several CRs performing intelligent spectrum sensing and access. We propose novel neighbor discovery and channel access rank allocation schemes that enable the CRs to collaborate and capitalize on spectral resources left unused by the multiple LUs in the network. We implement MA-LESSA on the DARPA Spectrum Collaboration Challenge (SC2) platform, manifesting superior performance over competitors in a real-world TDWR-UNII WLAN emulation [17]. Finally, we demonstrate the implementation feasibility of MA-LESSA in a network of ESP32 radios, and we illustrate the performance disparities between collaborative and opportunistic (non-cooperative or competitive) access through comparisons with algorithms in the multi-agent literature.

Related Work: We now discuss the most closely-related works in the state-of-the-art, summarized in Table I. Most prior algorithms for spectrum sensing and access operate under the assumption that the occupancy behavior of LUs is independent across both time and frequency [10]–[14], or exploit temporal correlation but fail to capitalize on frequency correlation [6]–[9]. These assumptions are not only impractical [16] but also imprudent because critical information aiding the accurate detection of white-spaces may be gleaned by exploiting the correlation in LUs' occupancy over *both* time and frequency, as done in this work.

Spectrum sensing algorithms exploiting both time-frequency correlation in LU occupancy are studied in [2]–[5]. Yet, the system detailed in [3] learns the time-frequency LU occupancy correlation structure offline using pre-loaded databases, which may be inefficient in non-stationary settings; in contrast, we propose a *concurrent learning and adaptation framework*, in which CRs learn a time-frequency correlation model of LUs' occupancy via an online Baum-Welch algorithm, and leverage this knowledge to concurrently optimize spectrum sensing and access (under sensing limitations). On the other hand, LESSA achieves performance superior to model-free DQN [2], owing to a parametric time-frequency Markov transition model. Contrasting our framework against black-box Deep Learning models [4], [5], our approach circumvents laborious data collection and pre-processing tasks, thanks to online learning of the parametric time-frequency correlation model underlying the LUs' occupancy behavior.

In terms of the observation model, a noiseless spectrum sensing setting is assumed in [3], which is unrealistic. In [6], LUs' spectrum occupancy is estimated directly via energy detection thereby ignoring errors in state estimation, while we incorporate a multi-subcarrier access decision based on the contemporary posterior belief probability distribution. Although the DQN framework in [2] revolves around a POMDP formulation, the uncertainty involved is assumed to be only due to sensing restrictions by the CR, while making no claims about system operations in noisy settings. Differently from these works, we assume a Hidden Markov Model (HMM) formulation in which the true LU occupancy states are hidden behind noisy observations at a CR's spectrum sensor. Hence, partial observability in our formulation is ~~due simultaneously~~ simultaneously due to channel sensing restrictions, a noisy observation model, and unknown LU occupancy dynamics.

All the above works fail to provide a mechanism to manage the trade-off between CRs' network throughput and LUs' interference; in contrast, LESSA enables this feature through a weighted reward metric that favors CRs' throughput and penalizes LUs' interference. Unlike non-adaptive sensing strategies [4], [5], [9], [12]–[14], our solution adapts the sensing action in each time-step to more effectively cope with spectrum sensing constraints, driven by transition model estimates and reward/penalty feedback.

Finally, analyzing the state-of-the-art in the multi-agent distributed CR domain, we find both collaborative [5], [6], [8], [11]–[14] as well as opportunistic [11] schemes for channel sensing and access. Drawing a contrast between our contributions and the systems detailed in these works, the authors in [8] make independence assumptions during a projection approximation to a lower-dimensional space; [13], [14] assume a time-frequency independence structure in LUs' occupancy, and employ energy detection at the CRs, i.e., no adaptive sensing and more importantly, no policy optimization; [12] focuses primarily on data aggregation strategies within the ensemble; [6] proposes a multi-agent Temporal Difference (TD) SARSA with Linear Function Approximation (LFA), but fails to detail neighbor discovery and channel access order allocation schemes, proposed in this paper; [11] details a collaborative scheme (greedy learning under pre-allocation) and an opportunistic one (g-statistics with ACKs), under the assumption of time-frequency independence in LUs' occupancy behavior; additionally, the opportunistic scheme in [11] relies on ACKs as a feedback mechanism to gauge the utility of an access decision, which imbibes unnecessary lag into the model; in contrast, our framework employs a threshold-based decision heuristic involving the posterior belief probability to evaluate the reward obtained from the executed access action: in addition to displaying superior performance, as illustrated in Section IV, this mechanism is easier to implement in real-world settings, as we demonstrate by realizing our solution on the DARPA SC2 emulator [18] and on an ad-hoc ESP32 network [19].

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

- We develop a LEarning-based framework for Spectrum Sensing and Access (LESSA) in a radio environment with LUs exhibiting time- and frequency- Markovian correlation in their occupancy behavior, and a CR attempting to detect and access unused spectral resources – under a noisy observation model with sensing restrictions.
- In Section III-A, we develop an online Baum-Welch algorithm [20] to learn the LUs' occupancy correlation model.
- In Section III-B, we concurrently leverage the learned model in a randomized PBVI algorithm known as PERSEUS [21] to devise an approximately optimal spectrum sensing and access policy; we alleviate its computational complexity by introducing fragmentation, belief update simplification heuristics via Hamming distance state filters and Monte-Carlo based methods.
- In single-agent settings, we demonstrate the superior performance of LESSA relative to relevant algorithms in the state-of-the-art (Section IV). Additionally, through computational time complexity bench-marking, we prove the enhanced scalability of our solution.
- In Section V, we extend LESSA to distributed Multi-Agent (MA-LESSA) deployments: adapting the Multi-band Directional Neighbor Discovery scheme described in [22] to distributed multi-agent CR deployments, we propose a novel neighbor discovery heuristic centered around RSSI thresholding; inspired by cluster fallback mechanisms (leader selection, broker failover, master-slave auto-configuration, and data replication) involved in Message Oriented Middleware [23], we propose a channel access rank allocation technique centered around a quorum-based preferential ballot algorithm. On this front, we demonstrate improved performance over both collaborative and opportunistic distributed multi-agent state-of-the-art; also, we exemplify its implementation feasibility on an ad-hoc WLAN testbed of ESP32 radios [19], [24].
- Finally, for multi-agent evaluations, we retrofit our proposed solution into the DARPA SC2 BAM! Wireless radio [25] to emulate its operations during the Active Incumbent scenario (TDWR-UNII WLAN) [17], and prove superior scoring performance over competing strategies [26]–[30]. We also perform computational time complexity analyses of our neighbor discovery and channel access rank allocation heuristics in emulations of highly-mobile real-world disaster relief (SC2 Payline [31]) and military deployment scenarios (SC2 Alleys of Austin [32]).

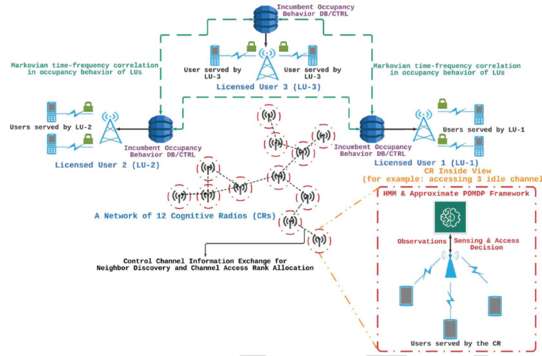


Fig. 1. The radio ecosystem under analysis: An exemplification of the system model detailed in Section II-A with $J_L = 3$ and $J_C = 12$: we first study deployments with $J_C = 1$ before extending our analyses to multi-agent settings.

The rest of this paper is organized as follows: Section II details the system model; Section III describes our algorithmic solutions; Section IV presents numerical evaluations for the single-agent case; Section V extends LESSA to multi-agent deployments ~~a distributed multi-agent setup~~, with numerical evaluations; finally, Section VI provides concluding remarks.

II. System Model

A. Signal Model

We consider a licensed network of J_L LUs, and J_C CRs attempting to exploit portions of the spectrum left unused by these LUs – as illustrated in Fig. 1. In Sections II, III, and IV, we focus on the single-agent case ($J_C = 1$); the multi-agent extension ($J_C{>}1$) is discussed in Section V. The spectrum of interest is discretized into K subcarriers of bandwidth W. The frequency-domain signal received at the CR's spectrum sensor in time-slot t at subcarrier k is

$$Y_k(t) = \sum_{j=1}^{J_L} H_{j,k}(t) X_{j,k}(t) + V_k(t), \tag{1}$$

where $X_{j,k}(t)$ is the frequency-domain signal of LU $j{\in}\{1,\ldots,J_L\}$ on subcarrier $k \in \{1,\ldots,K\}$, with $X_{j,k}(t) = 0$ if LU j is idle on subcarrier k; $H_{j,k}(t)$ denotes the channel on subcarrier k between LU j and the CR; and $V_k(t){\sim}\mathcal{CN}(0,\sigma_V^2)$ denotes Gaussian noise with variance σ_V^2, i.i.d across time and subcarriers. We assume an Orthogonal Frequency Division Multiple Access (OFDMA) strategy among the LUs: letting $j_{k,t}$ be the index of the LU that occupies subcarrier k in time-slot t, $X_k(t){\triangleq}X_{j_{k,t},k}(t)$ and $H_k(t){\triangleq}H_{j_{k,t},k}(t)$, we can rewrite (1) as

$$Y_k(t) = H_k(t) X_k(t) + V_k(t), \tag{2}$$

where $X_k(t) = 0$ if subcarrier k is not occupied in time-slot t. We model $H_k(t)$ as Rayleigh fading with variance σ_H^2 : $H_k(t){\sim}\mathcal{CN}(0,\sigma_H^2)$, i.i.d across time and subcarriers.

B. Time-Frequency Occupancy Correlation Structure

The LU's signal on subcarrier k in time-slot t is modeled as

$$X_k(t) = \sqrt{P_T} B_k(t) S_k(t), \tag{3}$$

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

$$\vec{B}(t) = [B_1(t), B_2(t), B_3(t), \ldots, B_K(t)]^\intercal \in \{0,1\}^K. \tag{4}$$

We assume that spectrum occupancy is correlated across time and subcarriers: LUs typically occupy a set of adjacent subcarriers (frequency correlation), repeating similar motifs in behavior over an extended period of time (temporal correlation) [16], [33], [34]. To capture temporal correlation, we model the evolution of $\vec{B}(t)$ over time as a Markov process

$$\mathbb{P}\Big(\vec{B}(t+1)|\vec{B}(j), \forall j \leq i\Big) = \mathbb{P}\Big(\vec{B}(t+1)|\vec{B}(t)\Big) \triangleq P_B\Big(\vec{B}(t+1)|\vec{B}(t)\Big), \tag{5}$$

with one-step transition probability $P_B(\vec{B}(t+1)|\vec{B}(t))$. Using the chain rule of conditional probability, we can further express it as

$$P_B\Big(\vec{B}(t+1)|\vec{B}(t)\Big) = \mathbb{P}\Big(B_1(t+1)|\vec{B}(t)\Big) \prod_{k=2}^{K} \mathbb{P}\Big(B_k(t+1)|\vec{B}_{1:k-1}(t+1), \vec{B}(t)\Big), \tag{6}$$

where $\vec{B}_{1:k-1}(t+1)$ is the spectrum occupancy state in subcarriers 1 to $k-1$ in time-slot $t+1$. We further assume a Markovian structure across frequency,

$$\mathbb{P}\Big(B_k(t+1)|\vec{B}_{1:k-1}(t+1), \vec{B}(t)\Big) = \mathbb{P}\Big(B_k(t+1)|B_{k-1}(t+1), \vec{B}(t)\Big),$$

i.e., $B_k(t+1)$ is independent of the spectrum occupancy in the non-adjacent subcarriers $\vec{B}_{1:k-2}(t+1)$, when conditioned on $\vec{B}(t)$ and on the spectrum occupancy in the adjacent subcarrier $B_{k-1}(t+1)$. This assumption reflects the intuition that the state of the adjacent subcarrier $k-1$ ($B_{k-1}(t+1)$) more directly affects the state of subcarrier k than non-adjacent subcarriers 1 to $k-2$. Replacing this expression into (6), we finally obtain

$$P_B\Big(\vec{B}(t+1)|\vec{B}(t)\Big) = \mathbb{P}\Big(B_1(t+1)|\vec{B}(t)\Big) \prod_{k=2}^{K} \mathbb{P}\Big(B_k(t+1)|B_{k-1}(t+1), \vec{B}(t)\Big). \tag{7}$$

We denote this model as *bottom-up frequency correlation*, since the state of subcarrier k depends on that of the adjacent lower subcarrier $k-1$, as opposed to the *top-down frequency correlation* where it depends on that of the adjacent upper subcarrier $k+1$. We remark that bottom-up and top-down frequency correlation models can be used interchangeably: in fact, replacing $\mathbb{P}(B_k(t+1)|B_{k-1}(t+1), \vec{B}(t)) = \mathbb{P}(B_{k-1}(t+1)|B_k(t+1), \vec{B}(t))\frac{\mathbb{P}(B_k(t+1)|\vec{B}(t))}{\mathbb{P}(B_{k-1}(t+1)|\vec{B}(t))}$ (Bayes' rule) in (7) and simplifying, we obtain the top-down frequency correlation model

$$P_B\Big(\vec{B}(t+1)|\vec{B}(t)\Big) = \mathbb{P}\Big(B_K(t+1)|\vec{B}(t)\Big) \prod_{k=1}^{K-1} \mathbb{P}\Big(B_k(t+1)|B_{k+1}(t+1), \vec{B}(t)\Big), \tag{8}$$

so that the two models can be directly mapped to each other. We now embed the bottom-up frequency correlation of (7) into a parametric form. Expressing $\vec{B}(t) = [B_k(t), \vec{B}_{-k}(t)]$, where $\vec{B}_{-k}(t)$ is the spectrum occupancy state in subcarriers other than k in time-slot t, we define $\forall u, v, w \in \{0,1\}, 2 \leq k \leq K$

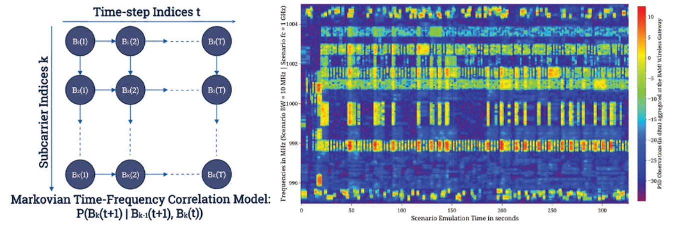
$$q_w \triangleq \mathbb{P}\Big(B_1(t+1) = 1|B_1(t) = w, \vec{B}_{-1}(t)\Big),$$

$$p_{u,v} \triangleq \mathbb{P}\Big(B_k(t+1) = 1|B_{k-1}(t+1) = u, B_k(t) \boxed{= v, \vec{B}_{-k}(t)\Big).}$$

Moved the entire equation to one line to improve readability (9)

Note that this parametric model assumes that $B_k(t+1)$ is independent of $\vec{B}_{-k}(t)$, when conditioned on $(B_{k-1}(t+1), B_k(t))$: intuitively, the state of subcarrier k in time-slot $t+1$ is most directly affected by the state on the same subcarrier in the previous time-slot t, as shown in Fig. 2a. In the following, we define the parameter vector $\vec{\theta} = [p_{0,0}, p_{0,1}, p_{1,0}, p_{1,1}, q_0, q_1]$, and express the dependence of the one-step transition model on $\vec{\theta}$ as $P_B(\vec{B}(t+1)|\vec{B}(t); \vec{\theta})$.

To experimentally validate the aforementioned parametric time-frequency correlation model, we performed a Bayesian Information Criterion (BIC) evaluation [35] on a dataset of Power Spectral Density (PSD) measurements from the DARPA SC2 Active Incumbent scenario emulation [17], depicted in Fig. 2b. This scenario constitutes a Terminal Doppler Weather Radar (TDWR) station, and several competitor CR networks [29], [30] (including our Purdue BAM! Wireless [25]) of Unlicensed National Information Infrastructure (UNII: 5GHz WLAN) Wi-Fi nodes. We define the BIC metric as $\text{BIC} = \Gamma \cdot \ln \nu - 2 \ln \mathbb{P}(\mathbf{B}|\vec{\theta}^*)$, where ν is the sample size, Γ is the number of model parameters (6 in the



(a) The correlation model (b) Active Incumbent PSD data

Fig. 2. Visualization of the LU occupancy time-frequency correlation structure (a); combined PSD plot of the occupancy behavior of a TDWR and competitors during the DARPA SC2 Active Incumbent scenario emulation (b).

proposed parametric model), $\mathbf{B}$ is the time-frequency binary occupancy matrix from the dataset, and $\vec{\theta}^*$ is the model parameter vector estimated from the dataset of PSD measurements, using the Baum-Welch algorithm detailed in Section III-A. We employed a $70-30$ training-test split to evaluate this 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 dedicated to the BIC evaluation. The parametric bottom-up time-frequency correlation model yields a BIC of 71.872, the best compared to other state-of-the-art models: time-frequency independence (98.840) [10]–[14]; time-only correlation (95.231) [6]–[9]; frequency-only correlation (76.879); notably, the *top-down* frequency correlation model, using a similar parameterization of (9), yields a similar BIC metric of 74.207 (the slightly different value is due to the parametric structure). This evaluation reveals that exploiting frequency correlation is more important than time correlation – and not surprisingly, exploiting *both* provides a better fit to the dataset. This assessment validates our conjecture that LUs in real-world deployments exhibit prominent spectrum occupancy patterns across both time and frequency, exploited by LESSA. Moreover, it further corroborates our previous observation that bottom-up and top-down correlation models can be used interchangeably.

C. Channel Sensing Model

The CR attempts to detect white-spaces and access them to deliver its network flows. Due to constraints on energy-efficiency and sensing/data aggregation times [14], it can sense a maximum of κ subcarriers in a time-slot, with $1 \leq \kappa \leq K$. Let $\mathcal{K}_t \subseteq \{1, 2, \dots, K\}$ be the set of subcarriers sensed by the CR at time t, with $|\mathcal{K}_t| \leq \kappa$. This selection is dictated by a sensing policy, described in Section II-D. After sensing the subcarriers listed in $\mathcal{K}_t$, the observation vector $\vec{Y}(t) = [Y_k(t)]_{k \in \mathcal{K}_t}$ is collected, with $Y_k(t)$ given in (2). Owing to (2), the i.i.d. assumptions on the noise $V_k(t)$, the transmitted symbols $S_k(t)$, and the frequency domain subcarriers $H_k(t)$, the probability density function (pdf) of $\vec{Y}(t)$ conditioned on $\vec{B}(t)$ and $\mathcal{K}_t$ is given by

$$
\begin{aligned}
&f\Big(\vec{Y}(t)|\vec{B}(t), \mathcal{K}_t\Big) = \prod_{k \in \mathcal{K}_t} f(Y_k(t)|B_k(t)), \text{where} \\
&Y_k(t)|B_k(t) \sim \mathcal{CN}\Big(0, \sigma_H^2 P_T B_k(t) + \sigma_V^2\Big). \qquad (10)
\end{aligned}
$$

D. POMDP Formulation and Spectrum Access

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. We now describe the POMDP operation, whose process flow is illustrated in Fig. 3 (the multi-agent features, shown in green, will be discussed in Section V).

Prior to gathering spectrum measurements in time-slot t, the POMDP state is given by the prior belief β_t, i.e., the probability distribution of the unknown spectrum occupancy state $\vec{B}(t)$, given the history of measurements obtained by the CR's spectrum sensor up to, but not including, time-slot t. Given β_t, the CR chooses a sensing action according to a sensing policy

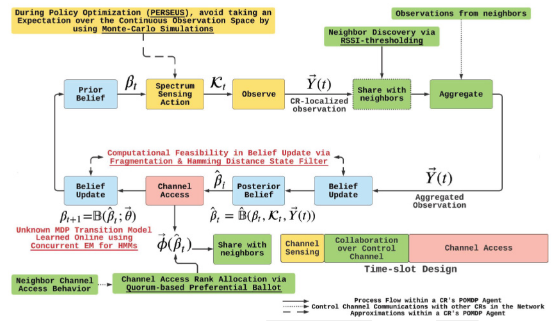


Fig. 3. The POMDP process flow as discussed in Section II-D – with neighbor discovery, channel access rank allocation, and time-slot design (shown in green) discussed in our multi-agent deployment analysis (Section V).

π, $\mathcal{K}_t = \pi(\beta_t) \in \mathcal{A}$, where $\mathcal{A}$ is the spectrum sensing space, i.e., all possible combinations in which κ subcarriers are chosen to be sensed in a time-slot (discussed in Section II-C); the CR then senses the subcarriers in $\mathcal{K}_t$, observes $\vec{Y}(t)$, and computes the posterior belief of $\vec{B}(t)$ using Bayes' rule, as

$$
\begin{aligned}
\hat{\beta}_t(\vec{B}) &= \mathbb{P}\Big(\vec{B}(t) = \vec{B}|\beta_t, \mathcal{K}_t, \vec{Y}(t)\Big) \\
&= \frac{f\Big(\vec{Y}(t)|\vec{B}, \mathcal{K}_t\Big)\beta_t(\vec{B})}{\sum_{\vec{B}' \in \{0,1\}^K} f\Big(\vec{Y}(t)|\vec{B}', \mathcal{K}_t\Big)\beta_t(\vec{B}')}. \qquad (11)
\end{aligned}
$$

Given $\hat{\beta}_t$, the CR then makes channel access decisions $\vec{\phi} \in \{0,1\}^K$, where $\phi_k = 1$ denotes the CR's decision to access the kth subcarrier, otherwise $\phi_k = 0$. We design $\vec{\phi}$ based on the reward metric

$$
R\Big(\vec{\phi}, \vec{B}(t)\Big) = \sum_{k=1}^{K} (1 - B_k(t))\phi_k - \lambda B_k(t)\phi_k. \qquad (12)
$$

Since $\vec{B}(t)$ is unknown, we compute its expectation based on the posterior belief $\hat{\beta}_t$ as

$$
\begin{aligned}
R\Big(\vec{\phi}, \hat{\beta}_t\Big) &= \mathbb{E}\Big[R\Big(\vec{\phi}, \vec{B}(t)\Big)|\vec{\phi}, \hat{\beta}_t\Big] \\
&= \sum_{k=1}^{K} \Big(1 - \hat{\beta}_{t,k}\Big)\phi_k - \lambda\hat{\beta}_{t,k}\phi_k, \qquad (13)
\end{aligned}
$$

where $\hat{\beta}_{t,k} = \sum_{\vec{B} \in \{0,1\}^K : B_k = 1} \hat{\beta}_t(\vec{B})$ is the posterior probability that subcarrier k is occupied by a LU. Note that, if the CR uses the kth subcarrier ($\phi_k(t) = 1$), it accrues a reward of 1 if the subcarrier is truly idle (with posterior probability $1 - \hat{\beta}_{t,k}$), and a penalty of $\lambda \geq 0$ if it is occupied (with posterior probability $\hat{\beta}_{t,k}$); it accrues no reward for not using a channel. Hence, (13) balances a trade-off between maximizing the CR's throughput (CR's transmissions on idle subcarriers), and minimizing the interference to the LUs (CR's transmissions on occupied ones); λ regulates such trade-off. The channel access decision is obtained as $\vec{\phi}^*(\hat{\beta}_t) = \arg\max_{\vec{\phi} \in \{0,1\}^K} R(\vec{\phi}, \hat{\beta}_t)$, yielding

$$
\phi_k^*\Big(\hat{\beta}_t\Big) = \mathcal{I}\Big[\hat{\beta}_{t,k} \leq (1 + \lambda)^{-1}\Big], \ \forall k = 1, \dots, K, \qquad (14)
$$

where $\mathcal{I}[\,\cdot\,]$ is the indicator function, yielding the optimized reward

$$R^*\left(\hat{\beta}_t\right) = \max_{\vec{\phi}\in\{0,1\}^K} R\left(\vec{\phi}, \hat{\beta}_t\right)$$

$$= \sum_{k=1}^{K} \max\left\{1 - (1+\lambda)\hat{\beta}_{t,k}, 0\right\}. \quad (15)$$

In other words, if the CR is confident that the kth subcarrier is idle ($\hat{\beta}_{t,k} < \frac{1}{1+\lambda}$), it accesses it; otherwise, it remains idle. The confidence level $\frac{1}{1+\lambda}$ is regulated by the penalty parameter λ.

After accessing the subcarriers based on the access decision $\phi_k^*(\hat{\beta}_t)$, the CR then updates the prior belief for the next time-slot $t+1$ based on the one-step transition model,

$$\beta_{t+1}(\vec{B}) = \sum_{\vec{B}'\in\{0,1\}^K} P_B\left(\vec{B}|\vec{B}';\vec{\theta}\right)\hat{\beta}_t\left(\vec{B}'\right), \quad (16)$$

and the process is repeated over time. Let

$$\hat{\beta}_t = \hat{\mathbb{B}}\left(\beta_t, \mathcal{K}_t, \vec{Y}(t)\right), \quad \beta_{t+1} = \mathbb{B}\left(\hat{\beta}_t;\vec{\theta}\right) \quad (17)$$

denote the functions that map the prior belief β_t to the posterior belief $\hat{\beta}_t$, and the latter to the next prior belief β_{t+1} (see Eqs. (11) and (16)). The goal is to devise a spectrum sensing policy (the spectrum access policy is given in (14)) to maximize the infinite-horizon discounted reward

$$V^\pi(\beta) = \mathbb{E}_\pi\left[\sum_{t=1}^{\infty}\gamma^t R^*\left(\hat{\beta}_t\right)\Big|\beta_0 = \beta\right], \quad (18)$$

where $0 < \gamma < 1$ is the discount factor, $\beta_0 = \beta$ (e.g., uniform) is the initial belief, and $\hat{\beta}_t$ is the posterior belief induced by the policy $\mathcal{K}_t = \pi(\beta_t)$ and the observation vector $\vec{Y}(t)$ via $\hat{\beta}_t = \hat{\mathbb{B}}(\beta_t, \mathcal{K}_t, \vec{Y}(t))$. The maximization problem is then stated as

$$\pi^* = \arg\max_\pi V^\pi(\beta), \quad (19)$$

yielding the optimal value $V^*(\beta)$ under the optimal spectrum sensing policy π^*. In principle, $V^*(\beta)$ can be determined via the value iteration algorithm $V_{i+1} = \mathcal{H}(V_i)$, initialized as $V_0(\beta) = 0$, which converges to V^* as $i \to \infty$ [36]. Here, $\mathcal{H}$ is the Bellman's operator, defined as $\forall\beta$

$$V_{i+1}(\beta) = \max_{\mathcal{K}\in\mathcal{A}} \mathbb{E}_{\vec{Y}|\vec{B}\sim\beta,\mathcal{K}}$$

Remove "X"

Move this part to the same line

Move "=" and everything after it to a new line

$$\left[R^*\left(\overbrace{\hat{\mathbb{B}}\left(\beta, \mathcal{K}, \vec{Y}\right)}^{\hat{\beta}}\right) + \gamma\cdot V_i\left(\overbrace{\mathbb{B}\left(\hat{\mathbb{B}}\left(\beta, \mathcal{K}, \vec{Y}\right);\vec{\theta}\right)}^{\beta_{next}}\right)\right]. \quad (20)$$

We term (20) the *backup* operation, which in exact value iteration should be solved for every belief β in the probability simplex. Intuitively, $V_i(\beta)$ and $V_{i+1}(\beta)$ represent the optimal finite horizon (of duration i and $i+1$, respectively) discounted values starting from belief β. To determine V_{i+1} from V_i, after choosing the spectrum sensing set $\mathcal{K}$ and observing $\vec{Y}$, the posterior belief is updated as $\hat{\beta} = \hat{\mathbb{B}}(\beta, \mathcal{K}, \vec{Y})$, the reward $R^*(\hat{\beta})$ is accrued, the next prior belief is updated as $\beta_{next} = \mathbb{B}(\hat{\beta};\vec{\theta})$, so that $\gamma V_i(\beta_{next})$ is the discounted future value accrued; the expectation is then taken with respect to $\vec{Y}$, whereas the maximization over $\mathcal{K}\in\mathcal{A}$ yields the optimal spectrum sensing action that maximizes the expected reward plus discounted future value. Yet, this direct approach is not applicable to our settings: 1) the transition model $P_B(\vec{B}(t+1)|\vec{B}(t);\vec{\theta})$ needed in (16) to update the prior β_{t+1} is unknown; 2) as the number of subcarriers of interest increases, the number of states of the underlying MDP scales exponentially, resulting in a high-dimensional belief space and high complexity. To address these two challenges:

- We incorporate an HMM EM estimator, i.e., the Baum-Welch algorithm, to learn the parameter vector $\vec{\theta}$ that defines the transition model $P_B(\vec{B}(t+1)|\vec{B}(t);\vec{\theta})$, while concurrently solving for the sensing and access policy. This is developed in Section III-A.
- We embed a low-complexity approximate PBVI algorithm known as PERSEUS [21], and embed it with fragmentation (into independent subsets of highly-correlated subcarriers), belief update simplification heuristics (Hamming distance state filters), and Monte-Carlo methods, developed in Section III-B.

III. LESSA Algorithms

In this section, we propose a Baum-Welch algorithm to estimate the parameter vector $\vec{\theta}$ that defines the Markov time-frequency correlation model. In Section III-B, we then use the estimated model in a randomized PBVI algorithm known as PERSEUS [21] to determine an approximately optimal sensing policy. Crucially, these algorithms are executed concurrently – a key feature to enable adaptation in non-stationary settings.

A. Occupancy Correlation Structure Estimation

Let τ be the learning period of the parameter estimation algorithm. Let $\mathbf{B} = [\vec{B}(t)]_{t=1}^{\tau}$ be the unknown sequence of states and $\mathbf{Y} = [\vec{Y}(t)]_{t=1}^{\tau}$ be the sequence of observations made at the CR's spectrum sensor from $t = 1$ to $t = \tau$, based on a generic spectrum sensing policy. We formulate the Maximum Likelihood Estimation (MLE) problem to estimate the vector $\vec{\theta}$ that defines the LU occupancy time-frequency correlation structure (detailed in Section II-B) as

$$\vec{\theta}^* = \arg\max_{\vec{\theta}} \log\left(\sum_{\mathbf{B}} \mathbb{P}\left(\mathbf{B}|\vec{\theta}\right) f(\mathbf{Y}|\mathbf{B}, \mathcal{K})\right). \quad (21)$$

Note that $f(\mathbf{Y}|\mathbf{B}, \mathcal{K}) = \prod_{t=1}^{\tau} f(\vec{Y}(t)|\vec{B}(t), \mathcal{K}_t)$ is the observation pdf conditional on the states, given by (10), whereas $\mathbb{P}(\mathbf{B}|\vec{\theta})$ can be expressed using the one-step parametric transition model as $\mathbb{P}(\mathbf{B}|\vec{\theta}) = \mathbb{P}(\vec{B}(1))\prod_{t=2}^{\tau} P_B(\vec{B}(t)|\vec{B}(t-1);\vec{\theta})$. We solve the MLE problem using the Baum-Welch algorithm, an EM algorithm for HMMs [37] that estimates $\vec{\theta}^{(i)}$ (i denotes the iteration index) by iterating through the E- and M- steps: given $\vec{\theta}^{(i)}$, the E-step computes

$$Q\left(\vec{\theta}|\vec{\theta}^{(i)}\right) = \mathbb{E}_{\mathbf{B}|\mathbf{Y},\vec{\theta}^{(i)}}\left[\log\left(\mathbb{P}\left(\mathbf{B}|\vec{\theta}\right) f(\mathbf{Y}|\mathbf{B}, \mathcal{K})\right)\right]; \quad (22)$$

afterwards, the M-step updates the estimate of $\vec{\theta}$ as

$$\vec{\theta}^{(i+1)} = \arg\max_{\vec{\theta}} Q\left(\vec{\theta}|\vec{\theta}^{(i)}\right). \tag{23}$$

Specifically, using the Markovian correlation structure, and neglecting additive terms independent of the optimization parameter $\vec{\theta}$ (indicated as $\propto$), the E-step can be rewritten as

$$Q\left(\vec{\theta}|\vec{\theta}^{(i)}\right) \propto \mathbb{E}_{\mathbf{B}|\mathbf{Y},\vec{\theta}^{(i)}}\left[\sum_{t=2}^{\tau} \log\left(P_B\left(\vec{B}(t)|\vec{B}(t-1),\vec{\theta}\right)\right)\right],$$

$$= \sum_{w\in\{0,1\}} \left[A_w^{(i)}(1)\log(q_w) + A_w^{(i)}(0)\log(1-q_w)\right]$$

$$+ \sum_{u,v\in\{0,1\}} \left[B_{u,v}^{(i)}(1)\log\left(p_{u,v}\right) + B_{u,v}^{(i)}(0)\log\left(1-p_{u,v}\right)\right],$$

where we have defined

$$A_w^{(i)}(b) \triangleq \sum_{t=2}^{\tau} \mathbb{P}\left(B_1(t) = b, B_1(t-1) = w|\mathbf{Y},\vec{\theta}^{(i)}\right),$$

$$B_{u,v}^{(i)}(b) \triangleq \sum_{t=2}^{\tau}\sum_{k=2}^{K} \mathbb{P}\left(B_k(t) = b, B_{k-1}(t) = u, \boxed{B_k(t-1)}\right.$$

Move here to the next line; Ensure proper equation alignment

$$\left. = v|\mathbf{Y},\vec{\theta}^{(i)}\right),$$

computed using the Forward-Backward algorithm [20]; the M-step can then be optimized in closed-form as

$$q_w^{(i+1)} = \frac{A_w^{(i)}(1)}{A_w^{(i)}(0) + A_w^{(i)}(1)}, \quad p_{u,v}^{(i+1)} = \frac{B_{u,v}^{(i)}(1)}{B_{u,v}^{(i)}(0) + B_{u,v}^{(i)}(1)}.$$

The computational time complexity of the Baum-Welch algorithm is $O(\tau K \tilde{T})$ [20], where $\tilde{T}$ is the number of iterations until convergence, which depends on the consistency of spectrum occupancy measurements, driven by our observation model and the CR's sensing limitations.

B. *The PERSEUS Algorithm*

LESSA solves for the spectrum sensing (and access, based on the reward maximization detailed in Section II-D) policy in parallel with the parameter estimation algorithm, employing its published iterative transition model estimates, until both algorithms converge. We use PERSEUS [21] to devise an approximately optimal sensing policy, primarily motivated by the following rationale. Unlike the Exhaustive Enumeration and the Witness algorithms in [36], PERSEUS does not perform the backup operation (20) exhaustively on every reachable belief point; instead, it *backs-up* only on a smaller subset of reachable belief points, and exploits this backup operation to improve the value of other reachable belief points, to achieve faster convergence; and instead of computing belief distances as in the PBVI algorithm [38], it computes a *finite* U-dimensional set of reachable beliefs $\tilde{\mathcal{B}}$ through an initial exploration phase, to balance computational complexity with accuracy (larger $U \Rightarrow$ improved accuracy but larger computational burden).

Despite being an approximate POMDP method which alleviates the computational overhead associated with the backup operation, PERSEUS applied to the spectrum sensing problem still possesses computational intractability challenges due to the high-dimensional spectrum occupancy state $\vec{B}\in\{0,1\}^K$ and continuous observation space: the computational cost scales exponentially with the number of subcarriers *K*. To alleviate these challenges, we embed PERSEUS with three novel low-complexity enhancements.

1) In updating the prior belief as in (16), we avoid iterating over all possible $\vec{B}' \in \{0,1\}^K$ by considering only those state transitions that involve a Hamming distance of at most $\delta\in\{1,2,\ldots,K\}$ subcarriers between two consecutive state vectors $\vec{B}(t)$ and $\vec{B}(t+1)$. This is practical because spectrum occupancy of LUs typically varies slowly over time, slower than the processing dynamics of the POMDP agent. Let $\mathcal{B}_\delta(\vec{B})\equiv\{\vec{B}'\in\{0,1\}^K : \zeta(\vec{B},\vec{B}') \leq \delta\}$ be the set of spectrum occupancy states with Hamming distance (ζ) at most δ from state $\vec{B}$. The next prior belief is then approximated as

$$\beta_{t+1} \approx \tilde{\mathbb{B}}\left(\hat{\beta}_t, \vec{\theta}\right), \text{where}$$

Move the equation number (24) here

$$\beta_{t+1}(\vec{B}) = \frac{\sum_{\vec{B}'\in\mathcal{B}_\delta\left(\vec{B}\right)} P_B\left(\vec{B}|\vec{B}';\vec{\theta}\right)\hat{\beta}_t\left(\vec{B}'\right)}{\sum_{\vec{B}''}\sum_{\vec{B}'\in\mathcal{B}_\delta\left(\vec{B}''\right)} P_B\left(\vec{B}''|\vec{B}';\vec{\theta}\right)\hat{\beta}_t\left(\vec{B}'\right)}, \tag{24}$$

where the normalization ensures that β_{t+1} is a probability distribution that sums to one.

2) We use a fragmentation technique to partition the set of K subcarriers into independent smaller sets of adjacent subcarriers; we then run PERSEUS independently on each one of these fragments, concurrently and in parallel, by employing multi-threading capabilities in software frameworks. For example, a radio environment with 18 subcarriers with a sensing constraint of 6 subcarriers per time-slot is fragmented into 3 independent fragments, each comprising 6 subcarriers correlated by the occupancy behavior of the corresponding LUs, and a sensing constraint of 2 subcarriers per time-slot. This fragmentation is practical because in a radio environment with multiple LUs, each LU is typically restricted to a portion (a set of adjacent frequency bands) of the spectrum, either by design or by bureaucracy. We let K' be the number of subcarriers, κ' be the sensing restriction in each of these fragments.

3) We avoid the expectation with respect to the continuous observation vector in the backup operation (20) by using a Monte-Carlo method: we generate N independent realizations of $\vec{Y}|\vec{B},\mathcal{K}$, based on the observation model (10); we then compute the argument of the expectation with respect to each of these realizations, followed by a sample average. As $N \to \infty$, this sample average converges to the conditional expectation in the backup operation.

The resulting *Monte-Carlo Fragmented PERSEUS with Hamming distance-based belief update* is shown in Algorithm 1 (run in parallel on each fragment of spectrum),

Algorithm 1 Monte-Carlo Fragmented PERSEUS With Hamming Distance State Filters

Input: Parameters $\vec{\theta}$; K' subcarriers, Fragment sensing limit κ'; Observation model (10); Target accuracy $\epsilon>0$; Belief space size U;
Output: Set of hyperplanes and spectrum sensing actions $\{(\vec{\alpha}^u, \mathcal{K}^u) : u = 1, \ldots, U\}$ per spectrum fragment.

1: Determine the set of *reachable beliefs* $\tilde{\mathcal{B}} \equiv \{\beta_u : u = 1, \ldots, U\}$;
2: **Initialization:** Hyperplanes $\vec{\alpha}_0^u = \mathbf{0}, \forall u$; actions $\mathcal{K}_0^u = \emptyset, \forall u$; iteration $i = -1$; $V_0(\beta) = 0$, $V_{-1}(\beta) = -\infty$;
3: **while** $|V_{i+1}(\beta) - V_i(\beta)| > \epsilon, \exists \beta \in \tilde{\mathcal{B}}$ **do**
4: $i \leftarrow i+1$; $\tilde{\mathcal{U}} \longleftarrow \{1, \ldots, U\}$; ▷ *Start new iteration*
5: **while** $\tilde{\mathcal{U}} \neq \{\cdot\}$ **do** ▷ *Improve all points in* $\tilde{\mathcal{U}}$
6: Pick u randomly from $\tilde{\mathcal{U}}$, and β_u from $\tilde{\mathcal{B}}$; $\tilde{\mathcal{U}} \leftarrow \tilde{\mathcal{U}} \setminus \{u\}$;
7: For each $\mathcal{K} \in \mathcal{A}$, compute hyperplane $\xi_{\mathcal{K}}^u$:
8:

$$\xi_{\mathcal{K}}^u(\vec{B}) = \frac{1}{N} \sum_{n=1}^{N} \Big[R(\vec{\phi}^*(\hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y}_n)), \vec{B}) \boxed{+}$$

Move "+" to this next line; Ensure proper alignment

$$\gamma \cdot \sum_{\vec{B}'} P_B(\vec{B}'|\vec{B}; \vec{\theta}) \xi_{\mathcal{K},n}^u(\vec{B}') \Big], \text{where}$$

9: $\vec{Y}_n \sim f(\cdot|\vec{B}, \mathcal{K})$ i.i.d. over $n = 1, \ldots, N$ (Eq. (10)), and
10: $\xi_{\mathcal{K},n}^u = \underset{\vec{\alpha}_i^{u'}, u' \in \{1,2,\ldots,|\tilde{\mathcal{B}}|\}}{\arg\max} \langle \mathbb{B}(\hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y}_n)), \vec{\alpha}_i^{u'} \rangle$;
11: $\mathcal{K}_{i+1}^u = \arg\max_{\mathcal{K} \in \mathcal{A}} \langle \beta_u, \xi_{\mathcal{K}}^u \rangle$,
12: $\vec{\alpha}_{i+1}^u = \xi_{\mathcal{K}_{i+1}^u}^u$, $V_{i+1}(\beta_u) = \langle \beta_u, \vec{\alpha}_{i+1}^u \rangle$; ▷ *Backup*
13: **for** $\forall u' \in \tilde{\mathcal{U}}$ **do**

Remove unnecessary white spaces in line 15 of Alg 1

14: **if** $\langle \beta_{u'}, \vec{\alpha}_{i+1}^u \rangle \geq V_i(\beta_{u'})$ **then**
15: $V_{i+1}(\beta_{u'}) = \langle \beta_{u'}, \vec{\alpha}_{i+1}^u \rangle, \vec{\alpha}_{i+1}^{u'} =$
$\boxed{\vec{\alpha}_{i+1}^u, \mathcal{K}_{i+1}^{u'} = \mathcal{K}_{i+1}^u;}$

Move to line 15 to have all of it in one line

16: $\tilde{\mathcal{U}} \longleftarrow \tilde{\mathcal{U}} \setminus \{u'\}$;
17: Return set of hyperplanes and associated spectrum sensing actions $\{(\vec{\alpha}^{*u}, \mathcal{K}^{*u}) \triangleq (\vec{\alpha}_i^u, \mathcal{K}_i^u)$: $u = 1, \ldots, U\}$.

described below. The key idea of PERSEUS is to exploit the fact that the value function V_i generated by the backup operation (20) is Piece-Wise Linear Convex (PWLC) [21], hence it can be approximated by an U-dimensional set of hyperplanes (each associated to a certain sensing action, Step 20) as

$$V(\beta_t) \approx \max_{u \in \{1,2,\ldots,U\}} \langle \beta_t, \vec{\alpha}^{*u} \rangle, \tag{25}$$

i.e., it is a PWLC function of β_t, where $\langle \beta, \vec{\alpha} \rangle = \sum_{\vec{B}} \beta(\vec{B}) \vec{\alpha}(\vec{B})$ denotes inner product, and $\{\vec{\alpha}^{*u} : u = 1, \ldots, U\}$ is the set of hyperplane vectors. Letting u_t be the maximizing index in (25), the approximately optimal spectrum sensing action is $\mathcal{K}^{*u_t}$, associated with the maximizing hyperplane $\vec{\alpha}^{*u_t}$. The goal of PERSEUS is to determine such a set of hyperplanes and associated actions. With these, the CR then operates as follows: with the prior belief β_t at time-slot t, it selects the spectrum sensing action $\mathcal{K}_t = \mathcal{K}^{*u_t}$; it then collects the observation $\vec{Y}(t)$, computes the posterior belief as $\hat{\beta}_t = \hat{\mathbb{B}}(\beta_t, \mathcal{K}_t, \vec{Y}(t))$, performs spectrum access as in (14), and updates the next prior belief as $\beta_{t+1} = \mathbb{B}(\hat{\beta}_t; \vec{\theta})$, and so on.

To determine these hyperplanes and associated actions, PERSEUS is first preceded by an initial phase of exploration to determine a set of representative belief points $\tilde{\mathcal{B}}$, by allowing the CR to randomly interact with the radio environment (Step 1). With $\tilde{\mathcal{B}}$ thus determined, the hyperplanes and associated actions are computed iteratively through Steps 3-19, after initializing them in Step 2. At iteration i, starting from the entire set of beliefs, i.e., all indices $\{1, \ldots, U\}$ (Step 4), the algorithm selects one belief at random, $\beta_u \in \tilde{\mathcal{B}}$ (Step 6). For this belief point, it then performs the backup operation similarly to (20), by searching through all possible spectrum sensing actions (Step 7) to determine the optimal one that maximizes the value (Step 11). The value of a certain spectrum sensing action $\mathcal{K}$ is determined in Steps 8-10. To understand these steps, it is helpful to note that $\xi_{\mathcal{K}}^u(\vec{B})$ is the value accrued starting from state $\vec{B}$ and sensing action $\mathcal{K}$, so that $\langle \beta_u, \xi_{\mathcal{K}}^u \rangle$ done in Step 11 represents an expectation with respect to the belief β_u, i.e., the expected value of the sensing action $\mathcal{K}$ from belief β_u. $\xi_{\mathcal{K}}^u(\vec{B})$ is comprised of two components: the instantaneous reward $R(\vec{\phi}^*(\hat{\mathbb{B}}(\beta_u, \mathcal{K}, \vec{Y}_n)), \vec{B})$ accrued in state $\vec{B}$, after observing $\vec{Y}_n$ and computing the spectrum access decision as in (14); the discounted future reward $\sum_{\vec{B}'} P_B(\vec{B}'|\vec{B}; \vec{\theta}) \xi_{\mathcal{K},\vec{Y}_n}^u(\vec{B}')$, obtained by averaging with respect to the one-step transition from $\vec{B}$ to $\vec{B}'$; here, $\xi_{\mathcal{K},n}^u$ is the hyperplane associated to the future value function, determined in Step 10 as the one that maximizes the value function (with PWLC structure) based on the next prior belief, computed via the Hamming distance-based method of (24). Finally, $\xi_{\mathcal{K}}^u(\vec{B})$ is obtained by taking an expectation with respect to the observations using the aforementioned Monte-Carlo method: Step 9 generates N independent realizations of $\vec{Y}$ given $\vec{B}$ and $\mathcal{K}$, so that $1/N \sum_n [\cdot]$ in Step 8 converges to $\mathbb{E}_{\vec{Y}|\vec{B},\mathcal{K}}[\cdot]$ as $N \to \infty$.

Another key approximation of PERSEUS is to limit as much as possible the number of backup operations of Steps 7-11 by adding Steps 12-17: here, the algorithm scans through the set of unimproved belief points indexed by $\tilde{\mathcal{U}}$; it then checks whether the new hyperplane α_{i+1}^u determined in the backup operation improves the value of any remaining belief points; if it does for belief $\beta_{u'}$ (Step 13), then this new hyperplane (and its associated sensing action) becomes the relevant hyperplane (and the relevant sensing action) for this belief point (Step 14), and u' is removed from $\tilde{\mathcal{U}}$ (Step 15), so that the backup operation is not done for $\beta_{u'}$ at iteration i. These sequence of operations (random choice from $\tilde{\mathcal{U}}$, backup, check for improvement and removal) are performed iteratively until the set $\tilde{\mathcal{U}}$ is empty (Step 5): this constitutes a single PERSEUS iteration. Multiple iterations are executed until the value iteration updates converge (Step 3).

The computational time complexity of Algorithm 1 is $O(\tilde{T}|\tilde{\mathcal{B}}|^2 2^{2K'})$ [21], where $\tilde{T}$ denotes the number of iterations until convergence. Note here that incorporating Hamming distance state filters to alleviate the computational intractability inherent in PERSEUS belief update further mitigates the exponential dependence on the fragment size.

IV. Numerical Evaluations for LESSA

In this section, we evaluate numerically LESSA and compare it against state-of-the-art algorithms. The simulated radio environment constitutes $J_L = 3$ LUs accessing a 2.88MHz spectrum, discretized into $K = 18$ subcarriers, each of bandwidth of $W = 160$kHz, and one CR, as illustrated in Fig. 1. The 3 LUs access these 18 subcarriers according to a time-frequency Markovian correlation structure defined in Section II-B, with parameters $p_{0,0} = 0.1, p_{0,1} = p_{1,0} = 0.3, p_{1,1} = 0.7, q_0 = 0.3, q_1 = 0.8$. LUs' and CR's transmitters and receivers are deployed randomly across an operational region: the three LUs' transmitters are placed at positions (−225, 200)m, (225, 200)m and (0, −300)m, at a height of 40m; the corresponding LU receivers are stationary nodes, placed randomly within a circular radius of 200m from the respective LU's transmitter. The CR's transmitter is located at position (0, 0)m at a height of 20m; the corresponding receiver is at ground level, and moves along a randomly generated trajectory within a radius of 100m from the CR's transmitter. As described below, we have also facilitated rate adaptation at the CR based on the perceived SINR, which may vary with time and channel indices. The CR is capable of sensing $\kappa = 6$ subcarriers per time-slot.

For each transmitter (Tx, LU or CR) and receiver (Rx; the intended one, or unintended, thus experiencing interference) pairs, we model the channel on the kth subcarrier as $\hbar_k = \sqrt{\psi}\omega_k$, where ψ and $\omega_k \in \mathbb{C}$ encapsulate the large- and small-scale channel variations, respectively – with $\mathbb{E}[|\omega_k|^2] = 1$. Given the angle of elevation between the Tx/Rx pair under consideration, $\chi \in (0, \frac{\pi}{2}]$, we generate the line-of-sight (LoS) condition with probability $P_{\mathrm{LoS}}(\chi) = \frac{1}{1+z_1 e^{-z_2(\chi - z_1)}}$ [39], so that the non-LoS (NLoS) condition occurs with probability $P_{\mathrm{NLoS}}(\chi) = 1 - P_{\mathrm{LoS}}(\chi)$, where z_1, z_2 are parameters specific to the propagation environment. Given the LoS or NLoS condition, and the distance d between Tx and Rx, we then generate the large- and small-scale channel conditions as follows. If the channel is LoS: the large-scale fading coefficient ψ is modeled as $\psi_{\mathrm{LoS}}(d) = \psi_0 d^{-\mu_L}$, where ψ_0 is the reference pathloss at a distance of 1m from the Tx and $\mu_L \geq 2$ is the LoS pathloss exponent; the small-scale fading coefficient ω_k is modeled as a Rician with K-factor $\mathbb{K}(\chi) = f_1 e^{f_2 \chi}$, where f_1, f_2 are parameters specific to the propagation environment. Conversely, if the channel is NLoS: the large-scale fading coefficient ψ is modeled as $\psi_{\mathrm{NLoS}}(d) = \iota \psi_0 d^{-\mu_N}$, where $\iota \in (0, 1]$ is the additional NLoS attenuation and $\mu_N \geq \mu_L$ is the NLoS pathloss exponent; the small-scale fading coefficient ω_k is modeled as Rayleigh distributed (i.e., Rician with K-factor $\mathbb{K}(\chi) = 0$). Throughout the simulation, we use $\mu_L = 2.0$, $\mu_N = 2.8$, $\iota = 0.2$, $W = 160$ kHz, $f_1 = 1.0$, $f_2 = 0.0512$, $z_1 = 9.12$, and $z_2 = 0.16$ [40].

With this channel model, the SINR between the ith transmitter (LU or CR) and its intended receiver on subcarrier k is given by

$$\mathrm{SINR}_{k,i} = \frac{|\hbar_{k,i,i}|^2 P_{k,i}}{\sigma_V^2 + \sum_{j \neq i} |\hbar_{k,j,i}|^2 P_{k,j}},$$

where $\hbar_{k,i,i}$ is the channel between the ith transmitter and its indented receiver, $P_{k,i}$ is the transmission power; σ_V^2 is the noise power and $\sum_{j \neq i} |\hbar_{k,j,i}|^2 P_{k,j}$ is the contribution from the active interferers: $\hbar_{k,j,i}$ is the channel between the jth interfering transmitter and the ith receiver, with transmission power $P_{k,j}$ ($P_{k,j} = 0$ if such transmitter is idle on subcarrier k). We can then define the link capacity for such transmitter-receiver pair on subcarrier k as

$$C_{k,i} = W \log_2\big(1 + \mathrm{SINR}_{k,i}\big).$$

We assume that the LUs transmit with fixed transmission rate of $\Phi_{\mathrm{LU}} = 0.9$Mbps (per subcarrier), and thus incur outage if $\mathrm{SINR} < 2^{\Phi_{\mathrm{LU}}/W} - 1$. We now detail the choice of the transmission rate at the CR, Φ_{CR}. Assuming that the large-scale fading coefficient ψ and the K-factor $\mathbb{K}$ are known throughout the simulation period (since they represent large-scale parameters, they can be reliably estimated), the outage probability for an interference-free link is given by [39]

$$\begin{aligned} P_{\mathrm{out}}(\Phi_{\mathrm{CR}}, \psi, \mathbb{K}) &= \mathbb{P}\big(C_{k,i} < \Phi_{\mathrm{CR}} | \psi, \mathbb{K}\big) \\ &= 1 - Q_1\left(\sqrt{2\mathbb{K}}, \sqrt{\frac{2(\mathbb{K}+1)\sigma_V^2}{\psi P_T}\left(2^{\frac{\Phi_{\mathrm{CR}}}{W}} - 1\right)}\right), \end{aligned}$$

where Q_1 denotes the standard Marcum Q-function. Herein, we select the rate for the CR as the one that maximizes the expected throughput $\Phi_{\mathrm{CR}} \cdot [1 - P_{\mathrm{out}}(\Phi_{\mathrm{CR}}, \psi, \mathbb{K})]$, obtained efficiently via a bisection method [41].

To evaluate the performance, we define the average CR's throughput over T time-slots as

$$C^{\mathrm{CR}} = \frac{1}{T} \sum_{t=1}^{T} \sum_{k=1}^{K} \Phi_{\mathrm{CR},k}(t) \phi_k(t) \mathcal{I}$$

Move to one line

Remove "X"

$$\cancel{\times} \left[\mathrm{SINR}_{\mathrm{CR},k}(t) \geq 2^{\frac{\Phi_{\mathrm{CR},k}(t)}{W}} - 1\right], \tag{26}$$

where $\Phi_{\mathrm{CR},k}(t)$ is the rate used by the CR on subcarrier k in time-slot t, based on the rate adaptation scheme described earlier, and $\vec{\phi}(t)$ denotes the spectrum access decision in time-slot t; we define the LUs' throughput over the same T time-slots, normalized by the number of transmissions as

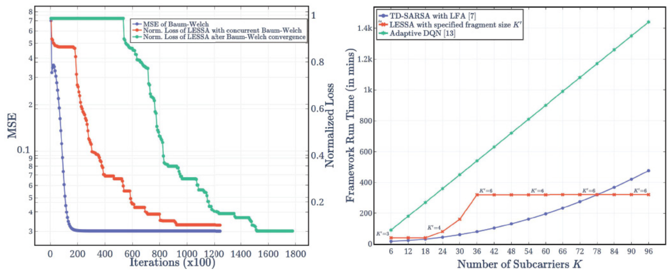
Move to one line

$$C^{\mathrm{LUs}} = \frac{\sum_{t=1}^{T} \sum_{k=1}^{K} \Phi_{\mathrm{LU}} B_k(t) \mathcal{I}\left[\mathrm{SINR}_{\mathrm{LU},k}(t) \geq 2^{\frac{\Phi_{\mathrm{LU}}}{W}} - 1\right]}{\sum_{t=1}^{T} \sum_{k=1}^{K} B_k(t)}, \tag{27}$$

where $\mathrm{SINR}_{\mathrm{LU},k}(t)$ is the SINR of the occupying LU, computed based on the channel realizations as described earlier.

In Fig. 4a, we plot the Mean Square Error (MSE) of the Baum-Welch algorithm versus the number of EM steps (i), $\|\vec{\theta} - \hat{\vec{\theta}}^{(i)}\|_2^2$, averaged over multiple independent realizations. The parameters are initialized as $p_{u,v} = 0.5, \forall u, v \in \{0, 1\}$ and $q_w = 0.5, w \in \{0, 1\}$. We observe that the MSE decreases over time, as the estimation progresses through the E- and M-steps [20], and converges to an MSE value of 0.03 after $\sim 5 \times 10^4$ iterations: this corresponds to an observation and estimation period of 160s, considering a typical time-slot duration of 3ms.

In Fig. 4(a), we also plot the convergence of the PERSEUS algorithm, using a discount factor of $\gamma = 0.9$ and a termination threshold of $\epsilon = 10^{-5}$, $\lambda = 1$, for both cases in which it is run concurrently with the Baum-Welch algorithm (red curve)



(a) Convergence visualization (b) Computational Bench-marking

Fig. 4. MSE convergence of the Baum-Welch algorithm to estimate $\vec{\theta}$, and normalized loss incurred by Algorithm 1 (a); computational time complexity bench-marking of LESSA against state-of-the-art (b).

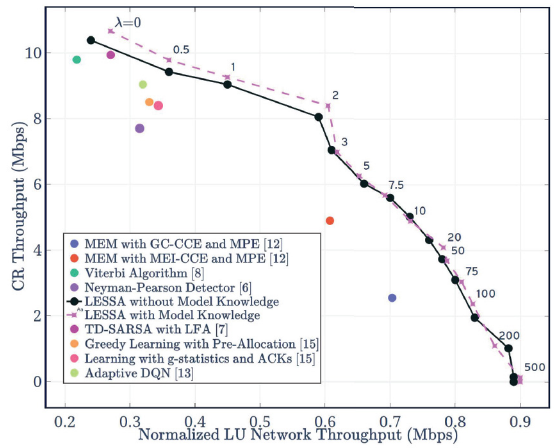


Fig. 5. Evaluation of CR and LU network throughputs for different values of λ with rate adaptation at the CR and with/without correlation model foreknowledge – along with comparisons with the state-of-the-art.

or after its convergence (green curve). To evaluate convergence, we use a normalized loss metric, defined as the loss of utility (defined in (12)) with respect to an oracle that performs spectrum access based on knowledge of the current occupancy state $\vec{B}(t)$, defined mathematically as

$$1-\frac{R\left(\vec{\phi}^*\left(\hat{\beta}_t\right),\vec{B}(t)\right)}{\max_{\vec{\phi}\in\{0,1\}^K} R\left(\vec{\phi},\vec{B}(t)\right)},$$

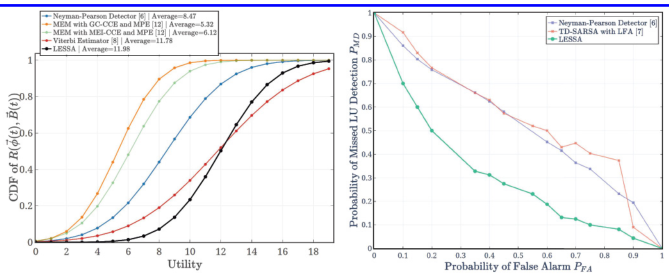
averaged out over time and multiple realizations, where $\vec{\phi}^*(\hat{\beta}_t)$ is the spectrum access decision defined in (14), based on the posterior belief $\hat{\beta}_t$ generated in our POMDP formulation, and $\max_{\vec{\phi}\in\{0,1\}^K} R(\vec{\phi},\vec{B}(t))$ is the Oracle reward, which uses knowledge of $\vec{B}(t)$. We find that the concurrent method converges in approximately half the time of the non-concurrent one. Remarkably, the normalized loss is around 5% after convergence, i.e., LESSA performs on par with an Oracle with knowledge of the current spectrum occupancy – a significant result considering the lack of *a priori* knowledge of the underlying Markov transition model and the noisy and constrained spectrum sensing environment. In addition, since the Oracle performs better than the *optimal POMDP* policy, this result demonstrates that solving for the *optimal POMDP* policy (a computationally intractable task) does not yield a tangible boost in spectrum white-space detection, thereby legitimizing the validity of the approximate POMDP approach proposed in this paper. Fig. 4(b) plots the run time of LESSA as a function of the number of subcarriers, against TD-SARSA with LFA [6] and Adaptive DQN [2]. The algorithms are run on a 2×12-core Intel Xeon Gold 6126 @ 2.6GHz compute node with 192GB RAM [42]: as the number of subcarriers increases, our solution scales better yielding a more computationally tractable performance relative to the other two, owing to fragmentation and belief update simplification heuristics. Interestingly, for $K \geq 36$ with fixed fragment size of $K' = 6$, the run time of LESSA flattens out: this is because LESSA is carried out in parallel across all 6-subcarrier fragments.

In Fig. 5, we compare the performance of LESSA with state-of-the-art algorithms, in terms of the CR's network throughput achieved vis-à-vis LUs throughput, as defined in (26) and (27), respectively. We note that LESSA covers the entire trade-off region by varying the penalty weight λ in the reward metric of the POMDP. Generally, Fig. 5 depicts a trend of increasing CR throughput and decreasing LUs' throughput (due to increased CR's interference), as the penalty λ is decreased. Therefore, our framework provides a crucial practical tool in CR MAC design: the ability to tune the trade-off between the throughput obtained by the CR and the interference caused by it to LU transmissions in the network. In contrast, the other state-of-the-art algorithms do not offer such capability, hence they attain a single point in the trade-off region. By comparing LESSA with unknown model (learned with the Baum-Welch algorithm) and LESSA with model known *a priori*, we observe that prior knowledge of the model offers a meagre 3.75% improvement in CR throughput for a given LU network throughput, compared to the proposed online concurrent model estimation and policy solver strategy – a testament to the accuracy of our estimator. Comparing LESSA with other state-of-the-art schemes, we observe consistent improvement in performance across the entire trade-off region. Specifically:[1]

- Minimum Entropy Merging (MEM) with Greedy Clustering (GC) based Channel Correlation Estimation (CCE) and Markov Process Estimation (MPE) [3]: Correlation Threshold 0.7; our solution offers a 115% improvement over this strategy;
- MEM with Minimum Entropy Increment (MEI) based CCE and MPE [3]: Correlation Threshold 0.7; our solution achieves 40% better performance over this strategy. In both GC and MEI implementations, our improved performance is owed to the POMDP formulation instead of an offline correlation-coefficient based clustering scheme employed in their work;
- Imperfect HMM-MAP State Estimation [9]: it uses the Viterbi algorithm; although [9] assumes time-only correlation and no sensing restrictions, we extend it to our setup to include *a priori* knowledge of the time-frequency

[1] Unless otherwise stated, all these algorithms have a spectrum sensing restriction of $\kappa = 6$; implementation details can be found in the respective references; the percentage improvements are all referred to improvements in CR's throughput provided by LESSA under the same LUs' throughput achieved by the state-of-the-art scheme under consideration.

"Figure Placement Comment": Use both columns of this double-column paper: Place this figure (Fig. 6) on the left column and Fig. 7 on the right column



(a) CDF of utility (b) ROC

Fig. 6. CDF of the utility $R(\vec{\phi}(t), \vec{B}(t))$ (see (12)), and comparison with the state-of-the-art (a); The receiver operating characteristics of LESSA versus those of a Neyman-Pearson Detector [13] and TD-SARSA with LFA [6].

correlation model, and sensing restriction of $\kappa = 6$; our solution attains a 6% boost over this strategy owing to an adaptive sensing policy exploited in our solution as opposed to a fixed one in their system, despite the lack of a-priori model knowledge;

- Neyman-Pearson Detection [13]: it assumes time-frequency independence, and no channel sensing restrictions, an AND fusion rule across 300 samples, and threshold determination via a false alarm probability of 30%; our solution offers a 26% enhancement over this strategy, by leveraging the time-frequency LU occupancy dynamics in our framework in contrast to the independence assumptions made in their model, despite the sensing restrictions of our model;
- Temporal Difference Learning via SARSA with Linear Function Approximation (LFA) [6]: we use a belief update heuristic constant of 0.9, a discount factor of 0.9, a fixed exploration factor of 0.01, and a raw false alarm probability of 5%; our solution exhibits a 3% superior performance over this strategy, owing to our additional exploitation of LU occupancy correlation across frequency as opposed to a purely temporal strategy employed in their model;
- Greedy Learning under Pre-Allocation [11]: it uses a time-varying exploration factor, and assumes independence across both time and frequency;
- Learning with g-statistics and ACKs [11]: our solution achieves a 13% boost in performance over this strategy. In both implementations of [11], LESSA offers enhancements by leveraging the time-frequency correlation structure in LUs' spectrum occupancy, in contrast to the independence assumptions made in their model;
- Adaptive Deep Q-Networks [2]: with Experiential Replay (Memory Size 10^6), 2048 input neurons, 4096 neurons with ReLU activation functions in each of the 2 hidden layers of the neural network, a MSE cost function with an Adam optimizer, a fixed exploration factor of 0.1, a learning rate of 10^{-4}, a batch Size of 32; our solution offers a 9% improvement over this strategy, owing to a parametric time-frequency correlation model that captures the correlation in LU occupancy, as opposed to a model-free policy optimization scheme used in their system.

In Fig. 6(a), we plot the Cumulative Distribution Function (CDF) of the reward $R(\vec{\phi}(t), \vec{B}(t))$, evaluated according to (13). We find that LEMMA achieves an average utility of 11.98 per time-step, 125% higher than that achieved by the MEM with GC based CCE and MPE algorithm from [3], 96% higher than that achieved by the MEM with MEI based CCE and MPE algorithm from [3], and 42% higher than that attained by the Neyman-Pearson Detector [13] (despite the latter having no sensing restrictions). Compared to the Viterbi algorithm [9] (= 11.78), our scheme achieves 2% higher utility (= 11.98), thanks to an adaptive sensing strategy.

Finally, in Fig. 6(b), we illustrate the receiver operating characteristic, i.e., the trade-off between false alarm probability ($P_{FA} = \mathbb{P}(\phi_k(t) = 0|B_k(t) = 0)$), i.e., the subcarrier is incorrectly detected as occupied, and it is therefore not used by the CR) versus missed detection probability ($P_{MD} = \mathbb{P}(\phi_k(t) = 1|B_k(t) = 1)$), i.e., the subcarrier is incorrectly detected as idle, and it is therefore used by the CR), averaged out over time-slots and subcarriers, for LESSA ($\kappa = 6$), and state-of-the-art TD-SARSA with LFA ($\kappa = 6$) [6] and Neyman-Pearson Detection (no sensing restrictions) [13]. Although the algorithms in [6] and [13] do not inherently possess means to regulate the trade-off between CR throughput and LU network interference, for the sake of this evaluation, we have modified these algorithms to enable such a mechanism: since [13] involves energy detection, false alarm constraints can be brought in by modifying the detection threshold; and we add this regulation to the observation and access decision reliability control logic in [6]. In line with the results of Fig. 5, we observe that LESSA consistently achieves lower P_{MD} (hence lower interference to LUs) across all ranges of P_{FA}. For instance, at a false alarm probability target of $P_{FA} = 0.7$, LESSA achieves $P_{MD} = 0.125$, as opposed to 0.36 by the Neyman-Pearson Detector (despite no sensing restrictions) [13] and 0.45 by TD-SARSA with LFA [6].

V. MA-LESSA: Algorithms and Evaluation

In this section, we extend LESSA to both centralized and distributed multi-agent deployments (MA-LESSA). With respect to distributed deployments, we propose novel neighbor discovery and channel access rank allocation schemes that are embedded into LESSA – as depicted in Fig. 3 – along with steps for sharing localized observations and access decisions with neighbors, as well as the corresponding data aggregation. The proposed neighbor discovery scheme involves an adapted version of the Multi-band Directional Neighbor Discovery (MDND) algorithm outlined in [22], with RSSI-thresholding based modifications; furthermore, inspired by quorum-based voting heuristics for cluster fallback mechanisms in Message Oriented Middleware frameworks [23], we detail an access order determination scheme for cooperative access among CRs in the ensemble. As a part of our centralized deployment analysis, we retrofit MA-LESSA into the Purdue BAM! Wireless CR network [25], with a gateway CR handling flow scheduling to individual nodes in the network, making multi-hop routing decisions, collaborating with other competitor networks, and ensuring that Quality of Service (QoS) mandates are being met throughout the scenario emulation period on the DARPA SC2 Colosseum [18].

Use Italics to emphasise this block instead of bold text to maintain homogeneity throughout the article

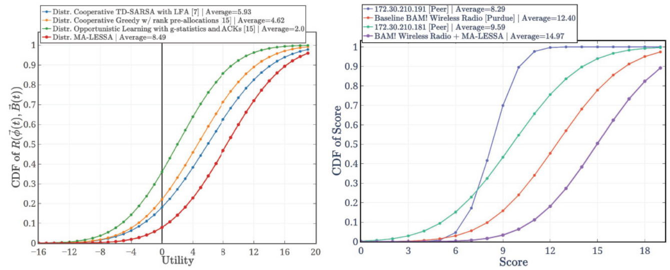
Neighbor Discovery via RSSI-Thresholding: The MDND scheme detailed in [22] employs the 2.4GHz Wi-Fi band as a control channel for neighbor discovery in 60GHz data environments. Adapting this scheme to multi-agent CR deployments, we designate the band-edges as the control channel on which CRs communicate and coordinate their neighborhoods for channel access order allocation and data aggregation. At a pre-determined schedule, each CR 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 CR checks if the expected RSSI of the radio signals corresponding to a certain node is above a threshold $\mathrm{RSSI}_{\mathrm{th}}$: if that is the case, it adds that node's identifier to its list of neighbors. The computational time complexity of the RSSI thresholding scheme for neighbor discovery is $O(J_C^2)$.

Channel Access Rank Allocation via Quorum-based Preferential Ballot: With a similar control channel strategy for channel access rank allocation, inspired by cluster fallback schemes in queueing environments in enterprise software systems [23], we employ a quorum-based preferential ballot scheme to determine the order in which the *estimated-idle* subcarriers are accessed by the CRs in the network. This procedure starts only after a quorum has been achieved, i.e., the number of neighbors identified by an CR should be equal to or exceed a node-specific pre-defined number. Over the control channel, each CR 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 CR 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 CR for a pre-specified consecutive period of time, a consensus has been reached, so that 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 CR, then the CR 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. The computational time-complexity of the quorum-based preferential ballot scheme for channel access rank allocation is $O(\tilde{T} J_C^2)$ [11], where $\tilde{T}$ corresponds to the number of iterations involved until a consensus is reached – which depends on the mobility patterns of these nodes along with the temporal evolution of the peer-to-peer link qualities.

A. Distributed MA-LESSA

Here, we evaluate the operational capabilities of MA-LESSA in distributed settings. As illustrated in Fig. 1, operating under the same signal and observation models as in Section II, consider a network of 3 LUs operating in an 18-subcarrier radio environment, with their occupancy behaviors governed by Markovian time-frequency correlation structure ($\vec{\theta}$), and 12 CRs intelligently trying to access white-spaces in the spectrum (cooperatively [6] or opportunistically [11]), with an added restriction of being able to sense only 1 subcarrier per CR per time-slot.

See "Figure Placement Comment" on previous page: Use both columns (Fig. 6 | Fig. 7) on a page



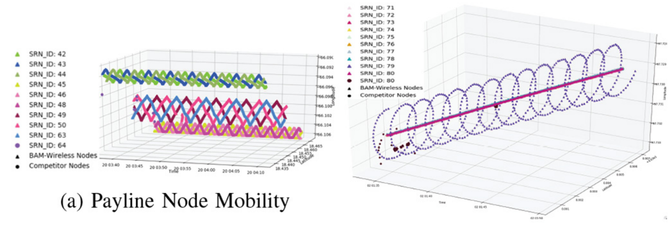
(a) Distributed MA-LESSA (b) Centralized MA-LESSA

Fig. 7. CDF of the utility $R(\vec{\phi}(t), \vec{B}(t))$ (see (12)) for distributed MA-LESSA against state-of-the-art (a); CDF of the Score achieved during the DARPA SC2 Active Incumbent scenario emulation: comparison between BAM! Wireless CR network design retrofitted with MA-LESSA and other competitor network radio designs (including the baseline BAM! Wireless CR network) (b).

Fig. 7(a) plots the CDF of the utility achieved by MA-LESSA against other distributed multi-agent schemes in the state-of-the-art. We find that MA-LESSA outperforms the distributed, cooperative, ϵ-greedy TD-SARSA with Linear Function Approximation framework from [6] by 43%; it also outperforms the distributed, cooperative, time-decaying ϵ-greedy algorithm with channel access rank pre-allocations from [11] by 84%; and it outperforms the distributed, opportunistic, g-statistics algorithm with ACKs (without channel access rank pre-allocations) from [11] by 324%. These enhancements can be attributed to the fact that MA-LESSA exploits the LUs' occupancy correlation structure across both time and frequency (vs time only of [6] and the independence of [11]) and it leverages collaboration among CRs in the network for access order determination and sensed data aggregation.

B. Centralized MA-LESSA: DARPA SC2 Emulations

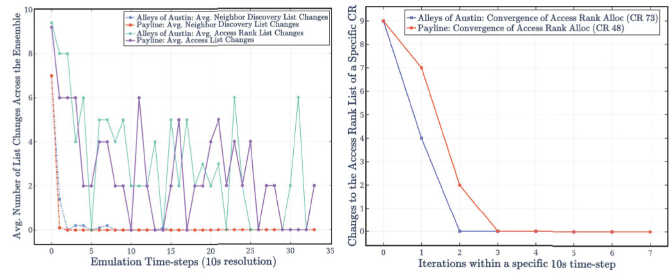
In order to evaluate the performance of MA-LESSA in real-world settings, we retrofit it into the MAC layer (channel & bandwidth allocation) of our BAM! Wireless radio [25], by leveraging the aggregated PSD measurements obtained at the gateway node of our BAM! Wireless network, as shown in Fig. 2(b). We analyze its operational capabilities in the DARPA SC2 Active Incumbent scenario [17] emulated on the Colosseum [43]. The DARPA SC2 Active Incumbent scenario consists of a Terminal Doppler Weather Radar (TDWR) system functioning as the LU, and 5 competitor networks (ours included), each comprising 2 UNII WLANs: 2 Access Points (APs) and 4 STAtions (STAs) per AP, serving as the CRs, in a 10MHz radio environment (995MHz to 1005MHz), for 330 seconds of emulation on the Colosseum [17]. 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



(a) Payline Node Mobility

(b) Alleys of Austin Node Mobility

Fig. 8. The mobility patterns of constituent nodes in the DARPA SC2 Payline (a) and Alleys of Austin (b) scenarios. A Standard Radio Node (SRN) is a CR in DARPA SC2.



(a) List Changes

(b) Access Rank Allocation

Fig. 9. Evaluation of neighbor discovery and channel access rank allocation: average number of neighbor discovery and channel access rank list changes across the ensemble (a); and convergence visualization of channel access rank allocation in a specific 10s time-step for specific CRs in the network (b).

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, then the Individual Mandates (IMs) associated with the flow are said to have been met.

Let $\mathcal{V}_t$ be the set of IMs achieved by a participant network in time-slot t, and let p_v be the points received for satisfying an IM $v \in \mathcal{V}_t$: we define the *score* of a participant network corresponding to a certain time-slot t as $\sum_{v \in \mathcal{V}_t} \mathrm{p}_v$ [17]. We evaluate the performance of MA-LESSA retrofitted into our standard BAM! Wireless radio [25], and compare it against a baseline channel and bandwidth allocation scheme that involves the weighted combination of PSD observations and collaboration data received from the other competing networks (titled "Baseline BAM! Wireless Radio [Purdue]"), and against the designs of other competitors (identified by their collaboration network registered IP address [26], "172.30.210.191 [Peer]" and "172.30.210.181 [Peer]"). Fig. 7(b) plots the CDF of the score achieved: remarkably "BAM! Wireless Radio + MA-LESSA" outperforms the baseline BAM! Wireless design by 21% in average score, and outperforms the competitors by 56% and 81%, respectively. This result shows the potential of MA-LESSA to optimize higher-layer metrics (the network score) thanks to improved detection of white-spaces. To evaluate the proposed neighbor discovery (RSSI thresholding) and channel access rank allocation (quorum-based preferential ballot) heuristics from a computational time complexity perspective, we retrofit these schemes into the control channel design, collaboration, and data aggregation modules of the Purdue BAM! Wireless radio [25], and analyse their feasibility in emulations of highly mobile real-world scenarios – namely, military deployments in the Alleys of Austin scenario [32] (urban: $5 \times [9 - \text{guardsmen} + 1 - \text{UAV}]$) and disaster relief deployments in the Payline scenario [31] (urban: $5 \times [9 - \text{EMTs} + 1 - \text{HQ}]$). Along with the scenario-specific node mobility patterns (Fig. 8), the results of these emulations are shown in Fig. 9: neighbor discovery list changes are minimal in spite of node mobility (with an RSSI threshold of 22dB), and distributed convergence of channel access rank allocation across the ensemble is achieved within the first few iterations in any given 10s time-step.

C. LESSA on an ESP32 WLAN Testbed

We employ 8 ESP32 radios [19], each embedded on a GCTronic e-puck2 robot [24], categorized into a network of 3 LUs (and their 3 corresponding sinks) occupying 6 subcarriers in the 2.4GHz Wi-Fi spectrum; and 2 independent CRs, each having the capability to sense only one channel at a time. With 2 CRs, the overall sensing restriction of the CR network is thus $\kappa = 2$ (work split over 2 ESP32 radios due to design limitations). The 3 LUs occupy the spectrum according to a Markovian time-frequency correlation structure (described by (7)), with parameters $p_{0,0} = 0.1$, $p_{0,1} = p_{1,0} = 0.3$, $p_{1,1} = 0.7$ and $q_0 = 0.3$, $q_1 = 0.8$. We simulate the occupancy behavior of the LUs according to this time-frequency correlation structure, and solve for the spectrum sensing and access policy using LESSA, by mimicking the observational capabilities of the actual ESP32 radios. Note that this step is performed on a PC.

The simulated LU occupancy behavior and the time-slot specific channel access decisions (derived through LESSA), are stored in databases for export onto the ESP32 network. In time-slot t, a peer-to-peer communication link l_{ij} is established between an ESP32 LU $j \in \{1, 2, 3\}$ and its designated sink $i \in \{1, 2, 3\}$, over a subcarrier $k_{l_{ij}} = k \in \{1, 2, \ldots, 6\}$, as determined by the exported LU occupancy database. These channel allocations are done orthogonally at the LUs, so that $k_{l_{ij}} \neq k_{l_{i',j'}}, \forall i, i' \in \{1, 2, 3\},\ j, j' \in \{1, 2, 3\}$. In the next synchronized time-slot $t+1$, this link l_{ij} moves to channel $k' \in \{1, 2, \ldots, 6\}$, again as determined by the LU's behavioral rules maintained in the exported occupancy database. This same procedure takes place for the other two LU communication links in every time-slot until the end of the evaluation period. Although the PC-based POMDP solver employs a single CR which can access 2 subcarriers at a time in order to deliver its flows (see the access part of the POMDP formulation in Section II-D), we employ 2 ESP32 CR 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, each able to access only one channel at a time. More specifically, we split the 2-channel access decision in time-slot t, as determined by the time-slot specific POMDP channel access database, into a 1-channel access action at each ESP32 CR radio. Next, based on whether the channel access at the 2 ESP32 CR radios was successful, we compute the success rate as $\frac{1}{2} \sum_{j=1}^{2} \mathcal{I}[B_{k_{CR_j}}(t) = 0]$, where $B_{k_{CR_j}}(t) \in \{0, 1\}$ is the occupancy variable of the channel accessed by CR j in time-slot t. The implementation of LESSA on this ad-hoc WLAN testbed demonstrates a channel access success probability of 96% across an evaluation period of 300s.

VI. Conclusion

In this paper, we formulate a learning-based spectrum sensing and access problem in resource-constrained radio ecosystems as an approximate POMDP, which leverages learning of the LU occupancy correlation model via the Baum-Welch algorithm and solving for an approximately optimal sensing and access policy via the PERSEUS algorithm. We propose fragmentation, Hamming distance state filter heuristics, and Monte-Carlo methods to alleviate the inherent computational complexity of PERSEUS. Through system simulations, we demonstrate the advantages of exploiting the correlation structure – as opposed to Neyman-Pearson Detection which assumes independence; adapting the spectrum sensing decision to enhance white-space detection – as opposed to Viterbi, which uses a fixed sensing strategy; and a parametric learning and adaptation framework – over model-free Deep Q-Network designs. 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 LU occupancy behavior. Our framework enables a critical feature in practical scenarios: the ability of the CR to regulate the interference caused to LUs, 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," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2021, pp. 1–6.

[2] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, "Deep reinforcement learning for dynamic multichannel access in wireless networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 2, pp. 257–265, Jun. 2018.

[3] 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 *Proc. IEEE Mil. Commun. Conf.*, Oct. 2014, pp. 405–410.

[4] L. Yu, Q. Wang, Y. Guo, and P. Li, "Spectrum availability prediction in cognitive aerospace communications: A deep learning perspective," in *Proc. Cogn. Commun. Aerosp. Appl. Workshop (CCAA)*, 2017, pp. 1–4.

[5] B. M. Pati, M. Kaneko, and A. Taparugssanagorn, "A deep convolutional neural network based transfer learning method for non-cooperative spectrum sensing," *IEEE Access*, vol. 8, pp. 164529–164545, 2020.

[6] J. Lundén, S. R. Kulkarni, V. Koivunen, and H. V. Poor, "Multiagent reinforcement learning based spectrum sensing policies for cognitive radio networks," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 5, pp. 858–868, Oct. 2013.

[7] L. Ferrari, Q. Zhao, and A. Scaglione, "Utility maximizing sequential sensing over a finite horizon," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3430–3445, Jul. 2017.

[8] N. Michelusi and U. Mitra, "Cross-layer estimation and control for cognitive radio: Exploiting sparse network dynamics," *IEEE Trans. Cogn. Commun. Netw.*, vol. 1, no. 1, pp. 128–145, Mar. 2015.

[9] C.-H. Park, S.-W. Kim, S.-M. Lim, and M.-S. Song, "HMM based channel status predictor for cognitive radio," in *Proc. Asia–Pac. Microw. Conf.*, Dec. 2007, pp. 1–4.

[10] K. Cohen, Q. Zhao, and A. Scaglione, "Restless Multi-Armed Bandits under time-varying activation constraints for dynamic spectrum access," in *Proc. 48th Asilomar Conf. Signals Syst. Comput.*, Nov. 2014, pp. 1575–1578.

[11] A. Anandkumar, N. Michael, and A. Tang, "Opportunistic spectrum access with multiple users: Learning under competition," in *Proc. 29th Conf. Inf. Commun. (INFOCOM)*, 2010, pp. 803–811.

[12] N. Michelusi, M. Nokleby, U. Mitra, and R. Calderbank, "Multi-scale spectrum sensing in dense multi-cell cognitive networks," *IEEE Trans. Commun.*, vol. 67, no. 4, pp. 2673–2688, Apr. 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 *Proc. IEEE EUROCON*, May 2009, pp. 1420–1425.

[14] S. Maleki, S. P. Chepuri, and G. Leus, "Energy and throughput efficient strategies for cooperative spectrum sensing in cognitive radios," in *Proc. IEEE 12th Int. Workshop Signal Process. Adv. Wireless Commun.*, Jun. 2011, pp. 71–75.

[15] M. Shafi *et al.*, "5G: A tutorial overview of standards, trials, challenges, deployment, and practice," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 6, pp. 1201–1221, Jun. 2017.

[16] S. Yin, D. Chen, Q. Zhang, M. Liu, and S. Li, "Mining spectrum usage data: A large-scale spectrum measurement study," *IEEE Trans. Mobile Comput.*, vol. 11, no. 6, pp. 1033–1046, Jun. 2012.

[17] DARPA-SC2-Freshdesk, *Active Incumbent Scenario Specifications*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2019. [Online]. Available: https://sc2colosseum.freshdesk.com/support/solutions/articles/22000239489-active-incumbent-

[18] M. Rosker, *Spectrum Collaboration Challenge (SC2)*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2018. [Online]. Available: https://www.darpa.mil/program/spectrum-collaboration-challenge

[19] *Espressif ESP32: A Different IoT Power and Performance*, Espressif Syst. (Shanghai) Company Ltd., Shanghai, China, 2019. [Online]. Available: https://www.espressif.com/en/products/hardware/esp32/overview

[20] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[21] M. T. J. Spaan and N. Vlassis, "Perseus: Randomized point-based value iteration for POMDPs," *J. Artif. Int. Res.*, vol. 24, no. 1, pp. 195–220, Aug. 2005.

[22] H. Park, Y. Kim, T. Song, and S. Pack, "Multiband directional neighbor discovery in self-organized mmwave ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 3, pp. 1143–1155, Mar. 2015.

[23] *Clusters: ActiveMQ Artemis Documentation*, ActiveMQ Artemis Document., Forest Hill, MD, USA, 2014. [Online]. Available: https://activemq.apache.org/components/artemis/documentation/latest/clusters.html

[24] *Epuck2 Specifications and General Wiki*, GCTronic e-puck2 Online Wiki, Lugano, Switzerland, 2020. [Online]. Available: https://www.gctronic.com/doc/index.php/e-puck2

[25] *Purdue University BAM! Wireless Radio*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2019. [Online]. Available: https://archive.darpa.mil/sc2/news/spectrum-collaboration-challenge-awards-four-teams-with-half-prizes

[26] DARPA SC2 GitLab CIL Schematics, *CIL Specifications*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2019. [Online]. Available: https://gitlab.com/darpa-sc2-phase3/CIL

[27] F. A. P. D. Figueiredo *et al.*, "SCATTER PHY: A physical layer for the DARPA spectrum collaboration challenge," in *Proc. IEEE Int. Symp. Dyn. Spectrum Access Netw. (DySPAN)*, 2019, pp. 1–6.

[28] R. J. Baxley and R. S. Thompson, "Team zylinium DARPA spectrum collaboration challenge radio design and implementation," in *Proc. IEEE Int. Symp. Dyn. Spectrum Access Netw. (DySPAN)*, 2019, pp. 1–6.

[29] D. Stojadinovic, F. A. P. de Figueiredo, P. Maddala, I. Seskar, and W. Trappe, "SC2 CIL: Evaluating the spectrum voxel announcement benefits," in *Proc. IEEE Int. Symp. Dyn. Spectrum Access Netw. (DySPAN)*, 2019, pp. 1–6.

[30] S. Giannoulis *et al.*, "Dynamic and collaborative spectrum sharing: The SCATTER approach," in *Proc. IEEE Int. Symp. Dyn. Spectrum Access Netw. (DySPAN)*, 2019, pp. 1–6.

[31] DARPA-SC2-Freshdesk-PE2, *Payline (2 Stage)*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2019. [Online]. Available: https://sc2colosseum.freshdesk.com/support/solutions/articles/22000234543-payline-2-stage-7065-

[32] DARPA-SC2-Freshdesk-CE, *Alleys of Austin With Points—5 Teams*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2019. [Online]. Available: https://sc2colosseum.freshdesk.com/support/solutions/articles/22000237879-pe2-alleys-of-austin-w-points-5-team-7013-

[33] R. I. C. Chiang, G. B. Rowe, and K. W. Sowerby, "A quantitative analysis of spectral occupancy measurements for cognitive radio," in *Proc. IEEE 65th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2007, pp. 3016–3020.

[34] 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 *Proc. 1st Int. Workshop Technol. Policy Accessing Spectr. (TAPAS)*, 2006.

[35] D. M. A. Haughton, "On the choice of a model to fit data from an exponential family," *Ann. Stat.*, vol. 16, no. 1, pp. 342–355, 1988.

[36] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artif. Intell.*, vol. 101, nos. 1–2, pp. 99–134, May 1998.

[37] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.*, vol. 37, no. 6, pp. 1554–1563, Dec. 1966.

[38] J. Pineau, G. Gordon, and S. Thrun, "Point-based value iteration: An anytime algorithm for POMDPs," in *Proc. 18th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2003, pp. 1025–1030.

[39] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, MA USA: Cambridge Univ. Press, 2005.

[40] Q. Feng, J. McGeehan, E. K. Tameh, and A. R. Nix, "Path loss models for air-to-ground radio channels in urban environments," in *Proc. IEEE 63rd Veh. Technol. Conf.*, vol. 6, 2006, pp. 2901–2905.

[41] Y. Sun, A. Baricz, and S. Zhou, "On the monotonicity, log-concavity, and tight bounds of the generalized marcum and nuttall q-functions," *IEEE Trans. Inf. Theory*, vol. 56, no. 3, pp. 1166–1186, 2010.

[42] D. Duplyakin *et al.*, "The design and operation of CloudLab," in *Proc. USENIX Annu. Tech. Conf. (ATC)*, Jul. 2019, pp. 1–14.

[43] *Radio Command and Control API*, DARPA Spectr. Collab. Challenge (SC2), Arlington, VA, USA, 2018. [Online]. Available: https://sc2colosseum.freshdesk.com/support/solutions/articles/22000220460-radio-command-and-control-c2-api

Bharath Keshavamurthy (Graduate Student Member, IEEE) received the B.E. degree in electronics and communication engineering from Visvesvaraya Technological University, India, in 2016, the M.S. degree in electrical and computer engineering from Purdue University, West Lafayette, IN, USA, in 2020. After 15 months in the Ph.D. program in Electrical and Computer Engineering at Purdue University, he is currently pursuing a Ph.D. in Electrical Engineering at Arizona State University, Tempe, AZ, USA. Previous industrial Research and Development roles include a Graduate Intern at DC-NXOS, Cisco Systems Inc; a Senior Software Engineer, OSS R&D, Nokia Digital Platforms; and a Networking Intern, Radio Access Networks, Ericsson R&D. As a Graduate Research Assistant with Purdue University, he worked on the BAM! Wireless Cognitive Radio Design team for DARPA SC2 from 2018 to 2019. His current research interests include multifaceted UAV fleet automation, 3-D coverage for 6G cellular networks, millimeter-wave V2X propagation modeling and experimentation, and massive MIMO system design.

Nicolò Michelusi (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees (Hons.) from the University of Padova, Italy, in 2006 and 2009, respectively, the M.Sc. degree in telecommunications engineering from the Technical University of Denmark, Denmark, in 2009, as part of the T.I.M.E. double degree program, and the Ph.D. degree from the University of Padova in 2013. From 2013 to 2015, he was a Postdoctoral Research Fellow with the Ming-Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, CA, USA, and from 2016 to 2020, he was an Assistant Professor with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA. He is currently an Assistant Professor with the School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ, USA. His research interests include 5G/6G wireless networks, millimeter-wave communications, stochastic and distributed optimization, and federated learning over wireless. He received the NSF CAREER Award in 2021. He was the Co-Chair of the Distributed Machine Learning and Fog Network Workshop at IEEE INFOCOM 2021, the Wireless Communications Symposium at IEEE GLOBECOM 2020, the IoT, M2M, Sensor Networks, and Ad-Hoc Networking Track at IEEE VTC 2020, and the Cognitive Computing and Networking Symposium at ICNC 2018. He is an Associate Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and a reviewer for several IEEE journals.