Fast and Efficient Online Selection of Sensors for Transmitter Localization

Arani Bhattacharya
IIIT Delhi
arani@iiitd.ac.in

Abhishek Maji
Hitachi Energy
amaji@kth.se

Jaya Prakash Verma Champati
IMDEA Networks Institute
jaya.champati@imdea.org

James Gross
KTH Royal Institute of Technology
james.gross@ee.kth.se

***Abstract*—The increase in cost and usage of RF spectrum has made it increasingly necessary to monitor its usage and protect it from unauthorized use. A number of prior studies have designed algorithms to localize unauthorized transmitters using crowdsourced sensors. To reduce the cost of crowdsourcing, these studies select the most relevant sensors a priori to localize such transmitters. In this work, we instead argue for online selection to localize such transmitters. Online selection can lead to more accurate localization using limited number of sensors, as compared to selecting sensors a priori, albeit at the cost of higher latency. To account for the trade-off between accuracy and latency, we add a constraint on the number of selection rounds. For the case where the number of rounds is equal to the number of selected sensors, we propose a heuristic based on Thompson Sampling and show using trace-driven simulation that it provides 23% better accuracy compared to a number of proposed baseline algorithms. For restricted number of rounds, we show that using conventional parallel version of the modified Thompson Sampling which selects equal number of sensors in each round results in a substantial reduction in accuracy. To this end, we propose a strategy of selecting decreasing number of sensors in subsequent rounds of the modified Parallel Thompson Sampling. Our evaluation shows that the proposed heuristic leads to only 3% reduction in accuracy in contrast to 22% using modified Parallel Thompson Sampling, when we select 50 sensors in 20 rounds.**

I. Introduction

As spectrum has become more expensive, understanding its usage patterns to better regulate its usage is getting important. A key technique used by most spectrum monitoring systems is to deploy a distributed set of sensors by crowdsourcing [1]–[7]. Analysis of the received signals from the distributed set of sensors can be used to detect and/or localize the wireless transmitters, including unauthorized ones. A number of such studies have been recently proposed, showing the feasibility of localizing unauthorized transmitters by deploying low-cost spectrum sensors [8], [9]. A major problem with such crowdsourced sensor deployment is that running them costs energy. Spectrum sensors consume energy as well as incur data cost as the sensor's output is sent to a cloud center for further processing [10], [11]. In some cases, users might also have to be given incentives to keep the sensors running [5], [8]. Thus, running these sensors continuously can quickly add to the overhead of monitoring spectrum. A technique for reducing the overhead cost is needed to manage this running cost.

Recent studies propose to reduce the costs by *selecting* the most relevant sensors [11], [12]. Such selection takes into account the fact that the sensors are noisy in nature, and then formulate sensor selection as a modified version of the stochastic set cover problem [12]. These studies depend on a hypothesis-driven Bayesian approach for localization, which can be used without any assumption on propagation models and is based on prior training of the joint probability distributions of sensor observations for each hypothesis. A hypothesis represents a potential location or configuration of the transmitter. Thus, the problem of localizing an unauthorized transmitter is reduced to finding the most likely hypothesis. The objective is to maximize accuracy of either detection or localization while adhering to a budget on the number of sensors. While this technique can reduce the cost of spectrum monitoring, it relies on static or a priori selection of sensors.

In contrast, in this paper, in addition to a budget on the number of sensors, we argue for sequential or *online* selection of sensors, which works by activating more number of relevant sensors by observing the output of a limited number of sensors that have been activated in previous rounds (illustrated in Figure 1). We model this problem of online sensor selection as a Constrained Partially Observable Markov Decision Problem (CPOMDP). We first deal with the special case where the number of rounds of selections is equal to the budget. This special case is closely related to the stochastic multi-armed bandit problem. In this case, we design a modified form of Thompson Sampling [13] called Hypotheses-based Thompson Sampling (HTS) and show that it provides higher accuracy than the proposed baseline techniques.

While HTS gives a good solution, as we will explain in §2, the latency in accurately localizing a transmitter scales linearly with the number of rounds, and therefore selecting a single sensor in each round incurs high latency. This latency occurs due to the repeated network delays involved in sequential selection of sensors. To circumvent this, we impose an additional limit on the number of rounds over which sensors can be selected. This can reduce the amount of network delay by querying sensors in parallel. Standard techniques such as Parallel Thompson Sampling (PTS) select an equal number of sensors in each round concurrently to handle such cases [14]. Instead, we show empirically, selecting different number of sensors in each round can significantly improve the accuracy of localization. We identify how such improvement in localization accuracy can be obtained by designing a heuristic called Asymmetric Modified Thompson Sampling (AMTS) that assigns budgets to each round.

We implement both HTS and AMTS on problem instances with large number of hypotheses. The runtimes of both HTS and AMTS take around $24\ ms$ of time per sensor on average on an instance with 4096 hypotheses. In contrast, an

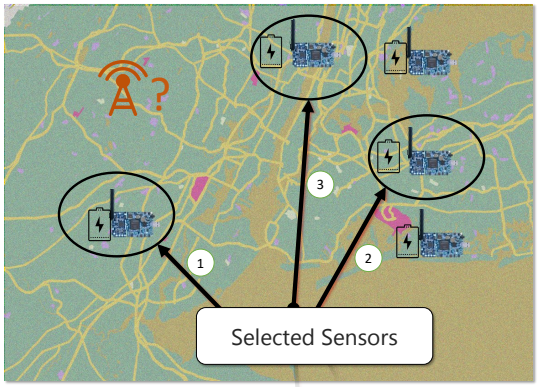

Fig. 1: An illustration of the online sensor selection system. The numbers alongside the arrows denote the sequence in which sensors are selected to localize a transmitter. Note that some sensors are never selected at all.

offline algorithm takes around $18ms$, whereas other baseline techniques such as greedy take over $1s$. We compare the performance of HTS with other baseline techniques, and show that it outperforms them by at least 22% higher accuracy for the case of unconstrained number of rounds. On the other hand, for the case of constrained number of rounds, AMTS outperforms hypotheses-based parallel thompson sampling (HPTS), a straightforward extension of HTS, by up to 19% higher accuracy.

We summarize our contributions as follows:

- We formulate the problem of online selection of sensors for localizing a transmitter which turns out to be a CPOMDP with exponentially large state space. Noting that POMDP is PSPACE-hard, for which the existence of an approximate algorithm is not guaranteed, we resort to proposing heuristic policies.
- We first study the relaxed version of the problem, where there is no restriction on the number of rounds. For this case, we show an optimal policy selects a single sensor in each round. We then design the heuristic HTS which selects one sensor per round.
- For the general problem, we modify HTS to design the baseline heuristic HPTS which selects equal number of sensors per round. To obtain a better performance, we propose AMTS which selects the number of sensors in the decreasing order with rounds.
- We evaluate both HTS and AMTS using large-scale trace-driven simulations, and show that they perform better in practice than other proposed baseline techniques.

The rest of this paper is organized as follows. We present the background and motivation of the problem, and formalize notations in §II. We propose baseline policies and HTS for the case of no restriction on the number of rounds in §III. We present AMTS in §IV and evaluate proposed heuristics in §V. We discuss related work in §VI and conclude in §VII.

II. Notation and Problem Statement

Problem Setting: We have a fixed area that we need to monitor for the presence/location of an transmission of interest. Such transmission could be either due to unauthorized utilization of RF spectrum [9], malfunctioning devices or malicious software [8].[1] Let $\mathbf{S}$ denote the set of spectrum sensors deployed or available (if attached to mobile devices) in the area at known locations. Each sensor $s \in \mathbf{S}$ can measure Received Signal Strength Indicator (RSSI), and when selected, reports RSSI to a central server, which estimates the location of the transmitter. We consider that the sensor observations are noisy, and denote the RSSI of sensor s by a random variable X_s and an observation received from it by x_s. Similarly, for a subset of sensors $\mathbf{T} \subseteq \mathbf{S}$, we use $\mathbf{X_T}$ to denote the random vector for RSSI and $\mathbf{x_T}$ to denote the observation vector from the sensors.

We represent potential locations of the transmitter of interest by hypotheses $H_0, H_1, \ldots, H_m$, where each hypothesis H_i represents a certain configuration (location and transmit power) of the transmitter of interest. We use the convention that the hypothesis H_0 corresponds to no transmitter of interest being present in the area. Since RSSI at a sensor is determined by its location relative to the transmitter of interest, the observations are directly related to the true hypothesis. As in other studies [12], we assume that the following inputs – obtained via a priori training, data gathering and/or analysis – are available:

- Prior probabilities of the hypotheses, i.e. $P_0(H_i)$, for each hypothesis H_i.
- Joint Probability Distribution (JPD) of sensors' observations for each hypothesis. More formally, for the set of sensors $\mathbf{S}$ and a given true hypothesis H_j, we assume $P(\mathbf{X_S}|H_j)$ is known. Note that this also gives us the JPD's of each subset $\mathbf{T} \subseteq \mathbf{S}$.

Given $\mathbf{x_S}$ and H_i, we consider $P(H_i|\mathbf{x_S})$ has Gaussian distribution $N(\mathbf{p}_i, \Sigma)$, with mean vector $\mathbf{p}_i$ and covariance vector Σ. Prior works have shown that the Gaussian distribution serves as a good approximation for such data [12]. Note that the covariance matrix remains same across hypotheses, since the correlation and noise are properties of the sensors. The mean vector $\mathbf{p}_i$ constitutes the mean RSSI values of all the sensors when H_i is true. We use $\mathbf{p}_{i,\mathbf{T}}$ to denote subset of $\mathbf{p}_i$ corresponding to the mean RSSI values of sensors from the set $\mathbf{T} \in \mathbf{S}$. In other words, $\mathbf{X_T} \sim N(\mathbf{p}_{i,\mathbf{T}}, \Sigma_\mathbf{T})$, where $\Sigma_\mathbf{T}$ is the covariance matrix for the sensors from the set $\mathbf{T}$. For convenience, we use $\mathbf{X_T} \sim N_T(\mathbf{p}_i, \Sigma)$, where $N_T(\mathbf{p}_i, \Sigma) = N(\mathbf{p}_{i,\mathbf{T}}, \Sigma_\mathbf{T})$.

Given the observations $\mathbf{x_T}$ from a selected set of sensors $\mathbf{T} \subseteq \mathbf{S}$, the posterior probability that hypothesis H_i is true is obtained using Bayes' rule:

$$P(H_i|\mathbf{x_S}) = \frac{P(\mathbf{x_S}|H_i)P_0(H_i)}{\sum_{j=0}^{m} P(\mathbf{x_S}|H_i)P_0(H_j)} \tag{1}$$

We select the hypothesis using Maximum a Posteriori (MAP) rule, which states that the hypothesis with the highest value of the posterior probability is most likely to be true. Formally given the observation vector $\mathbf{x_T}$, the most likely hypothesis H

[1] In this work, we assume that only a single transmission of interest is present in the area. Note that it is possible in our setting to have multiple other transmitters with known locations. Since separating out the power of such known transmitters is relatively simple [8], for simplicity we deal with a single transmitter of interest at an unknown location.

Algorithm 1 Algorithmic framework to select sensors

INPUT: Set of available sensors $\mathbf{S}$, budget B, priors $P(\mathbb{H})$
OUTPUT: Sequence of sensors $\mathbf{T}_1 \dots \mathbf{T}_K$

$k \leftarrow 1$
while $k \leq K$ **do**
  Select budget B_k for current stage, where $B_k \leq B$
  Select $\mathbf{T}_k$, where $T_k \subseteq \mathbf{S}$ and $|T_k| = B_k$
  Observe the values given by $\mathbf{T}_k$ to get the random vector $X_{\mathbf{T}_k}$
  Update $P(\mathbb{H}|R_k)$ to $P(\mathbb{H}|X_{R_{k+1}})$
  $\mathbf{S} \leftarrow \mathbf{S} \setminus T_k$
  $B \leftarrow B - B_k$
  $k \leftarrow k + 1$
end while
return $\mathbf{T}_1, \dots, \mathbf{T}_K$

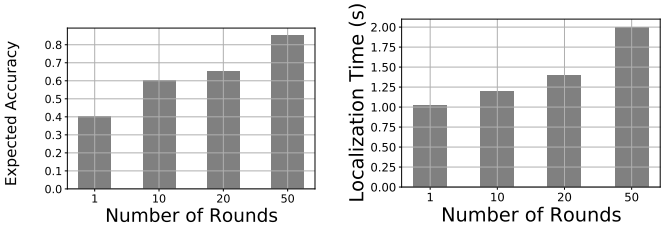

Fig. 2: Online selection of 50 sensors in varying number of rounds using Hypothesis-based Parallel Thompson Sampling. In each round, we account for the average processing time of 24ms (observed from our experiments), and assume that the network transmission delay is 20ms. We consider a fixed transmission delay of 20 ms per round even if multiple sensors are selected in a round. This is based on the fact that the data size of a sensor observation is small and multiple observations can be sent in parallel to the central server. We note that even in 20 rounds, the belief of the true hypothesis is on average less than HTS (50 rounds) by 18%. Having a single round is equivalent to offline sensor selection.

is given by:

$$H = \arg\max_{H_i} P(H_i|\mathbf{x_S}). \tag{2}$$

We collectively refer to the set of probability distributions $P(H_i|\mathbf{x_S})$ by the vector $P(\mathbb{H}|\mathbf{x_S})$. Note that initially, when no information is available, the prior probability $P_0(\mathbb{H})$ is a constant vector.

Motivation: Our goal is to select the most relevant subset of sensors $\mathbf{T} \subseteq \mathbf{S}$ that maximizes the posterior probability of the true hypothesis. Since selecting and observing each sensor's report costs energy and bandwidth [10], [11], we consider that the total number of sensors that can be selected cannot exceed a given budget $B < |S|$. The selection of B sensors can be performed in K rounds, where $K \in \{1, \dots, B\}$. There is an inherent trade-off between *accuracy* of localization (will be defined in a short while) and *latency* in detecting the transmission of interest which can be tuned by the choice of K. Note that the latency arises due to the delays involved with network transmission of the RSSI values and the processing required at the central server for computing the posterior probabilities. If all B sensors are selected in one round, i.e. if $K = 1$, then we will have low latency, but also lower accuracy. On the other hand, if we select B sensors in B rounds, i.e. if $K = B$, then we will have high latency, but much higher accuracy. The improvement in accuracy in this case results from exploiting the observations from the sensors in previous rounds for selecting the "best" sensors in the current round. We show this tradeoff by running multiple trials of the sensor selection problem in Figure 2 (details of the evaluation are discussed in §V), using the proposed Hypothesis-based Parallel Thomson Sampling (HPTS) algorithm. We note that when we select 50 sensors in 50 rounds, HPTS is equivalent to HTS, and we have an accuracy of 0.93. However, accuracy reduces to 0.68 when we select 50 sensors in 10 rounds. On the other hand, a reduction in the number of rounds, by increasing the number of sensor selected in each round, results in lower latency. For example, selecting 50 sensors in 50 rounds results in a latency of 2 s, whereas selecting them in 10 rounds results in 1.1 s latency. For thousands of sensors, this can lead to potential saving of minutes, which is essential for low-latency localization. Such low-latency localization can significantly improve spectrum allocation policies and lead to better allocation of spectrum in cognitive radio networks [15].

In our problem formulation, we capture the above trade-off for any given $K < B$. In k^{th} round, where $k \leq K$, let $\mathbf{T}_k$ denote the subset of sensors selected, then

$$\mathbf{T}_k \subseteq \mathbf{S} \setminus \bigcup_{l=0}^{k-1} \mathbf{T}_l, \tag{3}$$

where $\mathbf{T}_0 = \phi$, the empty set. We define $\mathbf{R}_k = \cup_{l=0}^{k-1} \mathbf{T}_l$.

Online Policy: An online policy is a series of actions $(\pi_1, \dots, \pi_K)$, where π_k specifies the *conditional distribution* for selecting a subset from $\mathbf{S} \setminus \mathbf{R}_k^\pi$ based on the observation vector $\mathbf{x}_{\mathbf{R}_k^\pi}$, where $\mathbf{R}_k^\pi$ is the set of sensors chosen under π until round $k-1$. In the sequel, we suppress π in the superscript when there is not ambiguity. A generic online policy is described in Algorithm 1.

Let H_j be the true hypothesis, then the probability, denoted by $P^j_{\pi_k}(\mathbf{x}_{\mathbf{T}_k}|\mathbf{x}_{\mathbf{R}_k})$, of observing the vector $\mathbf{x}_{\mathbf{T}_k}$ under a policy π in round k is determined by both $P(\mathbf{x}_{\mathbf{T}_k}|H_j)$ and the conditional probability $P_{\pi_k}(\mathbf{T}_k \subset \mathbf{S} \setminus \mathbf{R}_k|\mathbf{x}_{\mathbf{R}_k})$ specified by π_k for selecting the set $\mathbf{T}_k \subset \mathbf{S} \setminus \mathbf{R}_k$. Formally,

$$P^j_{\pi_k}(\mathbf{x}_{\mathbf{T}_k}|\mathbf{x}_{\mathbf{R}_k}) = P_{\pi_k}(\mathbf{T}_k \subset \mathbf{S} \setminus \mathbf{R}_k|\mathbf{x}_{\mathbf{R}_k}) P(\mathbf{x}_{\mathbf{T}_k}|H_j). \tag{4}$$

Using conditional probabilities, we obtain the probability of observing $\mathbf{x}_{\mathbf{R}_k}$ under π, when H_j is the true hypothesis, as follows:

$$P^j_\pi(\mathbf{x}_\mathbf{T}) = \prod_{k=1}^{K} P^j_{\pi_k}(\mathbf{x}_{\mathbf{T}_k}|\mathbf{x}_{\mathbf{R}_k}). \tag{5}$$

The *expected accuracy* of a given policy π is measured using the function $g(\pi)$, given by

$$g(\pi) = \sum_{j=0}^{m} \mathbb{E}_{\mathbf{x_T} \sim P^j_\pi(\mathbf{x_T})}[P(H_j|\mathbf{x_T})] P_0(H_j). \tag{6}$$

To understand function $g(\pi)$, consider that the prior probabilities for the hypotheses follow uniform distribution, i.e. $P_0(H_j = 1/(m+1))$, for all j. In this case, $g(\pi)$ quantifies the average over the expected posterior probabilities of the

hypothesis, where the expectation for each hypothesis involves the JPD of that hypothesis. In other words, for a given true hypothesis, we want the expected posterior probability of that hypothesis to be maximized by a policy so that the MAP algorithm outputs that hypothesis as the most likely hypothesis on an average. However, since the true hypothesis is not known we aim to maximize the summation of the expected posterior probabilities of the hypotheses (sum of the individual posterior probabilities weighted by the prior probabilities) which is given by $g(\pi)$[2].

Given B and K, we are interested in computing a π that solves the following problem:

$$\begin{aligned} \underset{\pi}{\text{Maximize}} \quad & g(\pi) \\ \text{s.t.} \quad & |\cup_{k=1}^{K} \mathbf{T}_k^{\pi}| = B. \end{aligned} \tag{7}$$

Note that the constraint in (7) embodies both constraints on the number of rounds and the total number of sensors selected. The number of rounds is constrained due to the amount of latency that is allowed, whereas the number of sensors is constrained by the available budget. Let g^* denote the optimum expected accuracy for (7).

Problem (7) can be formulated as a Constrained Partially Observable Markov Decision Problem (CPOMDP) with a state space that is exponentially large as it comprises of all possible realizations $\mathbf{x_T}$ for all $\mathbf{T} \subset \mathbf{S}$ such that $|\mathbf{T}| \leq B$. Given that solving a POMDP is PSPACE hard [16], the existence of an approximation algorithm is not guaranteed. Therefore, we resort to using a heuristics based on Thompson Sampling. It is worth noting that, a special case of our problem, where only one sensor needs to be selected, i.e. $B = 1$ (and allowing selection of a sensor multiple times), is related to the Multi-Armed Bandit (MAB) problem setting. However, extending the existing performance guarantees of Thompson Sampling – provided for MAB with respect to a regret function (cf. [17])– for (7) is a challenging and open problem.

III. Policy for Unconstrained Number of Rounds

We first relax the constraint on the number of rounds, i.e. assume that $K = B$ and explore different strategies to solve the relaxed problem below.

$$\begin{aligned} \underset{\pi}{\text{Maximize}} \quad & g(\pi) \\ \text{s.t.} \quad & |\cup_{k=1}^{B} \mathbf{T}_l^{\pi}| = B. \end{aligned} \tag{8}$$

Once we obtain a strategy for the relaxed problem, we then build on the solution of the relaxed problem to design a solution for the constrained problem.

A. Comparison of Sequential and Offine Policies

We first prove a few properties of our objective $g(\pi)$ that allows us to propose a solution of the relaxed problem. We note that, any offline algorithm which chooses a set $R \subseteq S$ – such that $|R| = B$ – in one shot cannot achieve an expected accuracy greater than g^*. To see this, the solution R provided by an offline algorithm is one feasible solution for (7). In particular, for this case, we have $T_1 = R$ and $T_k = \{\}$ for all $2 \leq k \leq K$, and the expected accuracy is given by

$$g(\{T_1\}) = \sum_{j=0}^{m} P(H_j|R_K)P_0(H_j).$$

Since the offline algorithm does not use observations, the prior probabilities are simply $P_0(H_j)$. The above argument asserts that an optimal offline solution can be achieved by an online algorithm. In the following proposition, we present a stronger statement that an optimal online policy belongs to the set of policies where a policy selects one sensor in each round, i.e. $|\mathbf{T}_k| = 1,\ \forall 1 \leq k \leq B$. We formalize this notion in the following theorem:

Proposition 1. *For the relaxed problem* (8)*, there exists an optimal online policy that selects one sensor in each round, i.e.* $|\mathbf{T}_k| = 1,\ \forall 1 \leq k \leq B$.

Proof: Let $\hat{\pi}$ be an optimal online policy that chooses more than one sensor in a round and achieves $g^* = g(\hat{\pi})$. We argue that, using $\hat{\pi}$ one can construct an online policy π^* which selects one sensor per round and also achieves g^*. To see this, the expected accuracies under two policies are equal if their conditional probabilities (defined in (5)) are equal. Now, for some hypothesis H_j and set $\mathbf{T}$ (with $|T| = B$), the conditional probability under $\hat{\pi}$ is given by

$$P_{\hat{\pi}}^{j}(\mathbf{x_T}) = \prod_{k=1}^{r} P_{\hat{\pi}_k}^{j}(\mathbf{x}_{\mathbf{T}_k}|\mathbf{x}_{\mathbf{R}_k}).$$

Since more than one sensor is selected in some round, say $\hat{k}$, we have $r < B$. To simplify the argument, we further assume that $\hat{\pi}$ selected one sensor in all rounds except in round $\hat{k}$, where it selected n sensors. We assign the same conditional probabilities $P_{\hat{\pi}_k}^{j}(.)$ to the policy π^* except in rounds $\hat{k}+1$ to $\hat{k}+n-1$; in these rounds, π^* chooses (randomly) one sensor per round from the selected set in round $\hat{k}$ using conditional probability $P_{\hat{\pi}_{\hat{k}}}^{j}(\mathbf{x}_{\mathbf{T}_{\hat{k}}}|\mathbf{x}_{\mathbf{R}_{\hat{k}}})$. In other words, π^* will be selecting with probability one sensor in rounds $\hat{k}+1$ to $\hat{k}+n-1$ from the same selected set (from round $\hat{k}$). Thus, by construction, $P_{\pi^*}^{j}(\mathbf{x_T})$ will be equal to $P_{\hat{\pi}}^{j}(\mathbf{x_T})$. This construction can be similarly done for other cases where $\hat{\pi}$ selects more than one sensor in multiple rounds. Since π^* has same conditional probabilities as $\hat{\pi}$, the result is proven. ■

B. Choices of Policies

Intuitively, selecting one sensor per round reveals relatively more information thus leading to make better selection of the sensors which is also asserted by Proposition 1. Therefore, we resort to policies that select one sensor per round. Furthermore, owing to the fact that a special case of our problem is related to MAB problem, we borrow some of the strategies used to solve an MAB problem, and discuss how they can be utilized for our problem. In the following, we describe different policies.

Greedy Policy: As a baseline comparison, we propose the following greedy policy. At round k, given the observation vector $\mathbf{x}_{\mathbf{R}_k}$, the policy takes action π_k which chooses exactly

[2]Note that other objective functions could also be potentially used, but we have left their utilization for future work.

one sensor c, i.e.

$$P_{\pi_k}(\mathbf{T}_k = \{c\}|\mathbf{x}_{\mathbf{R}_k}) = 1; P_{\pi_k}(\mathbf{T}_k = \{d\}|\mathbf{x}_{\mathbf{R}_k}) = 0, \forall\, d \in \mathbf{S}\backslash\{c\}, \tag{9}$$

where

$$c = \arg\max_{s \in \mathbf{S}\setminus R_k} \sum_{j=0}^{m} P(H_j|\mathbf{x}_{\mathbf{R}_k})[\mathbb{E}_{\mathbf{x}_s \sim N_s(\mathbf{p}_j,\Sigma)}[P(H_j|\mathbf{x}_{R_k} \cup x_s)]], \tag{10}$$

The sensor c is the one that increases the expected probability of the true hypothesis the most, where the expectation is with respect to the current beliefs of the hypotheses.

Although using the greedy policy is intuitive, it has a major drawback. The policy only looks at the *mean* improvement in accuracy. Since we only have imperfect knowledge about the position of the transmitters, there is also a variance of improvement in the posterior probability of the true hypothesis. The greedy policy ignores this variance and only chooses the sensor with the highest mean improvement. It is possible that with better knowledge of the transmitter location, the variance of the gain might reduce, and the sensor's actual gain turns out to be lower than that of some other sensor. In other words, the greedy algorithm relies excessively on the current beliefs without accounting for the the variance.

ϵ-Greedy Policy: One method that can be used to circumvent the disadvantage of greedy policy is to use a method called ϵ-greedy. In ϵ-greedy, we choose the sensor given by greedy policy with probability $1-\epsilon$, and choose a sensor randomly with probability ϵ. Note that we never choose any sensor from $\mathbf{R}_k$ that has been selected in previous round. The action π_k under ϵ-greedy uses the following conditional distribution given $\mathbf{x}_{\mathbf{R}_k}$.

$$\begin{aligned} &P_{\pi_k}(\mathbf{T}_k = \{c\}|\mathbf{x}_{\mathbf{R}_k}) = 1-\epsilon, \\ &P_{\pi_k}(\mathbf{T}_k = \{d\}|\mathbf{x}_{\mathbf{R}_k}) = \frac{\epsilon}{|\mathbf{S} \setminus \mathbf{R}_k \cup \{c\}|}, \forall d \in \mathbf{S} \setminus \mathbf{R}_k \cup \{c\}, \end{aligned} \tag{11}$$

where c is as defined in (10). Choosing a sensor randomly allows us to explore alternative hypotheses, whereas choosing a sensor greedily allows us to exploit our available information. This policy can circumvent the disadvantage of choosing the greedy sensor, and performs better in practice (experimentally shown in §V). The key drawback of this technique is that the value of ϵ needs to be carefully tuned, since a careful balance between greedy and random policy is needed. Next, we present the proposed HTS policy.

Hypotheses-based Thompson Sampling (HTS): The standard Thompson Sampling (cf. [17]) looks at the distribution of rewards of each possible action. From each of these distributions, it draws a sample and selects the action corresponding to the distribution from which the highest reward is drawn. However, computing the distributions of the rewards for each action is compute-intensive, since the distributions are correlated with the distributions of the hypotheses.

To handle this problem, we modify Thompson Sampling in the following way. Instead of computing the distribution of rewards, we take a realization $\mathbf{x}_{\mathbf{R}_k}$ of the sensors $\mathbf{R}_k$ that are selected before round k. This gives us the probabilities of each hypothesis, as perceived by the policy at k^{th} round, i.e. $P(H_j|\mathbf{x}_{\mathbf{R}_k}), \forall j = 0, \ldots, m$. Since the transmitter might be at any one of the $m+1$ locations, we randomly draw a sample $\mathcal{H}$ from the categorical distribution $[P(H_j|\mathbf{x}_{\mathbf{R}_k})]$, and compute the improvement in posterior probability if $\mathcal{H}$ were true. We pick the sensor c that increases the posterior probability of $\mathcal{H}$ the most. Mathematically, this action is defined as:

$$c = \arg\max_{s \in \mathbf{S}\setminus R_k} \mathbb{E}_{\mathbf{x}_s \sim N(p_j,\Sigma)}[P(\mathcal{H}|\mathbf{x}_{R_k} \cup x_s)] \tag{12}$$

This provides a natural exploration-exploitation tradeoff, since the hypothesis with the highest prior probability is selected more frequently, but the other less likely hypotheses are not completely ignored. Moreover, this technique has a much lower time complexity than all the other techniques, as it does not require computation of the mean or variance across all the hypotheses. We call this modified form of Thompson Sampling as HTS. We have experimentally observed that HTS provides the best result, when there is no constraint on the number of rounds, compared to all the three policies, and thus we utilize HTS for the rest of this paper.

Time Complexity: We note that the greedy and ϵ-greedy techniques require a total of m computations of the expected improvement in posterior probability for each sensor s. The posterior probability in Bayes' rule can be computed by a single multiplication, and thus can be done in constant time. Thus, the greedy and ϵ-greedy techniques take $O(m|\mathbf{S}|)$ number of computations for a single round. Since there are a total of K number of rounds, the time complexity of these algorithms is equal to $O(Km|\mathbf{S}|)$.

On the other hand, HTS requires only $O(|\mathbf{S}|)$ computations of the posterior probability in a single round, since it requires computation of for a single H_i in a particular round. Since it draws a single hypothesis $\mathcal{H}$, and drawing from a categorical distribution takes constant time, a single round takes $O(|\mathbf{S}|)$ computations. Repeating this across K rounds results in $O(K|\mathbf{S}|)$ time complexity, which is much lower than greedy and ϵ-greedy strategies.

Remark: An alternative approach to balance between exploration and exploitation is to take into account both mean and variance of posterior probability, using a technique called Upper Confidence Bound (UCB) [18]. The mean and variance can be taken into account by computing a weighted sum, as shown in the following equation:

$$\begin{aligned} c = \arg\max_{s \in \mathbf{S}} \sum_{j=0}^{m} P(H_j|\mathbf{x}_{\mathbf{R}_k})[\mathbb{E}_{\mathbf{x}_s \sim N_s(\mathbf{p}_j,\Sigma)}[P(H_j|\mathbf{x}_{R_k} \cup x_s))] \\ + \lambda \sigma_{\mathbf{x}_s \sim N_s(\mathbf{p}_j,\Sigma)}[P(H_j|\mathbf{x}_{R_k} \cup x_s)]], \end{aligned} \tag{13}$$

where λ is a weighing parameter. We choose the sensor c that has the highest value of the weighted sum.

However, computing the standard deviation of $P(H_j|\mathbf{x}_{\mathbf{R}_k} \cup x_s)$ is challenging. This is because the standard deviation depends on the joint probability distribution of the hypotheses. It is difficult to estimate how these probabilities are correlated. While it is possible to estimate these correlations by Monte Carlo simulations, this is too time consuming to be practical. Thus, we do not utilize this technique in this work.

Algorithm 2 HPTS

INPUT: Set of available sensors $\mathbf{S}$, budget B, priors $P(\mathbb{H})$
OUTPUT: Sequence of sensors $\mathbf{T}_1 \dots \mathbf{T}_K$

$k \leftarrow 1$
while $k \leq K$ **do**
  $B_k = \lfloor B/K \rfloor$ ▷ Allocating equal budgets (Step (1))
  $\mathbf{T}_k = \phi$
  while $|\mathbf{T}_k| \leq B_k$ **do**
    Select a hypothesis $\mathcal{H}$ using priors $\{P(H_j|\mathbf{x}_{\mathbf{R}_k})\}$
    $c = \arg\max_{s \in \mathbf{S} \setminus R_k \setminus \mathbf{T}_k} \mathbb{E}_{\mathbf{x}_s \sim N(p_j, \Sigma)}[P(\mathcal{H}|\mathbf{x}_{R_k} \cup x_s)]$ ▷ Select sensors (Step (2))
    $\mathbf{T}_k \leftarrow \mathbf{T}_k \cup \{c\}$
  end while
end while
  return $\mathbf{T}_1, \dots, \mathbf{T}_K$

IV. Policy for Constrained Number of Rounds

If we have limited number of rounds, where $K < B$, then we need to select more than one sensor in at least one round. In this case, we rewrite the probability of realizing some $\mathbf{x}_{\mathbf{T}_k}$ under policy π as:

$$P_{\pi_k}(\mathbf{T}_k \subset \mathbf{S} \setminus R_k|\mathbf{x}_{R_k}) = \sum_{B_k=1}^{B-|\mathbf{R}_k|} P_{\pi_k}(B_k) P_{\pi_k}(\mathbf{T}_k \subset \mathbf{S} \setminus R_k, |\mathbf{T}_k| = B_k|\mathbf{x}_{R_k}). \quad (14)$$

Thus, an action π_k can be seen as taking two distinct decisions. The first is to choose the budget for the round B_k, and the second is to choose $\mathbf{T}_k \subseteq \mathbf{S}$, where $|\mathbf{T}_k| = B_k$.

Baseline Solution: The simplest feasible solution is to modify Hypotheses-based Thompson Sampling (HTS) to select sensors in batches, where the batch sizes remain equal across rounds. We refer to this policy by HPTS and is described in Algorithm 2. As we will see in §5, HPTS does not perform well when the number of rounds is smaller. This is because allocating equal number of sensors in Step 1 of Algorithms 2 is not optimal. In the rest of this section, we modify Step (1) to design a better performing heuristic. Let $\bar{\pi}$ be any policy that selects different number of sensors in different rounds using HTS. To be precise, we define $\bar{\pi}$ same as in Algorithm 2, except for Step (1). We now formulate the number of sensors selected in each step as an optimization problem under $\bar{\pi}$.

We consider static assignment of budgets, i.e. in round k, the policy $\bar{\pi}$ assigns a fixed B_k for the number of sensors to be selected in round k. under $\bar{\pi}$, given B_k for round k, the sensors are selected according to Step (2) of Algorithm 2, and therefore the choice that is left to be made is the sequence of budgets $\{B_k\}$ that maximizes the expected accuracy. To get a solution to this optimization problem, we first make note of a trivial property of expected accuracy: selecting additional sensors can only increase the expected accuracy, and not reduce it. This implies that the summation of the sequence $\{B_k\}$ should be equal to B number of sensors (and not any fewer) over the K rounds.

We now make a second empirical observation. If $\{B_k\} = [B_1, \dots, B_K]$, then selecting additional sensors in round r gives a higher value of g than by selecting the same number of additional sensors in round r', where $r < r'$. Formally, we

Algorithm 3 Asymmetric Modified Thompson Sampling (AMTS) to select sensors

INPUT: Set of available sensors $\mathbf{S}$, budget B, priors $P(\mathbb{H})$
OUTPUT: Sequence of sensors $\mathbf{T}_1 \dots \mathbf{T}_K$

$k \leftarrow 1$
$B_1 \leftarrow B/2$
$B_K \leftarrow 1$
$d \leftarrow \frac{B_r - B_1}{K}$
while $k \leq K$ **do**
  $\mathbf{B_k} \leftarrow \mathbf{B_1} - \mathbf{d} \times (\mathbf{k} - \mathbf{1})$
  $\mathbb{T}_k \leftarrow \phi$
  while $b \leq B_k$ **do**
    Pick $\mathcal{H}$ with probabilities $P(H_0|\mathbf{x}_{\mathbf{R}_k}), \dots P(H_m|\mathbf{x}_{\mathbf{R}_k})$
    Choose c with the maximum gain of $\mathcal{H}$
    $\mathbb{T}_k \leftarrow \mathbb{T}_k \cup c$
    $b \leftarrow b + 1$
  end while
  Observe values given by $\mathbf{T}_k$ to get the random vector $\mathbf{x}_{\mathbf{T}_k}$
  Update $P(\mathbb{H}|x_{\mathbf{R}_k})$ to $P(\mathbb{H}|X_{\mathbf{R}_{k+1}})$ using Bayes rule
  $k \leftarrow k + 1$
end while
return $\mathbf{T}_1, \dots, \mathbf{T}_K$

denote $g(\bar{\pi}, \{B_k\})$ as the policy where the sequence $\{B_k\}$ is used as the budget in each round, while the method of selecting sensors in each round remains same. Then, we have:

$$g(\bar{\pi}, [B_1, \dots, B_r + a, \dots B_K]) \geq g(\bar{\pi}, [B_1, \dots, B_{r'} + a, \dots B_K]), \text{ where } r < r'. \quad (15)$$

Intuitively, this is true because the marginal gain in $g(\cdot)$ reduces with an increase in the number of rounds. Thus, choosing more sensors in the early rounds leads to a higher overall value of $g(\bar{\pi}, \mathbf{B})$. This observation holds for any sub-sequence $\mathbf{B}_a$ of any length. Applying this observation recursively, we note that the sequence of decreasing number of sensors provides higher $g(\bar{\pi}, [B_1, \dots, B_K])$ and therefore, we choose:

$$B_1 \geq B_2 \geq \dots \geq B_K \quad (16)$$

Heuristic to Select Number of Sensors Per Round: Expression (16) provides a set of solutions to our optimization problem. Based on this expression, we use the following assignment for the sequence. We set $B_1 = \lfloor B/2 \rfloor$, $B_K = 1$, and set the subsequence to an arithmetic progression (AP). The common difference of the AP is equal to $\frac{B-2}{2(K-1)}$. This gives us the following budget for each round:

$$B_k = \lfloor B/2 \rfloor - (k-1)\frac{B-2}{2(K-1)} \quad (17)$$

Equation (17) defines a sequence to assign budgets to each round. This sequence defines the heuristic, and is used in place of step (1) of HPTS. Note that other sequences satisfying the criteria are possible, but we empirically observe that this heuristic performs the best (detailed comparisons in §VB).

Asymmetric Modified Thompson Sampling (AMTS): We call the algorithm obtained by modifying HPTS as Asymmetric

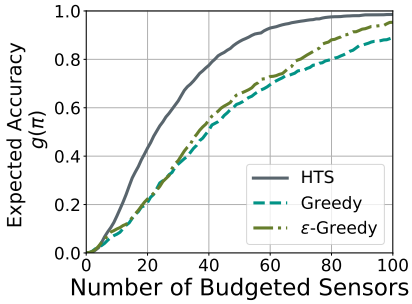

Fig. 3: Performance of HT, Greedy and ϵ-Greedy methods when the number of rounds is unlimited.

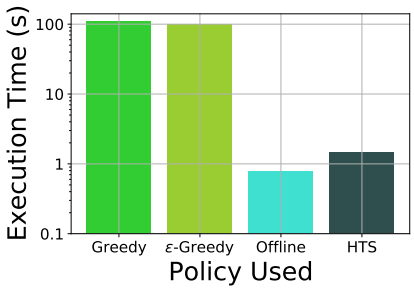

Fig. 4: Performance of HTS, Greedy and ϵ-Greedy in terms of execution time with a budget of 50 sensors.

Modified Thompson Sampling (AMTS), because of the asymmetric selection of sensors. We present the algorithm obtained by combining the two modifications in Algorithm 3. In Lines 2-3 we set the budget corresponding to first round B_1 to $\lfloor B/2 \rfloor$ and the budget corresponding to last round B_K to 1. We then compute the common difference of the arithmetic progression. For each round, we set the number of sensors according to an arithmetic progression (Line 6). We start with an empty subset $\mathbf{T}_k$ for round k (Line 7). Until the budget for the round is exhausted, we pick some hypothesis $\mathcal{H}$ drawn from the priors $P(H_0|\mathbf{x}_{\mathbf{R}_k}), \ldots, P(H_m|\mathbf{x}_{\mathbf{R}_k})$ (Line 9). It then selects c using the formula shown in Equation (12), and adds it to $\mathbf{T}_k$ (Lines 10-11). In this way, we keep selecting sensors for the round until the budget gets exhausted. Once the budget B_k gets exhausted, we observe the values or the output of the selected sensors $\mathbf{T}_k$, and compute the posterior probabilities. We repeat this process for each round, and finally return the sequence of sensors selected in each round. Thus, AMTS selects sensors in batches, depending on the number of available rounds.

V. Evaluation

Setting: We generate large-scale datasets using Longley-Rice propagation model using the tool SPLAT! [19]. SPLAT! is a widely used tool that uses landscape data from satellites as input to model the propagation of RF propagation of signals from cell phone towers. We take an area of 80×80 m^2, and divide it into 6400 grid cells, each of area $1m^2$. We ensure that the area has landscape features like hills to ensure that there are cases with non-line-of-sight data. We then simulate the presence of a transmitter in each of the individual grid cell, where the transmitter is at a height of $30m$ in each case. The transmitters transmit signals at different powers randomly selected in the range 25 to 35 dBm. We place a total of 500 sensors within this area, with the sensors all distributed randomly with uniform distribution.

We obtain the means of the JPD's by using the power values reported by SPLAT!. To obtain the standard deviations, we first obtain the datasets that are publicly available [20] and obtain the standard deviation in that dataset.

We implement each of the policies/algorithms in python 3.7. Since the algorithms are compute-intensive, we make extensive use of numpy and cupy to vectorize the compute-intensive operations. We run the simulations on an Intel Core i9-9900K CPU and nVidia Titan GPU. As discussed below, our algorithms run in the order of seconds in this environment.

A. Unconstrained Number of Rounds

We first evaluate the performance of HTS when the number of rounds is not constrained. In Section 3, we discussed a total of four common techniques – greedy, UCB, ϵ-greedy and HTS. We evaluate the performance of all these techniques, except UCB. We exclude UCB because we observed that its execution time is too slow for it to be feasible in practice, as running a single instance of the problem takes around 30 minutes. We run experiments with the budgeted number of sensors ranging from 1 to 100, and observe the value of $g(\pi)$ in each case. We simulate this for a total of 1000 instances, and then report the mean value.

Accuracy: Figure 3 shows the performance of these techniques. For the ϵ-greedy approach, we start with an ϵ value of 0.1 and then gradually reduces it linearly in steps of 0.01 with an increase in the number of selected sensors. We empirically observe that this tuning of ϵ provides the best performance of ϵ-greedy. We observe that HTS outperforms greedy by 22%, and ϵ-greedy by 20% for a budget of 60. The key reason why HTS does well is that it naturally balances the trade-off between exploration and exploitation. While ϵ-greedy performs better than greedy, this improvement is only up to 5%. This validates our choice of using HTS for online sensor selection.

Execution Time: We also look at the execution times of these approaches (Figure 4). Since the sensor selection is performed online, having a low execution time is critical. We observe that TS takes the least amount of time among the online approaches. HTS takes only around $1.52s$ to execute, compared to $105s$ for greedy and $100s$ for ϵ-greedy. Thus, HTS is 69 and 66 times faster than greedy and ϵ-greedy approaches. HTS is much faster because, as discussed in §3, it has a lower time complexity than the other techniques by a factor of m or the number of hypotheses. Since m is typically a large value (equal to 6400 in our experiment), this leads to a substantial reduction in execution time. We also note that HTS is only around 1.7 times slower than the pure offline selection. This shows that utilizing HTS is practical in a realistic scenario.

B. Constrained Number of Rounds

Comparison of AMTS and HPTS: We now evaluate the performance of the algorithms with constrained number of rounds. We compare the performance of our algorithm AMTS, with the baseline hypotheses-based Parallel Thompson Thompson (HPTS) as well as additional possible sequences. The other sequences include "Increasing", where the reverse sequence

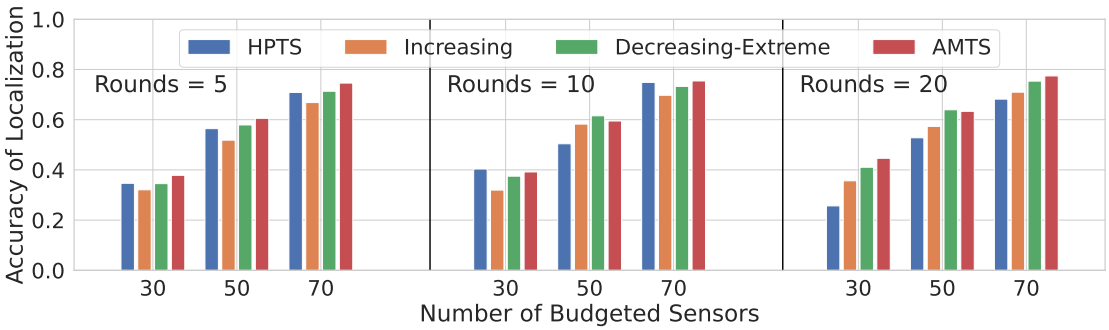

Fig. 5: Comparison of performance of HPTS and our method AMTS when number of rounds is (a) 5, (b) 10 and (c) 20. For the sake of comparison, we also show other possible heuristics "Increasing", where the reverse sequence of that of AMTS is used, and "Decreasing-Extreme", where the maximum number of sensors are selected in the first round.

of AMTS is used, and "Decreasing-Extreme", where the maximum number of sensors possible are selected in the first round itself. Figure 5 compares the accuracy of these techniques for budgets of 30, 50 and 70. We also show the performance for 5, 10 and 20 rounds. We note that AMTS outperforms HPTS by 0.19, 0.10 and 0.09 for budgets of 30, 50 and 70 respectively, when the number of rounds is equal to 20. We further observe that the accuracy of AMTS keeps increasing with an increase in the number of rounds, with the improvement of AMTS equal to 0.07 between rounds 20 and 5 for a budget of 30. While AMTS performs slightly worse than "Decreasing-Extreme" in a couple of cases and HPTS in a single case, these differences are less than 0.02, and thus are well within the margin of error. Thus, AMTS improves performance substantially when the budget is relatively low, whereas the number of rounds is equal to 20.

VI. Related Work

Crowdsourced Transmitter Localization: A number of studies have looked into the problem of crowdsourced transmitter localization and/or detection [1], [2], [8], [11], [21]. For example, [2] and [1] use cheap RTL-SDR sensors to localize transmitters, and benchmarks different algorithms for such localization. The work [21] proposes using UAV's or unmanned cars as spectrum policy enforcement agents and map this problem to a form of traveling salesman problem to reduce the amount of traveling involved. DeepMTL [7] and [6] utilize deep learning techniques to localize multiple transmitters. Like our work, SPLOT [8], [11] and [12] also focus on selecting the most relevant sensors. However, they all select the sensors offline to reduce latency of localization. In contrast, we map the problem of online selection that highlights the relationship between stochastic multi-armed bandit and online selection. SpecWatch [22] performs online selection of the channel band to be monitored, while assuming that the sensors are selected a priori. The work [23] also proposes online localization and shares the same objective of using the budgeted number of sensors, but utilizes the complementary approach of localization via Gaussian Process Regression. To the best of our knowledge, this is the first work that modifies Thompson Sampling to localize transmitters.

Online Feature Selection: A number of works in the machine learning literature study the problem of online feature selection [24]–[26]. For example, [25] propose using mutual information as a metric to select sensors to learn a Gaussian process. However, they also show that in case of noisy sensors, there are no performance guarantees. [26] show that performance guarantees using a greedy algorithm are possible only if the noise is limited. A number of other works also consider such selection as a case of "Value of Information" maximization problem [27]. We do not utilize these techniques because our objective function does not satisfy the criteria to get such performance guarantees owing to noisy sensors. We show in our experiments that our technique significantly outperforms the greedy algorithm.

Stochastic Multi-armed Bandits: There has been a lot of interest recently in stochastic multi-armed bandits (MAB) [28]. A number of techniques to solve MAB problems have been explored, including epsilon-greedy [28], Upper Confidence Bound [29] and Thompson Sampling. We have primarily focused on Thompson Sampling because of its low complexity and high accuracy. Thompson Sampling was first proposed in 1933 [13] and has recently drawn attention because of its good performance [30]. A recent study [14] proposed using Parallel Thompson Sampling (PTS) to speed up standard Thompson Sampling. Our algorithm modifies it to build a more accurate online sensor selection algorithm. A few recent works also deal with constrained bandits [31]. However, these studies currently do not utilize Thompson Sampling, instead preferring to use linear programming-based approaches, which are slower in practice. Finally, [32] deals with budgeted Thompson Sampling, but does not deal with concurrency. Our algorithm Asymmetric Modified Thompson Sampling (AMTS) uses these ideas to design a more accurate online sensor selection approach.

VII. Conclusion

We have formulated an online sensor selection for transmitter localization using a hypothesis-driven approach. In addition to limiting the number of sensors B that can be selected, we have considered a limit on the number of rounds K for selecting and acquiring the sensor observations to strike a trade-off between the expected accuracy and latency in detecting the presence of the transmitter. For the relaxed problem where $K = B$, the proposed HTS algorithm selects one sensor (per round), which increases the expected accuracy of a hypothesis that is chosen using the prior probability distribution at the start of the round. For the case where the number of rounds $K < B$, we have proposed the heuristic AMTS which chooses multiple sensors (per round) with decreasing number of sensors per round. We validated our approach using trace-driven simulation, and showed that both HTS outperforms the baseline greedy and ϵ-greedy techniques by 22% and 20% respectively. For limited rounds, AMTS also outperforms our

baseline HPTS by up to 20%. Moreover, both HTS and AMTS run in the order of seconds, making it feasible to do online selection.

REFERENCES

[1] A. Chakraborty, M. S. Rahman, H. Gupta, and S. R. Das, "Specsense: Crowdsensing for efficient querying of spectrum occupancy," in *IEEE INFOCOM 2017 - IEEE Conference on Computer Communications*, 2017. DOI: 10.1109/INFOCOM.2017.8057113 pp. 1–9.

[2] A. Nika, Z. Li, Y. Zhu, Y. Zhu, B. Y. Zhao, X. Zhou, and H. Zheng, "Empirical validation of commodity spectrum monitoring," in *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, ser. SenSys '16, 2016. DOI: 10.1145/2994551.2994557 p. 96–108.

[3] N. Kleber, M. Haenggi, J. Chisum, B. Hochwald, and J. N. Laneman, "Directivity in rf sensor networks for widespread spectrum monitoring," *IEEE Transactions on Cognitive Communications and Networking*, vol. Early Access, 2021. DOI: 10.1109/TCCN.2021.3124523

[4] S. Rajendran, R. Calvo-Palomino, M. Fuchs, B. Van den Bergh, H. Cordobes, D. Giustiniano, S. Pollin, and V. Lenders, "Electrosense: Open and big spectrum data," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 210–217, 2018. DOI: 10.1109/MCOM.2017.1700200

[5] S. Bayhan, A. Zubow, P. Gawłowicz, and A. Wolisz, "Smart contracts for spectrum sensing as a service," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 648–660, 2019. DOI: 10.1109/TCCN.2019.2936190

[6] A. Zubow, S. Bayhan, P. Gawłowicz, and F. Dressler, "Deeptxfinder: Multiple transmitter localization by deep learning in crowdsourced spectrum sensing," in *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, 2020. DOI: 10.1109/ICCCN49398.2020.9209727 pp. 1–8.

[7] C. Zhan, M. Ghaderibaneh, P. Sahu, and H. Gupta, "Deepmtl: Deep learning based multiple transmitter localization," in *2021 IEEE 22nd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, 2021. DOI: 10.1109/WoWMoM51794.2021.00017 pp. 41–50.

[8] M. Khaledi, M. Khaledi, S. Sarkar, S. Kasera, N. Patwari, K. Derr, and S. Ramirez, "Simultaneous power-based localization of transmitters for crowdsourced spectrum monitoring," in *Proceedings of the 23rd Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '17, 2017. DOI: 10.1145/3117811.3117845 p. 235–247.

[9] A. Bhattacharya, A. Chakraborty, S. R. Das, H. Gupta, and P. M. Djurić, "Spectrum patrolling with crowdsourced spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 1, pp. 271–281, 2020. DOI: 10.1109/TCCN.2019.2939793

[10] "Nsf workshop on spectrum measurements infrastructure," New York, NY, USA, Tech. Rep., 2016.

[11] A. Bhattacharya, A. Chakraborty, S. R. Das, H. Gupta, and P. M. Djurić, "Spectrum patrolling with crowdsourced spectrum sensors," *IEEE Transactions on Cognitive Communications and Networking*, pp. 1–1, 2019. DOI: 10.1109/TCCN.2019.2939793

[12] A. Bhattacharya, C. Zhan, A. Maji, H. Gupta, S. R. Das, and P. M. Djuricc, "Selection of sensors for efficient transmitter localization," *IEEE/ACM Transactions on Networking*, vol. Early Access, pp. 1–13, 2021. DOI: 10.1109/TNET.2021.3104000

[13] W. R. Thompson, "On the likelihood that one unknown probability exceeds another in view of the evidence of two samples," *Biometrika*, vol. 25, no. 3/4, pp. 285–294, 1933.

[14] K. Kandasamy, A. Krishnamurthy, J. Schneider, and B. Poczos, "Parallelised bayesian optimisation via thompson sampling," in *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Storkey and F. Perez-Cruz, Eds., vol. 84. Playa Blanca, Lanzarote, Canary Islands: PMLR, 09–11 Apr 2018, pp. 133–142.

[15] S.-F. Cheng, L.-C. Wang, C.-H. Hwang, J.-Y. Chen, and L.-Y. Cheng, "On-device cognitive spectrum allocation for coexisting urllc and embb users in 5g systems," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 171–183, 2021. DOI: 10.1109/TCCN.2020.3007890

[16] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of markov decision processes," *Math. Oper. Res.*, vol. 12, no. 3, p. 441–450, Aug. 1987.

[17] S. Agrawal and N. Goyal, "Analysis of thompson sampling for the multi-armed bandit problem," in *Proceedings of the 25th Annual Conference on Learning Theory*, vol. 23, 25–27 Jun 2012, pp. 39.1–39.26.

[18] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for gaussian process optimization in the bandit setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012. DOI: 10.1109/TIT.2011.2182033

[19] J. A. Magliacane, "Splat! a terrestrial rf path analysis application for linux/unix," Downloaded from https://www.qsl.net/kd2bd/splat.html, 2008.

[20] C. Zhan, H. Gupta, A. Bhattacharya, and M. Ghaderibaneh, "Efficient localization of multiple intruders in shared spectrum system," in *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, April 2020. DOI: 10.1109/IPSN48710.2020.00025 pp. 205–216.

[21] M. A. A. Careem, A. Dutta, and W. Wang, "Spectrum enforcement and localization using autonomous agents with cardinality," *IEEE Transactions on Cognitive Communications and Networking*, vol. 5, no. 3, pp. 702–715, 2019. DOI: 10.1109/TCCN.2019.2914915

[22] M. Li, D. Yang, J. Lin, M. Li, and J. Tang, "Specwatch: A framework for adversarial spectrum monitoring with unknown statistics," *Computer Networks*, vol. 143, pp. 176–190, 2018. DOI: 10.1016/j.comnet.2018.07.018

[23] A. Ghosh and A. Bhattacharya, "A gaussian process based technique of efficient sensor selection for transmitter localization," in *2021 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*. IEEE, 2021, pp. 1–1.

[24] J. Wang, P. Zhao, S. C. Hoi, and R. Jin, "Online feature selection and its applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 698–710, 2014. DOI: 10.1109/TKDE.2013.32

[25] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research*, vol. 9, 2008.

[26] Y. Chen, S. H. Hassani, A. Karbasi, and A. Krause, "Sequential information maximization: When is greedy near-optimal?" in *Proceedings of the Conference on Learning Theory (PMLR)*, 2015.

[27] S. E. Chick, J. Branke, and C. Schmidt, "Sequential sampling to myopically maximize the expected value of information," *INFORMS Journal on Computing*, vol. 22, no. 1, pp. 71–80, 2010.

[28] T. Lattimore and C. Szepesvári, *Bandit algorithms*. Cambridge University Press, 2020.

[29] A. Garivier and E. Moulines, "On upper-confidence bound policies for switching bandit problems," in *International Conference on Algorithmic Learning Theory*. Springer, 2011. DOI: 10.1007/978-3-642-24412-4_16 pp. 174–188.

[30] O. Chapelle and L. Li, "An empirical evaluation of thompson sampling," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011, pp. 2249–2257. [Online]. Available: http://papers.nips.cc/paper/4321-an-empirical-evaluation-of-thompson-sampling.pdf

[31] L. Tran-Thanh, A. Chapman, A. Rogers, and N. R. Jennings, "Knapsack based optimal policies for budget–limited multi–armed bandits," in *Twenty-Sixth AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, Sep 2012, pp. 1134–1140.

[32] Y. Xia, H. Li, T. Qin, N. Yu, and T.-Y. Liu, "Thompson sampling for budgeted multi-armed bandits," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.