This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

Cooperatively Learning Footprints of Multiple Incumbent Transmitters by Using Cognitive Radio Networks

Mihir Laghate and Danijela Cabric
Electrical Engineering Department, University of California, Los Angeles
Emails: mvlaghate@ucla.edu and danijela@ee.ucla.edu

***Abstract*—Energy measurements have conventionally been used for detecting the presence of signal energy but not for distinguishing incumbent users (IUs). In this work, soft and hard reports Gaussian mixture model learning algorithms are proposed to distinguish intermittently transmitting IUs and to find the footprints for each IU, i.e., the CRs that receive signals from each IU. Unlike existing methods for distinguishing IUs using single antenna CRs, the proposed methods do not require CR locations, channel models, or prior knowledge of the number of IUs or their protocol. The soft reports algorithm uses Mahalanobis distance to separate components while a two stage process learns components corresponding to individual IUs. The hard reports algorithm reduces computational complexity by learning unidimensional mixtures at each CR and fusing results using a novel algorithm for finding the maximum weight dominating set in a directed graph. MATLAB simulations of slotted ALOHA IU networks are used to evaluate the algorithms' performance as the number of CRs, IUs, and average activity are varied. In frequent collision scenarios, the soft reports algorithm has the best performance. However, NS3 simulations of 802.11n IU networks show that methods proposed to reduce false positives deteriorate detection performance due to channel capture effects.**

***Index Terms*—Spectrum sensing, Gaussian mixture model, footprints, maximum weight dominating set.**

I. Introduction

The paradigm of opportunistic spectrum usage has been motivated by the fact that existing or incumbent users (IUs) are actively using spectrum for a small fraction of time at a limited number of locations. For efficient dynamic spectrum access, cognitive radio (CR) networks analyze the spatial spectrum occupancy using various tools such as radio environment maps [1] and localization algorithms [2]. However, existing algorithms for these tasks are unable to disambiguate multiple IUs. Note that these algorithms can be extended to analyze the spatial spectrum occupancy of multiple coexisting IUs if they are input transmissions of just one IU at a time. Such an extension would be an example application of our algorithms.

Spectrum sensing literature has conventionally focused on CR networks where all CRs lie in the footprints of all the incumbents that it shares the spectrum with [3]. However, due to limited transmit powers, all cognitive radios (CRs) of a CR network may not be able to receive signals from all IUs. This motivates us to define the *footprint* of an IU as the set of CRs that can detect its transmissions. Extending the footprint from a set of CRs to a geographical area (where the IU's transmissions can be detected) has been studied extensively in radio environment map literature [1] and we consider it to be outside the scope of this paper. We focus on the identification of the footprints. Identifying the footprint of the IU helps reduce the dimensionality of the measurements used for the above mentioned algorithms and allows us to analyze two (or more) IUs with non-overlapping footprints at the same time.

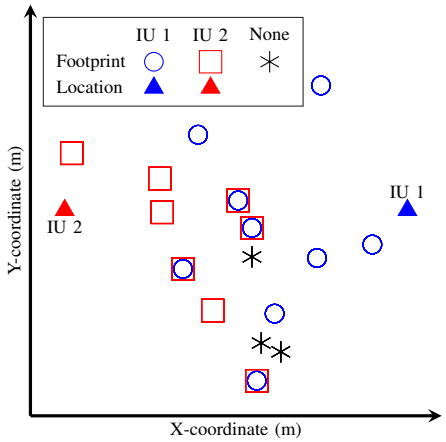


Fig. 1. Locations of 2 IUs and 16 CRs. Footprints inferred for a single realization of shadow fading channels correlated across space.

Apart from enabling the disambiguation of multiple IUs for sensing related tasks, the footprints of multiple IUs and the ability to identify the transmitting IU is assumed known in location aware routing literature [4] and for learning network topology [5].

A. Related Work

Existing work on disambiguating incumbent transmitters can be classified into three classes: 1) algorithms that distinguish between incumbents using prior knowledge and training, 2) algorithms that distinguish incumbents using orthogonality in one or more dimension, and 3) algorithms that distinguish incumbents without prior knowledge, extra hardware, or orthogonality in any dimension. The algorithms we propose belong to the last class of work.

The first class of algorithms assume that the identity of the incumbents are known in the form of transmitter identifiers, such as WiFi BSSID [6], or incumbent location, transmit power, and channel models [7]. Learning transmitter identifiers requires a significant amount of training. Identifying the current transmitter using CR and IU locations requires IUs to have isotropic radiation patterns and high accuracy prior information which may not be available [8]. More recently,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

[9] proposes a method that uses prior knowledge of channels between every pair of points on a given location grid and location tagged RSS measurements to estimate the number and locations of incumbents.

The second class of existing methods place restrictions on the incumbents in order to obtain orthogonality in one or more test dimension. Boundary estimation techniques from sensor network literature [10] and image processing techniques [11] have been extended to identify incumbent user locations and footprints when their footprints do not overlap. If the incumbents do have overlapping footprints, then they can be distinguished if they occupy non-overlapping frequency bands [12], [13]. If the incumbents also occupy overlapping frequency bands, then they can be distinguished and localized if they have distinct cyclostationary peaks [14]. Lastly, if the CRs have multiple antennas and a line-of-sight to the incumbent transmitters, then [15] proposes a method that uses direction of arrival to distinguish and localize them.

The last class does not assume any prior information about the incumbents and does not place restrictions on the incumbents' hardwares, protocols, or channels to CRs. [16] shows that the energy received at the CRs from multiple IUs can be modeled by a Gaussian mixture model (GMM) such that the each component of the GMM is the result of the transmissions of a subset of IUs transmitting simultaneously. The authors propose a method to learn this GMM and use energy detection on the learnt components' means to identify the footprints of each IU. However, the algorithm proposed in [16] does not learn the correlation between energy received at different CRs and, hence, overestimates the number of incumbents.

B. Challenges and Contributions

Our goal in this paper is to estimate the footprints for all IUs and identify the transmitter(s) of each detected transmission. We do not assume to know the number of IUs, locations of the CRs and the IUs, the channel propagation model, or features of the waveform transmitted by the incumbents. The lack of information makes our algorithms versatile.

Similar to the work in [16], our algorithms use multiple energy measurements over time to differentiate IUs using the fact that radios in communication systems transmit intermittently and not continuously. Hence, there will be times when exactly one IU is transmitting at one time. This is known as the separability assumption [17].

The first algorithm we propose fuses measurements of received energy values at the CRs, i.e., soft reports. The second fuses the CRs' identified transmitters for each detected transmission, i.e., hard reports.

Our soft reports algorithm is an online algorithm to learn a Gaussian mixture model of the received energy without knowing the number of components or the component parameters. The footprints of the IUs are estimated by energy detection on the sample means of particular components and the current transmitter(s) is (are) identified by the component that is most likely to have generated the received energy measurement. It improves on the algorithm proposed in [16] by estimating the off-diagonal entries of the covariance of the mixture components and introducing the concept of a candidate component. Compared to state of the art Gaussian mixture model learning algorithms, it significantly reduces computational complexity by recognizing that the means of a large fraction of the components are linear combinations of the means of the remaining mixture components. Unlike state of the art algorithms, it is not a recursive algorithm and does not require storage of received samples. Existing methods of learning mixture models are reviewed in detail in Section V.

For the hard reports algorithm, each CR uses a unidimensional Gaussian mixture learning algorithm to identify the IUs it receives. CRs report the transmitters for multiple measurements. The hard reports algorithm fuses these transmitter labels to identify the footprint of each IU. The challenge here is that since the CRs are not cooperating with each other, CRs may have different identifiers for the same transmitter. Furthermore, each CR is more likely to make a mistake in the identification because of the lack of diversity. The hard reports algorithm formulates the fusion problem as the problem of finding the maximum weight dominating set (MWDS) in a directed graph with edge weights. We propose a greedy algorithm to find the MWDS and, thus, infer the total number of incumbents and the labels that the CRs report for each. If a CR has a label for an IU, it belongs to the footprint of that IU.

The rest of the paper is organized as follows. Section II describes the notation, the system model, and the problem formulation. The performance metrics for evaluating the algorithms are proposed in Section III along with an algorithm to compute them. In Section IV, a Gaussian mixture model is fit to the system model. The soft reports algorithm is described in Section V while the hard reports algorithm is described in Section VI. The algorithms' performance is evaluated using MATLAB and NS3 simulations and discussed in Section VII. Conclusions and future work are described in Section VIII.

II. System Model and Problem Formulation

A. Notation

We use Φ to denote the cumulative distribution function of the zero mean unit variance Gaussian distribution. χ^2_{2T} denotes the χ^2 distribution with $2T$ degrees of freedom. For a positive integer M, we denote the set $\{1, 2, \ldots, M\}$ by $[M]$. An all-ones column vector is denoted by $\mathbb{1}$. Its length depends on the context. A unit norm vector with a 1 in the mth position is denoted by b_m. We denote the superset of a set $\mathcal{S}$ by $2^{\mathcal{S}}$. For a positive integer z, we define the map $\beta_z : \{0,1\}^z \to 2^{[z]}$ as $\beta_z(a) = \{m \in [z] : a_m = 1\}$. For a matrix C, we denote the submatrix consisting of the intersections of the rows indexed $\mathcal{I}_r$ and columns indexed $\mathcal{I}_c$ by $C_{\mathcal{I}_r,\mathcal{I}_c}$. We denote the indicator variable by $I_{\{\cdot\}}$.

B. System Model

Consider M incumbent transmitters indexed as $1, \ldots, M$. For $m \in [M]$, the mth incumbent transmitter transmits, at time t, a circular symmetric complex Gaussian signal $x_m[t] \sim \mathcal{CN}\left(0, \sigma_m^2\right)$ with unknown transmit power σ_m^2. As we shall describe below in detail, the CRs will repeatedly measure

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

TABLE I
NOTATION FOR SYSTEM MODEL

Symbol	Description
$a[n]$	Activity in nth frame
C	Matrix of means of source components
$e_k[n]$	Energy at kth CR during frame n
$\mathcal{F}_m$	Footprint of mth incumbent
K	No. of CRs
M	No. of IUs
N	No. of frames
T	No. of samples per frames
$\beta_z(\alpha)$	Indicator function for incumbents active according to a
$\nu_k[t]$	Noise at kth CR and time t
σ_m^2	Transmit power of mth user
σ_ν^2	Noise power vector.
τ_k	Threshold for footprint determination

energy received using frames of length T samples. During the nth frame, let the mth incumbent's activity be denoted by $a_m[n] \in \{0,1\}$ where $a_m[n] = 1$ if the mth incumbent is transmitting and 0 otherwise.

Assume that there are K CRs indexed as $1,\ldots,K$. Each CR measures the energy received at times $t_1 < t_2 < \cdots < t_N$ using frames of T samples each. The channel $h_{m,k}$ from the mth incumbent to the kth transmitter is assumed to be slow fading and, therefore, constant for the duration of the algorithm. The practicality of this assumption is discussed in Section VII-C. For $t \in \{t_n, t_n+1,\ldots,t_n+T-1\}$, $n \in [N]$, the kth CR receives

$$y_k[t] = \sum_{m=1}^{M} h_{m,k}\, a_m[n]\, x_m[t] + \nu_k[t] \tag{1}$$

where $\nu_k[t] \sim \mathcal{CN}(0,\sigma_{\nu_k}^2)$ is additive white Gaussian noise with known power $\sigma_{\nu_k}^2$. Known noise power is a common assumption in cognitive radio literature [18], [19]. Let $\sigma_\nu^2 \triangleq [\sigma_{\nu_1}^2,\ldots,\sigma_{\nu_K}^2]^T$. The energy measured at time t_n, i.e., in the nth frame, is given by

$$e_k[n] = \sum_{t=t_n}^{t_n+T-1} |y_k[t]|^2. \tag{2}$$

We define $e[n] \triangleq [e_1[n] \;\cdots\; e_K[n]]$.

We define $a[n] \triangleq [a_1[n] \;\cdots\; a_M[n]]$. Note that we do not need to assume independence of $a_{m_1}[n_1]$ and $a_{m_2}[n_2]$ for $m_1 \neq m_2$ and $n_1, n_2 \in [N]$. This permits the use of our algorithms to identify communicating incumbents that use medium access control (MAC) protocols. However, for our proposed algorithms to work, we shall make the following assumption on the activity $a[n]$: For each $m \in [M]$, there exists at least one $n \in [N]$ such that $a[n] = b_m$. In other words, each incumbent transmitter is observed to be transmitting alone at least once. This is known as a separability assumption [17].

C. Problem Formulation

Define the footprint $\mathcal{F}_m$ of the mth incumbent as

$$\mathcal{F}_m = \left\{k \in [K] : T|h_{m,k}|^2\sigma_m^2 + T\sigma_{\nu_k}^2 > \tau_k\right\} \tag{3}$$

where the thresholds τ_k may be obtained from standards such as IEEE 802.22 [20] or from the CR's noise power [21]. We choose it based on the noise power:

$$\tau_k \triangleq \frac{\sigma_{\nu_k}^2}{2}\chi_{2T}^{-2}(1-P_{\text{FA}}) \tag{4}$$

where P_{FA} is the desired probability of false alarm for energy detection. The goal of this work is to estimate M and $\mathcal{F}_1,\mathcal{F}_2,\ldots,\mathcal{F}_M$ from $e[1], e[2],\ldots,e[n]$. We assume that our proposed algorithms operate at a fusion center that all the CRs communicate their hard or soft reports with.

1) Soft Reports Algorithm: At the nth frame, the kth CR transmits $e_k[n]$ to the fusion center. At each frame, the fusion center uses the online algorithm proposed in Section V to compute an estimate $\hat{M}[n]$ of the number of incumbents and the footprints $\hat{\mathcal{F}}_1[n], \hat{\mathcal{F}}_2[n],\ldots,\hat{\mathcal{F}}_{\hat{M}[n]}[n]$ of each.

2) Hard Reports Algorithm: For the hard reports algorithm, the kth CR uses $e_k[1],\ldots,e_k[N]$ to estimate the number of incumbents it is receiving $\hat{M}_k$, identify the transmitters it can detect, and label each frame by the set of incumbents $\ell_k[n] \in 2^{[\hat{M}_k]}$ that it detects to be active. It also computes a confusion matrix $\hat{P}_k \in [0,1]^{\hat{M}_k\times\hat{M}_k}$ that estimates the probability of the CR misidentifying the transmitters for a frame. Unlike the soft reports algorithm, the CR communicates these estimates to the fusion center after N frames and not at each frame. After receiving N frames, the hard reports algorithm combines these reports to estimate the number of incumbents $\hat{M}$ and the footprints $\hat{\mathcal{F}}_1,\ldots,\hat{\mathcal{F}}_{\hat{M}}$ of each.

III. PERFORMANCE METRICS

The problem we tackle consists of a model size estimation problem, i.e., estimating the number of signals $\hat{M}$. It is also a parameter estimation problem because we are estimating the footprints $\hat{\mathcal{F}}_m$ of each signal $m \in [\hat{M}]$. Hence, we require two sets of performance metrics – one for estimating the number of signals and another for the quality of footprints.

Given a set $\{\hat{\mathcal{F}}_1,\hat{\mathcal{F}}_2,\ldots,\hat{\mathcal{F}}_{\hat{M}}\}$ of detected footprints, we define an injective map $\mu : [M] \to \{\emptyset\} \cup [\hat{M}]$ such that $\mu(m)$ is the index of the estimated incumbent that corresponds to the mth true incumbent and $\mu(m) = \emptyset$ if none of the estimated incumbents correspond to the mth true incumbent. We describe computing μ at the end of this section.

When estimating the number of incumbents detected, the algorithm may not distinguish between two or more incumbents or it may detect more incumbents than there actually are. To measure the first type of error, we compute the probability of misdetecting at least one incumbent:

$$P_{\text{det}}(\{\mathcal{F}_1,\ldots,\mathcal{F}_M\}) = P(\mu(m) \neq \emptyset \quad \forall\, m \in [M]). \tag{5}$$

The second type of error is measured by the average number of extra incumbents detected. It is normalized by the number of incumbents:

$$\hat{E}(\{\mathcal{F}_1,\ldots,\mathcal{F}_M\}) \triangleq \frac{1}{M}\left|\left\{m' \in [\hat{M}] : \mu(m) \neq m', \forall m \in [M]\right\}\right|. \tag{6}$$

The quality of the detected footprints is estimated by the number of CRs erroneously included in or excluded from the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

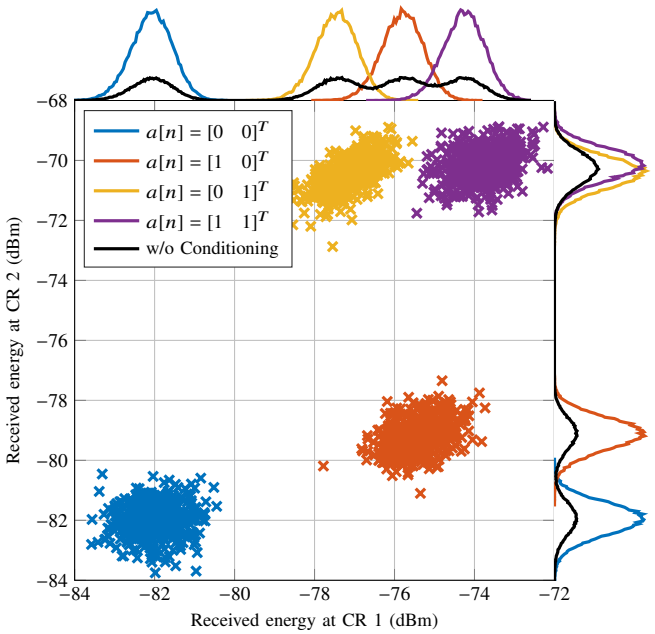


Fig. 2. Scatter plot of received energy at 2 CRs for different components compared to the marginal histograms of each component. Also shown is the histogram of the received energy without conditioning on the components. System: 2 IUs with average activity 0.5, 2 CRs measuring energy using frames of $T = 64$ samples each. Channels as per description in Section VII-A.

estimated footprint, i.e., the cardinality of the symmetric set difference between, say, $\mathcal{F}_m$ and $\hat{\mathcal{F}}_{\mu(m)}$:

$$\delta(\mu(m), m) \triangleq \left|\left(\hat{\mathcal{F}}_{\mu(m)} \backslash \mathcal{F}_m\right) \cup \left(\mathcal{F}_m \backslash \hat{\mathcal{F}}_{\mu(m)}\right)\right|. \tag{7}$$

Hence, we compute the errors in detected footprints as

$$\hat{F}\left(\{\mathcal{F}_1, \ldots, \mathcal{F}_M\}\right) \triangleq \sum_{\substack{m=1 \\ \mu(m)\neq\emptyset}}^{M} \frac{\delta(\mu(m), m)}{|\mathcal{F}_m|}. \tag{8}$$

Now, taking a step back, we describe the method of computing the function $\mu(m)$ for all $m \in [M]$. As noted above, the detected footprints may be erroneous. So, an estimated footprint $\hat{\mathcal{F}}_{m'}$ and a true footprint $\mathcal{F}_m$ are compared using $\delta(m', m)$. If $\delta(m', m) \leq \max\{1, |\mathcal{F}_m|/4\}$, then we say that the m'th estimated incumbent may correspond to the mth true incumbent. Multiple estimated incumbents may correspond to the mth true incumbent. In order to resolve such ambiguity we propose using a maximum weight bipartite graph matching algorithm to find μ.

We construct a bipartite graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ which has vertices $\mathcal{V} = \{\mathcal{F}_1, \ldots, \mathcal{F}_M, \hat{\mathcal{F}}_1, \ldots, \hat{\mathcal{F}}_{\hat{M}}\}$ and it has edges $(\mathcal{F}_m, \hat{\mathcal{F}}_{m'}) \in \mathcal{E}$ if and only if $\delta(m', m) \leq \max\{1, |\mathcal{F}_m|/4\}$. Furthermore, if the edge $(\mathcal{F}_m, \hat{\mathcal{F}}_{m'})$ exists, it is assigned weight $|\mathcal{F}_m| - \delta(m', m)$. We compute the maximum weight bipartite graph matching $\mathcal{E}' \subseteq \mathcal{E}$ on $\mathcal{G}$ using the algorithm proposed by [22] and using the implementation made available at [23]. The injective map μ is then defined as

$$\mu(m) = \begin{cases} m' & \text{if } (\mathcal{F}_m, \hat{\mathcal{F}}_{m'}) \in \mathcal{E}' \\ \emptyset & \text{otherwise.} \end{cases} \tag{9}$$

IV. Gaussian Mixture Model of the Received Energy

The two algorithms proposed in this paper are based on a Gaussian mixture model for the CRs' measurements of the received energy. We shall also describe certain properties of the components of this mixture model that are useful for reducing storage and computational complexity.

When conditioned on the activity vector for frame n, the received signal in that frame, i.e., at time t such that $t_n \leq t < t_n + T$, is a circularly symmetric complex Gaussian random variable:

$$y_k[t]|a[n] \sim \mathcal{CN}\left(0, \sum_{m=1}^{M} |h_{m,k}|^2 a_m[n] \sigma_m^2 + \sigma_{\nu_k}^2\right). \tag{10}$$

Therefore, the energy received during frame n conditioned on the activity vector $a[n]$ has a χ^2 distribution with $2T$ degrees of freedom:

$$e_k[n]|a[n] \sim \left(\sum_{m=1}^{M} |h_{m,k}|^2 a_m[n] \frac{\sigma_m^2}{2} + \frac{\sigma_{\nu_k}^2}{2}\right) \chi_{2T}^2. \tag{11}$$

The χ^2 distribution can be approximated by a Gaussian distribution as follows

$$e_k[n]|a[n] \sim \mathcal{N}\Bigg(T \sum_{m=1}^{M} |h_{m,k}|^2 a_m[n] \sigma_m^2 + T\sigma_{\nu_k}^2, \; T\left(\sum_{m=1}^{M} |h_{m,k}|^2 a_m[n] \sigma_m^2 + \sigma_{\nu_k}^2\right)^2\Bigg). \tag{12}$$

Hence, the received energy vector $e[n]$ given the activity vector $a[n]$ can be approximated by a jointly Gaussian distribution. Therefore, the received energy vector $e[n]$ has a Gaussian mixture distribution whose active component during frame n is decided by the activity $a[n]$. Note that $\beta_M(a[n]) \subseteq [M]$ incumbents are active when the activity vector is $a[n]$. Since $a[n] \in \{0, 1\}^M$, the Gaussian mixture distribution has 2^M components.

An example of the distribution of the received energy at 2 CRs receiving 2 IUs is shown in Fig. 2. The marginal distributions of the received energy are shown along the axes with and without conditioning on the activity vector. In this example, the conditional distributions of 3 of the 4 components do not have a clear separation at CR 1 while the conditional distributions of 2 of the 4 components overlap significantly at CR 2. However, all 4 components are clearly separated in the scatter plot of the received energy at both CRs.

We call the M components corresponding to the transmissions of single incumbents, i.e., $a[n]^T \mathbb{1} = 1$, as *source components*. Their means are collected as rows of a matrix C:

$$C \triangleq \begin{bmatrix} \mathbb{E}[e[n]|a[n] = b_1]^T \\ \mathbb{E}[e[n]|a[n] = b_2]^T \\ \vdots \\ \mathbb{E}[e[n]|a[n] = b_M]^T \end{bmatrix}. \tag{13}$$

Due to the additive nature of the received energy, the means of the remaining $2^M - M$ mixture components are deterministic functions of the means of the M source components. Using (12) and (13), the mean of a component corresponding to activity vector $\alpha \in \{0, 1\}^M$ is given by

$$\mathbb{E}[e[n]|a[n] = \alpha] = \alpha^T C - (\alpha^T \mathbb{1} - 1)\sigma_\nu^2. \tag{14}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

Furthermore, the variance of the component depends on its mean:

$$\mathrm{Var}\left(e_k[n]|a[n]=\alpha\right) = \frac{1}{T}\mathbb{E}^2\left[e_k[n]|a[n]=\alpha\right]. \tag{15}$$

The covariance of the component

$$\mathrm{Cov}\left(e_{k_1}[n], e_{k_2}[n]|a[n]=\alpha\right) = T\sum_{m=1}^{M}\left|h_{mk_1}\right|^2\left|h_{mk_2}\right|^2\alpha_m\sigma_m^4 + T\sum_{\substack{m_1,m_2=1\\ m_1\neq m_2}}^{M} h^*_{m_1k_1}h_{m_2k_1}h^*_{m_2k_2}h_{m_1k_2}\alpha_{m_1}\alpha_{m_2}\sigma^2_{m_1}\sigma^2_{m_2}.$$

is derived in the Appendix as (46).

Finally, we note that the Gaussian approximation motivates us to use the well-studied Mahalanobis distance for comparing a measurement of the received energy to already learnt components. We shall use this property while developing the soft reports algorithm.

V. Learning the Gaussian Mixture Distribution using Soft Reports

Mixture model learning algorithms have been developed in various frameworks. The expectation-maximization based algorithms, such as [24]–[26], are known to suffer from convergence issues. Method of moments based algorithms [27], [28] require prior knowledge of the number of mixture components and estimates of moments of up to $\Omega(2^M)$th order since the number of mixture components is 2^M. Accurate estimates of these moments may not be available or require computational and sample complexity exponential in the number of mixture components [29]. Multi-view methods such as [29] use only lower order moments but require each CR's measurement to have at least as many dimensions as the number of components. Methods based on matrix decomposition, such as [30], operate on univariate measurements but require that the number of CRs exceed the number of mixture components.

Our proposed soft reports algorithm is based on the online algorithm proposed in [16]. The improvements of the proposed soft reports algorithm over the algorithm proposed in [16] are explained at the appropriate points in the algorithm description below. Since the distribution described in Section IV clearly shows that two source components cannot have the same mean with different covariances, the proposed algorithm uses a separation criteria, i.e., a threshold on distance between the two components, to distinguish the components. The proposed algorithm is not recursive and, hence, avoids the corresponding computational complexity and the requirement of storing the received samples. The flow of the proposed algorithm is shown in Fig. 3.

A. Notation

We shall classify the components we learn as confirmed and candidates. Furthermore, each of the components are classified as source and sum components. The quantities for candidate components are denoted using a superscript c.

Let $\hat{M}[n] \in \mathbb{N}$ be the number of source components detected after processing frame n and assume that the source components are indexed as $1, 2, \ldots, \hat{M}[n]$. The means of these source components are denoted by the matrix $\hat{C}[n] \in \mathbb{R}_+^{\hat{M}[n]\times K}$ such that the mth row of $\hat{C}[n]$ is the sample mean of the mth detected source component.

TABLE II
Notation for Soft Reports Algorithm

Symbol	Description
$f^c[n]$	Frame at which candidate component is first observed
$s^c[n]$	No. of active measurements for candidate source components
$\hat{C}[n]$	Sample means of source components
$\hat{C}^C[n]$	Sample means of candidate source components
$D_M(e', \alpha, C')$	Mahalanobis metric
$\mathcal{D}$	Components to be deleted
$\hat{\mathcal{F}}_m[n]$	Footprint estimate of mth detected incumbent
$\hat{M}[n]$	No. of detected IUs
$\hat{M}^c[N]$	No. of detected candidate source components
$\mathcal{M}_m$	Set of sets that need to be merged together or deleted
$\hat{S}(\alpha)$	Estimated covariance matrix for α
$\eta[n]$	No. of measurements used to estimate sample mean of source components
$\eta^c[n]$	No. of measurements used to estimate sample mean of candidate source components
ω	No. of measurements for which candidate source components are on probation
$\hat{\sigma}_k^2(\alpha, \hat{C}[n-1])$	Variance of αth component inferred from source components $\hat{C}[n-1]$
υ	No. of frames candidate source component has to be active before confirmation
ζ	Threshold to detect outlier

We initialize the algorithm by using the received energy vector at the first frame n_0 where $e_k[n_0] > \tau_k$ for all $k \in [K]$ as the mean of the first candidate source component. To be precise, $\hat{M}^c[n] \triangleq 0$ if $n < n_0$ and $\hat{M}^c[n] \triangleq 1$ if $n = n_0$. Similarly, we initialize the sample means of the candidate components as $\hat{C}^c[n_0] = e[n_0]$, the measurement index at which it is seen first as $f_1^c[n_0] = n_0$, the number of times it is seen individually as $\eta^c[n_0] = 1$, and the number of times seen in total as $s^c[n_0] = 1$ while these quantities are undefined for $n < n_0$.

B. Comparing Energy Measurement to Learnt Components

To detect the active component, we compare received energy measurements to the already learnt confirmed and candidate components.

1) Estimation of Component Statistics: The mean of the component corresponding to the activity vector $\alpha \in \{0,1\}^{\hat{M}[n]}$ is estimated by

$$\alpha^T\hat{C}[n] - (\alpha^T\mathbb{1} - 1)\sigma_\nu^2 \tag{16}$$

based on (14). Based on (12), we can estimate $\mathrm{Var}(e_k[n]|a[n]=\alpha)$ by

$$\hat{\sigma}_k^2(\alpha, \hat{C}[n-1]) \triangleq \frac{1}{T}\left(\alpha^T\left(\hat{C}_{[\hat{M}[n]],k}[n-1] - T\sigma_{\nu_k}^2\mathbb{1}\right) + \sigma_{\nu_k}^2\right)^2. \tag{17}$$

The off diagonal elements of the theoretical value of $\mathrm{Cov}(e[n]|a[n]=\alpha)$ given by (46) in the Appendix cannot be estimated from the sample means of the received energy

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

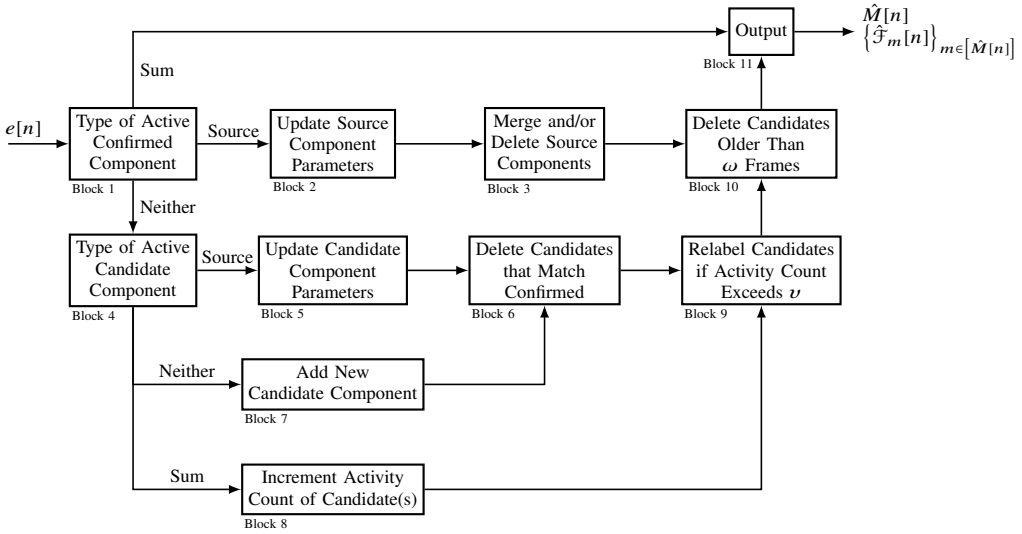


Fig. 3. Flow of soft reports algorithm. Blocks are numbered for ease of description in the text.

because its second summation depends on the complex channel coefficients. As an approximation, the algorithm proposed in [16] estimates it as a diagonal matrix with $\mathrm{Var}(e_k[n]|a[n] = \alpha)$ on the diagonal

$$\hat{S}_T(\alpha) \triangleq \mathrm{diag}\left(\hat{\sigma}_1^2\left(\alpha, \hat{C}[n-1]\right), \hat{\sigma}_2^2\left(\alpha, \hat{C}[n-1]\right), \ldots, \hat{\sigma}_K^2\left(\alpha, \hat{C}[n-1]\right)\right). \quad (18)$$

In this work, we propose estimating $\mathrm{Cov}(e[n]|a[n] = \alpha)$ as

$$\hat{S}(\alpha) \triangleq \frac{1}{T}\sum_{m=1}^{\hat{M}} \alpha_m \left(\hat{C}[n]^T b_m - T\sigma_\nu^2\right)^T \left(\hat{C}[n]^T b_m - T\sigma_\nu^2\right) + \hat{S}_T(\alpha) - \frac{1}{T}\sum_{m=1}^{\hat{M}} \alpha_m \left(\mathrm{diag}\left(\hat{C}[n]^T b_m - T\sigma_\nu^2\right)\right)^2 \quad (19)$$

by using the first sum in the theoretical value of the covariance as derived in (46) in the Appendix.

2) Distance Computation: In [16], a received sample $e[n]$ is compared to the component with activity vector α and mean $C' = \alpha^T \hat{C}[n-1] - (\alpha^T \mathbb{1} - 1)\sigma_\nu^2$ using $\hat{S}_T(\alpha)$ by computing the *minimum tail probability*:

$$D_T\left(e[n], \alpha, C'\right) \triangleq \max_{k\in[K]} \left\{ \Phi\left(\frac{e_k[n] - C'_{[\hat{M}[n]],k}}{\sigma_k(\alpha, C')}\right), 1 - \Phi\left(\frac{e_k[n] - C'_{[\hat{M}[n]],k}}{\sigma_k(\alpha, C')}\right)\right\}. \quad (20)$$

Hence, we denote the algorithm proposed in [16] as the minimum tail probability algorithm.

In this work, a received sample $e[n]$ is compared to that same component by computing the squared Mahalanobis distance

$$d_M\left(e[n], \alpha, C'\right) \triangleq \left(e[n] - C'\right)^T \hat{S}^{-1}(\alpha)\left(e[n] - C'\right) \quad (21)$$

and using the fact that the squared Mahalanobis distance has a χ^2 distribution with K degrees of freedom to compute its likelihood function

$$D_M\left(e[n], \alpha, C'\right) \triangleq \chi_K^2\left(d_M\left(e[n], \alpha, C'\right)\right). \quad (22)$$

In this paper, we shall refer to D_M as the Mahalanobis metric. The improved covariance estimation in (19) and the Mahalanobis metric reduce the probability of labelling outliers as new IUs but also reduce the probability of detecting a distinct IU as such, particularly for small K. This difference is described numerically in Section VII.

C. Adding a New Component

In essence, our algorithm checks if the latest measurement of received energy is an outlier when compared to the source and sum components learnt so far. If yes, then the algorithm should learn this measurement as a new source component. However, since the algorithm's estimate of the mean and variance of each component changes as more samples are received, it is possible that the measurement may be an outlier generated by a component that we have already learnt albeit with inaccurate statistics.

In such a scenario, the minimum tail probability algorithm of [16] adds a new source component and then merges multiple source components as their statistics become more accurate. This increases the total number of components, sum and source, exponentially. In turn, this increases the computational complexity of the algorithm exponentially. Instead, we propose that an outlying measurement received at frame n be learnt as a *candidate* source component that is confirmed to be a source component if it is detected to be active at at least υ for the next ω measurements. This step is intended to reduce the number of extra IUs detected.

Accordingly, the algorithm first compares the latest received measurement $e[n]$ to each confirmed source and sum component in Block 1:

$$\hat{D}_1 \triangleq \min_{\substack{\alpha\in\{0,1\}^{\hat{M}[n-1]} \\ \alpha^T\mathbb{1}>0}} D_M\left(e[n], \alpha, \hat{C}[n-1]\right) \quad (23)$$

and then to each candidate source and sum component in Block 4:

$$\hat{D}_2 \triangleq \min_{\substack{\alpha\in\{0,1\}^{\hat{M}[n-1]+\hat{M}^c[n-1]} \\ \alpha^T\mathbb{1}>0}} D_M\left(e[n], \alpha, \begin{bmatrix}\hat{C}[n-1] \\ \hat{C}^c[n-1]\end{bmatrix}\right). \quad (24)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

TABLE III
UPDATE EQUATIONS FOR BLOCKS 7 AND 10 OF SOFT REPORTS ALGORITHM IN FIG. 3

	$\hat{C}[n]$	$\hat{C}^c[n]$	$\eta[n]$	$\eta^c[n]$	$s^c[n]$	$f^c[n]$
Block 7	$\hat{C}[n-1]$	$\begin{bmatrix}\hat{C}^c[n-1]\\ e[n]^T\end{bmatrix}$	$\eta[n-1]$	$\begin{bmatrix}\eta^c[n-1]\\ 1\end{bmatrix}$	$\begin{bmatrix}s^c[n-1]\\ 1\end{bmatrix}$	$\begin{bmatrix}f^c[n-1]\\ n\end{bmatrix}$
Block 10	$\begin{bmatrix}\hat{C}[n-1]\\ \hat{C}^c_{\mathcal{C},[K]}[n]\end{bmatrix}$	$\hat{C}^c_{\mathcal{C}',[K]}[n]$	$\begin{bmatrix}\eta[n-1]\\ \eta^c_{\mathcal{C}}[n]\end{bmatrix}$	$\eta^c_{\mathcal{C}'}[n]$	$s^c_{\mathcal{C}'}[n]$	$f^c_{\mathcal{C}'}[n]$

If both $\hat{D}_1 > \zeta$ and $\hat{D}_2 > \zeta$, then a new candidate component is added in Block 7 using the update equations in Table III and by setting $\hat{M}^c[n] = \hat{M}^c[n-1] + 1$.

On the other hand, if $\hat{D}_1 \leq \zeta$, then the active component is determined as:

$$\hat{a} \triangleq \underset{\substack{\alpha\in\{0,1\}^{\hat{M}[n-1]}\\ \alpha^T\mathbb{1}>0}}{\operatorname{argmin}} D_M\left(e[n], \alpha, \hat{C}[n-1]\right) \tag{25}$$

and the component's parameters are updated in Block 2 if $\hat{a}^T\mathbb{1} = 1$ or the algorithm proceeds to Block 11 for output. If $\hat{D}_1 > \zeta$ but $\hat{D}_2 \leq \zeta$, a similar process is followed to update the candidate source component's parameters in Block 5 or to increment the activity counts in Block 8.

D. Updating Parameters of Components

The statistics of confirmed and candidate components are updated in Blocks 2 and 5 respectively. Let m be the index of the active confirmed (or candidate) component.

Block 2 updates the sample mean of the active component using $e[n]$.

$$\hat{M}[n] = \hat{M}[n-1] \tag{26}$$

$$\eta_m[n] = \eta_m[n-1] + I_{\lambda[n]=\{m\}} \tag{27}$$

$$\hat{C}_{m,[K]}[n] = \begin{cases} \frac{e[n]+(\eta_m[n]-1)\hat{C}_{m,[K]}[n-1]}{\eta_m[n]} & \text{if } \lambda[n] = \{m\} \\ \hat{C}_{m,[K]}[n-1] & \text{otherwise.} \end{cases} \tag{28}$$

Block 5 uses the same equation to update $\hat{M}^c$, η^c_m, and $\hat{C}^c$. Furthermore, Block 5 also updates the activity count:

$$s^c[n] = s^c[n-1] + \sum_{\substack{m'\in\lambda[n]\\ m'>\hat{M}[n-1]}} b_{m'-\hat{M}[n-1]} \tag{29}$$

where $\hat{a}$ is the activity vector that minimizes (24) and $\lambda[n] \triangleq \beta_{\hat{M}[n-1]+\hat{M}^c[n-1]}(\hat{a})$.

Finally, Block 8 increments the activity counts of the source components $\lambda[n]$ using (29).

E. Deleting or Merging Source Components

In Block 3, we delete any source components whose sample means are likely to have been generated by a sum components. For each $m \in [\hat{M}[n]]$, we find the sets

$$\mathcal{M}_m \triangleq \left\{\beta_{\hat{M}[n]}(\alpha) : \alpha \in \{0,1\}^{\hat{M}[n]}, \alpha^T b_m = 0, \alpha^T\mathbb{1} > 0, D_M\left(\hat{C}_{m,[K]}[n], \alpha, \hat{C}[n]\right) \leq \zeta\right\}. \tag{30}$$

The source components $\mathcal{D} \triangleq \{m : \exists\mathcal{N} \in \mathcal{M}_m \text{ such that } |\mathcal{N}| > 1\}$ are deleted. A similar check is performed on the candidate components in Block 6.

To merge multiple source components simultaneously, we construct an undirected graph with vertices $[\hat{M}[n]]\backslash\mathcal{D}$ and an edge between vertices corresponding to source components indexed m_1 and m_2 if $m_1 \in \mathcal{M}_{m_2}$. Source components that are connected are merged.

These checks are repeated until no more source components can be merged or deleted.

F. Maintenance of Candidate Components

In Block 9, we confirm those candidate source components that have an activity count greater than υ: $\mathcal{C} \triangleq \{m \in [\hat{M}^c[n-1]] : s^c_m[n] \geq \upsilon\}$.

In Block 10, the candidate source components older than ω are deleted. Hence, the new candidate source components are $\mathcal{C}' \triangleq \{m \in [\hat{M}^c[n-1]] : s^c_m[n] < \upsilon, n - f^c_m[n] < \omega\}$. The source components are updated as shown in Table III.

G. Output

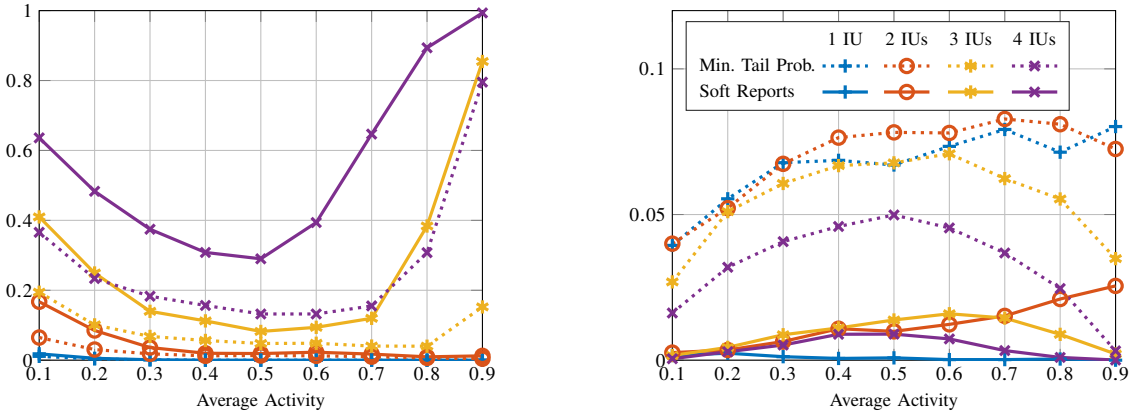
Finally, we infer the footprints of the identified incumbents using the sample means of the learnt source components:

$$\hat{\mathcal{F}}_m[n] \triangleq \left\{k : \hat{C}_{[\hat{M}[n]],k}[n](m) > \tau_k\right\} \quad m \in \{1,\ldots,\hat{M}[n]\}. \tag{31}$$

VI. HARD REPORTS ALGORITHM

The minimum tail probability metric proposed in [16] and restated in (20) distinguishes two incumbents if there is at least one CR k such that $\Phi\left(\frac{e'_k - C'_{[\hat{M}],k}}{\sigma_k(\alpha,C')}\right)$ or $1 - \left(\frac{e'_k - C'_{[\hat{M}],k}}{\sigma_k(\alpha,C')}\right)$ exceeds the threshold ζ. Hence, if we were to learn the marginal Gaussian mixture distribution from a single CR's received energy values, we might be able to distinguish these incumbents. In fact, Fig. 4 shows that a single CR is able to distinguish up to 3 incumbents with high accuracy using the soft reports algorithm. Motivated by this observation, we propose a fusion algorithm to fuse the inferences of multiple CRs each of which learns the marginal Gaussian mixture distribution locally. The CRs will transmit, to the fusion center, the number of incumbents it identifies, the estimated activity of each incumbent, and an estimated confusion matrix. Analogous to the hard and soft reports in spectrum sensing, the algorithm proposed in this section is a hard reports fusion algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking



(a) Probability of misdetecting at least one incumbent

(b) Normalized average number of extra incumbents

Fig. 4. Footprint learning by 1 CR. Parameters: $N = 200$ frames of $T = 64$ samples, $\zeta = 0.99$

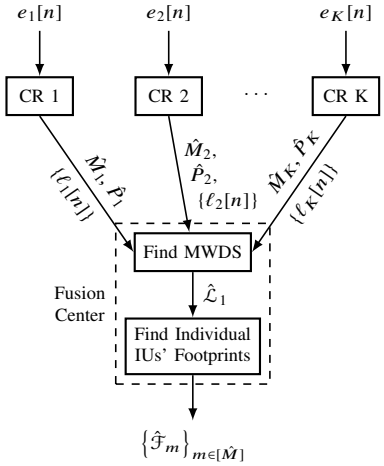


Fig. 5. Block diagram of hard reports algorithm.

TABLE IV
NOTATION FOR HARD REPORTS ALGORITHM

Symbol	Description
$\mathbf{0}$	K-tuple corresponding to noise only
$\mathcal{B}$	Edge set of $\mathcal{H}$
$\ell_k[n]$	Hard report from kth CR at time n
$v(L_1, L_2)$	Weight of edge in $\mathcal{H}$
$w(\mathcal{T})$	Weight of a dominating set $\mathcal{T}$
$D_S(L_1, L_2)$	Distance metric between tuples L_1, L_2
$\mathcal{H}$	Graph of reported tuples
$\mathcal{J}$	Weakly connected components of $\mathcal{H}$
$L[n]$	Reported tuple at time n
$\hat{\mathcal{L}}$	Search space in hard reports algorithm
$\hat{\mathcal{L}}_1$	Maximum weight dominating set of $\mathcal{H}$
$\hat{P}_k$	Confusion matrix for tuples reported by kth CR
$\mathcal{T}$	A dominating set in $\mathcal{H}$
$\mathcal{T}_D$	Set of dominated reported tuples in $\mathcal{H}$
$\mathcal{U}$	Set of reported tuples
γ	Threshold for hard reports algorithm

We shall begin by describing each CR's computations to generate the hard reports. Next, the problem of fusing these reports is formalized as the problem of finding the maximum weight dominating set of a graph constructed from the reports. We propose a greedy algorithm to solve this problem and apply it to the fusion problem. The block diagram of the algorithm is shown in Fig. 5.

A. Generation of Hard Reports by CR

Let each CR operate a unidimensional Gaussian mixture learning algorithm such as that proposed in [16] or the soft reports algorithm proposed in Section V. In this paper, we shall assume that the soft reports algorithm is used though the hard fusion algorithm does not have such a pre-requisite. Assume that the algorithm runs for N frames, the kth CR identifies $\hat{M}_k$ source components, and generates labels $\ell_k[n] \subseteq [\hat{M}_k]$ for each frame n according the source components detected to be active:

$$\ell_k[n] = \beta_{\hat{M}_k}\left(\underset{\substack{\alpha\in\{0,1\}^{\hat{M}_k}\\ \alpha^T\mathbb{1}>0}}{\operatorname{argmin}} D_M\left(e_k[n], \alpha, \hat{C}_{[\hat{M}_k],k}[N]\right)\right) \quad (32)$$

if $e_k[n] > \tau_k$ and $\ell_k[n] = \emptyset$ otherwise.

Furthermore, we assume that the CR estimates the confusion probability $\hat{P}_k : 2^{\hat{M}_k} \times 2^{\hat{M}_k} \to [0,1]$ such that $\hat{P}_k(\ell_1, \ell_2)$ estimates the probability that the set ℓ_1 is misclassified as the set ℓ_2, i.e.,

$$\hat{P}_k(\ell_1,\ell_2) = \int_{\tau_k}^{\infty}\left[I_{\left\{\ell_2=\beta_{\hat{M}_k}\left(\operatorname{argmin}_{\alpha\in\{0,1\}^{\hat{M}_k}} D_M\left(\epsilon,\alpha,\hat{C}_{[\hat{M}_k],k}[N]\right)\right)\right\}} \times\Phi\left(\frac{\epsilon-\hat{C}_{[\hat{M}_k],k}[N]}{\sigma_k\left(\sum_{m\in\ell_1} b_m, \hat{C}[N]\right)}\right)\right] \mathrm{d}\epsilon$$

The CR transmits $\hat{M}_k$, $\{l_k[n]\}_{n\in[N]}$, and $\hat{P}_k$ to the fusion center.

B. The Fusion Problem

For each incumbent, we wish to determine the labels assigned to it by each CR, i.e., for each incumbent we wish to find a tuple in the set

$$\mathcal{L} \triangleq 2^{[\hat{M}_1]} \times 2^{[\hat{M}_2]} \times \cdots \times 2^{[\hat{M}_K]}$$

such that the kth element of that tuple is the set of labels most likely to be detected by the kth CR when this particular incumbent is transmitting. In this section, we propose a method

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

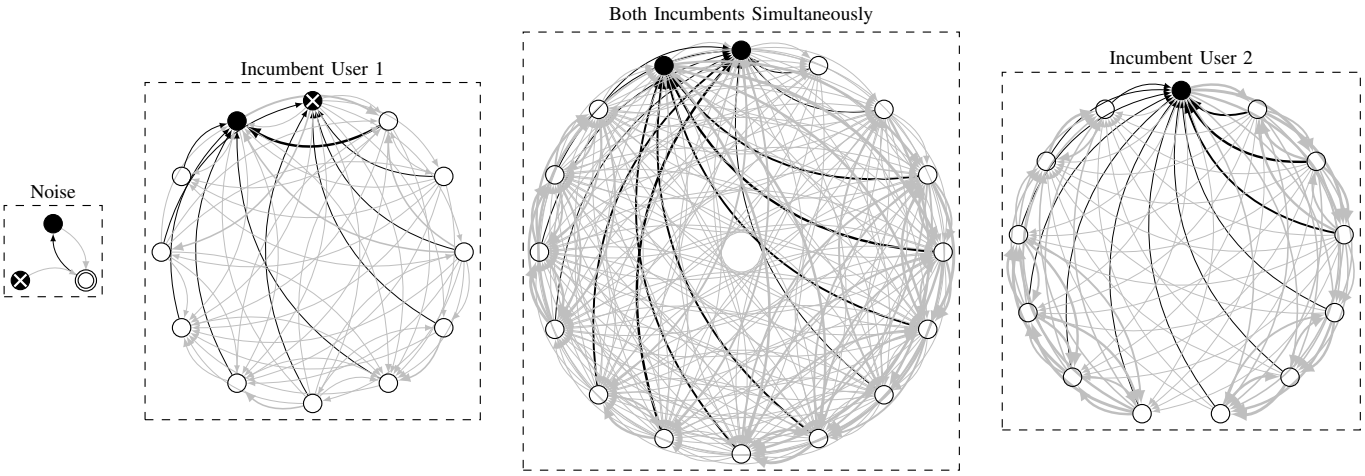


Fig. 6. An example of the graph $\mathcal{H}$ for a system with $K = 6$ CRs, $M = 2$ incumbents, and $N = 200$ frames of $T = 64$ samples. Edge thickness is proportional to the edge weights. Maximum weight dominating set indicated by vertices filled in black. Dark edges correspond to the edges used for computing the weight of the dominating set. Vertex corresponding to noise shown with a double border. Vertices that dominate sets that are active for less than 5% measurements are shown with a white cross.

to determine the tuples that correspond to each incumbent and the combination of incumbents transmitting simultaneously.

We collect the reported labels into a tuple

$$L[n] \triangleq (l_1[n], l_2[n], \ldots, l_K[n]) \in \mathcal{L} \tag{33}$$

which we refer to as the reported tuple at time n. If $L_1, L_2 \in \mathcal{L}$, then we define the distance D_S between them to be

$$D_S(L_1, L_2) \triangleq \max_{k \in [K]} 1 - \hat{P}_k\left(L_{1,k}, L_{2,k}\right). \tag{34}$$

We formulate our problem as follows: Find the minimal subset $\hat{\mathcal{L}} \subset \mathcal{L}$ such that

$$\min_{L' \in 2^{\hat{\mathcal{L}}}} D_S\left(S(L'), L[n]\right) < \gamma \quad \forall n \in [N] \tag{35}$$

where $S : 2^{\hat{\mathcal{L}}} \to \mathcal{L}$ such that $S(L')$ is defined as the tuple

$$S(L') \triangleq \left(\bigcup_{\ell \in L'} \ell_1, \bigcup_{\ell \in L'} \ell_2, \ldots, \bigcup_{\ell \in L'} \ell_K\right) \tag{36}$$

and $\gamma \in [0, 1]$ is a threshold set by the user. We set $\gamma = \zeta$ to ensure that the fusion algorithm differentiates between IUs if and only if at least one CR differentiates between them.

Even for a small number of CRs and few incumbents, the size of $\mathcal{L}$ is very large. For example, with 4 CRs and 3 incumbents, $|\mathcal{L}| = 2^{12}$ and $|2^{\mathcal{L}}| = 2^{2^{12}}$. Therefore, we simplify the problem by restricting the search space to the reported tuples: $\hat{\mathcal{L}} \subseteq \{\mathbf{0}\} \cup \mathcal{U}$ where $\mathcal{U} \triangleq \{L[1], L[2], \ldots, L[n]\}$ and $\mathbf{0}$ is the K-tuple in which all elements are empty sets. $\mathbf{0}$ is useful for identifying reported tuples that are the result of false alarms.

We divide the fusion problem into two stages. The first stage is to find the minimal subset $\hat{\mathcal{L}}_1 \subseteq \{\mathbf{0}\} \cup \mathcal{U}$ such that

$$\min_{L' \in \hat{\mathcal{L}}_1} D_S(L', L[n]) < \gamma \qquad \forall n \in [N]. \tag{37}$$

The vectors in $\hat{\mathcal{L}}_1 \backslash \{\mathbf{0}\}$ are the most likely report vectors generated by the CRs when different sets of incumbents are transmitting. Next, we find the minimal subset $\hat{\mathcal{L}} \subseteq \hat{\mathcal{L}}_1 \backslash \{\mathbf{0}\}$

$$\min_{L' \in \hat{\mathcal{L}}} D_S(S(L'), L'') < \gamma \qquad \forall L'' \in \hat{\mathcal{L}}_1 \backslash \{\mathbf{0}\} \tag{38}$$

These are the most likely report vectors generated by the CRs when individual incumbents are transmitting by themselves. Given $\hat{\mathcal{L}}_1$, we use brute force to solve (38) since the search space is significantly smaller ($O(2^{2^M})$) than for the first stage ($O(2^N)$). The computational complexity of this step is discussed in detail in Section VII.

C. Maximum Weight Dominating Set Model

We model the first stage (37) of our algorithm as a problem of finding the maximum weight dominating set in a directed graph. Consider a directed graph $\mathcal{H} = (\mathcal{U} \cup \{\mathbf{0}\}, \mathcal{B})$ which has vertex set $\mathcal{U} \cup \{\mathbf{0}\}$ and edge set $\mathcal{B}$. It has an edge $(L_1, L_2) \in \mathcal{B}$ if $D_S(L_1, L_2) < \gamma$. An example of the graph $\mathcal{H}$ is shown in Fig. 6. The edge (L_1, L_2) is assigned weight $v(L_1, L_2) \triangleq 1 - D_S(L_1, L_2)$. If edge $(L_1, L_2) \neq \mathcal{B}$, then we define $v(L_1, L_2) \triangleq 0$. We say that a vertex $L_2 \in \mathcal{U} \cup \{\mathbf{0}\}$ dominates another vertex $L_1 \in \mathcal{U} \cup \{\mathbf{0}\}$ if $(L_1, L_2) \in \mathcal{B}$. We say that a subset $\mathcal{T} \subset \mathcal{U} \cup \{\mathbf{0}\}$ is a dominating set if each $L_2 \in \mathcal{U} \cup \{\mathbf{0}\} \backslash \mathcal{T}$ is dominated by at least one $L_1 \in \mathcal{T}$. To such a dominating set, we assign the weight

$$w(\mathcal{T}) \triangleq \sum_{L_1 \in \mathcal{U} \cup \{\mathbf{0}\} \backslash \mathcal{T}} \left[\max_{L_2 \in \mathcal{T}} v(L_1, L_2)\right]. \tag{39}$$

We estimate $\hat{\mathcal{L}}_1$ as the maximum weight dominating set of $\mathcal{H}$. The advantage of maximizing a weight as defined above is that a minimal dominating set has higher weight than a non-minimal dominating set.

Existing literature does not consider the problem of maximum weight dominating sets in directed graphs. However, it is known that the problem of finding a dominating set in a directed graph is a NP-Hard problem [31]. Therefore, finding the maximum weight dominating set is also a NP-Hard problem. The other difference with existing literature is that the convention is to assign weights to the vertices rather than the edges. However, our problem requires that the weights be assigned to the edges and not to the vertices. Keeping this in mind, we propose a novel heuristic greedy algorithm to find the maximum weight dominating set in a directed graph with edge weights.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

Algorithm 1 Finding maximum weight dominating set in a directed graph

Input: $\mathcal{H} = (\mathcal{U} \cup \{\mathbf{0}\}, \mathcal{B})$, $v(L_1, L_2)$ for each $L_1, L_2 \in \mathcal{U} \cup \{\mathbf{0}\}$

1: $\hat{\mathcal{L}}_1 \leftarrow \emptyset$
2: $\mathcal{J} \leftarrow$ weakly connected components of $\mathcal{H}$
3: **for all** $J \in \mathcal{J}$ **do**
4: $\quad \mathcal{T} \leftarrow \{L_1 \in J : \nexists L_2 \in J$ such that $(L_1, L_2) \in \mathcal{B}\}$
5: $\quad$ **while** $\mathcal{T}$ does not dominate J **do**
6: $\quad\quad \mathcal{T}_D \leftarrow \{L_1 \in J \backslash \mathcal{T} : \exists L_2 \in \mathcal{T}$ such that $(L_1, L_2) \in \mathcal{B}\}$
7: $\quad\quad \mathcal{T} \leftarrow \mathcal{T} \cup \arg\max_{L \in J \backslash \mathcal{T}} \sum_{L' \in J \backslash \mathcal{T}_D} v(L', L)$
8: $\quad$ **end while**
9: $\quad \hat{\mathcal{L}}_1 \leftarrow \hat{\mathcal{L}}_1 \cup \mathcal{T}$
10: **end for**
11: $\hat{\mathcal{L}}_1 \leftarrow \hat{\mathcal{L}}_1 \backslash \{L_1 \in J \backslash \mathcal{T} : (L_1, \mathbf{0}) \in \mathcal{B}\}$

Output: $\hat{\mathcal{L}}_1$

Our proposed algorithm takes, as input, the directed graph $\mathcal{H}$ described above. The algorithm is summarized in Algorithm 1. First, we note that the dominating sets of each weakly connected component of $\mathcal{H}$ can be computed separately. For a weakly connected component J of $\mathcal{H}$, we initialize the maximum weight dominating set $\mathcal{T}$, in line 4, by those vertices that are not dominated by any other vertex. The loop in lines 5-8 grows the set $\mathcal{T}$ until all other vertices, i.e., set $\mathcal{T}_D$ computed in line 6, in J are dominated by $\mathcal{T}$. Specifically, we add the vertex from $J \backslash \mathcal{T}$ that maximizes the sum of the weights of the edges to the vertices not dominated by $\mathcal{T}$ yet. Finally, we remove from $\hat{\mathcal{L}}_1$ the vertices dominated by or the vertices that dominate the vertex corresponding to noise, i.e., $\mathbf{0}$.

Similar to the idea of candidate components in the soft reports algorithm, we remove, from $\hat{\mathcal{L}}_1$, vertices that dominate sets of vertices that are active for less than 5% of the measurements.

D. Estimating the Footprints

As mentioned earlier, the second stage (38) of the problem is solved by brute force. Having obtained $\hat{\mathcal{L}}_1$, each element of $\hat{\mathcal{L}}_1$ corresponds to an identified incumbent. Hence, the number of incumbents identified is estimated as $\hat{M} \triangleq |\hat{\mathcal{L}}|$. Indexing elements of $\hat{\mathcal{L}}_1$ as $1, 2, \ldots, \hat{M}$ is required to index the footprints of the identified incumbents. Let $L_m \in \hat{\mathcal{L}}_1$ be the mth element. Then, the footprint of the mth identified incumbent is

$$\hat{\mathcal{F}}_m \triangleq \{k \in [K] : L_{m,k} \neq \emptyset\}. \tag{40}$$

VII. Simulation Results and Discussion

In this section, we study the choice of algorithm parameters, the practicality of our constant channel assumption, the performance of our algorithm for various systems, an example application of localizing 4 IUs simultaneously, and the computational complexity of our algorithms.

We begin the discussion of our proposed algorithms by numerically evaluating the choice of our algorithms' parameters followed by discussing the practicality of our constant channel assumption. Then, we discuss the performance of our proposed algorithms for slotted ALOHA and 802.11 incumbent networks as the number of CRs and IUs is varied. Finally, we discuss the computational complexity of our proposed algorithms.

A. Simulation System

We generated $a[n]$ through 2 systems. For the first system, similar to slotted ALOHA [32], each $a_m[n]$ was generated as a Bernoulli i.i.d. random variable in MATLAB. Up to 4 IUs were simulated in a 2000m×2000m area. For the second system, NS-3[1] was used to simulate up to 2 coexisting 802.11n networks each with 1 AP and 2 STAs in a 2000m×2000m area. The STAs are located randomly in a 100m radius disc around the AP. UDP flows were set up on both downlink and uplink for sending 2000 byte packets at regular intervals. The start times and durations of transmit events were logged with a timestamp. Activities $a_m[n]$ were generated by sampling these events every 150μs as per the discussion below in Section VII-C.

Using the generated $a[n]$, we used MATLAB to generate complex Gaussian incumbent signals, a shadow fading channel with path loss 4, standard deviation 6 dB, and spatial correlation as modeled in [33] with $\alpha = 0.1$. The IUs' transmit power is set as $\sigma_m^2 = 20$dBm and the CRs have a noise power of $\sigma_{\nu_k}^2 = -100$dBm. The received signals at CRs are generated as per (1). The threshold for the footprint definition is obtained from (4) with $P_{\text{fa}} = 0.01$. Unless mentioned otherwise, $N = 200$ frames of length $T = 32$ samples are used for energy measurement and the threshold $\zeta = 0.99$. For the soft reports algorithm, candidate source components are labeled as confirmed if they are detected to be active for $\upsilon = 5$ of the latest $\omega = 100$ frames.

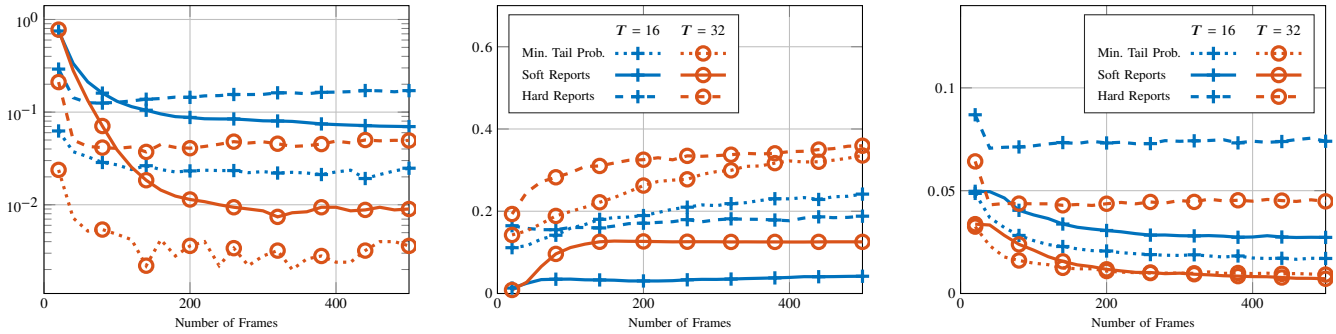
Note that throughout this section, plots of probability of misdetecting at least one IU are lower bounded by 10^{-4} in order to permit using the log scale. Since at most 5000 realizations were used for any experiment, this is a valid lower bound.

B. Design of Algorithm Parameters

The separability assumption and the introduction of candidate components mean that the probability of detecting all IUs increases with N. In addition, increasing the number of samples per frame T helps distinguish mixture components. Hence, Fig. 7(a) shows that increasing T and N help improve performance for all three algorithms. Increasing T also increases the number of extra IUs detected, as shown in Fig. 7(b) because the increase in variance of the components increases the number of outliers. Another point to note from Fig. 7 is a common trend that the minimum tail probability algorithm detects all IUs with higher probability but also detects a larger number of extra IUs than the soft reports algorithm. Particularly, note that the average number of extra incumbents detected by the minimum tail probability algorithm does not converge with increasing N. The hard reports algorithm suffers in performance doubly because it uses a more error-prone

[1] We used commit eb6cd95 from the ns-3-dev git repository in order to use the latest SpectrumWifiPhy implementation. The MonitorSnifferTx trace was modified to obtain the physical layer transmit duration for each packet.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking



(a) Probability of misdetecting at least one IU (b) Normalized average number of extra IUs (c) Quality of detected footprints

Fig. 7. Error in detecting incumbents when varying T and N for $M = 3$ IUs, $K = 4$ CRs, 0.4 average activity for each IU.

method to generate the labels and its inability to use the off-diagonal elements of the covariance matrix as the soft reports algorithm does. This causes detection of a larger number of extra IUs than the soft reports algorithm.

The quality of footprints detected also improves with increasing N and T due to the increased averaging, as shown in Fig. 7(c). Due to the lack of space, we will be focusing only on the probability of detecting all IUs and the average number of extra IUs detected.

The minimum distance required between two components in order to distinguish them as separate incumbents increases with the threshold ζ. Hence, as seen in Fig. 8, reducing ζ increases the probability of detecting all incumbents while also increasing the normalized average number of extra incumbents detected. This effect is more prominent in the minimum tail probability algorithm because it does not use candidate components. We choose $\zeta = 0.99$ to minimize the number of extra IUs detected since the drop in probability of detecting all IUs is hardly 0.01.

C. Assumption of Constant Channel

An important assumption in our algorithm is that the channels between IUs and CRs are constant, i.e., the channel coherence time is greater than the time required to collect N energy measurements. We shall now estimate the measurement time for CRs using 6MHz wide bands, similar to IEEE 802.22 [20], and find the minimum coherence time supported by our algorithm.

Time required to collect a single energy measurement of T samples is $T/6\mu$s, i.e., 5.33μs for $T = 32$. To ensure that different IUs are transmitting in different energy measurements, we assume a fixed time interval between successive measurements. A suitable sensing time interval depends on the medium access control protocol of the IU systems because that controls the transmission duration and interval between successive transmissions. For the sake of argument, if the IUs belong to the commonly used IEEE 802.11 standards, the time interval between successive transmissions IUs is at least 150μs and the transmission duration of each packet is of the order of 100μs. Hence, we assume that measurements occur periodically at 150μs. Hence, $N = 200$ measurements require only 30ms.

Thus, the coherence time of the channel should be at least 30ms. The corresponding Doppler spread should be at most 33.33Hz. For comparison, the maximum Doppler shift in the IEEE 802.22 channel model is 2.5Hz [34] and that for IEEE 802.11n is 6Hz [35].

D. Performance for Different Number of CRs, IUs, and Average Activity

In this section, we would like to show the gain from cooperation between CRs and the effect of collisions between IUs.

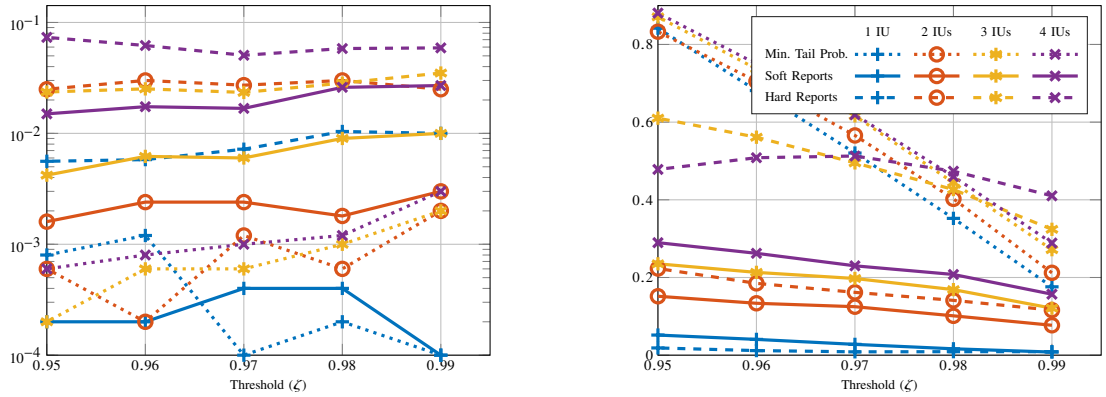
First, Fig. 9 shows that the probability of misdetecting even one IU decreases rapidly as the number of CRs increases. This results from the additional diversity of increasing the number of CRs. The Mahalanobis metric used for the soft reports algorithm also uses this diversity to reduce the number of extra IUs detected as seen in Fig. 9(b). On the other hand, the idea of distinguishing IUs if even one CR has an outlier measurement that the hard reports algorithm inherits in (34) from the minimum tail probability algorithm's distance metric (20) causes both algorithms to detect a large number of extra IUs. As the number of CRs increases, it also reduces the hard report algorithm's ability to combine erroneous reports into the same connected component of $\mathcal{H}$. Hence, it is unable to detect all IUs as the number of CRs increases.

To evaluate the performance of our algorithms as the frequency of collisions increase, we vary the average activity of all IUs. As seen in Fig. 10(a), the soft reports and minimum tail probability algorithm are approximately unaffected by the average activity until it reaches 0.6. Beyond 0.6, the number of collisions between 3 or 4 IUs makes the separability assumption begin to fail. Hence, sum components are wrongly labeled as source components and the probability of detecting all IUs reduces while the number of extra IUs detected increases. The candidate components used for the soft reports algorithm also increases the probability of misdetecting IUs at extremely high average activity. Hence, the soft reports algorithm should be used rather than the minimum tail probability algorithm unless the average activity is very high.

E. Example: Sensing IEEE 802.11n Networks

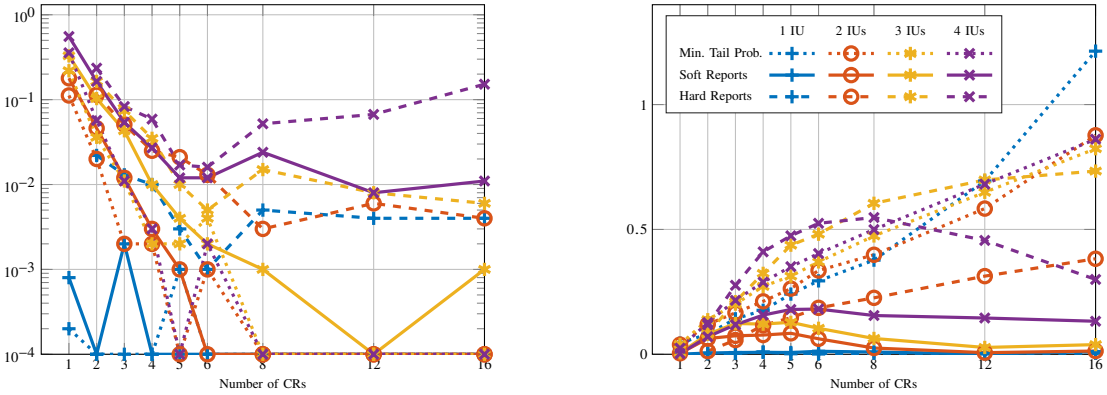
We simulated multiple 802.11n IU networks in NS3 to show the performance of our algorithms when the IUs use a realistic communication system. We see multiple features of the 802.11n MAC protocol affecting the performance of the proposed algorithms in Fig. 11. The MAC protocol reduces the number of collisions. But the problem of channel capture [36]

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking



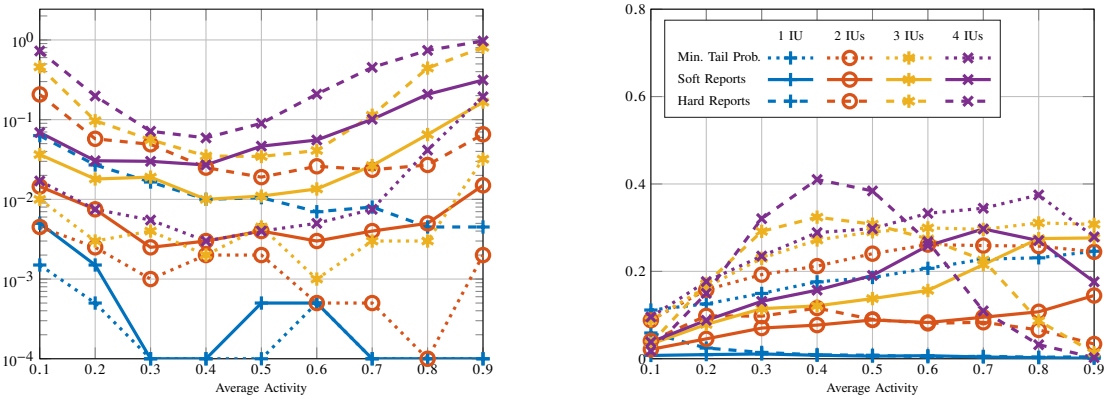
(a) Prob. of misdetecting at least one incumbent (b) Normalized average number of extra incumbents

Fig. 8. Effect on detection performance when threshold ζ is varied from 0.95 to 0.99. Parameters: M = 1 to 4 IUs, K = 4 CRs, 0.4 average activity, N = 200 frames of T = 32 samples each.



(a) Prob. of misdetecting at least one incumbent (b) Normalized average number of extra incumbents

Fig. 9. Effect on detection performance when CRs are increased from 1 to 16 for up to 4 IUs with average activity 0.4. Hard reports algorithm not run for 1 CR system. Parameters: N = 200 frames of T = 32 samples each.



(a) Prob. of misdetecting at least one incumbent (b) Normalized average number of extra incumbents

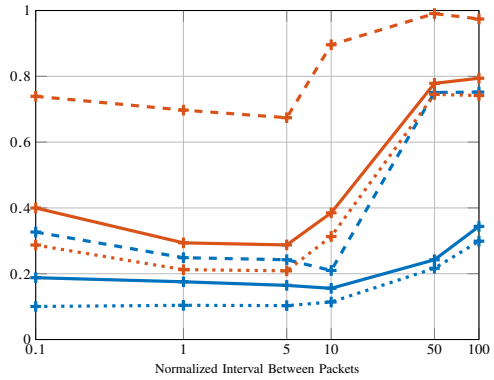
Fig. 10. Performance as a function of average activity for up to 4 incumbents and 4 CRs.

combines with the requirement of a minimum number of active measurements and candidate components to make our proposed algorithms ignore some components. The hard reports algorithms suffers the most because of the same reason as described for Fig. 9: the distance measure between reported label vectors separates labels into separate components if even a single CR makes a mistake and these components get ignored because they have less than 5% active measurements. Instead of trying to optimize this 5% parameter value, we believe we need to design a distance metric that weights all CRs' measurements instead of finding only the outlier.

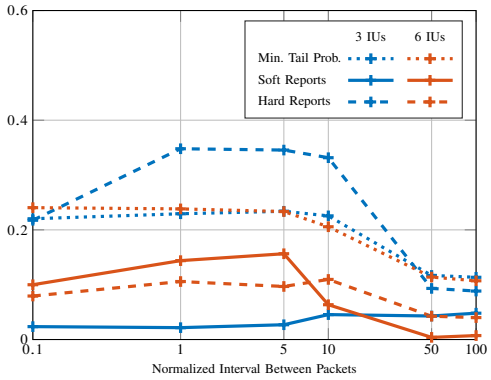
F. Comparison to Existing Work

Apart from [16], methods proposed in existing literature do not consider multiple IUs with spatially overlapping footprints. For comparison on a simpler system, we implemented the distributed boundary estimation (DBE) algorithm proposed in [10] that finds anisotropic non-overlapping footprints. In addition to received energy, the DBE algorithm assumes geographical location for clustering and pair wise communication channels for message exchanges. We simulated a single IU located at the center of a 2000m $\times$ 2000m area with 100 CRs distributed uniformly around it. The footprint detected by the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking



(a) Prob. of misdetecting at least one incumbent. Legend same as (b).



(b) Normalized average number of extra incumbents

Fig. 11. Performance of sensing 802.11n networks as the time interval between arrival of packets at IUs is varied. The time interval between packets is normalized by the transmit duration of a single packet. System: 100 topologies of up to 3 co-channel networks each consisting of 1 AP and 2 STAs were simulated for 3 seconds, packet traces were sampled every 150μs to obtain 100 time sequences of 200 frames each, 6 CRs used for sensing.

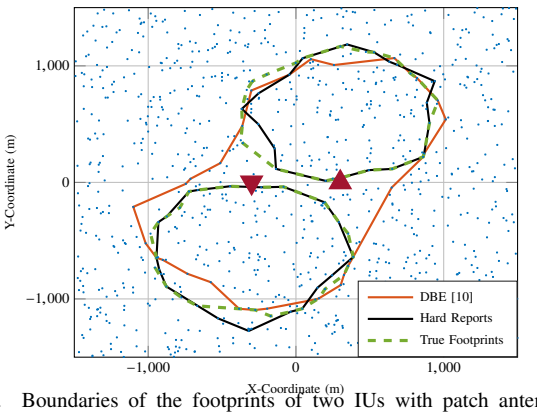


Fig. 12. Boundaries of the footprints of two IUs with patch antennas as detected by the hard reports algorithm and the DBE algorithm proposed in [10]. Dots represent CR locations and triangles represent IU locations.

hard reports algorithm had, on average over 1000 Monte Carlo realizations, 8.16% error while the footprint detected by the DBE algorithm had 24.45% error.

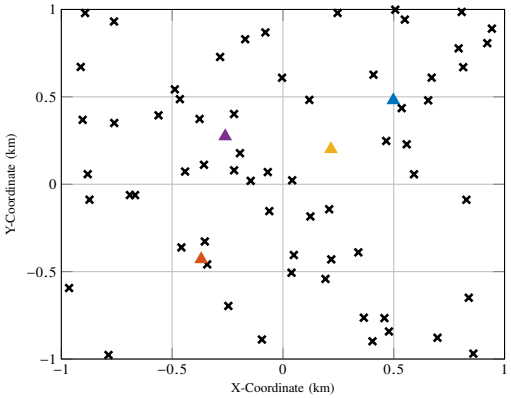
Furthermore, as the example in Fig. 12 shows, the DBE algorithm is not able to distinguish two IUs even though their footprints do not overlap. For Fig. 12, we simulated two IUs with patch antennas directed in opposite directions being sensed by 1000 CRs uniformly distributed in a square area of side 3000m. A channel model with path loss exponent 4 and no fading was used to ensure a compact footprint as modeled in [10].

G. Example Application: Localization of Multiple IUs

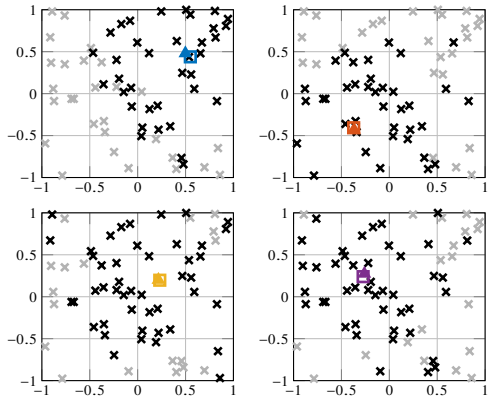
Localization of multiple IUs is a practical application of our proposed algorithms. The source components' sample means as estimated by the soft reports algorithm can be used as input for received energy based localization algorithms. For our example scenario, we use the relative weighted localization algorithm from [37]. We estimate the location of the mth IU as

$$\hat{l}_m \triangleq \sum_{k=1}^{K} \frac{\max\left\{\hat{C}_{\mu(m),k} - \sigma^2_{\nu_k}, 0\right\} p_k}{\sum_{k=1}^{K} \max\left\{\hat{C}_{\mu(m),k} - \sigma^2_{\nu_k}, 0\right\}} \tag{41}$$

where $p_k \in \mathbb{R}^2$ is the location of the kth CR. An example of 4 IUs coexisting in space and spectrum is shown in Fig. 13. Fig. 13(a) shows the locations of all the IUs and CRs and



(a) Locations of all CRs (crosses) and IUs (triangles)



(b) Darker crosses mark CRs in footprint of IU, triangle marks location of IU, and square marks estimated location of IU

Fig. 13. Weighted Centroid Localization of 4 IUs using 200 measurements at 64 CRs. The component means estimated by the soft reports algorithm are used as input to the WCL algorithm.

Fig. 13(b) shows the identified footprints of each IUs and the estimated locations of the corresponding IUs.

H. Computational Complexity

1) Soft Reports Algorithm: Consider the computational complexity of one iteration of the soft reports algorithm. When all the source components of the mixture distribution have been learned, the algorithm reaches Blocks 4-9 only $O(1-\zeta)$ fraction of the measurements. Therefore, the computational

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

TABLE V
COMPUTATIONAL COMPLEXITY OF SOFT REPORTS ALGORITHM

Block	Dominant Operation	Unit	Complexity
1, 4	Computing D_M	Per Component	$O(2^M K^2)$
2, 5	Inverting Covariance	Per Component	$O(2^M K^3)$
3, 6	Exhaustive Search	Per Source Component	$O(M2^{M-1})$
7-11	Linear Search	Per Source Component	$O(M)$

complexity of the algorithm is dominated by the Blocks 1, 2, 3, and 10.

When the mixture distribution is being learnt and some of the true components are classified as candidate components, Blocks 2 and 3 are replaced by Blocks 5 and 6 with the same complexity.

For each block in the soft reports algorithm, Table V lists the operation that dominates the computational complexity and the type of component it iterates over. Based on this table, we conclude that the worst case computational complexity of each iteration of the soft reports algorithm is dominated by the updates of the parameters of the components and is $O(2^M K^3)$.

Thus, the soft reports algorithm has a computational complexity of $O(T2^M K^3)$ for T iterations.

2) Hard Reports Algorithm: The computational complexity for the CRs to learn the source components of the energy they receive is $O(T\sum_{k=}^{K} 2^{M_k})$.

The weakly connected components $\mathcal{J}$ for the graph $\mathcal{H}$ are computed by a breadth first search requiring $O(|\hat{\mathcal{L}}|)$ operations. Consider a component J of the expected $O(2^M)$ weakly connected components. Since each CR is testing a multihypothesis problem with a univariate test statistic, we can assume that each CR is most likely to report one of three hypotheses – the correct hypothesis and the two hypotheses that have the highest overlap with the correct hypothesis in terms of densities. Extending this idea to K CRs and assuming conditional independence amongst their reports, we can approximate $|J| = O(3^K)$.

The breadth first search to check whether $\mathcal{T} \subseteq J$ is a dominating set requires $O(|J|)$ operations. $\mathcal{T}_D$ can be computed simultaneously. Finding the best node to add to $\mathcal{T}$ requires $O(|J|^2)$ operations. This last step is repeated at most $|J|$ times in the worst case but only once or twice in practice. Hence, determining the dominating set of $\mathcal{H}$ requires $O(2^M 9^K)$ operations.

The second stage of the algorithm has a smaller search space, i.e., $\binom{|\hat{\mathcal{L}}_1|}{2} = O(2^M)$ pairs of components. Hence, the computational complexity of this stage is $O(2^{2M-1})$. The complexity of this stage exceeds the complexity of the first stage (in the big-O notation) only if $M > 1 + K\log_2 9$.

In summary, the computational complexity of the hard reports algorithm is dominated by the search for the maximum weight dominating set of $\mathcal{H}$ which requires $O(2^M 9^K)$ operations.

VIII. CONCLUSION AND FUTURE WORK

We have proposed two centralized methods to use received energy at multiple CRs to identify multiple IUs without knowledge of their number, location, communication protocol, and channel conditions. We have shown that by using multidimensional statistical analysis but with assumptions similar to that of energy detection, we can distinguish sources and the spatial footprints of each. The soft reports algorithm improves performance as compared to the minimum tail probability algorithm by better estimating the off-diagonal entries of the covariance matrix of received energy and by introducing the concept of candidate components to reduce the effect of statistical outliers. The hard reports algorithm shows that the computation can be split between the CRs and the fusion center while limiting the communication overhead.

Simulation results showed that the algorithms can detect multiple IUs with high probability for a wide variety of systems. The soft reports algorithm has the best performance for sensing IU networks using slotted ALOHA like MAC protocols because it deals with collisions well. Comparisons to an existing boundary estimation algorithm exemplify the ability of our proposed algorithms to learn non-circular footprints of multiple IUs.

However, NS3 simulations of 802.11n network highlight the need for better methods to reduce the number of extra IUs detected than simply thresholding the number of active measurements for each IU. For the soft reports algorithm, data-driven approaches of covariance estimation such as shrinkage may be useful. The hard reports algorithm would benefit from a weight-based distance metric instead of finding outliers. Finally, the simplicity of the energy measurements make our proposed algorithms amenable to experimental verification.

APPENDIX
DERIVATION OF COVARIANCE OF RECEIVED ENERGY

The covariance of the energy received at two distinct CRs indexed k_1 and k_2 given the activity $a[n]$ can be written as follows, for $t \in \{t_n, \ldots, t_n + T - 1\}$, using the fact that $y_k[t']$ and $y_k[t'']$ are independent if $t' \neq t''$.

$$\begin{aligned}
&\mathrm{Cov}\left(e_{k_1}[n], e_{k_2}[n]|a[n]\right) \\
&= \mathbb{E}\left[e_{k_1}[n]e_{k_2}[n]|a[n]\right] - \mathbb{E}\left[e_{k_1}[n]|a[n]\right]\mathbb{E}\left[e_{k_2}[n]|a[n]\right] \\
&= \mathbb{E}\left[\left(\sum_{t'=t_n}^{t_n+T-1}|y_{k_1}[t']|^2\right)\left(\sum_{t''=t_n}^{t_n+T-1}|y_{k_2}[t'']|^2\right)\middle|a[n]\right] \\
&\quad - \mathbb{E}\left[\left(\sum_{t'=t_n}^{t_n+T-1}|y_{k_1}[t']|^2\right)\middle|a[n]\right]\mathbb{E}\left[\left(\sum_{t''=t_n}^{t_n+T-1}|y_{k_2}[t'']|^2\right)\middle|a[n]\right] \\
&= T\,\mathrm{Cov}\left(|y_{k_1}[t]|^2, |y_{k_2}[t]|^2\middle|a[n]\right) \qquad (42)
\end{aligned}$$

Since the time index t and frame index n do not change in this derivation, we will skip them for ease of reading now. First, we compute the following.

$$\begin{aligned}
&\mathbb{E}\left[|y_{k_1}[t]|^2|y_{k_2}[t]|^2\middle|a[n]\right] \\
&= \mathbb{E}\left[\left|\sum_{m=1}^{M}h_{mk_1}a_m x_m + \nu_{k_1}\right|^2\left|\sum_{m=1}^{M}h_{mk_2}a_m x_m + \nu_{k_2}\right|^2\middle|a\right] \\
&= \mathbb{E}\left[\left|\sum_{m=1}^{M}h_{mk_1}a_m x_m\right|^2\left|\sum_{m=1}^{M}h_{mk_2}a_m x_m\right|^2\middle|a\right] \\
&\quad + \left(\sum_{m=1}^{M}|h_{mk_1}|^2 a_m\sigma_m^2\right)\sigma_{\nu_{k_1}}^2 + \left(\sum_{m=1}^{M}|h_{mk_2}|^2 a_m\sigma_m^2\right)\sigma_{\nu_{k_2}}^2 \\
&\quad + \sigma_{\nu_{k_1}}^2\sigma_{\nu_{k_2}}^2 \qquad (43)
\end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCCN.2017.2710309, IEEE Transactions on Cognitive Communications and Networking

The last equality above uses the definition of $\sigma^2_{\nu_k}$ and (11). The first term may be simplified as:

$$\begin{aligned}
&\mathbb{E}\left[\left|\sum_{m=1}^{M} h_{mk_1} a_m x_m\right|^2 \left|\sum_{m=1}^{M} h_{mk_2} a_m x_m\right|^2 \middle| a\right] \\
&= \mathbb{E}\left[\sum_{m_1,m_2,m_3,m_4=1}^{M} h^*_{m_1k_1} h_{m_2k_1} h^*_{m_3k_2} h_{m_4k_2} x^*_{m_1} x_{m_2} x^*_{m_3} x_{m_4}\right] \\
&= \sum_{\substack{m_1,m_2=1 \\ m_1\neq m_2}}^{M} |h_{m_1k_1}|^2 |h_{m_2k_2}|^2 a_{m_1} a_{m_2} \sigma^2_{m_1} \sigma^2_{m_2} \\
&\quad + \sum_{\substack{m_1,m_2=1 \\ m_1\neq m_2}}^{M} h^*_{m_1k_1} h_{m_2k_1} h^*_{m_2k_2} h_{m_1k_2} a_{m_1} a_{m_2} \sigma^2_{m_1} \sigma^2_{m_2} \\
&\quad + \sum_{m=1}^{M} |h_{mk_1}|^2 |h_{mk_2}|^2 a_m \, \mathbb{E}\left[|x_m|^4\right] \qquad (44)
\end{aligned}$$

Using (43), (44), (11), and $\mathbb{E}\left[|x_m|^2\right] = 2\sigma^4_m$, we can compute the required covariance:

$$\begin{aligned}
&\mathrm{Cov}\left(|y_{k_1}[t]|^2, |y_{k_2}[t]|^2 \middle| a[n]\right) \\
&= \mathbb{E}\left[|y_{k_1}[t]|^2 |y_{k_2}[t]|^2 \middle| a[n]\right] \\
&\quad - \mathbb{E}\left[|y_{k_1}[t]|^2 \middle| a[n]\right] \mathbb{E}\left[|y_{k_2}[t]|^2 \middle| a[n]\right] \\
&= \sum_{m=1}^{M} |h_{mk_1}|^2 |h_{mk_2}|^2 a_m[n] \sigma^4_m \\
&\quad + \sum_{\substack{m_1,m_2=1 \\ m_1\neq m_2}}^{M} h^*_{m_1k_1} h_{m_2k_1} h^*_{m_2k_2} h_{m_1k_2} a_{m_1}[n] a_{m_2}[n] \sigma^2_{m_1} \sigma^2_{m_2}.
\end{aligned} \qquad (45)$$

Therefore, by substituting (45) into (42), we get

$$\begin{aligned}
&\mathrm{Cov}\left(e_{k_1}[n], e_{k_2}[n] | a[n]\right) \\
&= T \sum_{m=1}^{M} |h_{mk_1}|^2 |h_{mk_2}|^2 a_m[n] \sigma^4_m \\
&\quad + T \sum_{\substack{m_1,m_2=1 \\ m_1\neq m_2}}^{M} h^*_{m_1k_1} h_{m_2k_1} h^*_{m_2k_2} h_{m_1k_2} a_{m_1}[n] a_{m_2}[n] \sigma^2_{m_1} \sigma^2_{m_2}.
\end{aligned} \qquad (46)$$

REFERENCES

[1] M. Höyhtyä, A. Mämmelä, M. Eskola, M. Matinmikko, J. Kalliovaara, J. Ojaniemi, J. Suutala, R. Ekman, R. Bacchus, and D. Roberson, "Spectrum Occupancy Measurements: A Survey and Use of Interference Maps," *IEEE Commun. Surv. Tutor.*, vol. 18, no. 4, pp. 2386–2414, Fourthquarter 2016.

[2] B. Li, S. Li, A. Nallanathan, and C. Zhao, "Deep Sensing for Future Spectrum and Location Awareness 5G Communications," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 7, pp. 1331–1344, Jul. 2015.

[3] L. Wei, P. Dharmawansa, and O. Tirkkonen, "Multiple Primary User Spectrum Sensing in the Low SNR Regime," *IEEE Trans. Commun.*, vol. 61, no. 5, pp. 1720–1731, May 2013.

[4] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Netw.*, vol. 9, no. 3, pp. 228 – 248, 2011.

[5] P. Tilghman and D. Rosenbluth, "Inferring Wireless Communications Links and Network Topology from Externals Using Granger Causality," in *IEEE MILCOM*, Nov. 2013, pp. 1284–1289.

[6] I. Bisio, M. Cerruti, F. Lavagetto, M. Marchese, M. Pastorino, A. Randazzo, and A. Sciarrone, "A Trainingless WiFi Fingerprint Positioning Approach Over Mobile Devices," *IEEE Antennas Wirel. Propag. Lett.*, vol. 13, pp. 832–835, 2014.

[7] M. Xie, W. Zhang, and K.-K. Wong, "A geometric approach to improve spectrum efficiency for cognitive relay networks," *IEEE Trans. Wirel. Commun.*, vol. 9, no. 1, pp. 268–281, Jan. 2010.

[8] I. F. Akyildiz, W.-Y. Lee, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *IEEE Commun. Mag.*, vol. 46, no. 4, pp. 40–48, Apr. 2008.

[9] B. A. Jayawickrama, E. Dutkiewicz, M. Mueck, and Y. He, "On the Usage of Geolocation-Aware Spectrum Measurements for Incumbent Location and Transmit Power Detection," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 8177–8189, Oct. 2016.

[10] Y. Zhang, W. P. Tay, K. H. Li, and D. Gaiti, "Distributed Boundary Estimation for Spectrum Sensing in Cognitive Radio Networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 11, pp. 1961–1973, Nov. 2014.

[11] R. Martin and R. Thomas, "Algorithms and bounds for estimating location, directionality, and environmental parameters of primary spectrum users," *IEEE Trans. Wirel. Commun.*, vol. 8, no. 11, pp. 5692–5701, Nov. 2009.

[12] M. Bradonjić and L. Lazos, "Graph-based criteria for spectrum-aware clustering in cognitive radio networks," *Ad Hoc Netw.*, vol. 10, no. 1, pp. 75 – 94, 2012.

[13] R. Eletreby, H. Elsayed, and M. Khairy, "CogLEACH: A spectrum aware clustering protocol for cognitive radio sensor networks," in *CROWNCOM 2014*, Jun. 2014, pp. 179–184.

[14] S. Chaudhari and D. Cabric, "Cyclic weighted centroid localization for spectrally overlapped sources in cognitive radio networks," in *IEEE GLOBECOM*, Dec. 2014, pp. 935–940.

[15] J. Wang and D. Cabric, "A cooperative DoA-based algorithm for localization of multiple primary-users in cognitive radio networks," in *IEEE GLOBECOM*, Dec. 2012, pp. 1266–1270.

[16] M. Laghate and D. Cabric, "Identifying the presence and footprints of multiple incumbent transmitters," in *2015 49th Asilomar Conference on Signals, Systems and Computers*, Nov. 2015, pp. 146–150.

[17] D. Donoho and V. Stodden, "When Does Non-Negative Matrix Factorization Give a Correct Decomposition into Parts?" in *Advances in Neural Information Processing Systems*. MIT Press, 2004, pp. 1141–1148.

[18] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, vol. 55, no. 4, pp. 523–531, Apr. 1967.

[19] Z. Quan, S. Cui, and A. Sayed, "Optimal Linear Cooperation for Spectrum Sensing in Cognitive Radio Networks," *IEEE J. Sel. Top. Signal Process.*, vol. 2, no. 1, pp. 28–40, Feb. 2008.

[20] "IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Policies and procedures for operation in the TV Bands," *IEEE Std 80222-2011*, pp. 1–680, Jul. 2011.

[21] F. Digham, M.-S. Alouini, and M. K. Simon, "On the Energy Detection of Unknown Signals Over Fading Channels," *IEEE Trans. Commun.*, vol. 55, no. 1, pp. 21–24, Jan. 2007.

[22] Z. Galil, "Efficient Algorithms for Finding Maximum Matching in Graphs," *ACM Comput Surv*, vol. 18, no. 1, pp. 23–38, Mar. 1986.

[23] D. Saunders, "Weighted maximum matching in general graphs," http://www.mathworks.com/matlabcentral/fileexchange/42827-weighted-maximum-matching-in-general-graphs, Aug. 2015.

[24] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum Likelihood from Incomplete Data via the EM Algorithm," *J. R. Stat. Soc. Ser. B Methodol.*, vol. 39, no. 1, pp. 1–38, 1977.

[25] D. Ververidis and C. Kotropoulos, "Gaussian Mixture Modeling by Exploiting the Mahalanobis Distance," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 2797–2811, Jul. 2008.

[26] J. J. Verbeek, N. Vlassis, and B. Kröse, "Efficient Greedy Learning of Gaussian Mixture Models," *Neural Computation*, vol. 15, no. 2, pp. 469–485, Feb. 2003.

[27] K. Fukunaga and T. E. Flick, "Estimation of the Parameters of a Gaussian Mixture Using the Method of Moments," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-5, no. 4, pp. 410–416, Jul. 1983.

[28] N. Gravin, J. Lasserre, D. V. Pasechnik, and S. Robins, "The Inverse Moment Problem for Convex Polytopes," *Discrete Comput. Geom.*, vol. 48, no. 3, pp. 596–621, 2012.

[29] A. Anandkumar, D. Hsu, and S. Kakade, "A Method of Moments for Mixture Models and Hidden Markov Models," in *Proceedings of the 25th Annual Conference on Learning Theory (COLT)*, Jun. 2012.

[30] S. C. Brubaker and S. Vempala, "Isotropic PCA and Affine-Invariant Clustering," in *Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science*, ser. FOCS '08, Washington, DC, USA, 2008, pp. 551–560.

[31] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.

[32] C. Namislo, "Analysis of Mobile Radio Slotted ALOHA Networks," *IEEE J. Sel. Areas Commun.*, vol. 2, no. 4, pp. 583–588, Jul. 1984.

[33] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electron. Lett.*, vol. 27, no. 23, pp. 2145–2146, Nov. 1991.

[34] "WRAN Channel Modeling," IEEE, Tech. Rep. 802.22-05/0055r7, 2005.

[35] V. Erceg, L. Schumacher, P. Krytsi, and et. al, "TGn Channel Model," IEEE, Tech. Rep. 802.11-03/940r4, May 2004.

[36] C. Ware, J. Judge, J. Chicharo, and E. Dutkiewicz, "Unfairness and capture behaviour in 802.11 adhoc networks," in *IEEE International Conference on Communications. ICC 2000.*, vol. 1, 2000, pp. 159–163 vol.1.

[37] J. Wang, P. Urriza, Y. Han, and D. Cabric, "Weighted Centroid Localization Algorithm: Theoretical Analysis and Distributed Implementation," *IEEE Trans. Wirel. Commun.*, vol. 10, no. 10, pp. 3403–3413, 2011.