Spectrum Patrolling with Crowdsourced Spectrum Sensors

Arani Bhattacharya, Ayon Chakraborty, Samir R. Das, Himanshu Gupta, Petar M. Djurić[†]
Stony Brook University
Email: {arbhattachar, aychakrabort, samir, hgupta}@cs.stonybrook.edu, [†]petar.djuric@stonybrook.edu

***Abstract*—We use a crowdsourcing approach for RF spectrum patrolling, where heterogeneous, low-cost spectrum sensors are deployed widely and are tasked with detecting unauthorized transmissions while consuming only a limited amount of resources. We pose this as a signal detection problem where the individual sensor's detection performance may vary widely based on their respective hardware or software configurations, but are hard to model using traditional approaches. Still an optimal subset of sensors and their configurations must be chosen to maximize the overall detection performance subject to given resource (cost) limitations. We present the challenges of this problem in crowdsourced settings and propose a set of methods to address them. These methods use data-driven approaches to model individual sensors and exploit mechanisms for sensor selection and fusion while accounting for their correlated nature. We present performance results using examples of commodity-based spectrum sensors and show significant improvements relative to baseline approaches.**

I. Introduction

With growing realization of mobile communication's impact on the nation's economic prosperity, RF spectrum has emerged as an important natural resource that is in limited supply [1]. While various spectrum sharing models are being developed to improve spectrum usage, 'spectrum patrolling' to detect unauthorized spectrum use is emerging as a critical technology [2]. Such unauthorized uses can take many forms, such as lower-tier devices accessing spectrum reserved for higher tier devices in a tiered spectrum sharing model [3], unauthorized devices accessing licensed spectra using software radios, or denials of service attacks. Techniques must be developed to detect such unauthorized accesses and large-scale spectrum monitoring is one effective way to do this.

However, large-scale spectrum monitoring using lab-grade spectrum analyzers is not scalable, given that such devices cost anywhere from several thousands to tens of thousands of US$ depending on the exact capability and require availability of AC power. Several recent papers have proposed to address this scalability issue by deploying low-cost, small form-factor, low-power spectrum sensors in large numbers perhaps using a crowdsourcing paradigm [4, 5, 6].[1] The overall monitoring performance achieved by a large number of such low-cost sensors can exceed that of a handful of lab-grade spectrum analyzers while costing several orders of magnitude less [4]. Due to this reason, there is a growing body of literature in studying the performance characteristics of commodity-based inexpensive sensors [8, 9, 6].

Although using inexpensive, commodity-grade sensors in large numbers may provide a very encouraging cost-performance tradeoff, use of a crowdsourcing paradigm brings in certain management problems. Spectrum patrolling must involve signal detection. It is unlikely that all deployed sensors will be used in specific detection tasks [4]. Only a subset will be typically employed ensuring that a required level of detection performance is achieved. This conserves the backhaul bandwidth and also energy when the sensors are battery operated (e.g., when mobile phones serve as spectrum sensors [8]). In case of multiple sensing needs in the same geographical space (e.g., detecting specific signals in multiple spectrum bands), sensors may need to be configured to engage in one specific task as their processing powers may not be sufficient for multiple concurrent signal detection tasks. The broad goal of this work is to *develop mechanisms to select the right set of sensors that optimizes the performance of detection task for a given cost.* There are two sub-problems that arise: 1) modeling individual sensor performance and cost for given configurations, 2) fusing data from multiple sensors and selecting the optimal subset to maximize detection performance subject to cost limitations (or, minimizing cost subject to a given detection performance). While these problems are not entirely new in a general sense, the specific nature of crowdsourced spectrum patrolling problem makes them challenging.

Challenge 1 – Modeling Individual Sensors: Fundamentally spectrum sensors must perform a signal detection task in form of a binary hypothesis testing (intruding transmitter present/absent). Detection performance is usually characterized by standard metrics like *probability of detection* (P_D) and *false alarm rate* (P_{FA}). Assigning a specific sensor to a specific sensing task and choosing specific configurations, requires accurate estimation of its P_D and P_{FA} metrics and cost for such configurations. Modeling of the cost depends on the scenario and can include, e.g., energy cost, backhaul data cost or any form incentives to be paid to the owner of the sensor. However, given the heterogeneity and diversity of spectrum sensors, in a crowdsensing paradigm estimating such metrics accurately is challenging. Existing literature

[1] There is at least one commercially successful crowdsourced application of spectrum sensing. FlightAware [7] deploys low-cost sensors via crowdsourcing to detect signals from aircrafts flying overhead.

extensively uses so-called first principles modeling approach that could miss various forms of imperfections (e.g., clock skew, I/Q imbalance, RF front end non-linearity) and noises common in commodity platforms. Even when they are able to account for those, they require knowledge of internal details of the sensor or separate calibration efforts. These are either not practical or do not scale well. More specifics of these issues are discussed in Section II.

Instead of relying on first principles models, we use a data-driven (blackbox) approach where models are created based on data from prolonged observation of the sensor. This type of approach is getting traction in other communities such as industrial process control where first-principles approaches are not practical for largely similar reasons (see, e.g., [10]). We abstract out the observable and easily quantifiable parameters of a sensor, its operating environment or runtime configuration. We use machine learning methods that treat the internal sensor hardware information (otherwise inaccessible) as hidden variables. This gives our methodology a direct and practical advantage over involved analytical models. Second, such models get richer with time and can easily accommodate new sensors without the need of explicitly calibrating them, an otherwise impossible task.

Challenge 2 – Sensor Selection and Fusion: Once individual sensors are modeled, we must select the subset of sensors (and their configurations if they are configurable) to achieve the best cost-performance tradeoff, i.e., the best detection performance for a given total cost (or minimum cost for a given desired performance). Here, the local sensor decisions (target present/absent) are to be combined into a global 'fused' decision. Thus, a fusion rule is needed. While there is a very rich literature on sensor fusion and developing optimal fusion rules, most techniques in the literature assume that *sensor decisions are conditionally independent.* This is not true for spectrum sensors, where their decisions could be correlated depending on the sensor locations. The reason is that sensors located in the same neighborhood are likely to face the same fading environment, resulting in correlations in their observations/decisions. The case for correlated observations have been indeed studied (see, e.g., [11, 12, 13]). But these methods are either too complex computationally to implement in practical systems and/or require prior knowledge of the correlation structure (e.g., in terms of higher-order moments of the sensor observations under each hypothesis [11] or spatial correlation coefficient [14]). Also, these techniques do not help addressing the sensor selection problem.

To handle this problem, we use a variant of sensor selection from machine learning literature called Maximum Relevance Minimum Redundancy (mRMR) [15]. This technique first assigns a value to each sensor by considering both its probability of detection, and its correlation with the other sensors. It uses an adaptive greedy selection where the value of each sensor is computed at each step, and then the sensor with the highest value is taken. While this does not guarantee an optimal subset, experiments on a large variety of datasets have shown that it works well in practice. Our evaluation shows that it works significantly better than a baseline technique that does not take correlation into account.

Contributions: Figure 1 pictorially describes the overall approach with pointers to various sections of the paper. Overall, we make two sets of contributions. *First,* we develop a systematic approach for data-driven models of spectrum sensors engaged in signal detection (Section III). The model takes the sensor's configuration and SNR as input and estimates detection performance and cost (we use energy to model cost in this work). We precede this modeling approach by highlighting limitations of traditional first-principles based analytical modeling approaches (Section II) and demonstrate improved model performance using the proposed data-driven approach using actual spectrum sensor hardware. *Second,* we develop a technique for the sensor selection and fusion problem taking into account the fact that the spectrum sensors are not conditionally independent (Section IV). The proposed feature selection based technique is suitable for crowdsourcing as it does not require information that is hard to obtain or estimate. We show that the overall detection performance improves significantly relative to baseline techniques.

An earlier version of this work was published in IEEE INFOCOM 2018 [16]. In contrast to this work, here we propose a more robust version of sensor selection, based on feature selection borrowed from the machine learning literature. The sensor selection algorithm in the previous version assumed that sensors were placed in a way that they could be partitioned into independent sets by clustering. Unlike the previous version, our sensor selection technique does not make any assumption about the distribution of the sensor locations. We have also included a more extensive evaluation of the performance of our selection and fusion algorithms in this version.

II. Modeling Detection Performance

The spectrum sensor detects the absence or presence of an intruding transmitter's signal. The corresponding hypotheses are denoted as H_0 (absence) and H_1 (presence) respectively. Raw sensed samples from the sensor are fed to the corresponding detection algorithm on board of the sensor that computes a *sensing metric*. The sensing metric is compared against a threshold (S_T) to output a binary decision. This is the local decision of the sensor.

Performance Metrics: Given H_1, the rate at which the sensor detects the transmitter is known as the *probability of detection* (P_D). Second, given H_0, the rate at which the sensor incorrectly flags the presence of a transmitter is known as the *probability of false alarm* (P_{FA}). Figure 2 demonstrates the basic working principle. The sensing metric has two different distributions under hypotheses H_0 and H_1. Under H_0, the distribution reflects noise. P_D and P_{FA} depends on the selection of S_T. Varying S_T varies both P_D and P_{FA} between 0 and 1. This produces the *receiver operating characteristics* (ROC) curve. Specifying P_{FA} (common case) also determines P_D as per the ROC curve. However, the ROC curve itself would look different if the distributions of the sensing metric shown in Figure 2(a) change. This is possible when the signal power from the transmitter changes (due to a different location, e.g.). We further discuss details of the distribution later in this section.

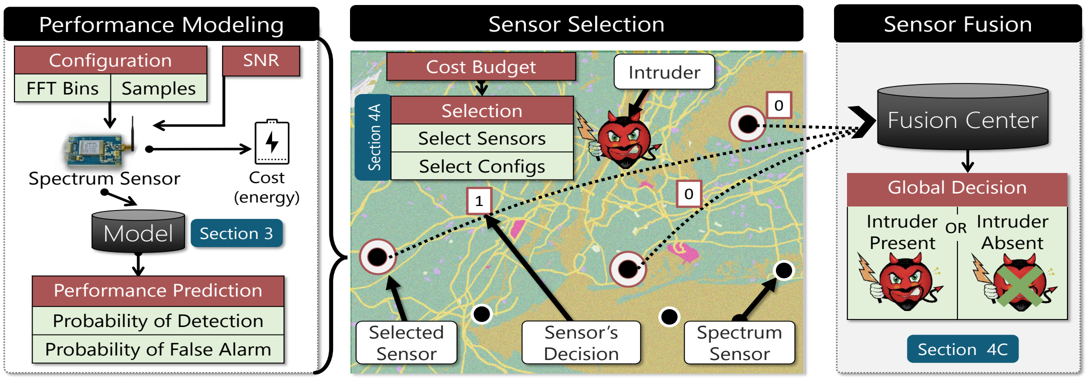
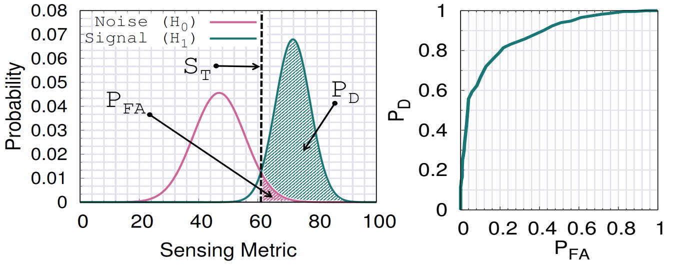


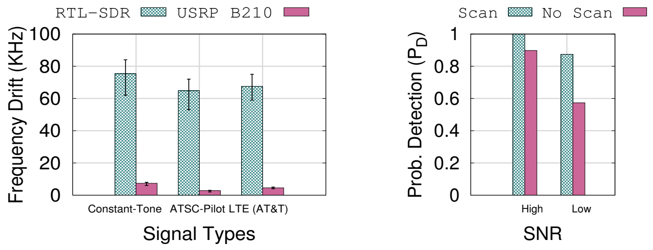
Figure 1: Overview of the proposed technique. Performance models of individual spectrum sensors are created first using a data-driven approach. Then sensor selection using a feature selection based approach. Finally individual sensor decisions are fused together to get a global decision. The figure indicates the different steps along with the section numbers where they are described.



(a) Detector Basics

(b) ROC Curve

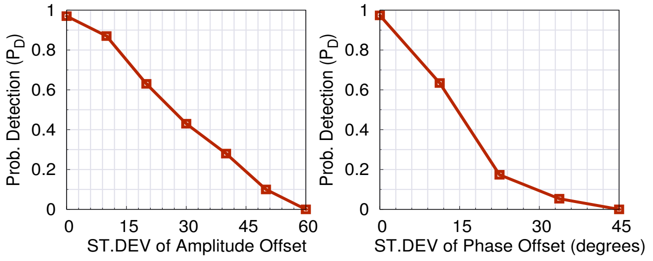
Figure 2: Working principle of a detector. S_T denotes the *threshold* of the sensing metric. Increasing S_T reduces P_D but also reduces P_{FA} as per the ROC curve.



(a) Higher clock-skew in cheaper devices

(b) Clock skew deteriorates P_D

Figure 3: Unpredictable clock skew makes frequency offset calculation harder, resulting in poorer signal detection performance.



(a) Effect of Amplitude Offset

(b) Effect of Phase Offset

Figure 4: We demonstrate the effect of I/Q imbalance in deteriorating the performance of simple waveform-based detector algorithm used in detecting an ATSC pilot tone.

Challenges: Estimating an optimal value of S_T is straightforward when the distributions of the sensing metric for H_0 or H_1 (Figure 2(a)) are known or can be accurately estimated. Unfortunately, this is not the case in practice. The distributions depend on a variety of factors including the detection algorithm, specifics of the sensor hardware, SNR or SINR at the sensor location, number of sensed samples, FFT resolution and so on. Common detection algorithms are energy-based, waveform or feature-based, autocorrelation or cyclostationary-based. Existing analytical techniques [17, 18, 19] can help model such algorithms to estimate an optimal S_T. However, such models typically result in significant estimation errors [17, 18]. The reasons are as follows. First, many of these models make idealistic assumptions about the distribution of the signal or noise or the noise associated with sensor hardware. For example, [20] shows that the performance of a sensor actually depends on both the signal parameters and the amount of RF front-end non-linearities of the sensors. Second, complex models do exist that take into account such factors [20, 21], but it is seldom possible to parameterize them correctly. This is due to the uncertainty in the hardware itself or inaccessible components that make reliable measurements impossible. Third, even when such measurements are possible manual calibration of individual sensors does not scale well, especially in the context of crowdsourcing.

We provide two sets of benchmarking experiments to highlight the challenges.

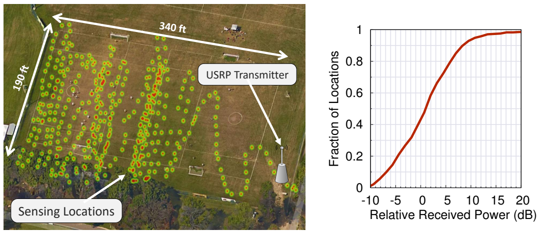
***Clock-skew*:** As an example, we study the clock skew associated with the local oscillator (LO) in the sensor. The frequency set in LO tunes the sensor to the desired frequency. However, the LO-frequency drifts giving rise to clock skew. To understand the nature of such drifts in commodity sensor hardware, we use two different spectrum sensors based on RTL-SDR and USRPB210. These sensors are chosen due to their low-power, small form factor nature [8]. They are both USB-powered and could be driven by an embedded CPU board or even a smartphone. Three test signals are used for detection. The first two are constant frequency tones in the 915 MHz band and the pilot tone of an ATSC signal (DTV band). In both cases we observe a non-trivial frequency drift that varies widely across individual sensor instances. For the third, we use an LTE downlink signal from a real network (AT&T) using these sensors and recorded the frequency correction needed in order to decode the synchronization signals. The results are summarized in Figure 3(a). In most cases RTL-SDR suffers from an appreciable clock skew which is less prevalent in more expensive hardware like USRP. In Figure 3(b) we show the impact of such clock-skew in detecting an ATSC signal. The ATSC signal has a pilot tone located at an offset of 310 KHz that is expected by our waveform based detector algorithm. We create two variations of the algorithm that expects the pilot tone (i) exactly at the 310 KHz offset and (ii) ≈100 KHz surrounding the expected location that it scans. In a low SNR scenario, scanning provides almost a 50% improvement in P_D compared to the detector that expects the pilot at a fixed offset demonstrating the impact of the clock skew problem.

***I/Q imbalance*:** Apart from clock skew, I/Q imbalance and RF front-end non linearities are other prominent issues. I/Q imbalance is introduced as a result of mismatch between the in-phase (I) and quadrature (Q) signal paths of the RF receive chain. For example, phase difference between the I and Q components is not always exactly 90° which results in an amplitude and phase offset in an I/Q sample. Since we do not have direct control over the radio circuitry, we simulate I/Q imbalance by adding amplitude and phase offsets to real I/Q traces obtained for an ATSC signal using a RTL-SDR device. For both cases, we use an offset drawn from a zero-mean Gaussian with a standard deviation as shown in Figure 4. We report the detection rate of the ATSC signal using a waveform-based detector that identifies the ATSC pilot signal. As the I/Q imbalance becomes more prominent, it becomes impossible to detect the signal. Although I/Q imbalance can be addressed directly in the hardware [21] we expect that crowdsourced spectrum sensors may use inexpensive hardware unable to do such corrections.

As mentioned earlier, while such problems can be accounted for by applying models that 'corrects' for such errors, these models are based on the 'first principles' approach. These models can only be applied after knowing specific sensor-specific parameters (e.g., characteristics of frequency drift, whether the algorithm scans, or nature of I/Q imbalance, etc). However, this information may not be available in a crowd-sourcing scenario given significant possible heterogeneity.

III. Data-Driven Performance Modeling

To address the problem of scalable modeling of heterogeneous sensors, we borrow from the concept of data-driven soft sensors utilized in industrial processes [10, 22]. Industrial processes find it impossible to use first principles models for their physical and chemical processes. These models are often idealized (e.g., they assume steady state behavior) or require parameters that are hard to obtain. Instead, data-driven soft sensors models are gaining ground that take an alternative blackbox approach where massive amount of collected data is used to model and predict the industrial process behavior in realistic conditions using statistical or machine learning techniques (see, e.g., [10, 22]).



(a) ≈ 1000 sensing locations (b) Received power CDF

Figure 5: Spectrum sensor data collection

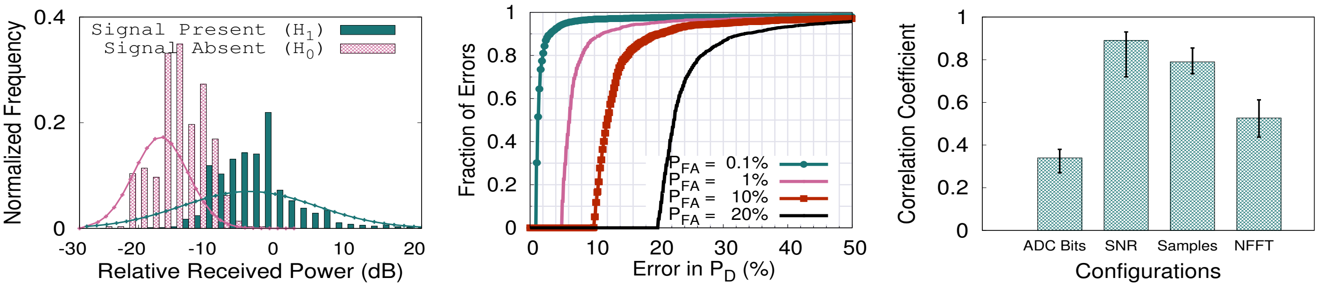
In the following we present our approach for the data-driven analysis using an example dataset. We first describe our dataset, quantify the errors associated with first-principles-based analytical models and then present our data-driven performance model of spectrum sensors.

A. Dataset

We collect spectrum sensor measurements in an outdoor setting within the university campus. As shown in Figure 5(a), we setup a USRP B210 based transmitter that transmits a constant tone in the 915 MHz band and collect sensing data (I/Q samples) using three RTL-SDR and two USRP B210 devices. We collect 1M samples at every location and our sensing area covers approximately 1000 locations within a 190×340 ft^2 region (Figure 5(a)). The distribution (under H_1) of the received power is also shown in Figure 5(b). We bias our data collection towards relatively lower SNR zones so as to have more variations in detection performance. We also note that detection is much easier if the received power is higher. Thus we only consider cases where the intruder uses relatively low power to avoid detection. Using the same set of sensors we also collect a *noise dataset* by turning off the transmitter. This data corresponds to the distribution for H_0.

For every location we employ three different detection algorithms (energy, feature and autocorrelation based) [8] both on the signal and the noise dataset. We vary two key parameters of the algorithm that directly influence $P_D - P_{FA}$ as well as energy cost in the sensor [8]: (i) N, number of sensed samples and (ii) $NFFT$, resolution of the FFT. N and $NFFT$ are varied from 32 (2^5) to 4096 (2^{12}) by repeated doubling with the constraint of $N \geq NFFT$ (36 configurations). We introduce heterogeneity in the resolution of sensed samples by changing the number of bits per sample. We produce additional data sets of 14, 12, 10 and 6 bit samples by ignoring the least significant bits from the collected 16 bit samples. Note that this depends on the resolution of the ADC in the sensor and heavily influences the dollar cost.

Across all locations, detection algorithms running with different configurations (≈650K in all) we obtain the sensing metrics for H_0 and H_1 respectively. For each location and for every possible configuration at that location, we repeat the detection experiment 1000 times by selecting a contiguous chunk of N samples from the respective 1 M samples starting at a random offset. This gives us 1000 instances of the sensing metrics under the same configuration and we compute P_D and P_{FA} for a given value of the sensing threshold, S_T. By varying S_T, we obtain the *ground truth* ROC curves for all such configurations across all locations.



(a) Empirical vs. Analytical Model for a fixed configuration

(b) Error in estimating P_D

(c) Factor analysis for the estimation error

Figure 6: Estimation errors associated with the analytical models and their dependency on the sensor's operating environment or configurations. The median estimation error in P_D can be as high as 25%. Higher errors are highly associated to low SNR operating environments.

B. Limitations of Analytical Models

Before directly delving into the internals of the data driven model, we first demonstrate the limitations of first-principles-based analytical models using our dataset. Due to space restriction we are not able to explain individual variations of the analytical models we use but will explain the general conclusions and trends. Figure 6(a) shows two histograms of the sensing metric corresponding to H_0 and H_1 obtained by using the energy-based detector algorithm (N = 2048, NFFT = 1024). We use the analytical model for energy-based detector to estimate the distributions for H_0 and H_1 for the same location. Figure 6(a) visually shows the difference between *ground truth* and the *estimated* distributions. In Figure 6(b), we present the estimation errors for different values of P_{FA}. Note that the median error can be as high as 25% in many cases. We observe that the distribution of signal and noise are close to each other in case of low SNR scenarios, leading to higher probabilities of error. We also show (Figure 6(c)) the correlation of such errors to the sensing configurations. Unlike other factors, the number of ADC-bits does not show a very high degree of correlation. This may be because we attempt to detect a simple tone at a constant power in this study.

C. Data-Driven Performance Model

Given the relatively poor performance of parametric models, we make use of 'training data' collected from spectrum sensors to take a non-parametric data-driven approach. Note that this data is already labeled as it has been obtained using our own sensors and transmitters at specific defined configurations. Essentially, the task of the model is to determine an optimal sensing threshold, S_T^{opt} that maximizes P_D for a given P_{FA}. For training the model we use feature vectors of the form V: <Algorithm, N, NFFT, B, SNR, P$_{FA}^{target}$>, where P_{FA}^{target} is the allowable false alarm rate. Algorithm refers to the signal detection algorithm the sensor runs that uses N, B-bit samples and involves an NFFT-bin FFT. We use energy, waveform and autocorrelation based detection algorithms. SNR refers to the signal-to-noise ratio of the signal at the sensor's location. Every vector V$_i$ is mapped to a corresponding $S_T^{opt_i}$ in the training examples. Note that we do not explicitly take into account internal hardware details unlike the involved analytical models [23, 24]. We explore off-the-shelf machine learning techniques using the scipy package [25] to learn the estimator for S_T^{opt}. Out of several popular techniques we tried out, the Support Vector Regressors (SVR) using RBF kernel works best in our case. We have also explored deep-learning methodologies [10] using convolutional neural networks (CNN); however, the amount of training data required to get reasonable estimation performance is significant. This makes CNN impractical in our case, and we adopt SVR for creating the performance model.

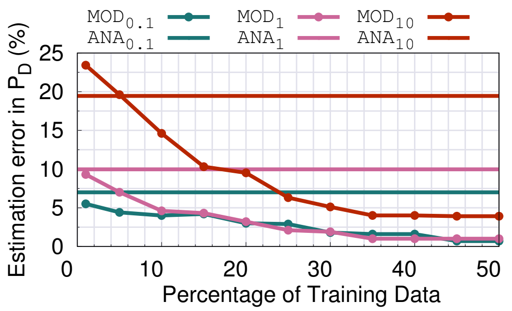


Figure 7: Performance of the data-driven models $MOD_{P_{FA}}$ compared to the analytical models $ANA_{P_{FA}}$ for different values of P_{FA}.

Validation: We validate the performance of our data-driven model in Figure 7. Given the configuration of the sensor and the SNR it operates in, our model predicts the optimal threshold S_T^{opt} that maximizes P_D for a given P_{FA}. We use the sensor traces and the model predicted $\widehat{S_T^{opt}}$ to compute $\widehat{P_D}$ for a given P_{FA}. The relative error of $\widehat{P_D}$ with respect to P_D is reported. We restrict our evaluation to sensor traces that has moderate to low SNR values as under such scenarios the models are error prone. We show estimation error in P_D for P_{FA} equal to 0.1%, 1% and 10% respectively. The data-driven models are indicated by $MOD_{P_{FA}}$ in Figure 7. We also present the estimation errors of the analytical models ($ANA_{P_{FA}}$) for the same set of data points (low/moderate SNRs). In all cases after our model is moderately trained, we reduce our estimation error by a significant margin with respect to the analytical models. For instance, MOD_{10} outperforms ANA_{10} by $\approx 12\%$ for a training set of size 20%. With more training samples the estimation error of our model becomes negligible and we see a clear improvement over the analytical performance models.

IV. Sensor Selection and Fusion

The approach described in the previous section gives us the power to estimate the detection performance of an individual sensor deployed in the wild without explicitly calibrating it. In this section we use such models to optimize the (network-wide or global) detection rate. This is done by selecting an optimal set of sensors (and their configurations such as number of ADC bits, number of samples or FFT bins etc.) and fusing their local decisions into a *network-wide (global)* decision. This needs a simultaneous solution of sensor selection and sensor fusion problems. As discussed in Section I, a wide body of literature exists that propose mathematical techniques to fuse sensor decisions to optimize certain detection performance metrics (typically Bayes risk). In a widely used method proposed by Chair and Varshney [26] that we will also use, an optimal fusion rule is developed to minimize the sum of false alarm and missed detection rates, but specifically for the case when the sensors are conditionally independent.

As explained in Section I, the conditional independence assumption does not hold for spectrum sensors. It is hard to account for correlated sensor observations with existing techniques due to complexity or unavailable parameters of the crowdsourced spectrum sensors. We develop an alternative feature selection-based approach below that we will demonstrate to perform well in practice.

A. Sensor Selection

To optimize performance under the constraint of a cost budget, we need to select a set of sensors $\mathbf{S}$ that collectively offers the best network-wide detection performance. Let $P_D(\mathbf{S})$ denote the probability that the set of sensors $\mathbf{S}$ detects an intruder. We denote the selection of a sensor by setting the decision variable $z_i = 1$, otherwise we set $z_i = 0$. Let C_i denote the cost of utilizing sensor S_i. Our objective is to maximize the probability of detection while keeping the cost within a fixed budget B:

$$\text{Maximize } P_D(\mathbf{S}) \text{ subject to: } \sum_{S_i \in \mathbf{S}} z_i C_i \leq B. \tag{1}$$

Sensor Ranking: Solving this optimization is a known NP-hard problem, since the sensors are correlated. This is mainly because quantifying the effect of the correlations on performance of the set of sensors is difficult. To solve this optimization problem, we utilize a variant of a commonly used feature selection technique from the machine learning literature, known as Maximum Relevance Minimum Redundancy (mRMR) [15]. In this technique, the sensors are ranked based on their contribution to $P_D(\mathbf{S})$. However, a measurement of the contribution of a single sensor needs to take into account two distinct factors:

- **Relevance:** A sensor is more relevant if its data is more frequently used to detect an intruder. Based on the feature selection literature, we measure the relevance of a sensor by looking at the mutual information between the sensor reading and the presence of intruder. Let X_i be a random variable denoting the local decision given by sensor S_i. Also, let U be a random variable denoting if an intruder is actually present at location j. Both X_i and U are binary random variables. Then, the relevance of S_i is measured by the mutual information between X_i and U, $I(X_i, U)$:

$$I(X_i, U) = \sum_{x_i \in \{0,1\}} \sum_{u \in \{0,1\}} P(X_i = x_i, U = u) \tag{2}$$

$$\log \frac{P(X_i = x_i, U = u)}{P(X_i = x_i) P(U = u)} \tag{3}$$

  The value of $I(X_i, U)$ is 1 if X_i and U are perfectly correlated, and 0 if they are completely independent. The probability terms can be estimated if we have sufficient data representative of the cases of intruders being present as well as absent. Thus, $I(X_i, U)$ is a measure of how relevant the sensor S_i is in detecting the presence of the intruder (denoted by U).
- **Redundancy:** Assuming we already have selected a set of sensors, we need a way to measure if adding a new sensor adds any new information. A common approach of measuring the redundancy of a sensor S_k with respect to a subset $\mathbf{T} \subseteq \mathbf{S}$ is to measure the amount of information given by the new sensor about the output of the subset:

$$R(S_k, \mathbf{T}) = \frac{1}{|\mathbf{T}|} \sum_{S_i \in \mathbf{T}} I(X_i, X_k) \tag{4}$$

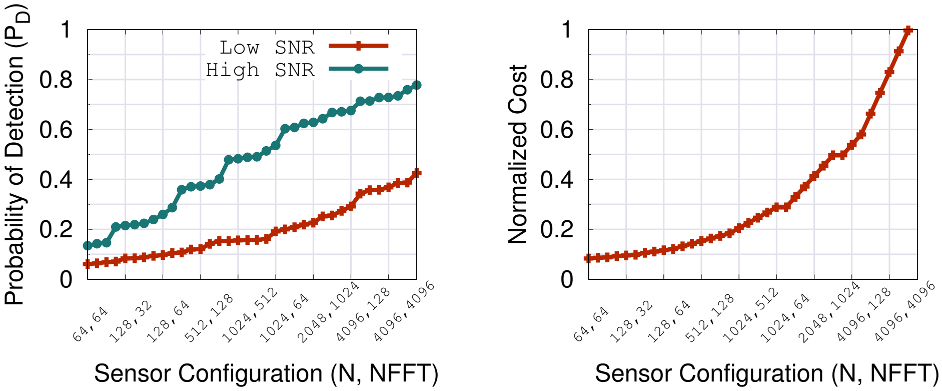
If two sensors S_i and S_k are spatially close to each other, then the mutual information of these two sensors will be high. Once again, it is possible to estimate the mutual information by looking at the outputs of each individual sensors for both H_0 as well as H_1. In this case, selecting both the sensors leads to a high degree of redundancy. Thus, there is little value in adding one such sensor to a subset containing the other.

To account for both relevance and redundancy, the actual value (denoted by $V(S_k, \mathbf{T})$) of adding a sensor S_k is the difference between the mutual information and redundancy. Mathematically, we write this as:

$$V(S_k, \mathbf{T}) = I(X_k, U) - R(S_k, \mathbf{T}). \tag{5}$$

Sensor selection schemes: For each sensor $S_i \in \mathbf{S}$ and a subset of the sensor set $\mathbf{T} \subseteq \mathbf{S}$, we now have a fixed value $V(S_i, \mathbf{T})$. We also have a fixed cost C_i for each sensor $S_i \in \mathbf{S}$. This is a feature selection problem with linear cost constraints, which is NP-hard in general. We first solve it in the simple case where the sensors are homogeneous in terms of configurations, and follow this up with heterogeneous configurations.

Homogeneous Sensors (HOMS): We assume all sensors are identical and have the same configuration. Hence their costs are equal and we assume unit cost for every sensor, i.e., $C_i = 1$. In this case we first iterate across all the sensors and select the sensor with the highest V_i. We add this sensor to the subset $\mathbf{T}$. In the next iteration, we recompute the values of V_i for all the remaining sensors, and again select the maximum. In this way, we keep selecting sensors until we reach the budget for the number of sensors allowed.



(a) Configuration vs. P_D for P_{FA} = 0.1%

(b) Configuration vs. Normalized Cost

Figure 8: (a) P_D for different configurations of the sensor under low and high SNR. (b) Sensor cost model (N is number of samples, NFFT is number of FFT bins).

Heterogeneous Sensors (HETS): In this case the sensors have heterogeneous configurations that are preconfigured for every sensor and cannot be changed. Accordingly, the sensor's cost C_i is a function of its configuration as demonstrated in Figure 8(b). Depending on the sensor's configuration, C_i can vary anywhere from the minimum cost value to 1. In this case, we pick the sensor having the highest value-to-cost ratio V_i/C_i, and add it to the subset $\mathbf{T}$. We recompute the values of V_i/C_i again, and keep picking the sensor with the highest value and adding it to $\mathbf{T}$ until again we exceed the budget. We summarize this algorithm in Algorithm 1.

Algorithm 1 HETS: Heterogeneous Sensor Selection.

```
Input: Value of sensors $V$, cost of sensors $C$, cost budget $B$
Output: Optimal selection $A$
$R = 0$ /* R stores the cost of sensors selected so far. */
$\mathbf{T} \leftarrow \phi$
while $R < B$ do
    $j \leftarrow \arg\max_{i=1}^{N} V_i/C_i$
    $\mathbf{T} \leftarrow \mathbf{T} \cup \{S_j\}$
    $R \leftarrow R + C_j$
    $V_j \leftarrow 0$
end while
return $\mathbf{T}$
```

Reconfigurable Sensors (RES): Here, the sensor can adopt a specific configuration from a pool of available configurations. Here the task is not only to select the sensors but also determine the configuration of the sensor that it should adopt. We again compute the value-to-cost ratios V_i/C_i for each configuration, and select the one that provides the highest. However, in the next iteration, we repeat the procedure after excluding the sensor that has already been selected in the previous step. We repeat this procedure until no sensor can be selected within the budgeted cost.

Analysis of our Technique: We note that our technique is a heuristic, i.e. it does not provide any guarantee of optimal or approximate performance. To understand this, we consider a case where there are three sensors S_1, S_2 and S_3 available for selection, with values V_1, V_2, V_3 and costs C_1, C_2 and C_3, respectively. We also have a budget of 2. Let the sensors S_1 and S_2 as well as S_2 and S_3 be strongly correlated with each other. Assume that V_2 is slightly greater than V_1 and V_3. In this case, it is obvious that selecting S_1 and S_3 is better. However, our algorithm first chooses S_2 and then any of S_1 or S_3. Since both S_1 and S_3 are strongly correlated with S_2, this can give a solution that is arbitrarily bad. Note that this can be extended to any number of sensors, in the special case where there are groups of 3 sensors with each of them having the same configuration. Thus, in the worst case, our algorithm can give a solution that is arbitrarily bad. However, we show in our evaluation that our heuristic performs well in a large number of different cases.

Time Complexity: To understand the time complexity of our technique, we note that selecting a single sensor requires iterating over all the sensors to compute each sensor's relevance. This requires $O(|\mathbf{S}|)$ time. It also requires iterating over all the selected sensors. Since the number of selected sensors is always less than the budget B, this requires B time. Thus, a single selection requires $O(|\mathbf{S}| \times B)$ time. This needs to run B times to fill the budget, and so the total time complexity of our technique is $O(|\mathbf{S}| \times B^2)$.

B. Sensor Fusion

We now have a selection of sensors and their configurations. We use the Chair-Varshney optimal sensor fusion rule [26] that fuses the local decisions of the individual sensors into a global (fused) decision to minimize the error rate. However, Chair-Varshney sensor fusion rule assumes that the sensor decisions are conditionally independent. This is not true in practice in our case, since the intruder can arrive at any location within the area, which affects the sensor local decisions.

To resolve this limitation, we apply this fusion rule repeatedly for each possible location of the intruder. We note that for a particular location of the intruder, the sensor local decisions are conditionally independent. Formally, assume that U_i is the local decision (1 or 0) of the sensor S_i, if the intruder signal is detected or not detected (respectively) by this sensor given the intruder is at location j. Using [26], we compute the fused decision $D_{L=j}$ of the sensors given this location of the intruder as:

$$D_{L=j} = \sum_{P_{Di,L=j} > P_{FAi}} [U_i \log \frac{P_{Di,L=j}}{P_{FAi}} + (1 - U_i) \log \frac{1 - P_{Di,L=j}}{1 - P_{FAi}}] \quad (6)$$

The summation above is for all selected sensors. $P_{Di,L=j}$ is the probability of detection of sensor S_i for an intruder at location j. $U_{L=j} > 0$ indicates presence of the intruder (at location j), otherwise it is considered absent. Note that we only consider sensors with probability of detection higher than the probability of false alarm from a particular location l, since only those sensors are close enough to give meaningful information. To estimate the presence of an intruder anywhere, we first compute the values of $D_{L=j}$ for all possible locations j. We conclude that there is an intruder anywhere only if at least one of these $D_{L=j}$'s is positive. Otherwise, we conclude that no intruder is present.

V. Evaluation

We simulate a $1000\,\text{m} \times 1000\,\text{m}$ grid where we randomly deploy 100 spectrum sensors. The sensors can choose among 36 different configurations. Each configuration corresponds to the tuple (N, NFFT), N being the number of I/Q samples and NFFT, the resolution of the FFT in the sensor's detection algorithm. N, NFFT $\in \{2^5, 2^6, \cdots, 2^{12}\}$, where N $\geq$ NFFT. For each sensor, we set $P_{FA} = 1\%$ (or 0.01) and obtain the P_D from our data-driven performance model (MOD_{10}). The sensors have a cost model as mentioned in Figure 8. Next, we simulate an intruder in the grid. The intruder is represented by a wireless transmitter with a transmit power of 10 dB. We use the log-normal model to compute RSS at all the sensor locations. We make the intruder's *prior map* realistic to account for different factors such as terrain information or proximity to residential or navigable areas. We create the *prior map* directly from a snapshot of Google map's satellite imagery data. To remove intricate details (e.g., buildings, texture) in the image, we apply Gaussian blur, a well known image filtering technique. Next we resize the image to a dimension of 100×100 to emulate our grid. We make the prior probability of the transmitter to be present in a certain cell $< i, j >$ proportional to the pixel intensity at $< i, j >$. Figure 9 shows our prior map. For all simulations we sample the intruder's location 10 K times from the *prior map* that we use to obtain weights for our sensor selection algorithms. Every time the intruder appears, the selected sensors attempt to determine its presence with their respective values of P_D. The fused decision is compared to the ground truth. We compute the detection rate for the given instance of selected sensor by simulating the intruder 1000 times. We also compute the false alarm rate by simulating another 1000 cases where no intruder is present.

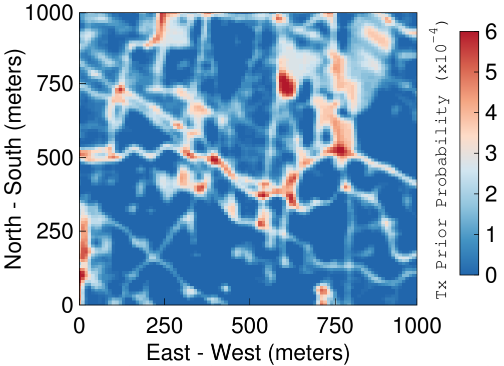
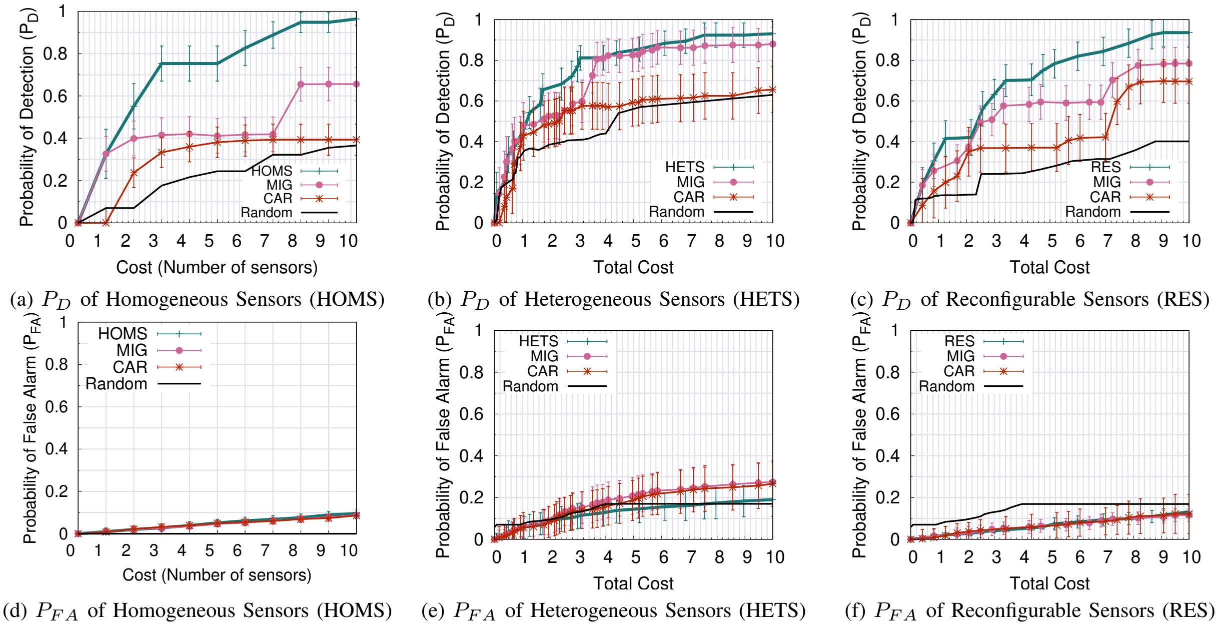


Figure 9: Transmitter prior map of the area obtained from a satellite image.

A. Performance of Sensor Selection Algorithm

We compare the performance of our sensor selection algorithms with that of two baseline algorithms. As baseline, we first run a *random* selection algorithm where we pick the sensors randomly with uniform probability. We then run a greedy algorithm where we pick the best sensors (the ones with highest relevance) without accounting for their correlation. We refer this algorithm as **m**utual **i**nformation based **G**reedy (MIG). When the sensors are homogeneous, *MIG* selects the sensors for which the prior probabilities are the highest. For other cases, *MIG* selects sensors in decreasing order of their V_i/C_i ratios. Finally, we also run the sensor selection algorithm proposed in our earlier work [16], which first segments the entire grid into clusters, and then uses ranking of sensors across each cluster. We refer this technique as **C**lustering **a**nd **R**anking (CAR).

Observation: Figure 10 shows the performance in terms of P_D and P_{FA} obtained by the sensors selected by our algorithms compared to baseline heuristics across different cost budgets. We show both the mean performance and the standard deviation at each of the data points. For HOMS, we consider the number of sensors as the cost, i.e., $C_i = 1$. However for HETS and RES, the cost $C_i \in [min_{cost}, 1]$. We note that our algorithms perform significantly better compared to *MIG*, *CAR* as well as *Random* schemes, especially at medium values of the budget. For all cases, till a budget of 1, our algorithms perform similar to the *MIG* scheme. This is because both of them select sensors only from the cluster with high prior probability. When we increase the budget above 2, the *MIG* method keeps selecting from the same cluster, since it does not consider the effect of correlation. For instance, at a budget of 5, 3 and 4, HOMS, HETS and RES outperform the *MIG* scheme by 91%, 10% and 15%, respectively. Note that our algorithm also performs much better than the random selection in each of the cases. The lower gain in the case of HETS can be explained by observing that a larger number of lower cost sensors provides higher probability of detection than a fewer number of expensive sensors. For less expensive sensor configurations, the amount of correlation is also lower, since their individual P_D's fall more sharply with a reduction in power. Thus, our algorithms, because of removal of redundancy, improves performance the most when the budget constraint requires intelligent selection of sensors.



(a) P_D of Homogeneous Sensors (HOMS) (b) P_D of Heterogeneous Sensors (HETS) (c) P_D of Reconfigurable Sensors (RES)

(d) P_{FA} of Homogeneous Sensors (HOMS) (e) P_{FA} of Heterogeneous Sensors (HETS) (f) P_{FA} of Reconfigurable Sensors (RES)

Figure 10: Performance comparison of the three proposed schemes with *MIG*, *CAR* and *Random* baseline heuristics. For each data point, we show the mean value and the standard deviation. We do not show the standard deviation of the *Random* scheme for clarity since it has a higher value.

We also note that the increase in the values of P_D also leads to increase in P_{FA}. However, this increase in the value of P_{FA} is relatively small, as it is always less than 0.1 in case of HOMS and RES, and less than 0.2 in case of HETS. Our algorithm also provides either lower or equal values of P_{FA} compared to each of the baseline techniques.

B. Performance of Our Fusion Rule

We compare the performance of the Chair Varshney fusion rule with a baseline technique. To compare, we run the same simulation and selection process as HETS, but run both Chair Varshney fusion rule and a baseline technique. Our baseline technique concludes that there is an intruder if a total of k out of $|\mathbf{S}|$ sensors give output 1, where the best value of k is chosen by simulation.

Observation: Figure 11 shows the probability of detection using Chair Varshney and the baseline technique. We find that Chair Varshney performs better in all the cases, with the performance rising with increase in number of sensors. Thus, the Chair Varshney rule is 91.2% accurate when just 8 sensors are present, whereas using k out of N sensors just gives 65.8% accuracy. This is because Chair Varshney is able to consider the individual performance of each of the sensors, whereas the baseline technique always considers all sensors as equivalent. The contributions of the individual sensors need to be considered for good detection performance. This further confirms our claim that the Chair-Varshney rule is optimal. This also shows that having information about probability of detection of individual sensors is important for accurate sensor fusion. Thus, our data-driven technique of evaluating sensors behavior is necessary to improve the accuracy of detection.

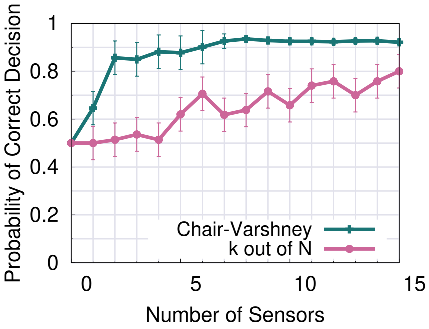


Figure 11: Performance of Chair-Varshney Fusion rule (our method) compared to a baseline k out of N rule.

VI. Related Work

Shared spectrum architectures need to enforce suitable policies to control spectrum access among secondaries [27, 28]. On the other hand, with the advent of cheaper radio hardware the licensed spectrum is prone to unauthorized use [29]. This makes the problem of spectrum patrolling important. Dutta and Chiang [2] introduce the concept of crowdsourced enforcement of spectrum policies. Vaze and Murthy [30] also localize transmitters using binary sensors similar to our study. However, unlike our work, they do not consider the effect of correlation among sensors and do not consider the cost of utilizing sensors. Bhattacharya et al. [31] propose reducing the cost of spectrum sensing by using FPGA-based sensors.

Performance of low cost spectrum sensors: The authors in [2] assume complete knowledge about the performance of crowdsourced sensors which is not practical. [2] also assumes the sensors to be homogeneous which is generally not true in a crowdsourced environment. Spectrum monitoring using cheap crowdsourced sensors is not new [4, 9, 6] but they do not provide any insights regarding performance or reliability of sensing. We also show that analytical techniques [32] that model the sensor's detection performance are often simplistic and error prone. [21, 23] build upon the analytical techniques providing corrections for hardware related aspects like I/Q imbalance, RF front-end non-linearities etc. Inspired by [22, 10], we use a data-driven approach to create performance models of heterogeneous spectrum sensors.

Sensor Selection and Fusion: A good amount of literature exists that study the problem of selecting sensors and combining the decisions of multiple sensors. Joshi and Boyd [33] show a method of selecting sensors using convex optimization, and empirically show that their results are usually close to optimal. Shamaiah et al. [34] propose a greedy selection of sensors that is close to optimal. Unlike our work, these studies consider data that follow normal distribution. To select sensors in the presence of intruders, we utilize a feature selection technique commonly used in the machine learning literature. This technique, known as maximum relevance minimum redundancy (MRMR) [15], is widely used to select relevant features when the features are correlated.

Combining the data of multiple sensors is a well-known problem in sensor networks. We utilize the rule provided by Chair and Varshney [26] which optimizes the overall performance when the individual sensor outputs are conditionally independent of one another. Different techniques of fusing multiple sensor decisions are presented in [35]. Some studies have also looked at the problem of distributed spectrum monitoring. Ghasemi and Sousa [3] propose using collaborative sensing across multiple sensors to better monitor spectrum. Dasari et al. [36] showed that detection of intermittent transmitters can be significantly improved by fusing the decisions of multiple sensors. Our work builds upon these studies to focus on detecting the presence of spectrum intruders.

VII. Conclusion

In this work we address the problem of spectrum patrolling using crowsourced heterogeneous sensors. To the best of our knowledge this is the first work that models the performance of a spectrum sensor in a data-driven way. Our model provides significant improvement over state-of-the-art 'whitebox' models. Next we address the problem of sensor selection and fusion of heterogeneous sensors deployed over a region of interest to improve intrusion detection performance within a cost budget. We investigate different scenarios of homogeneous, heterogeneous and reconfigurable sensors. Our sensor selection algorithms perform significantly better than reasonable baseline heuristics. We highlight challenges of the patrolling problem in a cost-effective fashion using crowdsourced sensors and develop mechanisms to address them.

Acknowledgement

This work is partially supported by NSF grants AST-1443951, CNS-1642965 and by the MSIT, Korea, under the ICT Consilience Creative Program (IITP-2017-R0346-16-1007). The authors would like to acknowledge the help of Jian Xu and Professor Aruna Balasubramanian for access to their servers. We are also grateful to the anonymous reviewers for their constructive feedback.

References

[1] S. Huang, X. Liu, and Z. Ding, "Opportunistic spectrum access in cognitive radio networks," in *Proc. IEEE INFOCOM*, 2008.

[2] A. Dutta and M. Chiang, ""See Something, Say Something" crowdsourced enforcement of spectrum policies," *IEEE Trans. on Wireless Communications*, vol. 15, no. 1, pp. 67–80, 2016.

[3] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *Proc. IEEE DySPAN*, 2005.

[4] A. Chakraborty, M. S. Rahman, H. Gupta, and S. R. Das, "Specsense: Crowdsensing for efficient querying of spectrum occupancy," in *Proc. IEEE INFOCOM*, 2017.

[5] T. Zhang, N. Leng, and S. Banerjee, "A vehicle-based measurement framework for enhancing whitespace spectrum databases," in *Proc. ACM Mobicom*, 2014.

[6] R. Calvo-Palomino, D. Giustiniano, V. Lenders, and A. Fakhreddine, "Crowdsourcing spectrum data decoding," 2017.

[7] "FlightFeeder for Android, FlightAware." http://flightaware.com/adsb/ android/.

[8] A. Chakraborty, U. Gupta, and S. R. Das, "Benchmarking resource usage for spectrum sensing on commodity mobile devices," in *Proc. ACM HotWireless*, 2016.

[9] A. Nika, Z. Li, Y. Zhu, Y. Zhu, B. Y. Zhao, X. Zhou, and H. Zheng, "Empirical validation of commodity spectrum monitoring," in *Proc. ACM SenSys*, 2016.

[10] W. Yan, D. Tang, and Y. Lin, "A data-driven soft sensor modeling method based on deep learning and its application," *IEEE Trans. on Industrial Electronics*, 2017.

[11] M. Kam, Q. Zhu, and W. S. Gray, "Optimal data fusion of correlated local decisions in multiple sensor detection systems," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 28, no. 3, pp. 916–920, 1992.

[12] E. Drakopoulos and C.-C. Lee, "Optimum multisensor fusion of correlated local decisions," *IEEE Trans. on Aerospace and Electronic Systems*, vol. 27, no. 4, pp. 593–606, 1991.

[13] J. Unnikrishnan and V. V. Veeravalli, "Cooperative sensing for primary detection in cognitive radio," *IEEE Journal of selected topics in signal processing*, vol. 2, no. 1, pp. 18–27, 2008.

[14] A. S. Cacciapuoti, I. F. Akyildiz, and L. Paura, "Correlation-aware user selection for cooperative spectrum sensing in cognitive radio ad hoc networks," *IEEE JSAC*, vol. 30, no. 2, pp. 297–306, 2012.

[15] H. Peng, F. Long, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1226–1238, 2005.

[16] A. Chakraborty, A. Bhattacharya, S. Kamal, S. R. Das, H. Gupta, and P. M. Djuric, "Spectrum patrolling with crowdsourced spectrum sensors," in *Proc. IEEE INFOCOM*, 2018.

[17] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, pp. 523–531, 1967.

[18] F. F. Digham, M.-S. Alouini, and M. K. Simon, "On the energy detection of unknown signals over fading channels," *IEEE Trans. on Communications*, pp. 21–24, 2007.

[19] J. Chen, A. Gibson, and J. Zafar, "Cyclostationary spectrum detection in cognitive radios," in *Cognitive Radio and Software Defined Radios: Technologies and Techniques, 2008 IET Seminar on*. IET, 2008.

[20] A.-A. A. Boulogeorgos, H. A. B. Salameh, and G. K. Karagiannidis, "Spectrum sensing in full–duplex cognitive radio networks under hardware imperfections," *IEEE Trans. on Vehicular Technology*, vol. 66, no. 3, pp. 2072–2084, 2017.

[21] A. Gokceoglu, S. Dikmese, M. Valkama, and M. Renfors, "Enhanced energy detection for multi-band spectrum sensing under rf imperfections," in *Proc. IEEE CROWNCOM*, 2013.

[22] C. Shang, F. Yang, D. Huang, and W. Lyu, "Data-driven soft sensor development based on deep learning technique," *Journal of Process Control*, vol. 24, no. 3, 2014.

[23] A.-A. Boulogeorgos, H. B. Salameh, and G. Karagiannidis,

"Spectrum sensing in full–duplex cognitive radio networks under hardware imperfections," *IEEE Transactions on Vehicular Technology*, 2016.

[24] A. Gokceoglu, S. Dikmese, M. Valkama, and M. Renfors, "Energy detection under IQ imbalance with single-and multi-channel direct-conversion receiver: Analysis and mitigation," *IEEE JSAC*, 2014.

[25] E. Jones, T. Oliphant, P. Peterson *et al.*, "SciPy: Open source scientific tools for Python," 2001–, [Online; accessed Aug 29, 2019]. [Online]. Available: http://www.scipy.org/

[26] Z. Chair and P. Varshney, "Optimal data fusion in multiple sensor detection systems," *IEEE Trans. on Aerospace and Electronic Systems*, 1986.

[27] J.-M. Park, J. H. Reed, A. Beex, T. C. Clancy, V. Kumar, and B. Bahrak, "Security and enforcement in spectrum sharing," *Proc. IEEE*, vol. 102, no. 3, pp. 270–281, 2014.

[28] X. Jin, J. Sun, R. Zhang, Y. Zhang, and C. Zhang, "Specguard: Spectrum misuse detection in dynamic spectrum access systems," in *Proc. IEEE INFOCOM*, 2015.

[29] "FCC Proposes Levying Huge Fine on New York Police Radio Jammer, ARRL news," http://bit.ly/2uPsB1P.

[30] R. Vaze and C. R. Murthy, "Multiple transmitter localization and whitespace identification using randomly deployed binary sensors," *IEEE Transactions on Cognitive Communications and Networking*, vol. 2, no. 4, pp. 358–369, 2016.

[31] A. Bhattacharya, H. Chen, P. Milder, and S. R. Das, "Quantifying energy and latency improvements of fpga-based sensors for low-cost spectrum monitoring," in *2018 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, Oct 2018, pp. 1–5.

[32] R. Tandra and A. Sahai, "SNR walls for signal detection," *IEEE Journal of Selected Topics in Signal Processing*, pp. 4–17, 2008.

[33] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.

[34] M. Shamaiah, S. Banerjee, and H. Vikalo, "Greedy sensor selection: Leveraging submodularity," in *Decision and Control (CDC), 2010 49th IEEE Conference on*. IEEE, 2010, pp. 2572–2577.

[35] B. Ao, Y. Wang, L. Yu, R. R. Brooks, and S. Iyengar, "On precision bound of distributed fault-tolerant sensor fusion algorithms," *ACM Computing Surveys (CSUR)*, p. 5, 2016.

[36] M. Dasari, M. B. Atique, A. Bhattacharya, and S. R. Das, "Spectrum protection from micro-transmissions using distributed spectrum patrolling," in *International Conference on Passive and Active Network Measurement*. Springer, 2019, pp. 244–257.

Arani Bhattacharya received his B.Tech and M.Tech degrees in Computer Science from West Bengal University of Technology and Indian Statistical Institute in 2011 and 2013 respectively. Since 2013, he has been pursuing his PhD in Computer Science at Stony Brook University.

Arani's general research interest is in wireless networks and distributed computing. He is passionate about integrating both model and data-driven techniques into working systems and studying their performance tradeoffs. His current focus includes exploring techniques of efficient monitoring of radio frequency spectrum, and improving scalability of IoT applications.

Ayon Chakraborty received his B.E. degree in computer science from Jadavpur University, India in 2011. After finishing his PhD in computer science from Stony Brook University, NY, USA in 2017, he joined the Mobile Communications and Networking group in NEC Laboratories America as a researcher. He is broadly interested in designing mobile systems that interact with and interpret the physical world, spanning both algorithm design as well as system prototyping.

Samir R. Das (S'90–M'94) received the PhD degree in computer science from the Georgia Institute of Technology, Atlanta, GA, USA, in 1994. Since 2002, he has been with Stony Brook University, Stony Brook, NY, USA, where he is a Professor with the Department of Computer Science and currently serves as the Department Chair. His current research interests include mobile/wireless networking protocols, systems, and performance evaluation. Samir Das was a recipient of the U.S. National Science Foundations CAREER Award in 1998. He has been on the Editorial Board of several major journals, including the IEEE Trans. on Networking and the IEEE Trans. on Mobile Computing. He also co-chaired premier conferences, such as ACM MobiCom and ACM MobiHoc. He routinely serves on the Program Committees of top networking and mobile computing conferences.

Himanshu Gupta received the B.Tech. degree in computer science and engineering from the Indian Institute of Technology (IIT) Bombay, Mumbai, India, in 1992, and the M.S. and Ph.D. degrees in computer science from Stanford University, Stanford, CA, USA, in 1999. He is currently an Associate Professor with the Department of Computer Science, Stony Brook University, Stony Brook, NY, USA. His recent research activities focus on theoretical issues in wireless networking, sensor networks, and database systems.

Petar M. Djurić (M'90–SM'99–F'06) received the B.S. and M.S. degrees in electrical engineering from the University of Belgrade, Belgrade, Yugoslavia, respectively, and the Ph.D. degree in electrical engineering from the University of Rhode Island, Kingston, RI, USA. He is a SUNY Distinguished Professor and currently a Chair of the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, USA. His research has been in the area of signal and information processing with primary interests in the theory of machine learning; Monte Carlo-based methods; signal and information processing over networks; signal modeling, detection, and estimation; RFID and the IoT. He has been invited to lecture at many universities in the United States and overseas. Prof. Djuri was a recipient of the IEEE Signal Processing Magazine Best Paper Award in 2007 and the EURASIP Technical Achievement Award in 2012. In 2008, he was the Chair of Excellence of Universidad Carlos III de Madrid-Banco de Santander. From 2008 to 2009, he was a Distinguished Lecturer of the IEEE Signal Processing Society. He has been on numerous committees of the IEEE Signal Processing Society and of many professional conferences and workshops. He was the first Editor-in-Chief of the IEEE TRANSACTIONS ON SIGNAL AND INFORMATION PROCESSING OVER NETWORKS. Prof. Djurić is also a Fellow of EURASIP.