Learning Optimal Sniffer Channel Assignment for Small Cell Cognitive Radio Networks

Lixing Chen[†], Zhuo Lu[*], Pan Zhou[‡], Jie Xu[†]
[†]Department of Electrical and Computer Engineering, University of Miami. Email: {lx.chen, jiexu}@miami.edu.
[*]Department of Electrical Engineering, University of South Florida. Email: zhuolu@usf.edu.
[‡]School of Cyber Science and Engineering, Huazhong University of Science and Technology. Email: panzhou@hust.edu.cn.

***Abstract*—To cope with the exploding mobile traffic in the fifth generation cellular network, the dense deployment of small cells and cognitive radios are two key technologies that significantly increase the network capacity and improve the spectrum utilization efficiency. Despite the desirable features, small cell cognitive radio networks (SCRNs) also face a higher risk of unauthorized spectrum access, which should not be overlooked. In this paper, we consider a passive monitoring system for SCRNs, which deploys sniffers for wireless traffic capture and network forensics, and study the optimal sniffer channel assignment (SCA) problem to maximize the monitoring performance. Unlike most existing SCA approaches that concentrate on user activity, we highlight the inherent error in wireless data capture (i.e. imperfect monitoring) due to the unreliable nature of wireless propagation, and propose an online-learning based algorithm called OSA (Online Sniffer-channel Assignment). OSA is a type of contextual combinatorial multi-armed bandit learning algorithm, which addresses key challenges in SCRN monitoring including the time-varying spectrum resource, imperfect monitoring, and uncertain network conditions. We theoretically prove that OSA has a sublinear learning regret bound and illustrate via simulations that OSA significantly outperforms benchmark solutions.**

I. Introduction

The fifth generation (5G) cellular network is becoming a major driver of many emerging technologies, e.g. the smart city, the intelligent transportation and the Internet of Things, which is expected to hail a new era of faster data speed, and greater ability to move massive data and support a diverse set of services. To meet the explosive growth of mobile data traffic and improve the spectrum efficiency in 5G networks, a variety of solutions have been proposed [1], among which the dense deployment of small cells and Cognitive Radio (CR) are two promising technologies [2]. As a key enabler of 5G, small cells dramatically increase the network capacity by exploiting spatial reuse and dynamic spectrum management [3], [4]. In addition, CR further enables low-cost opportunistic spectrum access for unlicensed users without causing harmful interference to the incumbent licensed users. Although the small cell cognitive radio network (SCRN) promises more flexible and efficient spectrum utilization, it also faces a higher risk of spectrum attack and unauthorized spectrum usage. In a line of recent works, several spectrum attack patterns and their countermeasures in CR have been investigated, such as spectrum sensing falsification [5] and primary user emulation [6]. To deal with these security threats, efficient traffic monitoring and network forensic methods [7] have to be developed, and a fundamental building block of these methods is wireless data sniffing, which is attracting increasing research attention recently [8], [9]. The idea is to deploy a dedicated set of hardware devices, called *sniffers*, to capture transmissions of wireless devices or activities of interference sources in their vicinity, and store packet level or PHY layer information in the trace file for analysis. Because a sniffer has a limited spectrum bandwidth for monitoring while most wireless networks utilize multiple contiguous or non-contiguous channels, a key problem in wireless sniffing is *sniffer channel assignment* (SCA), which concerns with determining which set of channels each sniffer should sniff.

The SCA problem has been investigated in traditional broadband multi-channel wireless networks to achieve the optimal sniffer-channel matching [10], [11] or in macrocell CR networks to deal with opportunistic access behaviors of secondary users [12], [13]. However, SCRN poses many new challenges for the SCA policy design. Firstly, unlike macrocell CR networks where the spectrum resource of a macro base station is relatively stable across time, the available spectrum resource for each small cell is time-varying as a result of the dynamic spectrum resource management [4] for handling the spatially and temporally volatile small cell mobile traffic. This time-varying spectrum resource results in much more complicated behavior of CR users for the opportunistic spectrum access, and many existing SCA strategies in macrocell [13], [14] cannot handle well the time-varying spectrum resource (more detailed discussion will be given in the Related Works). Secondly, most previous studies on SCA overlook the monitoring conditions by assuming perfect monitoring (i.e., all packets transmitted on a channel can be captured by a sniffer). In fact, imperfect monitoring is more common in practice because of the unreliable nature of wireless propagation. Particularly, the mmWave transmission used in 5G small cells is more sensitive to blockage and shadowing [15], which further increases the unreliability of wireless propagation and makes imperfect monitoring an important challenge for SCA in SCRNs. Thirdly, sniffer assignment has to be determined with many uncertainties in the network conditions, e.g., the users' traffic pattern and packet capture probabilities, which

L. Chen and J. Xu's work is supported in part by the Army Research Office under Grant W911NF-18-1-0343.

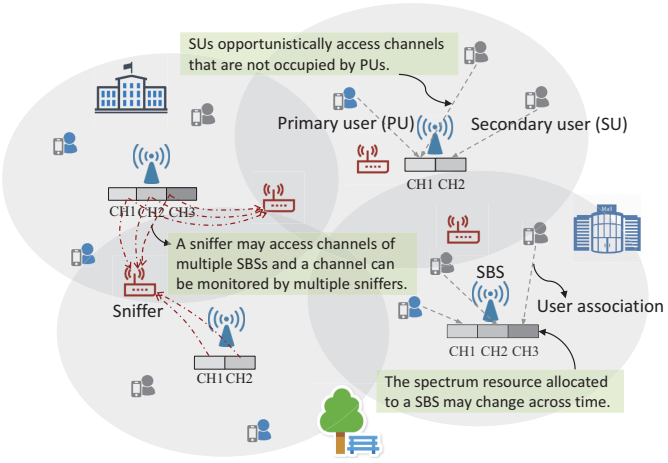


Fig. 1: Passive monitoring system for SCRN. The gray rectangles represent the channels operated by SBSs, which vary over time; a red dash line indicates that a channel can be monitored by a sniffer.

affect the sniffing performance in *a priori* unknown ways. Therefore, a learning-based SCA policy is much preferred to a solely optimization-based policy.

In this paper, we investigate the SCA problem in SCRNs and propose a new learning-based SCA algorithm, called Online Sniffer-channel Assignment (OSA). OSA is designed based on a novel bandit learning framework called Contextual Combinatorial Multi-Armed Bandit (CC-MAB) [16]. OSA is "contextual" because it lets sniffers use their own channel conditions, i.e., signal-to-interference-plus-noise ratio, as context information to infer the probability of successfully capturing packets (thereby featuring imperfect monitoring); OSA is "combinatorial" because it chooses a subset of channels to monitor and tries to maximize the monitoring reward by optimizing the (possibly redundant) sniffer assignment. One salient feature of CC-MAB compared to other MAB algorithms is that it allows the arms (i.e., the available channels to monitor for sniffers) to be different in each round and hence is suitable for the time-varying spectrum resource of small cells. Fig. 1 illustrates a passive monitoring system for SCRN considered in this paper. The main contributions of this paper are summarized as follow:

1) We formulate an SCA problem for SCRNs, which takes into account various special issues in SCRNs including time-varying spectrum resource of small cells, imperfect monitoring, and uncertainties in network condition. In particular, we allow redundant sniffer-channel assignment, i.e., multiple sniffers can be assigned to one channel for increased packet capture probability. The goal of the monitoring system is to maximize the amount of traffic captured by the sniffers.

2) Assuming that the packet capture probabilities of sniffers on all channels are known, we show that our SCA problem is a matroid-constrained submodular maximization problem. A greedy algorithm is developed to approximate the optimal solution with a performance guarantee.

3) With *a priori* uncertain knowledge of the packet capture probability of sniffers on each channel, the SCA problem is cast as a contextual combinatorial bandit learning problem. An online learning algorithm called Online Sniffer-channel Assignment (OSA) is developed, which uses the channel condition of sniffers as context and learns the packet capture probabilities over time for sniffer assignment. We analytically bound the utility loss, termed *regret*, of OSA compared to the oracle solution that knows exactly the packet capture probabilities. A sublinear regret bound is proved for the proposed OSA algorithm, which implies that OSA is able to produce asymptotically optimal sniffer assignment decisions.

The rest of this paper is organized as follows: Section II reviews related works. Section III presents the system model for the SCA problem in SCRNs. Section IV gives an oracle solution to the formulated SCA problem. Section V designs an online sniffer-channel assignment algorithm based on CC-MAB. Section VI shows the simulation results, followed by the conclusion in Section VII.

II. Related Work

Applying cognitive radio to small cells has been studied in the context of heterogeneous networks, known as the cognitive small cell [17], [18], where the macrocells are considered as the primary network and small cells are regarded as the secondary cognitive network that serves unlicensed users. By contrast, SCRNs considered in our work are allocated with spectrum resource for serving both licensed and unlicensed users by following a medium access control mechanism [19]. Our paper designs a passive monitoring system for SCRNs.

Sniffer Channel Assignment (SCA) is a fundamental building block for passive monitoring systems. In early studies, the SCA problem is simply formulated as a resource allocation problem [10], [11] by assuming the complete statistical knowledge of users' traffic and is solved by optimization-based approaches. To relax the assumption of users' traffic information, learning-based solutions have been investigated [8], [12], [20] for SCA problems. For example, [12] proposed a non-parametric estimation method to infer the traffic pattern of secondary users in cognitive radio networks which is then used to optimize the sniffer channel assignment; authors in [8] characterize the user activities using a stochastic process with unknown parameters and solve the SCA problem using the stochastic bandit learning; authors in [20] used support vector regression to predict the packet arrival time of secondary users and employed a greedy scheduling scheme to navigate the candidate channels. The above works dealt with the uncertainty in user activities only and overlooked the imperfectness of monitoring. Imperfect monitoring of sniffers is gaining increasing attraction in recent works. [9] introduces the capture probability of packet to characterize the imperfectness of monitoring in the SCA problem. However, it still assumes that the statistical knowledge on the packet capture probability on channels is available. However, these capture probabilities are actually unknown to the monitoring system. In this paper, we design an online learning algorithm to learn the packet capture probabilities on channels and optimize sniffer channel assignment with the learned knowledge.

Multi-armed bandit (MAB) has been widely studied to address the critical tradeoff between exploration and exploitation in sequential decision making under uncertainty [21]. The MAB learning method has been applied in the SCA problems. For example, [9] uses the non-stochastic MAB to learn the activity of secondary users, especially for the misbehaving ones. Authors in [13] consider the switching cost in SCA and proposed a multi-agent multi-arm partial information problem with linear parameterized pay-off which guarantees a logarithmic regret. However, the MAB algorithms proposed in the above works cannot be intuitively extended to address our SCA problem in SCRNs since those algorithms require a fixed set of arms, i.e. the channels can be monitored by a sniffer should not change across time. This is less likely to be true in SCRNs with real-time spectrum resource management. Our algorithm is a contextual combinatorial multi-armed bandit learning algorithm, which uses the SINR of sniffers on channels as context and learns the packet capture probability for sniffer-channel pairs with similar context rather than for each sniffer-channel pair. In this way, we could learn with time-varying spectrum resource in SCRNs.

III. System Model

A. Small Cell Cognitive Radio Network

We consider a small cell network consisting of a set of small cell base stations (SBSs) indexed by $\mathcal{N} = \{1, 2, \ldots, N\}$. The operational time line is discretized into time slots $t = 1, 2 \ldots, T$. In each time slot t, the cellular network operator allocates the spectrum resource to each small cell using the state-of-the-art spectrum allocation scheme (e.g. [4]). The spectrum resource allocated to SBS $n \in \mathcal{N}$ is represented by a set of channels $\mathcal{H}_n^t$, which can be opportunistically accessed by unlicensed or *Secondary Users* (SUs) as long as they do not cause any interference to licensed or *primary users* (PU). Note that $\mathcal{H}_n^t$ is indexed by the time slot t, indicating that the available channels for each SBSs can change over time. Due to the spatial spectrum reuse, different SBSs may have channels operated on the same frequency band. Therefore, we represent a unique channel by two attributes: the operating SBS and the frequency band of the channel. For an arbitrary channel k, we let $\mathring{n}(k)$ denote the index of SBS that operates channel k and $\mathring{f}(k)$ denote its channel frequency band. Based on these two attributes, we assign an index for each unique channel and denote all channels in the whole network as $\mathcal{H}^t = \{1, 2, \ldots, K^t\}$, where K^t is the total number of channels in time slot t.

B. Passive Monitoring System

An independent passive monitoring system deploys a set of sniffers in the SCRN for sensing channels and capturing packets. Each sniffer is equipped with a single antenna, which allows it to sense/capture traffic over a single channel at one time. Similar to [20], the sniffers are categorized into two types: *inspection sniffers* and *monitoring sniffers*. All sniffers are connected to a central controller of the monitoring system for centralized decision making. Inspection sniffers are operated in the time scale of *medium access control frame* [19] (e.g., 100ms) and periodically sense channels to gain channel usage statistics of SUs and PUs, which can be used for analysis and prediction of traffic pattern in a time slot. The monitoring sniffers are operated in the time scale of spectrum resource allocation, i.e., the discretized time slots (e.g. few seconds), and capture packets transmitted on the channel. The slower time scale for operating the monitoring sniffers is due to the delay incurred by the online learning algorithm which requires information gathering and exchange between monitoring sniffers and the central controller. This delay may result in large switching costs if monitoring sniffers are operated frame-wise. This paper focuses on the assignment of monitoring sniffers with the help of information gathered by inspection sniffers. Therefore, a sniffer will be referred to as the monitoring sniffer unless noted otherwise. Let $\mathcal{S} = \{1, 2, \ldots, S\}$ be the set of (monitoring) sniffers. Each sniffer $s \in \mathcal{S}$ has a set of channels that can be assigned to, denoted by $\mathcal{C}_s^t \subseteq \mathcal{H}^t$, depending on their relative locations to the SBSs. The objective of the monitoring system is to capture as much traffic (weighted by the importance of PUs and SUs) as possible. Whether the traffic on a channel can be captured depends on the following three factors:

(a) Sniffer assignment: The premise of traffic monitoring on a channel is that there is at least one sniffer assigned to that channel. In each time slot t, let $\boldsymbol{a}^t = \{a_1^t, a_2^t, \ldots, a_S^t\}$ denote the assignment decisions of all sniffers, where $a_s^t \in \mathcal{C}_s^t \cup \{null\}$ is the assignment decision of sniffer s. The assignment can take $null$ value, which means that sniffer s will not monitor any channel. For ease of the notation, we introduce a channel index mapping $\mathring{k}(a_s^t)$ which maps the assigned channel of sniffer s to the unique channel index $k \in \mathcal{H}^t$. Let $\mathcal{A}^t$ be the set of all feasible sniffer assignment decisions in time slot t.

(b) User activities: During a time slot, the traffic on each channel comes from both PUs and SUs who share the available spectrum by following a medium access control (MAC) mechanism [19]. Fortunately, the traffic pattern prediction for PUs and SUs has been widely studied, see [22] and the reference therein. We assume the monitoring system uses state-of-the-art prediction algorithms to acquire the traffic composition based on the channel usage statics gathered by inspection sniffers and gives the fraction of PU traffic and SU traffic on channel, denoted by $m_k^{P,t}$ and $m_k^{S,t}$, respectively. The monitoring system may have different preferences for the packets of PUs and SUs. Let $\epsilon_k^{P,t}$ and $\epsilon_k^{S,t}$ be the importance weights for PU traffic and SU traffic on channel k, which are determined by the operating SBSs of the channel. Then, the importance weight of a channel can be calculated as $w_k^t = \epsilon_k^{P,t} m_k^{P,t} + \epsilon_k^{S,t} m_k^{S,t}$.

(c) Packet capture probability: As highlighted before, we consider imperfect monitoring because wireless traffic capture is unreliable. Specifically, we introduce the packet capture probability θ_k^t as the probability that a packet transmitted on channel k in time slot t can be successfully captured by the assigned sniffers. Note that θ_k^t depends on many factors, including the number of sniffers assigned to the channel as

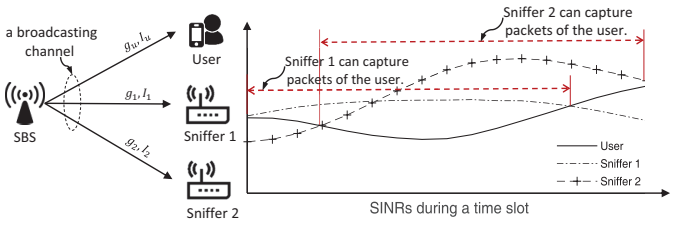


Fig. 2: Illustration of secrecy channel capacity.

well as the channel conditions of both users and assigned sniffers. In the following, we will explain in detail the impacts of these factors on the packet capture probability.

One key factor that affects packet capture probabilities on a channel is the signal-to-interference-plus-noise ratio (SINR) of sniffers and users on the channel. According to the theory of secrecy channel capacity, the information can be reliably decoded from packets received by a sniffer only if the SINR of the sniffer on the channel is larger than or equal to the SINR of the user [23], [24]. To be specific, if sniffer s is assigned to a channel $k \in \mathcal{C}_s^t$ and let $\text{SINR}_{u,k} = \frac{Pg_{u,k}}{I_{u,k}+N_0}$ and $\text{SINR}_{s,k} = \frac{Pg_{s,k}}{I_{s,k}+N_0}$ be the SINRs of a user and a sniffer on channel k, where P is the transmission power, $g_{u,k}$ and $g_{s,k}$ are the channel gains of the user and the sniffer on the channel, $I_{u,k}$ and $I_{s,k}$ are the interference power of the user and the sniffer, and N_0 is the noise power, then the sniffer can reliably decode the information from the received packets only if $\text{SINR}_{s,k} \geq \text{SINR}_{u,k}$ holds. We say that a packet is successfully captured if the user's packet is received by the sniffer with $\text{SINR}_{s,k} \geq \text{SINR}_{u,k}$. Because channel state varies over time even within a time slot (as illustrated in Fig. 2), the sniffer may sometimes be able to capture packets but sometimes may not. Therefore, we define the *non-outage probability* $p_{s,k}^t$ of sniffer s on channel $k \in \mathcal{C}_s^t$ in time slot t as:

$$p_{s,k}^t = \Pr\left\{\text{SINR}_{s,k}^t \geq \text{SINR}_{u,k}^t\right\}, \tag{1}$$

Suppose a single sniffer s is assigned to channel k, then the packet capture probability ϕ_k^t on channel k equals the non-outage probability $p_{s,k}^t$ of sniffer s on channel k. However, the monitoring system can apply redundant sniffer assignment and assign multiple sniffers to one channel to increase the capture probability of packets transmitted on the channel. This is because the traffic patterns on channels can be very different from each other and the monitoring system may want to increase the packet capture probability on important channels. Let $\mathcal{S}_k^t(\boldsymbol{a}^t) = \left\{s \in \mathcal{S} \mid k(a_s^t) = k, a_s^t \in \boldsymbol{a}^t, k \in \mathcal{H}^t\right\}$ be the set of sniffers assigned to channel k in time slot t. Assuming that SINRs of users and sniffers on channels evolve independently and let $\boldsymbol{p}^t = \{p_{s,k}^t\}_{\forall s,k \in \mathcal{C}_s^t}$ collect the non-outage probabilities of all possible sniffer-channel pairs, then the packet capture probability $\theta_k^t(\boldsymbol{a}^t;\boldsymbol{p}^t)$ on channel k can be calculated as:

$$\theta_k^t(\boldsymbol{a}^t;\boldsymbol{p}^t) = \begin{cases} 1 - \prod_{s \in \mathcal{S}_k^t(\boldsymbol{a}^t)}(1 - p_{s,k}^t), & \text{if } \mathcal{S}_k^t(\boldsymbol{a}^t) \neq \varnothing \\ 0, & \text{if } \mathcal{S}_k^t(\boldsymbol{a}^t) = \varnothing \end{cases} \tag{2}$$

C. System Utility and Problem Formulation

In each time slot t, the monitoring system aims to monitor a set of channels and maximize the monitoring reward by choosing a sniffer assignment decision $\boldsymbol{a}^t$. For ease of problem formulation, we assume that all the channels are fully used in a time slot and the data throughputs on channels are the same. Then, the utility of the monitoring system in each time slot t is defined as the sum of packet capture probabilities on all channels, weighted by the importance of the channel:

$$u^t(\boldsymbol{a}^t;\boldsymbol{p}^t) = \textstyle\sum_{k \in \mathcal{H}^t} w_k^t \theta_k^t(\boldsymbol{a}^t;\boldsymbol{p}^t) \tag{3}$$

However, note that the above problem can be easily extended to consider only active channels and different data throughputs since this information can be acquired or predicted by inspection sniffers [22]. Consider a total number of T time slots, the monitoring system aims to maximize the cumulative utility by choosing a sniffer assignment action in every time slot. Formally, our SCA problem is formulated as:

$$\textbf{P1} \quad \max_{\boldsymbol{a}^1,\boldsymbol{a}^2,\ldots,\boldsymbol{a}^T} \ \textstyle\sum_{t=1}^T u^t(\boldsymbol{a}^t;\boldsymbol{p}^t) \tag{4a}$$

$$\text{s.t. } a_s^t \in \mathcal{C}_s^t \cup \{null\}, \forall s, \forall t \tag{4b}$$

The above formulation is neat, but solving **P1** is not as easy as it may appear. The main reason is that the utility $u^t(\boldsymbol{a}^t;\boldsymbol{p}^t)$ depends on the non-outage probability $p_{s,k}^t$ of sniffers on channels, which changes over time based on SINRs of sniffers/users and is uncertain to the monitoring system when determining the sniffer assignment. Therefore, the SCA problem is not a conventional optimization problem but requires joint online learning and optimization. In the following sections, we will first give a solution to the SCA problem, assuming that the non-outage probability of sniffers on each channel is perfectly known at the beginning of each time slot. Later, we will develop an online learning algorithm for sniffer assignment based on the framework of CC-MAB.

IV. Sniffer Channel Assignment with Oracle Information

In this section, we give an Oracle solution to the SCA problem by assuming that the non-outage probabilities of sniffers on channels, i.e., $p_{s,k}^t, \forall s$ and $k \in \mathcal{C}_s^t$, are known to the monitoring system. In this ideal case, the utility function $u^t(\boldsymbol{a}^t;\boldsymbol{p}^t)$ is perfectly known and hence we solve the SCA problem via an optimization approach. The long-term optimization problem **P1** is fully decoupled across time slots and hence can be divided into T independent subproblems, one for each time slot t as follows:

$$\textbf{P2} \quad \max_{\boldsymbol{a}^t} \ u^t(\boldsymbol{a}^t;\boldsymbol{p}^t) = \textstyle\sum_{k \in \mathcal{H}^t} w_k^t \theta_k^t(\boldsymbol{a}^t;\boldsymbol{p}^t) \tag{5a}$$

$$\text{s.t. } a_s^t \in \mathcal{C}_s^t \cup \{null\} \tag{5b}$$

The problem **P2** is actually a combinatorial optimization problem where the monitoring system picks a set of channels for monitoring and optimizes packet capture probability on these channels by choosing an appropriate sniffer assignment decision. While exhaustive search can always find the optimal solution, the complexity can be high when there are many sniffers and channels in the monitoring system. To address this problem, we design an efficient approximation algorithm

to solve **P2** in polynomial runtime. The performance guarantee of the approximation algorithm will also be presented.

A. Approximation of Oracle Solution

To solve the per-slot problem **P2**, we first show that **P2** is a Matroid-Constrained Submodular Maximization (MCSM) problem. Below gives the definition of the MCSM problem:

Definition 1 (MCSM)**.** *Given a monotone submodular function $f : 2^X \to \mathbf{R}_+$ and a matroid $\mathcal{M} = (X, \mathcal{I})$, an MCSM problem is* $\max_{S \in \mathcal{I}} f(S)$.

In the above definition, a matroid $\mathcal{M} = (X, \mathcal{I})$ is a system of *independent sets* where X is a finite set (called the ground set) and $\mathcal{I}$ is the set of independent subsets of X with the following properties: 1) $\varnothing \in \mathcal{I}$ and at least one subset of X is independent; 2) For each $A' \subset A \subset X$, if $A \in \mathcal{I}$, then $A' \in \mathcal{I}$. 3) If $A, B \in \mathcal{I}$, and $|A| > |B|$, then $\exists x \in A \backslash B$ such that $B \cup \{x\} \in \mathcal{I}$.

Theorem 1. *The subproblem **P2** of sniffer channel assignment is a matroid-constrained submodular maximization problem.*

Proof. For the subproblem **P2**, the ground set X of matroid $\mathcal{M} = (X, \mathcal{I})$ is the set of channels $X = \cup_{s \in \mathcal{S}} \mathcal{C}_s^t$ and $\mathcal{I} = \{I_1, I_2, \dots\}$ consists of subsets of X (i.e., $I_1 \subseteq X, I_2 \subseteq X, \dots$) where all $I \in \mathcal{I}$ contains at most one channel from C_s^t for each $s \in \mathcal{S}$. By the definition of I, it can be written as $I = \cup_{s \in \mathcal{S}} a_s^t$, s.t. $a_s^t \in \mathcal{C}_s^t, \forall s$. In other words, $\mathcal{I}$ is the set of all feasible sniffer assignment decisions. It is easy to verify that $\mathcal{I}$ satisfies all three above properties. It remains to show that $u(\boldsymbol{a}^t; \boldsymbol{p}^t)$ is a submodular function. The utility function $u(\boldsymbol{a}^t; \boldsymbol{p}^t)$ is a weighted sum of packet capture probabilities on channels. Clearly, that packet capture probability function in (2) is submodular and therefore the utility $u(\boldsymbol{a}^t; \boldsymbol{p}^t)$ is a weighted sum of submodular function, which is also a submodular function. This concludes the proof. □

A simple greedy algorithm (in Algorithm 1) is quite natural for solving the MCSM problem in **P2**. The greedy algorithm determines the assignment decision of sniffer sequentially starting with the all-$null$ decisions. In each iteration, it assigns a sniffer to a channel that gives the largest incremental utility. Since each sniffer can monitor at most one channel and each iteration decides the assignment decision for one sniffer, the algorithm is guaranteed to terminate in at most S iterations.

Based on the classic results for MCSM problems in [25], the greedy algorithm guarantees to yield a $1/2$-approximation:

Lemma 1. *In an arbitrary time slot t, let $\boldsymbol{a}^{*,t}$ be the sniffer assignment derived by the greedy algorithm and $\boldsymbol{a}^{opt,t}$ be the optimal sniffer assignment decision for **P2**, we will have* $u^t(\boldsymbol{a}^{*,t}; \boldsymbol{p}^t) \geq \frac{1}{2} u(\boldsymbol{a}^{opt,t}; \boldsymbol{p}^t)$.

Proof. The proof follows [25] and hence is omitted. □

We use the greedy algorithm to approximate the optimal sniffer assignment decision with oracle information on the sniffers' non-outage probabilities. Note that the actual performance of the greedy algorithm is usually much better than the 1/2 approximation ratio. However, the 1/2-approximation guarantee of the greedy algorithm is a critical basis in designing our online learning algorithm in the next section.

Algorithm 1 Greedy Algorithm

Input: Sniffer set $\mathcal{S}$; non-outage probabilities $p_{s,k}^t, \forall s, k \in \mathcal{C}_s^t$; weights of channels w_k^t.
Initialization: $a_s^t = null, \forall s$; $\tilde{\mathcal{S}} \leftarrow \mathcal{S}$; $\tilde{u} \leftarrow 0$.
while $\tilde{\mathcal{S}} \neq \varnothing$ **do**
  for all $s \in \tilde{\mathcal{S}}$ **do**
    for all $k \in \mathcal{C}_s^t$ **do**
      Set $\tilde{\boldsymbol{a}}^t \leftarrow \boldsymbol{a}^t$ and $\tilde{a}_s^t \leftarrow k$;
      Calculated the utility $\tilde{u} \leftarrow u(\tilde{\boldsymbol{a}}^t; \boldsymbol{p}^t)$;
      if $\tilde{u} \geq u_{\max}$ **then**
        $s_{\max} \leftarrow s$, $u_{\max} \leftarrow \tilde{u}$, $k_{\max} \leftarrow k$;
  Set $a_{s_{\max}}^t \leftarrow k_{\max}$, $\tilde{\mathcal{S}} = \tilde{\mathcal{S}} \setminus \{s_{\max}\}$;
return $a_s^t, \forall s$

V. Online Sniffer Channel Assignment under Uncertainty

In the previous section, we discussed the oracle solution for the SCA problem by assuming that the non-outage probability of each sniffer-channel pair is perfectly known. However, in practice, the monitoring system does not know these non-outage probabilities in advance. Recall that the non-outage probability of a sniffer-channel pair depends on the SINR of sniffer and users on the channel. While it is difficult, if not impossible, for sniffers to know SINRs of users, the sniffers can easily acquire their own SINR on the channels [26]. Therefore, we take the SINR on the sniffer side as *context* and use this information to help infer the non-outage probability. Let $\phi_{s,k}^t$ be the context (i.e. SINR) of sniffer s on channel $k \in \mathcal{C}_s^t$. Without loss of generality, we normalize $\phi_{s,k}^t$ in a bounded space $\Phi = [0, 1]$ using min-max feature scaling. The context of all sniffers on channels are collected in $\boldsymbol{\phi}^t = \{\phi_{s,k}^t\}_{\forall s, k \in \mathcal{C}_s^t}$. The non-outage probability of sniffer s on channel k is a random variable parameterized by the context $\phi_{s,k}^t$. We slightly abuse the notation of non-outage probability and define the context-aware non-outage probability $p_{s,n}(\phi_{s,k}^t)$, where $p_{s,n} : \Phi \to [0, 1]$ is a mapping that maps the context $\phi_{s,k}^t$ of sniffer s on channel $k \in \mathcal{C}_s^t$ (operated by SBS $n = \mathring{n}(k)$) to a non-outage probability. Note that there is a mapping function $p_{s,n}$ for each sniffer-SBS pair (s, n) because factors determining the SINR of sniffers and users on a channel are usually location-dependent, e.g., the distance between the sniffer and the operating SBS, the surrounding environment, and the user population density (which affects the interference of users). In addition, the locations of SBSs and sniffers are usually fixed and therefore the learned statistical knowledge is useful. We further define $\mu_{s,n}(\phi_{s,k}^t) \triangleq \mathbb{E}\left[p_{s,n}(\phi_{s,k}^t)\right]$ as the expected value of $p_{s,n}(\phi_{s,k}^t)$.

Based on the context-aware non-outage probability, we design our Online Sniffer-channel Assignment (OSA) algorithm, which learns the underlying connection between non-outage

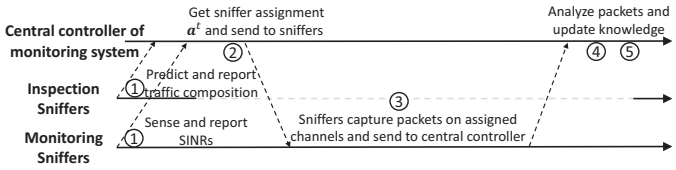


Fig. 3: Time diagram for the SCA problem with OSA.

probabilities and sniffers' SINRs, and optimizes the sniffer assignment using the learned knowledge. Before presenting our algorithm, we first introduce the performance measurement of an online learning policy. The performance of an online learning policy is measured by the utility loss compared with the Oracle solution, termed as *regret*. Suppose a policy gives the decision sequence $\{\boldsymbol{a}^1, \boldsymbol{a}^2, \ldots, \boldsymbol{a}^T\}$, then its regret is:

$$R(T) = \sum_{t=1}^{T} u(\boldsymbol{a}^{\text{opt},t}; \boldsymbol{p}^t) - \sum_{t=1}^{T} u(\boldsymbol{a}^t; \boldsymbol{p}^t) \tag{6}$$

where $\boldsymbol{a}^{\text{opt},t}$ is the optimal oracle sniffer assignment decision for **P2** in time slot t. This definition of regret is used when the optimal oracle solutions are derivable. However, as discussed in the previous section, we use the greedy algorithm to efficiently approximate the optimal oracle solution for **P2** instead of finding the optimal oracle solution. Therefore, we employ the definition of δ-regret which is often used in bandit learning with approximation algorithms [27]. Consider a δ-approximation algorithm (i.e., the solution $\boldsymbol{a}^t$ derived by the approximation algorithm satisfies $u^t(\boldsymbol{a}^t; \boldsymbol{p}^t) \geq \frac{1}{\delta} u(\boldsymbol{a}^{\text{opt},t}; \boldsymbol{p}^t)$) for **P2**, the δ-regret is defined by:

$$R^{\delta}(T) = \sum_{t=1}^{T} \frac{1}{\delta} u(\boldsymbol{a}^{\text{opt},t}; \boldsymbol{p}^t) - \sum_{t=1}^{T} u(\boldsymbol{a}^t; \boldsymbol{p}^t) \tag{7}$$

Since the greedy algorithm for **P2** has an approximation ratio of 1/2, δ equals 1/2 for OSA. The definition of δ-regret essentially compares the utility of a policy with the lower bound of the approximated oracle solution.

A. CC-MAB for Online Sniffer-Channel Assignment

Now, we present our OSA algorithm for online sniffer-channel assignment. The primary goal of OSA is to guarantee a sublinear regret bound $R^{\delta}(T) = O(T^{\gamma})$ with $\gamma < 1$ so that the policy is asymptotically optimal as $\lim_{T\to\infty} R^{\delta}(T)/T = 0$. We will later prove that the proposed OSA algorithm has a sublinear regret bound with appropriate algorithm parameters.

Fig. 3 illustrates the time diagram of OSA in each time slot. The monitoring system operates sequentially as follows: 1) sniffers sense SINRs $\boldsymbol{\phi}^t$ on channels and report them to the central controller of the monitoring system, meanwhile, the central controller determines the channel weights based on the predicted traffic composition reported by inspection sniffers. 2) The monitoring system assigns each sniffer to a channel according to the observed context $\boldsymbol{\phi}^t$ and the designed OSA algorithm. 3) Each sniffer captures packets on the assigned channel and sends them to the central controller. 4) The central controller analyzes the captured packets and observes non-outage fractions (defined as a ratio of the number of captured packets on a channel to the number of total packets transmitted on that channel). 5) The observed non-outage fractions will be used to update the counters and experiences.

Algorithm 2 Online Sniffer Allocation (OSA)

Input: T, β_T, $Q(t)$.
Initialization: $\mathcal{L}_T, C_{s,n}(l) = 0, \mathcal{E}_{s,n}(l) = \varnothing, \hat{\mu}_{s,n}(l) = 0, \forall s \in \mathcal{S}, n \in , l \in \mathcal{L}_T$;
for $t = 1, \ldots, T$ **do**:
  Sniffers observe contexts on channels $\boldsymbol{\phi}^t$;
  Identify $\mathcal{C}_s^{\text{ue},t}(\boldsymbol{\phi}^t)$ and $\mathcal{S}_s^{\text{ue},t}(\boldsymbol{\phi}^t)$, and estimate the non-outage probabilities $\hat{\boldsymbol{p}}^t$ based on $\boldsymbol{\phi}^t$;
  if $\mathcal{S}_s^{\text{ue},t}(\boldsymbol{\phi}^t) \neq \varnothing$ **then** ▷ *Exploration*
    for $s \in \mathcal{S}_s^{\text{ue},t}(\boldsymbol{\phi}^t)$ **do**:
      $a_s^t \leftarrow$ randomly assign sniffer s to a channel in $\mathcal{C}_s^{\text{ue},t}(\boldsymbol{\phi}^t)$;
    if $\{\mathcal{S} \setminus \mathcal{S}_s^{\text{ue},t}(\boldsymbol{\phi}^t)\} \neq \varnothing$ **then**
      Determine $a_s^t, \forall s \in \{\mathcal{S} \setminus \mathcal{S}^{\text{ue},t}(\boldsymbol{\phi}^t)\}$ by solving the problem in (10);
  else ▷ *Exploitation*
    Determine the sniffer assignment $\boldsymbol{a}^t$ by solving the problem in (11);
  for $s \in \mathcal{S}$ **do**: ▷ *Update*
    Identify operating SBS $n = \mathring{n}(a_s^t)$ and the context interval l that $\phi_{s,a_s^t}^t$ belongs to;
    Observe non-outage fraction p of sniffer s on the assigned channel a_s^t;
    Update estimation: $\hat{\mu}_{s,n}(l) \leftarrow \frac{\hat{\mu}_{s,n}(l) C_{s,n}(l) + p}{C_{s,n}(l)+1}$;
    Update counters: $C_{s,n}(l) \leftarrow C_{s,n}(l) + 1$;

The pseudocode of OSA is presented in Algorithm 2. It starts by creating partition $\mathcal{L}_T$ of the context space Φ, which quantizes $\Phi = [0, 1]$ into intervals of similar contexts. Specifically, OSA determines the quantization step length $\frac{1}{\beta_T}$ based on the given time horizon T. The partition $\mathcal{L}_T$ splits Φ into β_T intervals with identical size $\frac{1}{\beta_T}$. Here, β_T is a parameter to be designed in OSA. Each sniffer $s \in \mathcal{S}$ keeps a counter $C_{s,n}^t(l)$ for each SBS $n \in \mathcal{N}$ and each interval $l \in \mathcal{L}_T$. For the tuple (s, n, l) of a counter $C_{s,n}^t(l)$, we define an assignment event $V_{s,n,l}$ which represents an assignment decision satisfying the three following conditions: 1) the sniffer s is assigned to a channel $k \in \mathcal{C}_s^t$; 2) the channel k is operated by SBS n, i.e. $\mathring{n}(k) = n$; 3) the context of sniffer s on channel k belongs to l, i.e. $\phi_{s,k}^t \in l$. The counter $C_{s,n}^t(l)$ records the number of times that the event $V_{s,n,l}$ happens up to time slot t. Each sniffer s also keeps an experience $\mathcal{E}_{s,n}^t(l)$ for each SBS n and each interval l, which stores the observed non-outage fraction when assignment event $V_{s,n,l}$ happens. Fig. 4 illustrates a context partition and counter/experience update. Based on the observed non-outage fractions in $\mathcal{E}_{s,n}^t(l)$, the estimated non-outage probability for an assignment event $V_{s,n,l}$ is:

$$\hat{\mu}_{s,n}^t(l) = \frac{1}{C_{s,n}^t(l)} \sum\nolimits_{p \in \mathcal{E}_{s,n}^t(l)} p. \tag{8}$$

Consider a time slot t and a channel $k \in \mathcal{C}_s^t$ for sniffer s

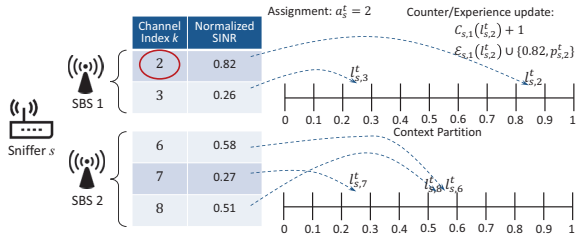


Fig. 4: Context partition and counter/experience update. Sniffer s have 5 channels in $\mathcal{C}_s^t$ with $\{2,3\}$ operated by SBS 1 and $\{6,7,8\}$ operated by SBS 2. The illustration assumes sniffer s is assigned to channel 2 and the observed non-outage fraction is $p_{s,2}^t$.

with operating SBS $n = \mathring{n}(k)$, let $l_{s,k}^t$ be the interval that the context $\phi_{s,k}^t$ belongs to, then the estimated non-outage probability of sniffer s on channel k is $\hat{p}_{s,k}^t = \hat{\mu}_{s,n}^t(l_{s,k}^t)$. Let $\hat{\boldsymbol{p}}^t = \{\hat{p}_{s,k}^t\}_{\forall s,k\in\mathcal{C}_s^t}$ collect all the estimated non-outage probabilities. To decide the sniffer assignment, OSA first checks whether these intervals have been explored sufficiently often such that the estimated non-outage probability is accurate enough to use. Therefore, we define the *under-explored* intervals $\mathcal{L}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t)$ for sniffer s in time slot t as:

$$\mathcal{L}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t) \triangleq \left\{ l \in \mathcal{L}_T \,\middle|\, \begin{array}{l} \exists \phi_{s,k}^t \in \boldsymbol{\phi}^t, \phi_{s,k}^t \in l, n = \mathring{n}(k) \\ \text{and } C_{s,n}^t(l) \le Q(t) \end{array} \right\} \tag{9}$$

where $Q(t)$ is a deterministic, monotonically increasing control function, which is an input of OSA to be designed for determining whether the amount of collected historical data in an experience $\mathcal{E}_{s,n}^t(l)$ is enough to produce an accurate non-outage probability estimation. We also collect the under-explored channels for each sniffer s, defined as $\mathcal{C}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t) \triangleq \{k \in \mathcal{C}_s^t \mid l_{s,k}^t \in \mathcal{L}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t)\}$. OSA is either in an exploration or exploitation phase based on the under-explored channels.

1) Exploration: If there exists a sniffer s with non-empty $\mathcal{C}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t)$, then OSA enters exploration. In exploration, the sniffers have two possible types: sniffers that have under-explored channels, denoted by $\mathcal{S}^{\mathrm{ue},t}(\boldsymbol{\phi}^t) \triangleq \{s \in \mathcal{S} \mid \mathcal{C}_s^{\mathrm{ue},t}(\boldsymbol{\phi}^t) \neq \varnothing\}$, and sniffers that have no under-explored channels, denoted by $\mathcal{S}^{\mathrm{ed},t}(\boldsymbol{\phi}^t) \triangleq \mathcal{S} \setminus \mathcal{S}^{\mathrm{ue},t}(\boldsymbol{\phi}^t)$. For a sniffer $s \in \mathcal{S}^{\mathrm{ue},t}$, it is randomly assigned to one of its under-explored channels. For sniffers in $\mathcal{S}^{\mathrm{ed},t}$, their assignment decisions are jointly optimized based on **P2** with other sniffers taking *null* decisions.

$$\max_{\boldsymbol{a}^t} \quad \sum_{k\in\mathcal{H}^t} w_k^t \theta_k^t(\boldsymbol{a}^t; \hat{\boldsymbol{p}}^t)) \tag{10a}$$
$$\text{s.t.} \quad a_s^t \in \mathcal{C}_s^t \cup \{null\}, \forall s \in \mathcal{S}^{\mathrm{ed},t}(\boldsymbol{\phi}^t) \tag{10b}$$
$$a_s^t = null, \forall s \in \mathcal{S}^{\mathrm{ue},t}(\boldsymbol{\phi}^t) \tag{10c}$$

Note that in problem (10), the sniffer assignment decision is made based on the estimated non-outage probabilities $\hat{\boldsymbol{p}}^t$.

2) Exploitation: If all sniffers have empty $\mathcal{C}_s^{\mathrm{ue},t}$, then OSA enters exploitation. The sniffer assignment decision is derived by solving **P2** with estimated non-outage probabilities $\hat{\boldsymbol{p}}^t$.

$$\max_{\boldsymbol{a}^t} \quad \sum_{k\in\mathcal{H}^t} w_k^t \theta_k^t(\boldsymbol{a}^t; \hat{\boldsymbol{p}}^t) \tag{11a}$$
$$\text{s.t.} \quad a_s^t \in \mathcal{C}_s^t \cup \{null\}, \forall s \in \mathcal{S} \tag{11b}$$

At the end of each time slot, the monitoring system observes the non-outage fractions of sniffers on the assigned channels and updates the corresponding counter and experience. Note that the time index of counters and experiences is dropped in the pseudocode due to the recursive update.

B. Parameter Design and Regret Analysis

Next, we design the algorithm parameters β_T and $Q(t)$ and give a regret upper bound for OSA. The regret analysis is carried out based on the assumption that the expected non-outage probability of sniffer-channel pairs are similar if they have similar context. This assumption is formalized by the Hölder condition as follows:

Assumption 1 (Hölder Condition)**.** *For a sniffer s and an SBS n, there exists $L > 0$, $\alpha > 0$ such that for any contexts $\phi, \phi' \in \Phi$ observed by sniffer s on channels operated by SBS n, it holds that $|\mu_{s,n}(\phi) - \mu_{s,n}(\phi')| \le L|\phi - \phi'|^\alpha$.*

Assumption 1 is needed for the regret analysis, but it should be noted that OSA can also be applied if this assumption does not hold. However, a regret bound might not be guaranteed in this case. Given the parameters β_T and $Q(t)$ in Theorem 2, we have a sublinear regret upper bound of OSA:

Theorem 2 (Regret Upper Bound)**.** *Let $Q(t) = t^{\frac{2\alpha}{3\alpha+1}}\log(t)$ and $\beta_T = \lceil T^{\frac{1}{3\alpha+1}} \rceil$. If Hölder condition holds true, the upper bound of expected δ-regret $\mathbb{E}[R^\delta(T)]$ is:*

$$\mathbb{E}[R^\delta(T)] = O\left(NS^2 w^{\max} T^{\frac{2\alpha+1}{3\alpha+1}} \log(T)\right) \tag{12}$$

where $w^{\max}$ is the maximum weight of a channel.

Proof. See in Appendix A. □

The regret upper bound given in Theorem 2 is sublinear. In addition, the regret bound is valid for any finite time T and hence can be used to characterize the convergence speed.

VI. Simulation

A. Simulation Setup

We simulate a small cell network covering a 1200×1200m area. The network consists of 8 SBSs and the communication radius of each SBS is 200m. The transmission power of

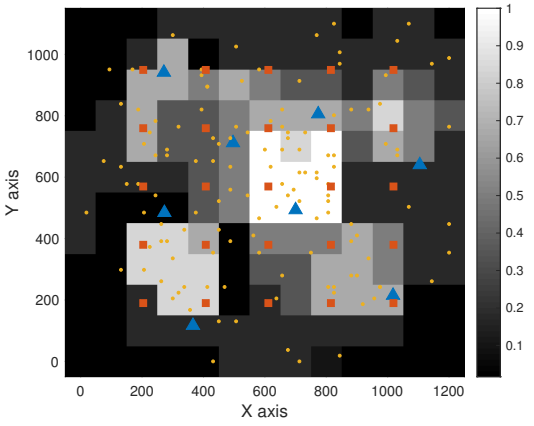


Fig. 5: Deployment of SBSs, sniffers, and users. The background color is the user population density; the blue triangles are SBSs, red squares are sniffers, yellow circles are users in a time slot.

SBSs is P = 20dBm. The channel gain is calculated by the free-space path loss with Gaussian random shadowing. 25 sniffers are deployed in a grid layout. Each sniffer is able to monitor the channels operated by SBSs that are within the 200m communication range. The number of user devices in each time slot is a random variable that follows a Poisson distribution with arrival rate 120 per time slot and these users are randomly deployed according to the population density in the service area. A deployed user is a primary user with probability 0.3. Fig. 5 shows the deployment of the SBSs, sniffers, user population densities, and user devices in a certain time slot. The parameter α for Hölder condition is set to 1 and the time horizon is $T = 10^4$, which give $\beta_T = 10$ in OSA.

B. Performance Comparison With MAB Benchmarks

We compare the cumulative reward and regret of OSA with the following benchmarks:

1) Oracle: Oracle knows precisely the non-outage probabilities of sniffers on all channels. In each time slot, Oracle uses the greedy algorithm (Algorithm 1) to approximate the optimal solution for problem **P2**.

2) UCB: UCB [28] is a classical MAB algorithm (non-contextual and non-combinatorial) that achieves the logarithmic regret bound when only one arm is played in each round. Since UCB cannot work with the time-varying arm set, we fix the spectrum resource for SBSs. In addition, each sniffer runs UCB independently without combinatorial optimization.

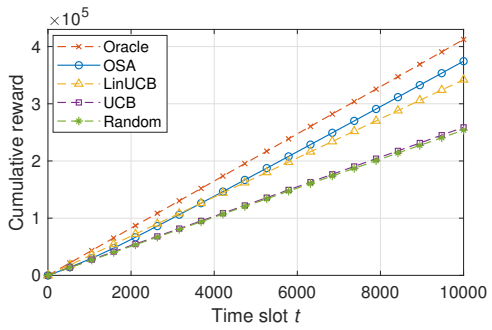
3) LinUCB: LinUCB [29] is a contextual-variant of UCB. It assumes that the reward of an arm is a linear function of observed contexts. In our simulation, the context information for LinUCB is the SINRs of sniffers and weights of channels.

4) Random: The Random algorithm assigns a sniffer randomly to one of its accessible channels in each time slot.

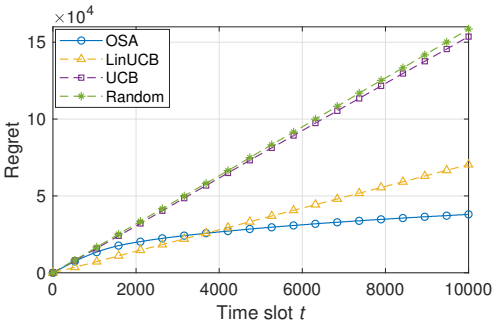
Fig. 6 compares the cumulative utility and regret achieved by OSA and other 4 benchmarks over a total of T time slots. From Fig. 6(a), we see that Oracle, as expected, achieves the highest cumulative utility and gives an upper bound to other learning policies. Among the others, OSA outperforms the other benchmarks and achieves a close-to-Oracle cumulative utility. The benefit of including sniffers' SINR as the context in OSA can be appreciated by comparing the performances of context-aware algorithms (OSA and LinUCB) and context-unaware algorithm (UCB and Random). It can be observed that context-aware algorithms achieve much higher cumulative reward than the context-unaware algorithms. In addition, we see that the cumulative utility of UCB is almost the same as Random. The malfunction of UCB is due to the time-varying spectrum resource and SINR-dependent utilities in passive monitoring. Fig. 6(b) depicts the regret incurred by OSA, LinUCB, UCB, and Random. We see that OSA is the only policy that achieves the sublinear regret while the regrets of other policies grow linearly over time.

C. Sub-optimal OSA variants

To show the importance of considering imperfect monitoring and allowing redundant sniffer assignment, we further



(a) Cumulative utility.



(b) Regret.

Fig. 6: Performance comparison with other MAB policies.

compare with the following two sub-optimal OSA variants:

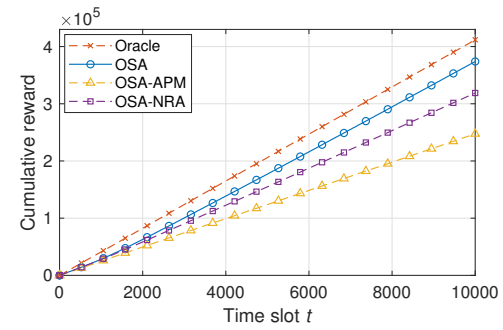
1) OSA with Assumed Perfect Monitoring (OSA-APM): OSA-APM assumes the perfect monitoring when assigning the sniffers [12]. In this case, the monitoring system does not learn non-outage probabilities and assumes perfect monitoring when deciding the sniffer assignment. However, when the sniffers monitor the channels, the packet capture probability is determined by the SINRs of sniffers and users.

2) OSA with Non-Redundant Assignment (OSA-NRA): OSA-NRA considers a non-redundant sniffer assignment [30] where the monitoring system assigns at most one sniffer to a channel. In this case, the utility maximization problem in each time slot becomes a matching problem.

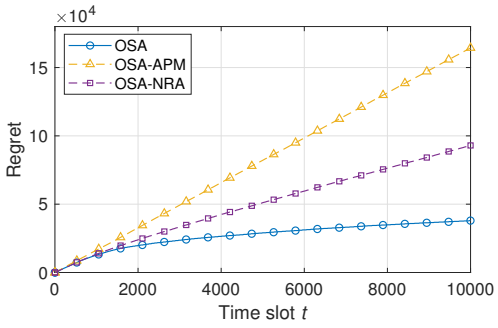
Fig. 7 shows the cumulative utility and regret of OSA when applied in the above two settings. In general, we see that considering the imperfect monitoring and redundant sniffer assignment helps improve the monitoring reward. The role of imperfect monitoring is more critical (providing 35.7% utility increase) compared to the redundant sniffer assignment (providing 18.4% utility increase). Actually, the benefit of redundant sniffer assignment depends on the number of sniffers in the monitoring system, which will be analyzed next.

D. Impact of the number of sniffers

Fig. 8 shows the cumulative utilities achieved by Oracle, OSA, and OSA-NRA after 10^4 time slots. Clearly, for all three policies, the monitoring system can achieve a higher cumulative utility with more sniffers. This is simply because more channels can be monitored or more sniffers can be assigned to monitor one channel if there are more sniffers in the monitoring system. In particular, the benefit of redundant



(a) Cumulative utility.



(b) Regret.

Fig. 7: Running OSA in different application settings.

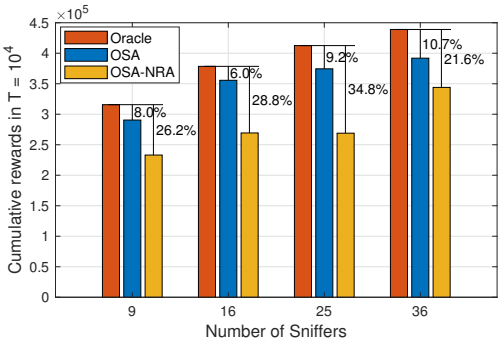


Fig. 8: Impact of the number of deployed sniffers.

sniffer assignment (i.e., the utility gap between OSA and OSA-NRA) depends on the number of sniffers. As can be observed, when the number of sniffers increases from 9 to 25, the benefit of redundant sniffer assignment gradually increase since the sniffer assignment becomes more flexible with more sniffers. However, if the sniffers are too many (e.g. $S = 36$), the benefit of redundant assignment decreases since channels are already monitored by multiple sniffers and adding more sniffers or making better sniffer assignment decisions will not provide much improvement due to the submodular utility function.

VII. Conclusion

This paper studied the optimal sniffer-channel assignment for a passive monitoring system for SCRNs with imperfect monitoring. A contextual combinatorial bandit learning algorithm was developed to learn the relationship between sniffer channel SINRs and the packet capture probabilities, and uses the learned knowledge to optimize the sniffer channel assignment on-the-fly. Our approach departs from traditional optimization-based approaches, which is able to work in systems with uncertain information. The developed algorithm addresses many practical challenges for SCRN packet monitoring, is easy to implement and achieves provably asymptotically optimal performance.

Appendix A
Proof of Theorem 2

Proof. The proof for Theorem 2 is similar to that in our previous work [31]. Due to space limitation, we only provide a sketch of the proof. The expected δ-regret can be divided into two parts: $\mathbb{E}[R^{\delta}(T)] = \mathbb{E}[R^{\delta}_{\text{explore}}(T) + R^{\delta}_{\text{exploit}}(T)]$, where $\mathbb{E}[R^{\delta}_{\text{explore}}(T)]$ and $\mathbb{E}[R^{\delta}_{\text{exploit}}(T)]$ are the regrets incurred by exploration and exploitation, respectively. The regret of these two parts will be bounded separately. The key idea for bounding the exploration regret is to ensure that the number of time slots that OSA enters exploration phase is sublinear. Therefore, given that maximum utility loss in each exploration phase, the total regret incurred by exploration is sublinear. For the exploitation regret, we need to prove the gap between the utilities incurred by the assignment decision of OSA and Oracle is sublinear in each time slot.

In fact, the leading order of $\mathbb{E}[R^{\delta}(T)]$ is determined by $\mathbb{E}[R^{\delta}_{\text{exploit}}(T)]$ and therefore we only give detailed proof for bounding $\mathbb{E}[R^{\delta}_{\text{explore}}(T)]$. Suppose time slot t is an exploration phase, then according to the algorithm design, there exists a sniffer s that has non-empty $\mathcal{L}_s^{\text{ue},t}$. Clearly, there can be at most $\lceil Q(t) \rceil$ exploration phases for exploring an interval l for a sniffer-SBS pair (s,n) up to time slot t. In each of these exploration phases, let $U^{\max,t} \triangleq \max_{\boldsymbol{a},\boldsymbol{a}' \in \mathcal{A}^t} |u^t(\boldsymbol{a};\boldsymbol{p}^t) - u^t(\boldsymbol{a}';\boldsymbol{p}^t)|$ be the maximum utility loss for picking a wrong sniffer assignment decision. Let $w^{\max}$ be the maximum weight for a channel. Since S sniffers can only monitor at most S channels, $U^{\max,t}$ is bounded by $w^{\max}S$. There are at most $\lceil Q(T) \rceil$ exploration phases for a interval l of a sniffer-SBS pair (s,n) up to T time slots, β_T intervals in the partition $\mathcal{L}_T$, and SN sniffer-SBS pairs, then $\mathbb{E}[R^{\delta}_{\text{explore}}(T)]$ is bounded by:

$$\mathbb{E}[R^{\delta}_{\text{explore}}(T)] \leq \frac{1}{2} SN\beta_T w^{\max} S \lceil Q(t) \rceil. \tag{13}$$

The upper bound for $\mathbb{E}[R^{\delta}_{\text{exploit}}(T)]$ is derived based on the Hölder condition and Chernoff-Hoeffding bound [32]. Due to the space limitation, this part of proof is omitted here.

By substituting the control function $Q(T) = T^{\frac{2\alpha}{3\alpha+1}} \log(T)$ and the parameter $\beta_T = \lceil T^{\frac{1}{3\alpha+D}} \rceil$ into (13), we have:

$$\begin{aligned}
\mathbb{E}[R^{\delta}(T)] &\leq \frac{1}{2} SN \lceil T^{\frac{1}{3\alpha+1}} \rceil w^{\max} S \lceil T^{\frac{2\alpha}{3\alpha+1}} \log(T) \rceil \\
&\leq \frac{1}{2} SN 2T^{\frac{1}{3\alpha+1}} w^{\max} S (T^{\frac{2\alpha}{3\alpha+1}} \log(T) + 1) \\
&= NS^2 T^{\frac{1}{3\alpha+1}} w^{\max} (T^{\frac{2\alpha}{3\alpha+1}} \log(T) + 1)
\end{aligned} \tag{14}$$

The second inequality in (14) is because $\lceil T^{\gamma} \rceil \leq 2T^{\gamma}$. The leading order of (14) is the same to that in Theorem 2. □

REFERENCES

[1] P. Demestichas, A. Georgakopoulos, D. Karvounas, K. Tsagkaris, V. Stavroulaki, J. Lu, C. Xiong, and J. Yao, "5g on the horizon: Key challenges for the radio-access network," *IEEE vehicular technology magazine*, vol. 8, no. 3, pp. 47–53, 2013.

[2] D. Wang, B. Song, D. Chen, and X. Du, "Intelligent cognitive radio in 5g: Ai-based hierarchical cognitive cellular networks," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 54–61, 2019.

[3] Q. C. Li, H. Niu, A. T. Papathanassiou, and G. Wu, "5g network capacity: Key elements and technologies," *IEEE Vehicular Technology Magazine*, vol. 9, no. 1, pp. 71–78, 2014.

[4] Y. Liu, L. Lu, G. Y. Li, Q. Cui, and W. Han, "Joint user association and spectrum allocation for small cell networks with wireless backhauls," *IEEE Wireless Communications Letters*, vol. 5, no. 5, pp. 496–499, 2016.

[5] F. R. Yu, H. Tang, M. Huang, Z. Li, and P. C. Mason, "Defense against spectrum sensing data falsification attacks in mobile ad hoc networks with cognitive radios," in *MILCOM 2009-2009 IEEE Military Communications Conference*. IEEE, 2009, pp. 1–7.

[6] S. Chen, K. Zeng, and P. Mohapatra, "Hearing is believing: Detecting wireless microphone emulation attacks in white space," *IEEE transactions on mobile computing*, vol. 12, no. 3, pp. 401–411, 2011.

[7] E. S. Pilli, R. C. Joshi, and R. Niyogi, "Network forensic frameworks: Survey and research challenges," *digital investigation*, vol. 7, no. 1-2, pp. 14–27, 2010.

[8] P. Arora, C. Szepesvári, and R. Zheng, *Sequential learning for optimal monitoring of multi-channel wireless networks*. IEEE, 2011.

[9] J. Xu, Q. Wang, K. Zeng, M. Liu, and W. Liu, "Sniffer channel assignment with imperfect monitoring for cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 3, pp. 1703–1715, 2015.

[10] D.-H. Shin and S. Bagchi, "Optimal monitoring in multi-channel multi-radio wireless mesh networks," in *Proceedings of the tenth ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2009, pp. 229–238.

[11] A. Chhetri, H. Nguyen, G. Scalosub, and R. Zheng, "On quality of monitoring for multi-channel wireless infrastructure networks," in *Proceedings of the eleventh ACM international symposium on Mobile ad hoc networking and computing*. ACM, 2010, pp. 111–120.

[12] Q. Yan, M. Li, F. Chen, T. Jiang, W. Lou, Y. T. Hou, and C.-T. Lu, "Non-parametric passive traffic monitoring in cognitive radio networks," in *2013 Proceedings IEEE INFOCOM*. IEEE, 2013, pp. 1240–1248.

[13] T. Le, C. Szepesvari, and R. Zheng, "Sequential learning for multi-channel wireless network monitoring with channel switching costs," *IEEE Transactions on Signal Processing*, vol. 62, no. 22, pp. 5919–5929, 2014.

[14] Y. Xue, P. Zhou, T. Jiang, S. Mao, and X. Huang, "Distributed learning for multi-channel selection in wireless network monitoring," in *2016 13th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*. IEEE, 2016, pp. 1–9.

[15] H. Shokri-Ghadikolaei, C. Fischione, G. Fodor, P. Popovski, and M. Zorzi, "Millimeter wave cellular networks: A mac layer perspective," *IEEE Transactions on Communications*, vol. 63, no. 10, pp. 3437–3458, 2015.

[16] L. Chen, J. Xu, and Z. Lu, "Contextual combinatorial multi-armed bandits with volatile arms and submodular reward," in *Advances in Neural Information Processing Systems*, 2018, pp. 3247–3256.

[17] M. Wildemeersch, T. Q. Quek, C. H. Slump, and A. Rabbachin, "Cognitive small cell networks: Energy efficiency and trade-offs," *IEEE Transactions on Communications*, vol. 61, no. 9, pp. 4016–4029, 2013.

[18] H. Zhang, C. Jiang, N. C. Beaulieu, X. Chu, X. Wang, and T. Q. Quek, "Resource allocation for cognitive small cell networks: A cooperative bargaining game theoretic approach," *IEEE Transactions on Wireless Communications*, vol. 14, no. 6, pp. 3481–3493, 2015.

[19] L. Gavrilovska, D. Denkovski, V. Rakovic, and M. Angjelichinoski, "Medium access control protocols in cognitive radio networks: Overview and general classification," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 4, pp. 2092–2124, 2014.

[20] S. Chen, K. Zeng, and P. Mohapatra, "Efficient data capturing for network forensics in cognitive radio networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 22, no. 6, pp. 1988–2000, 2014.

[21] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in applied mathematics*, vol. 6, no. 1, pp. 4–22, 1985.

[22] J. Marinho and E. Monteiro, "Cognitive radio: survey on communication protocols, spectrum decision issues, and future research directions," *Wireless networks*, vol. 18, no. 2, pp. 147–164, 2012.

[23] P. K. Gopala, L. Lai, and H. El Gamal, "On the secrecy capacity of fading channels," in *2007 IEEE International Symposium on Information Theory*. IEEE, 2007, pp. 1306–1310.

[24] J. Barros and M. R. Rodrigues, "Secrecy capacity of wireless channels," in *2006 IEEE International Symposium on Information Theory*. IEEE, 2006, pp. 356–360.

[25] G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák, "Maximizing a monotone submodular function subject to a matroid constraint," *SIAM Journal on Computing*, vol. 40, no. 6, pp. 1740–1766, 2011.

[26] S. Liu, G. Xing, H. Zhang, J. Wang, J. Huang, M. Sha, and L. Huang, "Passive interference measurement in wireless sensor networks," in *The 18th IEEE International Conference on Network Protocols*. IEEE, 2010, pp. 52–61.

[27] L. Chen, A. Krause, and A. Karbasi, "Interactive submodular bandit," in *Advances in Neural Information Processing Systems*, 2017, pp. 141–152.

[28] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning*, vol. 47, no. 2-3, pp. 235–256, 2002.

[29] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 661–670.

[30] J. Xu, Q. Wang, R. Jin, K. Zeng, and M. Liu, "Secondary user data capturing for cognitive radio network forensics under capturing uncertainty," in *2014 IEEE Military Communications Conference*. IEEE, 2014, pp. 935–941.

[31] L. Chen and J. Xu, "Task replication for vehicular cloud: Contextual combinatorial bandit with delayed feedback," in *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 2019, pp. 748–756.

[32] W. Hoeffding, "Probability inequalities for sums of bounded random variables," in *The Collected Works of Wassily Hoeffding*. Springer, 1994, pp. 409–426.