Distributed boundary estimation for spectrum sensing in cognitive radio networks

Zhang, Yi; Tay, Wee Peng; Li, Kwok Hung; Gaïti, Dominique

2014

Zhang, Y., Tay, W. P., Li, K. H., & Gaïti, D. (2014). Distributed boundary estimation for spectrum sensing in cognitive radio networks. IEEE Journal on Selected Areas in Communications, 32(11), 1961-1973.

https://hdl.handle.net/10356/81706

https://doi.org/10.1109/JSAC.2014.1411RP08

Distributed Boundary Estimation for Spectrum Sensing in Cognitive Radio Networks

Yi Zhang, *Student Member, IEEE*, Wee Peng Tay, *Member, IEEE*,
Kwok Hung Li, *Senior Member, IEEE*, Dominique Gaïti, *Member, IEEE*

Abstract

In a cognitive radio network, a primary user (PU) shares its spectrum with secondary users (SUs) temporally and spatially, while allowing for some interference. We consider the problem of estimating the no-talk region of the PU, i.e., the region outside which SUs may utilize the PU's spectrum regardless of whether the PU is transmitting or not. We propose a distributed boundary estimation algorithm that allows SUs to estimate the boundary of the no-talk region collaboratively through message passing between SUs, and analyze the trade-offs between estimation error, communication cost, setup complexity, throughput and robustness. Simulations suggest that our proposed scheme has better estimation performance and communication cost trade-off compared to several other alternative benchmark methods, and is more robust to SU sensing errors, except when compared to the least squares support vector machine approach, which however incurs a much higher communication cost.

Index Terms

Cognitive radio, boundary estimation, spectrum sensing.

I. Introduction

A cognitive radio (CR) network improves radio spectrum utilization by permitting unlicensed secondary users (SUs) to access the same spectrum when the licensed primary users (PUs) are not using it, or when SU transmissions do not interfere significantly with the PUs. Various

This research was supported in part by the MOE AcRF Tier 1 Grant RG25/10. Y. Zhang, W. P. Tay, and K. H. Li, are with the Nanyang Technological University, Singapore. E-mail: yzhang29@e.ntu.edu.sg, {wptay,ekhli}@ntu.edu.sg. Dominique Gaïti is with University of Technology of Troyes, France. E-mail: dominique.gaiti@utt.fr.

spectrum sensing methods have been proposed, including centralized [1], distributed [2] and relay-assisted cooperative detection schemes [3].

In this paper, we consider the *spatial* usage diversity of the PU by letting the PU fix an interference temperature limit that allows for interference from SUs in its licensed spectrum below a threshold [4], [5]. This translates to a no-talk region around the PU, in which SUs opportunistically transmit only if the PU is not transmitting [4], [6]. SUs outside of this no-talk region can transmit regardless of whether the PU is active or not. Our main goal is to develop a distributed algorithm that allows SUs to cooperatively determine the boundary of the no-talk region.

In [6]–[8], different spectrum sharing regions, including a primary exclusive region and the no-talk region, are defined. However, all these works assume that the propagation path loss between the PU and SUs are isotropic, and all regions are assumed to be circular. Bounds on the radius of each region are given based on interference and outage considerations, which are characterized in terms of propagation parameters like path loss exponents. In practice, the propagation environment may be very difficult to model quantitatively, and the no-talk region is unlikely to be circular. Therefore, in this work, we develop boundary estimation methods for the no-talk region without relying on extensive assumptions about the shape of the region.

Boundary estimation is widely used in different sensor networking applications, and has been extensively studied. Methods based on node degrees [9], connectivity information [10], and topology information [11] have been proposed to estimate the coverage region of a sensor network. Although the definition of a no-talk region was comprehensively addressed in [4], little work has been done to estimate the boundary of this region. A classifier-based cooperative boundary detection algorithm for estimating the no-talk region using support vector machines (SVM) [12] has been proposed in [13]. A computational geometry method based on convex hulls has also been utilized for boundary estimation in [13]. All these works however assume that sensors send their local information to a fusion center, and boundary estimation is performed at the fusion center. In a cognitive radio network, constraints on energy and bandwidth usually restrict SUs from communicating with a single fusion center effectively. Localized edge detection algorithms based on statistical, image processing and classification methods have been proposed in [14] to allow a sensor to locally decide whether or not it is located on or near a boundary. A distributed Bayesian algorithm has also been proposed to determine event regions [15], [16].

These methods however do not make use of cooperation between sensors. A hierarchical tree-based estimation method using recursive dyadic partitioning [17] and a dynamic boundary tracking algorithm that combines spatial and temporal estimation techniques [18] have been proposed for boundary estimation in ad-hoc networks. However, this method is again centralized, and does not consider the smoothness of the estimated boundary.

In this paper, we consider the cooperative estimation of the PU's no-talk region by exploiting local communications amongst SUs. Our main contributions are the following:

1) We propose a distributed boundary estimation method based on the distributed learning framework of [19], and with additional smoothness constraints. Sensors outside the estimated no-talk region are allowed to transmit even if the PU is transmitting.
2) We provide approximate theoretical bounds for the communication cost incurred by our proposed method and the expected estimation error, so that the approximate optimal SU density can be inferred. This is useful for randomly allocating SUs to estimate the no-talk regions of multiple PUs transmitting over different frequency bands. We note that our theoretical performance analysis is not considered in [19], and to the best of our knowledge, is new.
3) We derive order bounds for the setup complexity of our proposed method, and expressions for the throughput achievable by the PU and SUs.
4) Simulations suggest that our proposed boundary estimation algorithm have better trade-offs in the throughput and setup communication cost than various other boundary estimation algorithms in the literature, and is more robust to SU sensing errors except when compared to the centralized least squares SVM (LS-SVM) method, which however incurs a much higher communication cost.

Our method allows better spatial usage of the spectrum and improves the overall system throughput, albeit at the cost of estimating the boundary. For a stationary PU, this is a fixed cost that does not contribute significantly to the overall operational energy cost. An example is the use of CR systems in the Internet-of-Things framework [20], where devices like electrical appliances are fixed and CRs in the devices allow opportunistic use of the cluttered spectrum.

The rest of this paper is organized as follows. In Section II, we introduce our system model and problem formulation. In Section III, we propose a distributed boundary estimation algorithm for estimating the boundary of $\mathcal{R}$. In Section IV, we analyze the trade-offs between communication

cost and estimation error of our boundary estimation method for a Poisson field of SUs, and determine its setup complexity and throughput. Simulation results are provided in Section V. Finally, we conclude in Section VI.

II. System Model

Suppose that there is one PU and N SUs in a bounded region $A \subset \mathbb{R}^d$.[1] We say that the PU is active if it is transmitting in its licensed spectrum. Suppose that the PU is located at x_0. We assume that all wireless channels are symmetric, and define the no-talk region [4] of the PU to be the set $\mathcal{R} = \{x \in \mathbb{R}^d : P_0 - L(x, x_0) > \theta_0\}$, where P_0 is the transmit power of the PU, $L(x, x_0)$ is the average propagation loss function between the PU and a SU located at x, and θ_0 is a fixed threshold. The average propagation loss can be modeled as $L(x, x_0) = l(\|x - x_0\|) + S(x, x_0) + F(x, x_0)$,[2] where $l(\|x - x_0\|)$ is the power attenuation due to the distance $\|x - x_0\|$ between a SU at location x and the PU at location x_0, $S(x, x_0)$ represents the average shadowing effect, and $F(x, x_0)$ is the average power loss due to multipath fading. We suppose that the PU can tolerate an average interference below the fixed threshold θ_0 so that SUs outside of $\mathcal{R}$ can utilize the PU spectrum regardless of whether it is active or not. SUs within the no-talk region $\mathcal{R}$ are required to refrain from using the PU spectrum if the PU is transmitting. Note that the threshold θ_0 is chosen to include a safety margin or budget for the propagation loss due to shadowing and fading, and other parameters like the average density of SUs. The reader is referred to [4] for a detailed discussion of the different considerations involved in defining the no-talk region of a PU.

In this paper, we aim to estimate the no-talk region $\mathcal{R}$, or equivalently the boundary of $\mathcal{R}$, in order to facilitate spatial spectrum sharing between the PU and SUs. The average propagation loss $L(x, x_0)$ for a SU at position x depends on various factors including the terrain, the type and number of reflectors and attenuators between the PU and SU, and other ambient environmental factors. The propagation loss function is thus difficult to determine to good accuracy in practice, and therefore we assume that $L(x, x_0)$ is unknown, and adopt a learning approach to estimate the region $\mathcal{R}$ solely based on the received power at the SUs. We make the following assumptions.

[1] In most applications, $d = 2$ or 3, which corresponds to SUs scattered over a geographical region or in a building respectively.

[2] All power quantities are expressed in dB.

Assumption 1:

(a) Communications by SUs are over relatively shorter distances than the PU, and hence the transmit power of each SU is at most P_0. Multiple SUs can share the PU's spectrum spatially (see Figure 1).[3]

(b) The region $\mathcal{R}$ is compact, and has a smooth[4] boundary.

(c) Time can be discretized into intervals and the PU is active in each interval with known probability $\pi \in [0, 1]$, independently across intervals.[5]

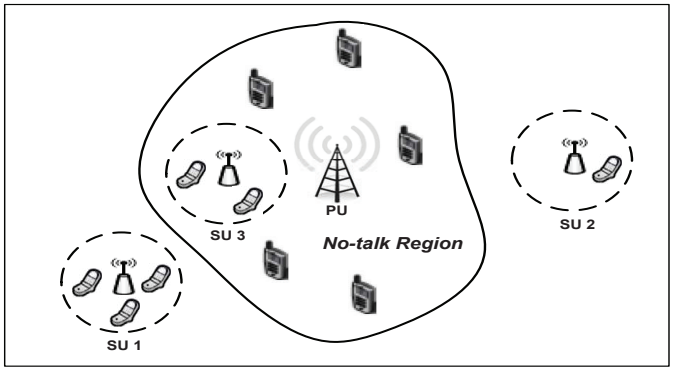

Fig. 1. Spatial spectrum sharing between PU and multiple SUs. SU 1 and 2 can use the licensed spectrum of the PU without spectrum sensing. SU 3 can only utilize the spectrum when the PU is inactive.

We let each SU sample the PU licensed spectrum over a sufficiently long calibration phase in order to perform boundary estimation of $\mathcal{R}$. We assume that $\mathcal{R}$ has a smooth boundary in Assumption 1(b) to avoid the case where a temporary degradation in the channel between the PU and a SU during the calibration phase may incur a large estimation error. This assumption is also valid in most practical situations, except when there are strong attenuators close to the boundary, in which case our solution leads to an estimated no-talk region larger than the actual one. In our problem formulation (cf. Section III-B), we will not impose Assumption 1(b) strictly, but rather adopt a simpler constraint to approximate it. In addition, for practicality, we require that the estimation algorithm is distributed, with each SU having access only to local information (its own observations and information from its neighbors).

[3]Various spectrum sharing schemes have been described in [21].

[4]Formally, this means that the boundary is parameterizable and differentiable in that parameter.

[5]Although we restrict our analysis to the case where the PU active probability π is the same across intervals, our analysis can be easily generalized to the case where each interval has a different PU active probability.

Suppose that SU i is located at x_i, for $i = 1, \ldots, N$, and suppose that the calibration period for SU i is divided into J observation intervals, where the PU is active with probability π during each interval independently (cf. Assumption 1(c)). Let $Y_i[j]$ be the signal sample obtained by SU i in interval j, where $j = 1, \ldots, J$. If the PU is inactive during interval j, we have

$$Y_i[j] = W_i[j],$$

where the noise variables $W_i[j]$ are independent zero mean Gaussian random variables with variance σ_W^2. If the PU is active during interval j, we have

$$Y_i[j] = X_i[j] + W_i[j],$$

where $X_i[j]$ is the PU signal, which is assumed to be a Gaussian random variable with zero mean, and independent of $W_i[j]$. The variance or power of $X_i[j]$ is then given by $\mu_i = P_0 - L(x_i, x_0)$.

Let $\mathcal{H}_0 : \mu_i \leq \theta_0$ and $\mathcal{H}_1 : \mu_i > \theta_0$ be the hypotheses that SU i is outside and within the no-talk region $\mathcal{R}$, respectively. For the sake of generality, we assume that the PU signal modulation scheme is unknown to the SUs. Therefore, SUs are constrained to use energy detection methods [4], [22] in order to perform the hypothesis test. For this purpose, SU i forms the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |Y_i[j]|^2$ and uses the following threshold rule to determine the hypothesis:

$$T_i \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \theta, \tag{1}$$

where θ is chosen so that the false alarm probability is below a predefined threshold $\alpha \in (0, 1)$. Since μ_i is unknown a priori, we need to make further approximations in order to determine θ. The mean of T_i is given by

$$m(\mu_i) = \pi\mu_i + \sigma_W^2,$$

and applying Wald's identity [23], we obtain after some algebra, the variance of T_i is

$$\eta(\mu_i) = \frac{1}{J}\left(\mu_i^2\pi(3-\pi) + 2\sigma_W^2(\sigma_W^2 + 2\pi)\right).$$

Assuming that J is sufficiently large, the central limit theorem [23] allows us to approximate the distribution of T_i as a Gaussian distribution with mean $m(\mu_i)$ and variance $\eta(\mu_i)$. Since both $m(\cdot)$ and $\eta(\cdot)$ are increasing functions, the threshold θ can now be chosen to ensure that the false alarm constraint in the worst case situation is satisfied, by setting

$$\mathcal{Q}\left(\frac{\theta - m(\theta_0)}{\sqrt{\eta(\theta_0)}}\right) = \alpha,$$

where $\mathcal{Q}(\cdot)$ is the complementary cumulative distribution function for the standard normal distribution. We note that the test (1) and the choice of θ do not require knowledge of the PU transmit power P_0 or the locations of the PU and SUs.

When there are multiple PUs transmitting in the same spectrum, the no-talk region is the union of all PUs' no-talk regions. Suppose there are $N' > 1$ PUs, and that all PU signals are uncorrelated. Let $Y_{ip}[j]$ be the signal sample received by SU i from the PU p in the interval j. For a sufficiently large J and $p \neq p'$, we have $\frac{1}{J}\sum_{j=1}^{J} Y_{ip}[j]Y_{ip'}[j] \approx 0$, and the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |\sum_{p=1}^{N'} Y_{ip}[j]|^2$ can be approximated as $\frac{1}{J}\sum_{p=1}^{N'}\sum_{j=1}^{J} |Y_{ip}[j]|^2$. A threshold for T_i similar to that in (1) can be found to determine if SU i is within the no-talk region of at least one PU.

Let u_i be the decision of SU i for the test (1), where $u_i = -1$ if it decides in favor of $\mathcal{H}_0$, and $u_i = 1$ otherwise. Recall that x_i is the position of SU i, and let $y_i = (x_i, u_i)$, for $i = 1, \ldots, N$. Our aim is to learn a function $f : \mathbb{R}^d \mapsto \mathbb{R}$ based on the collection of pairs $y = \{y_i : i = 1, \ldots, N\}$, so that $f(x) \geq 0$ or $f(x) < 0$ if a SU location x is inside or outside the region $\mathcal{R}$ respectively, and with $\{x : f(x) = 0\}$ corresponding to the boundary of $\mathcal{R}$. In the same spirit as statistical learning theory [24], we can regard each y_i as being drawn independently from the same joint distribution $p(x, u)$ (which is unknown because the path loss $L(x, x_0)$ is unknown). Then, in estimating the boundary of $\mathcal{R}$, we hope to obtain a function f with small generalization error

$$\mathcal{E} = \mathbb{E}[(f(X) - U)^2], \tag{2}$$

where (X, U) has joint distribution $p(x, u)$. If $\mathcal{E}$ is large, the throughput at the PU deteriorates because of interference from SUs that wrongly believe themselves to be in $\mathcal{R}^c$.[6] Therefore, we are interested to study the trade-offs in throughputs with $\mathcal{E}$ and the communication cost of performing the boundary estimation.

In a centralized estimation algorithm, the data y is sent to a fusion center, which trains a global function. Such centralized algorithms suffer from several disadvantages, including the need to select a site for the fusion center, the susceptibility of the whole network to a single point of failure at the fusion center, the need for significant processing power and memory storage

[6]We use $\mathcal{R}^c = A \backslash \mathcal{R}$ to denote the complement of $\mathcal{R}$ in the region of interest A.

at the fusion center, and the use of long range communications as the area A becomes large. In this paper, we consider distributed algorithms, in which each SU communicates only with neighboring SUs to collaboratively estimate the boundary of $\mathcal{R}$.

For the convenience of the reader, we list some commonly used notations in Table I. Some of these notations have been defined in this section, while the remaining ones will be defined formally in the sequel where they first appear. In addition, we adopt the following definitions. For SU i and cluster C, we use $i \in C$ to mean $x_i \in C$. The number of SUs in y belonging to C is given by $|C|$. The indicator function $\mathbf{1}\{S\}$ equals to 1 if the statement S is true and 0 otherwise.

TABLE I

SUMMARY OF NOTATIONS USED

Symbol	**Definition**
A	region of interest containing the PU and all SUs
$\mathcal{R}$	no-talk region of PU
N	number of SUs in region of interest A
δ	broadcast range of a SU
p_h	probability of a SU to become a cluster head
γ	threshold to determine if a cluster is a boundary cluster
M	number of boundary clusters
C_j	the jth boundary cluster, $j = 1, \ldots, M$
$\mathcal{B}$	set of boundary clusters $\{C_1, \ldots, C_M\}$
$\mathcal{N}(C_j)$	set of neighboring clusters of C_j
f_{C_j}	local boundary estimation function of cluster C_j

III. Distributed Boundary Estimation

In this section, we propose a distributed boundary estimation algorithm that determines the boundary of the set $\mathcal{R}$ based on message passing between SUs. The SUs are grouped into clusters, and most communications are over relatively short ranges within clusters. Each cluster has a SU that serves as the cluster head. The cluster head communicates with SUs inside its cluster, performs most of the necessary computations required for distributed estimation of the boundary, and communicates with other cluster heads. Cluster heads thus expend more energy than typical SUs inside the cluster. Incentives can be designed to compensate cluster heads; an example being given higher priority to access the spectrum. Such incentive mechanisms are out of the scope of our current work, and will not be discussed here.

Our distributed boundary estimation procedure consists of the following steps.

(i) *Formation of clusters.* Each SU independently nominates itself to be a cluster head with probability p_h. A cluster head broadcasts a message over a control channel to all SUs within a distance δ to inform them of their inclusion into the cluster. To avoid collisions amongst cluster heads, a carrier sense multiple access protocol [25] is used. Note that a cluster head can also belong to another cluster, and a SU can belong to multiple clusters.

(ii) *Boundary cluster identification.* We design a metric to identify those clusters that lie close to the boundary of the set $\mathcal{R}$. We call these the boundary clusters.

(iii) *Distributed boundary estimation.* Messages are exchanged between members of a boundary cluster and its cluster head. In addition, messages are exchanged between cluster heads of neighboring boundary clusters to collaboratively estimate the boundary of $\mathcal{R}$.

In the following subsections, we describe steps (ii) and (iii) in detail.

A. Boundary Cluster Identification

Let C be a cluster, and $\mathcal{U}^- = \frac{1}{|C|}\sum_{i\in C}\mathbf{1}\{u_i = -1\}$ to be the fraction of SUs in cluster C with $u_i = -1$. The clusters within $\mathcal{R}$ have a higher probability of $1-\mathcal{U}^-$ being much larger than $\mathcal{U}^-$, while the reverse is true for clusters that are far from the PU. To identify those clusters that are close to the boundary of $\mathcal{R}$, we let $\mathcal{S} = \max(\mathcal{U}^-, 1-\mathcal{U}^-)$, and say that C is a boundary cluster if and only if $\mathcal{S} \leq \gamma$, where γ to be a fixed threshold. If C is not a boundary cluster, the cluster head declares it to be within $\mathcal{R}$ if $\mathcal{U}^- < 1/2$, and outside vice versa. We call those clusters in the former class *inside* clusters, and those in the latter class *outside* clusters.

B. Distributed Boundary Estimation

To learn a function f that can be used to determine if a new SU location x (not necessarily belonging to the training data y) is within or without $\mathcal{R}$, we consider the following approach. A SU at location x queries its cluster head to check the types of cluster it belongs to (recall that a SU may belong to multiple clusters). If it belongs to an inside cluster, we let $f(x) = 1$ and declare that it belongs to $\mathcal{R}$. If it does not belong to any inside clusters, and it belongs to a boundary cluster, it uses a local function, which we describe below, to determine its location status. Finally, if it is not within any inside or boundary clusters, we let $f(x) = -1$, and declare that the SU is in $\mathcal{R}^c$.

Let $\mathcal{B} = \{C_1, \ldots, C_M\}$ be the set of boundary clusters. The boundary clusters collaboratively estimate the boundary of $\mathcal{R}$ based on local information and message exchanges between cluster heads. We use the reproducing kernel Hilbert space (RKHS) [19], [26] formulation to obtain a function that distinguishes a location x to be inside or outside $\mathcal{R}$. However, since we do not assume that there is a central authority to perform the estimation, we consider instead finding a collection $\{f_{C_j}\}$ of local functions, each corresponding to a boundary cluster. If x is not within an inside cluster and it belongs to a boundary cluster, we let the estimation function $f(x)$ take the value $f_C(x)$ where C is chosen randomly from the set of boundary clusters containing x.

Let H_K be a RKHS corresponding to a kernel $K(\cdot, \cdot)$ that serves as a similarity measure between two SU locations. We restrict to kernels that are radial basis functions (RBF), i.e., those kernels that can be expressed as functions of the Euclidean distance between two SUs. For each $C_j \in \mathcal{B}$, let $\mathcal{N}(C_j)$ be the set of indices k with $j \neq k$ and $|C_k \cap C_j| \neq 0$. We call those clusters in $\mathcal{N}(C_j)$ the neighboring clusters of C_j. Our goal is to

$$\min \quad \sum_{i \in \cup_{j=1}^{M} C_j} (z_i - u_i)^2 + \sum_{m=1}^{M} \nu_m \|f_{C_m}\|_{H_K}^2 + \sum_{m=1}^{M} \sum_{k \in \mathcal{N}(C_m)} \eta_m \epsilon_{m,k}^2 \tag{3}$$

subject to

$$f_{C_m} \in H_K, \ \forall \ C_m \in \mathcal{B}, \tag{4}$$

$$z_i = f_{C_m}(x_i), \ \forall \ i \in C_m, C_m \in \mathcal{B}, \tag{5}$$

$$\epsilon_{m,k} = \frac{1}{|C_m|} \sum_{i \in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|} \sum_{i \in C_k} f_{C_k}(x_i), \ \forall \ k \in \mathcal{N}(C_j), C_j \in \mathcal{B}, \tag{6}$$

where $\|\cdot\|_{H_K}$ is the norm of H_K, and ν_m, η_m, for $j = 1, \ldots, M$, are positive constants. The minimization in (3) is over all variables z_i, f_{C_m} and $\epsilon_{m,k}$. The constraints (4) require that the local classifier f_{C_m} from each boundary cluster C_m is chosen from the RKHS H_K. The constraints (5) ensure that if a SU belongs to multiple boundary clusters, the classification result remains the same regardless of the local classifier used. Finally, the constraints (6) ensure that the estimated boundary is smooth (cf. Assumption 1(b)).

The kernel least squares minimization problem (3) is similar to that proposed in [19], which considers a general distributed learning framework, but without additional constraints like (6). The reference [19] also provides a distributed method to iteratively obtain the optimizers $\{f_{C_m}\}$ by message exchanges between cluster heads. In the following, we show that their distributed

algorithm can be adapted to our minimization problem (3). Our argument is similar to that in [19], and treats the minimization in (3) as projections onto closed convex subspaces of a Hilbert space. This can be done because of the successive orthogonal projection (SOP) theorem [27], which we state below without proof.

Theorem 1: Let $\{\Lambda_m\}_{m=1}^M$ be a set of closed, convex, and affine subsets of a RKHS H, and whose intersection $\Lambda = \bigcap_{m=1}^M \Lambda_m$ is nonempty. For any $v_0 \in H$, let v^* be the orthogonal projection of v_0 onto Λ, and for all $n \geq 1$, let v_n be the orthogonal projection of v_{n-1} onto $\Lambda_{(n \mod M)}$. Then, $\lim_{n\to\infty} \|v_n - v^*\| = 0$.

Suppose that $S = \sum_{m=1}^M |\mathcal{N}(C_m)|$ is the total number of variables $\epsilon_{m,k}$ where $m = 1, \ldots, M$ and $k \in \mathcal{N}(C_m)$. Let $H = \mathbb{R}^N \times H_K^M \times \mathbb{R}^S$. In the sequel, to avoid cluttered notations, we let $v = ((z_i), (f_m), (\epsilon_{m,k})) \in H$ to denote an element from H with the understanding that the index i runs from 1 to N, the index m runs from 1 to M, and the index $k \in \mathcal{N}(C_m)$ for each m. We let H be a Hilbert space by letting the squared norm of v be

$$\|v\|_H^2 = \sum_{i=1}^N |z_i|^2 + \sum_{m=1}^M \nu_m \|f_{C_m}\|_{H_K}^2 + \sum_{m=1}^M \sum_{k\in\mathcal{N}(C_m)} \eta_m \epsilon_{m,k}^2.$$

For each $j = 1, \ldots, M$, let

$$\Lambda_j = \Big\{((z_i), (f_m), (\epsilon_{m,k})) \in H : z_i = f_{C_m}(x_i), \forall\ i \in C_m, \text{ and}$$
$$\epsilon_{m,k} = \frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|} \sum_{i\in C_k} f_{C_k}(x_i), \forall\ k \in \mathcal{N}(C_m)\Big\}.$$

It can be shown that Λ_j is a closed subspace of H. Then, the minimization problem (3) is equivalent to finding the projection of $(u_1, \ldots, u_N, 0, \ldots, 0)$ onto the closed and convex set $\Lambda = \cap_{j=1}^M \Lambda_j$.

As pointed out in [19], instead of directly finding the projection onto Λ, Theorem 1 allows us to iteratively project onto each Λ_m, for $m = 1, \ldots, M$. The SOP algorithm first finds the projection v_1 of $(u_1, \ldots, u_N, 0, \ldots, 0)$ onto Λ_1, then finds the projection of v_1 onto Λ_2, and so on. Projections are performed over all Λ_m, $m = 1, \ldots, M$, with multiple iterations over the indices m. Suppose that at an iteration, we seek to project $v = ((\tilde{z}_i), (\tilde{f}_j), (\tilde{\epsilon}_{j,k}))$ onto Λ_m. This

is equivalent to

$$\min \quad \sum_{i\in C_m} (f_{C_m}(x_i) - \tilde{z}_i)^2 + \nu_m \left\| f_{C_m} - \tilde{f}_{C_m} \right\|_{H_K}^2 + \eta_m \sum_{k\in\mathcal{N}(C_m)} (\epsilon_{m,k} - \tilde{\epsilon}_{m,k})^2 \qquad (7)$$

subject to

$$f_{C_m} \in H_K,$$
$$\epsilon_{m,k} = \frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|} \sum_{i\in C_k} f_{C_k}(x_i), \ \forall\ k \in \mathcal{N}(C_m).$$

The minimization in (7) is over f_{C_m} and $\epsilon_{m,k}$, and involves only data from C_m and its neighboring clusters. It is thus a *local* optimization problem. Suppose that $(f^*_{C_m}, (\epsilon^*_{m,k})_{k\in\mathcal{N}(C_m)})$ is the optimizer for (7). The projected point is then given by $v^* = ((z^*_i), (f^*_j), (\epsilon^*_{j,k}))$, where

$$z^*_i = \tilde{z}_i \text{ if } i \notin C_m, \text{ and } z^*_i = f^*_{C_m}(x_i) \text{ if } i \in C_m,$$
$$f^*_{C_j} = \tilde{f}_{C_j} \text{ and } \epsilon^*_{j,k} = \tilde{\epsilon}_{j,k} \text{ if } j \neq m.$$

The messages that cluster C_m passes to a neighboring cluster C_k are $\{z^*_i : i \in C_m \cap C_k\}$ and $\frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i)$, where the first message represents its current best estimate of $\{u_i : i \in C_m \cap C_k\}$ subject to the constraints (4)-(6), and serves as the "training labels" [19] for the SUs in both clusters. The second message encodes the average value achieved by $f^*_{C_m}$, and allows C_k to adjust its own classifier to improve the smoothness of the estimated boundary. The following result is a direct consequence of the Representer Theorem [28], and its proof is omitted. It characterizes the form of the optimal solution $f^*_{C_m}$ for (7).

Proposition 1: For each $C_m \in \mathcal{B}$, the optimal solution to the minimization problem (7) is given by

$$f^*_{C_m}(x) = \sum_{i\in C_m} \beta_{m,i} K(x, x_i).$$

Furthermore, if the kernel $K(\cdot,\cdot)$ is a radial basis function, the computation of $f_{C_m}(x)$ requires only knowledge of $\|x - x_i\|$, for all $i \in C_m$.

From (7) and Proposition 1, to train the classifier for a cluster $C_m \in \mathcal{B}$, we require the cluster head to know $\|x_i - x_j\|$, for all $i, j \in C_m$. This can be obtained using various ranging techniques. Examples include methods in which each SU i broadcasts a pilot signal with known transmit power, or exchange messages with timestamps [29]. Our distributed boundary estimation

algorithm is formally stated in Algorithm 1, which we call the DBE algorithm. The following proposition shows that the classifiers in the DBE algorithm converges.

Proposition 2: For each C_m, where $m = 1, \ldots, M$, the sequence $(f_{C_m}^t)$ in line 9 of the DBE algorithm converges as number of iterations $t \to \infty$.

Proof: Since each Λ_m is a closed subspace of H, and their intersection $\Lambda = \cap_m \Lambda_m$ is nonempty, the result follows from Theorem 1. ■

The DBE algorithm presented in Algorithm 1 assumes that boundary cluster heads are synchronized so that local projections can be performed sequentially. We note however that it is still possible to achieve convergence if after a boundary cluster head has performed its local projection, it randomly chooses a neighboring boundary cluster head to pass information to. The chosen neighboring cluster head then repeats the same procedure. We call this the randomized DBE algorithm. Let $\mathcal{G}$ be the graph with vertex set $\mathcal{B}$, which has an edge between C_i and C_j if they are neighboring clusters. We have the following convergence result.

Proposition 3: Suppose that $K(u, u) \leq \kappa^2$ for all $u \in H_K$, and $\mathcal{G}$ is connected. The estimation error in the randomized DBE algorithm converges to $\mathbb{E}[(f^*(X) - U)^2]$ where f^* is an optimal solution to (3).

Proof: Let f_n be the estimation function at the n-th projection of the randomized DBE algorithm. Since $\mathcal{G}$ is connected, the random sequence of chosen cluster heads is an irreducible and recurrent Markov chain so that every cluster head appears infinitely often in the random sequence. From [30], the sequence f_n is weakly convergent to an optimal estimation function f^*. Since weakly convergent sequences are bounded [31], we have $|f_n(x)| \leq \|f_n\|_{H_K} \sqrt{K(x, x)} \leq \kappa \|f_n\|_{H_K}$ is bounded. From the dominated convergence theorem [23] and the reproducing property of H_K, we obtain

$$\begin{aligned} \lim_{n\to\infty} \mathbb{E}[(f_n(X) - U)^2] &= \mathbb{E}[\lim_{n\to\infty} (\langle f_n, K(\cdot, X)\rangle_{H_K} - U)^2] \\ &= \mathbb{E}[\lim_{n\to\infty} (f^*(X) - U)^2], \end{aligned}$$

where $\langle \cdot, \cdot \rangle_{H_K}$ is the inner product of H_K, and the proof is now complete. ■

IV. Performance Analysis

In this section, we first analyze the trade-off between communication cost and estimation error in the DBE algorithm. Then, we propose a two-step approach to spatial spectrum sensing

Algorithm 1 Distributed Boundary Estimation (DBE)

Initialization:

- $\tilde{z}_i = u_i$, for $i = 1, \ldots, N$,
- $f^0_{C_j} = 0$, $m_{j,k} = 0$ and $\tilde{\epsilon}_{j,k} = 0$, for all $C_j \in \mathcal{B}$, $k \in \mathcal{N}(C_j)$.
- $t_{\max}$ = maximum number of iterations

for each $C \in \mathcal{B}$ **do**
  for each $i \in C$ **do**
    Compute $K(x_i, x_j)$ by measuring $\|x_i - x_j\|$ for all $j \in C$. Send computed values to the cluster head.
  end for
end for
for $t = 1, \ldots, t_{\max}$ **do**
  for $j = 1, \ldots, M$ **do**
    Solve (7) by setting $f_{C_j}(x) = \sum_{i \in C_j} \beta_{j,i} K(x, x_i)$, and minimizing over $(\{\beta_{j,i} : i \in C_j\}, \epsilon_{j,k})$. Let $f^t_{C_j}$ be the optimal solution for f_{C_j}.
    Update

- $\tilde{z}_i = f^t_{C_j}(x_i)$, and send $\tilde{z}_i$ to all $k \in \mathcal{N}(C_j)$.
- $\tilde{\epsilon}_{j,k} = \epsilon_{j,k}$,
- $m_{j,k} = \frac{1}{|C_j|} \sum_{i \in C_j} f^t_{C_j}(x_i)$, and send $m_{j,k}$ to all $k \in \mathcal{N}(C_j)$.

  end for
end for

based on the DBE algorithm, and compare its setup complexity and throughput with that of the traditional fusion center (FC) approach.

A. Communication Cost and Estimation Error

We let the SU locations be distributed as a homogeneous Poisson point process Π in $\mathbb{R}^d$ with rate λ, and assume that the region of interest A has unit d-dimensional volume. Since we do not have any prior knowledge of the SU locations, it is reasonable to assume that SUs are located independently and randomly. The homogeneous Poisson point process captures this assumption

and has been widely adopted in the literature to model the distribution of ad hoc communicating devices [32], [33]. The Poisson point process also makes the mathematical analysis tractable, which provides insights into the system performance in practical scenarios. In Section V-C, we present simulation results for a specific case when SUs are not distributed according to a homogeneous Poisson point process.

We consider the trade-off between communication cost and the estimation error resulting from the boundary estimation as the rate λ varies, and we determine an approximate optimal density for the SUs that minimizes a weighted sum of the communication cost and estimation error. Finding the optimal density is useful in the case where there are multiple PUs, and random subsets of SUs may be chosen to estimate the boundary of each PU. Intuitively, as SUs become more dense, the expected communication cost increases because the number of SUs in each cluster and the number of boundary clusters increase, but the expected estimation error decreases due to the availability of more training examples. In the following, because of technical difficulties, we present *heuristic* approximations to both the expected communication cost and estimation error, and determine the optimal density by minimizing a weighted sum of these approximations. We present simulation results in Section V to verify that the approximate optimal density found is close to the true optimal one.

For simplicity, we assume that the boundary cluster heads all come from a fixed region $\mathcal{D}$ with volume $b > 0$, that this region contains the boundary of $\mathcal{R}$, and that it is sufficiently small so that certain approximations, which we describe below, hold. In finding the optimal density, we will see that the region $\mathcal{D}$ need not be known in advance. We summarize some of the notations introduced in this section in Table II for ease of reference.

TABLE II

SUMMARY OF NOTATIONS USED

Symbol	**Definition**
λ	rate of SU location Poisson point process
$\mathcal{D}$	approximate region in A containing all boundary clusters
b	volume of the region $\mathcal{D}$
$B_x(\delta)$	disk of radius δ centered at x
v_d	volume of $B_0(1)$ in $\mathbb{R}^d$
$g(r)$	communication cost function between two SUs distance r apart
p_B	approximate probability a cluster centered in $\mathcal{D}$ is a boundary cluster (see (9))
κ	$K(u,u) \leq \kappa^2$ for all $u \in H_K$

1) Communication Cost: Suppose that the cost of sending a message from a SU at position x to another at position x' is given by a non-negative function $g(\|x - x'\|)$ with $g(0) = 0$. In many wireless applications, this cost is modeled by the power required to achieve a given signal to noise ratio at the receiver, and $g(r)$ is a function of the form cr^ζ, where $c > 0$ and $\zeta \in [2, 5]$. Let a disk of radius δ centered at x be denoted as $B_x(\delta)$, and let v_d be the volume of a unit disk in $\mathbb{R}^d$. The expected communication cost can be found by considering the intra-cluster communication cost and the inter-cluster communication cost separately. The intra-cluster cost is incurred when SUs within a cluster communicate with their cluster head. Let the cluster head of cluster C_j be $\bar{x}_j$. The intra-cluster cost is then given by

$$\begin{aligned}\mathbb{E}\left[\sum_{j=1}^{M}\sum_{i\in C_j} g(\|x_i - \bar{x}_j\|)\right] &= \mathbb{E}[M]\ \mathbb{E}[|C_1|]\ \mathbb{E}\left[g(\|x\|)\mathbf{1}\{x \in B_0(\delta)\}\right] \\ &= \lambda^3 b p_h v_d \delta^d\ G(\delta), \end{aligned} \tag{8}$$

where the first equality follows from Wald's identity [23], the expected number of boundary clusters is given by $\mathbb{E}[M] = p_h \lambda b$, and

$$G(\delta) = \int_{B_0(\delta)} g(\|x\|)\ \mathrm{d}x.$$

The inter-cluster communication cost is incurred when boundary cluster heads exchange messages during the execution of the DBE algorithm. Cluster heads form a marked Poisson process with rate $p_h\lambda$. Let $p_B(x)$ be the probability that a cluster C with cluster head at $x \in \mathcal{D}$, is a boundary cluster. We make the following approximations in order to compute $p_B(x)$: (i) we assume that the boundary cluster test in Section III-A does not include the observation at the cluster head; (ii) we replace the number of SUs $|C|$ in one cluster by $\mathbb{E}[|C|] = v_d\delta^d$; and (iii) we assume that every SU in a cluster has the same probability $\bar{\alpha} = 1 - \alpha$ of declaring itself to be in $\mathcal{R}^c$ (this assumption is exact for those SUs in $\mathcal{R}^c$, and approximately true for all SUs in a boundary cluster if the cluster radius δ is sufficiently small). It can be shown that declaring a cluster C to be a boundary cluster is equivalent to requiring that $|C|(1-\gamma) \le \mathcal{U}^- \le |C|\gamma$. Under the above approximations, we then have for $x \in \mathcal{D}$,

$$p_B(x) \approx \sum_{(1-\gamma)v_d\delta^d \le k \le \gamma v_d \delta^d} \frac{\bar{\alpha}^k e^{\bar{\alpha}}}{k!} \triangleq p_B. \tag{9}$$

The expected inter-cluster communication cost is then given by

$$\begin{aligned}
&t_{\max}\mathbb{E}\left[\sum_{x,x'\in\mathcal{D}} g(\|x-x'\|)\mathbf{1}\{B_x(\delta), B_{x'}(\delta)\in\mathcal{B}, \|x-x'\|\le 2\delta\}\right] \\
&\le t_{\max}\mathbb{E}\left[\sum_{x,x'\in\mathcal{D}} g(\|x-x'\|)\mathbf{1}\{\|x-x'\|\le 2\delta\}p_B\right] \\
&= p_B p_h^2\lambda^2 t_{\max}\int_{\mathcal{D}}\int_{B_x(2\delta)} g(\|x-x'\|)\mathrm{d}x'\mathrm{d}x \\
&= \lambda^2 b p_h p_B t_{\max} G(2\delta), \qquad (10)
\end{aligned}$$

where the penultimate equality follows from two applications of the Slivnyak-Mecke Theorem [34].

From (8) and (10), the total expected communication cost per SU in $\mathcal{D}$ is then upper bounded by

$$C(\lambda) = \lambda^2 p_h v_d \delta^d G(\delta) + \lambda p_h p_B t_{\max} G(2\delta). \qquad (11)$$

2) Estimation Error: To evaluate the estimation error $\mathcal{E}$ in (2), we consider

$$\mathbb{E}[(f^*(X)-U)^2\mathbf{1}\{X\in\mathcal{D}\}] = b\mathcal{E}_{\mathcal{D}}, \qquad (12)$$

where

$$\mathcal{E}_{\mathcal{D}} = \mathbb{E}[(f^*(X)-U)^2 \mid X\in\mathcal{D}], \qquad (13)$$

and f^* is the solution to (3) given the data $y = \{(x_i, u_i) : i = 1,\ldots,N\}$. Compared to $\mathcal{E}$ in (2), we have ignored the estimation errors incurred in clusters close to the PU or far away from the no-talk region boundary. This is because for sufficiently large rate λ, these errors are largely dependent on the detection threshold instead of the rate.

Unfortunately, to the best of our knowledge, finding high probability bounds for the generalization error of learning problems like (3) is an open problem, because of correlations in the loss functions for the clusters due to constraints (5) and (6). We therefore make a simplification by dropping these constraints in our analysis, and assume the boundary clusters perform their learning *independently* of each other. Furthermore, for a boundary cluster C_j, let $f^*_{C_j}$ be the local

estimation function corresponding to f^*, and we approximate (13) using

$$\begin{aligned}\tilde{\mathcal{E}}_{\mathcal{D}} &= \mathbb{E}\left[\frac{1}{M}\sum_{j=1}^{M}(f^*_{C_j}(X)-U)^2\mathbf{1}\{X\in C_j\}\mid X\in\mathcal{D}\right]\\ &= \frac{v_d\delta^d}{bM}\sum_{j=1}^{M}R_j,\end{aligned}$$

where $R_j = \mathbb{E}[(f^*_{C_j}(X)-U)^2 \mid X\in C_j]$.

We assume that the kernel K satisfies the bound $K(u,u)\le\kappa^2$ for all $u\in H_K$, and for some constant $\kappa>0$. We also assume that $\nu_j=\nu|C_j|$ for all $j=1,\dots,M$, and some positive constant ν. We first state two lemmas, the first of which follows from the Chernoff bound, and the second from Lemma 23, Theorems 12 and 22 of [35]. We omit their proofs here.

Lemma 1: For any measurable set C, let $N(C)$ and $\mu(C)$ be the count function and mean measure of the Poisson point process Π, respectively. For any $\varepsilon>0$, we have

$$\mathbb{P}(|N(C)-\mu(C)|\ge\varepsilon)\le 2e^{-\frac{1}{4}\varepsilon^2\mu(C)}.$$

Lemma 2: Suppose that $K(u,u)\le\kappa^2$ for all $u\in H_K$. Then, for any $j=1,\dots,M$, and any $\varepsilon>0$, with probability at least $1-\varepsilon$ over the random draw of the data y, we have

$$R_j\le\frac{1}{|C_j|}\sum_{i\in C_j}(f^*_{C_j}(x_i)-u_i)^2+\frac{4\kappa^2}{\nu|C_j|}(\frac{\kappa}{\sqrt{\nu}}+1)^2+(\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{\ln(1/\varepsilon)}{2|C_j|}}.$$

For simplicity, we approximate $M\approx p_h\lambda b$. From Lemma 1, if λ is sufficiently large, we have for any region C, $(1-\varepsilon)\mu(C)\le N(C)\le(1+\varepsilon)\mu(C)$ with high probability. Therefore, by choosing λ to be large enough, with probability at least $1-\varepsilon$, where $\varepsilon\in(0,1)$, we have for all $j=1,\dots,M$,

$$\begin{aligned}R_j\le{}&\frac{1}{\lambda(1-\varepsilon)v_d\delta^d}\sum_{i\in C_j}(f^*_{C_j}(x_i)-u_i)^2+\frac{4\kappa^2}{\nu\lambda(1-\varepsilon)v_d\delta^d}(\frac{\kappa}{\sqrt{\nu}}+1)^2\\&+(\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{\ln(p_h\lambda b/\varepsilon)}{2\lambda(1-\varepsilon)v_d\delta^d}}.\end{aligned}$$

Using the probability union bound, we have with probability at least $1-\varepsilon$,

$$\begin{aligned}\tilde{\mathcal{E}}_{\mathcal{D}}\le{}&\frac{1}{\lambda(1-\varepsilon)bM}\sum_i(f^*(x_i)-u_i)^2+\frac{4\kappa^2}{\nu\lambda(1-\varepsilon)b}(\frac{\kappa}{\sqrt{\nu}}+1)^2\\&+\frac{1}{b}(\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{v_d\delta^d\ln(p_h\lambda b/\varepsilon)}{2\lambda(1-\varepsilon)}}.\end{aligned}\tag{14}$$

Furthermore, Lemma 23 of [35] yields

$$(f^*_{C_j}(x) - u)^2 \leq (\frac{\kappa}{\sqrt{\nu}} + 1)^2,$$

for all $j = 1, \ldots, M$, $x \in \mathcal{D}$, and $u \in \{-1, 1\}$. This implies that with probability one, $\tilde{\mathcal{E}}_{\mathcal{D}}$ is upper bounded by the right hand side of (14) plus $\varepsilon(\kappa/\sqrt{\nu} + 1)^2$.

We aim to find $\lambda > 0$ that minimizes a weighted sum of the communication cost upper bound (11) and the estimation error upper bound given by (12) and (14). The objective function to be minimized is given by

$$\begin{aligned} & C(\lambda) + \beta \left(\frac{F_3}{\lambda} + F_4 \sqrt{\frac{\ln \lambda}{\lambda}} \right) \\ & = F_1 \lambda^2 + F_2 \lambda + \beta \left(\frac{F_3}{\lambda} + F_4 \sqrt{\frac{\ln \lambda}{\lambda}} \right), \end{aligned} \tag{15}$$

where $\beta > 0$ is a constant, and

$$\begin{aligned} F_1 &= p_h v_d \delta^d G(\delta), \\ F_2 &= p_h p_B t_{\max} G(2\delta), \\ F_3 &= \frac{4\kappa^2}{\nu} (\frac{\kappa}{\sqrt{\nu}} + 1)^2, \\ F_4 &= (\frac{8\kappa^2}{\nu} + 1)(\frac{\kappa}{\sqrt{\nu}} + 1)^2 \sqrt{\frac{(1-\varepsilon) v_d \delta^d}{2}}. \end{aligned}$$

We have made the approximation that $\ln(p_h b/\varepsilon)/\lambda \approx 0$ when λ is sufficiently large so that the value of b need not be known a priori. The optimal rate can be found by setting the derivative with respect to λ of (15) to zero (it is clear that there is a positive minimizer) to obtain

$$4F_1 \lambda^{\frac{5}{2}} + 2F_2 \lambda^{\frac{3}{2}} + \beta(-2F_3 \lambda^{-\frac{1}{2}} + F_4((\ln \lambda)^{-\frac{1}{2}} - (\ln \lambda)^{\frac{1}{2}})) = 0,$$

the solution of which can be computed numerically. To find the optimal SU density that minimizes the communication cost subject to the constraint that the estimation error is below a given level is equivalent to (15), where β is a Lagrange multiplier.

B. Setup Complexity and Throughput

Let $\hat{\mathcal{R}}$ be the estimator for $\mathcal{R}$ produced by the DBE algorithm. Those SUs outside of $\hat{\mathcal{R}}$ can utilize the spectrum without performing spectrum sensing, while SUs inside of $\hat{\mathcal{R}}$ perform

collaborative spectrum sensing by sending their local sensing decisions to a fusion center. For convenience, we call our two-step approach the DBE-spectrum sensing (DBE-SS) method. We analyze the complexity and throughput of the DBE-SS method and the traditional FC method, where all SUs send their local sensing decisions to a fusion center. In the FC method, SUs do not know the boundary of $\mathcal{R}$, therefore the spectrum is utilized by the SUs only if the fusion center declares that the PU is inactive.

We first consider the complexity of performing boundary estimation using the DBE algorithm. Recall that each SU nominates itself to be a cluster head with probability p_h, and each cluster is covered by a disk of radius δ. Therefore, there are on average $O(\delta^d)$ SUs in a cluster and line 4 in the DBE algorithm has complexity $O(\delta^{2d})$. The optimization problem (7) can be viewed as a convex quadratic program with $O(\delta^d)$ constraints, with complexity $O(\delta^d \delta^{3d}) = O(\delta^{4d})$. The expected number of boundary clusters is bounded by $O(p_h N)$, therefore the overall expected complexity of the DBE algorithm is $O(p_h N \delta^{4d})$. On the other hand, in the FC approach, SUs route their local decisions to a fusion center using a minimum spanning tree (MST). If we assume that the underlying communication network is formed by joining any two SUs that are within distance δ of each other, then the complexity of setting up a MST (with global knowledge of the whole network topology) is $O(N\delta^d)$. Clearly, the DBE algorithm has higher complexity than the fusion center setup if $p_h > \delta^{-3d}$.

We now compare the throughput of the DBE-SS method with that achieved by the FC approach. We make several assumptions to simplify the analysis. Suppose that all SUs transmit at the same power $P_s < P_0$, and that in any given area, at most a fraction q of the SUs can share the spectrum. We assume additive white Gaussian noise channels with noise power N_0. We also assume that interference between the PU and the SUs outside of $\mathcal{R}$ is negligible, while the throughput for SUs in $\mathcal{R}$ when the PU is active is negligible. Then, the throughput of a SU in the absence of the PU is [36] $\bar{R}_0 = \log_2\left(1 + \frac{P_s}{N_0}\right)$. We further assume that each SU uses a periodic frame structure of duration T, which includes a sensing duration of τ. Suppose that the PU is active with probability π. Then, the average throughput per user under the FC method is

$$R_s^{\text{FC}} = (1-\pi) q \bar{R}_0 (1 - \frac{\tau}{T})(1 - Q_f^{\text{FC}}),$$

where Q_f^{FC} is the false alarm rate at the fusion center. As the fusion center does not know the receiver operating characteristic (ROC) of each sensor, it uses a simple k-out-of-N rule to fuse

the SUs' local decisions. The false alarm is given by[7]

$$Q_f^{\text{FC}} = \sum_{i=1}^{k} \binom{qN}{i} \alpha^i (1-\alpha)^{qN-i}. \tag{16}$$

The value of k is chosen so that the probability of detection $Q_d^{\text{FC}} \geq \bar{\beta}$, for some fix threshold $\bar{\beta}$.

On the other hand, the average throughput per user for the SUs under DBE-SS is

$$R_s = (1-\pi) q \bar{R}_0 \left(\frac{|\hat{\mathcal{R}}|}{N} (1 - \frac{\tau}{T})(1 - Q_f) + 1 - \frac{|\hat{\mathcal{R}}|}{N} \right) + \pi q \bar{R}_0 \left(1 - \frac{|\mathcal{R} \cup \hat{\mathcal{R}}|}{N} \right),$$

where Q_f is the false alarm rate for the SUs inside $\hat{\mathcal{R}}$, with the same minimum probability of detection $\bar{\beta}$, and can be computed in a similar manner as (16). It can be shown that if $\hat{\mathcal{R}} \approx \mathcal{R}$ and $|\mathcal{R}|/N$ is sufficiently small, then $R_s > R_s^{\text{FC}}$. This is intuitively clear as there is little benefit to perform boundary estimation if $\mathcal{R}$ covers almost all the SUs.

Next, we derive the throughput for the PU. In the FC approach, the PU average throughput is

$$R_p^{\text{FC}} = \pi \left(Q_d^{\text{FC}} \log_2 \left(1 + \frac{P_0}{N_0} \right) + (1 - Q_d^{\text{FC}}) \log_2 \left(1 + \frac{P_0}{N_0 + qNP_s} \right) \right),$$

where Q_d^{FC} is the detection probability at the fusion center. Because of estimation errors in the DBE-SS scheme, $\mathcal{R} \backslash \hat{\mathcal{R}}$ may be non-empty. Interference from SUs in this set is bounded by $N_1 = q|\mathcal{R} \backslash \hat{\mathcal{R}}| P_s$. The PU average throughput under the DBE-SS scheme is then given by

$$R_p = \pi \left(Q_d \log_2 \left(1 + \frac{P_0}{N_0 + N_1} \right) + (1 - Q_d) \log_2 \left(1 + \frac{P_0}{N_0 + q|\mathcal{R}|P_s} \right) \right),$$

where $Q_d \geq \bar{\beta}$ is the detection probability for the SUs inside $\hat{\mathcal{R}}$. In Section V, we present simulation results to compare the throughput R_s under the DBE-SS scheme with R_s^{FC} under the FC method, when $R_p = R_p^{\text{FC}}$.

V. Simulation Results

In this section, we present simulation results to verify the performance of the DBE algorithm and the DBE-SS method. In each simulation run, 1000 sensors are uniformly distributed in a region A of size 5×5 km^2, with the PU (e.g., a TV transmitter) located at the center of the region. We use the standard CCIR model [37] for the path loss. For each data point, we perform 1000 simulation runs using the parameters in Table III.

[7]To avoid cluttered expressions, we assume that qN is always an integer.

TABLE III

SIMULATION PARAMETERS

Parameters	**Value**	**Parameters**	**Value**
PU transmit power P_0	40 dBm	SU transmit power P_s	10 dBm
PU transmitter antenna height	30 m	SU transmitter antenna height	3 m
PU transmit antenna gain	6 dBd	SU transmit antenna gain	0 dBd
PU active probability π	0.3	SU coverage radius δ	0.25 km
interference threshold θ_0	-75 dBm	SU false alarm probability α	0.05
frame duration T	20 ms	shadow fading standard deviation σ	6 dB
sensing time τ	1 ms	learning parameters $\nu_1 = \ldots = \nu_M$	1
communication cost function $g(r)$	r^2	learning parameters $\eta_1 = \ldots = \eta_M$	10
noise power N_0	-10 dBm	learning kernel $K(x, x')$	$e^{-0.5\|x-x'\|^2}$
estimation error parameter ε	0.01	trade-off weight β	500

A. Estimation Error and Communication Cost

We compare the communication cost incurred and the estimation performance of the DBE algorithm with that of various benchmark algorithms, including the following:

1) Centralized boundary estimation algorithm based on LS-SVM [12]: a global classifier is trained based on information from all SUs in the boundary clusters.
2) Centralized image processing based seeded region growing (SRG) algorithm [38]: we regard the decision u_i of each SU i as a pixel gray level in a binary image and segment the image by growing a region from a seed point using an intensity mean measure.
3) Distributed Bayesian event region detection (ERD) algorithm [15], [16]: a threshold decision scheme is applied to correct the errors of local SU decisions. We refer the reader to [15] for details.

The estimation performance is evaluated according to (2), normalized by four times the area of $\mathcal{R}$. Since the estimation function f takes values close to 1 or -1, the normalized estimation error is approximately the area in which misclassification occurs, expressed as a fraction of the area of $\mathcal{R}$. The communication cost is computed by assuming that each message passed between two SUs a distance r apart incurs a cost of $g(r) = r^2$.

Figure 2 shows the normalized estimation error and communication cost for each algorithm when choosing different values for p_h, which is the probability that each SU independently nominates itself to be a cluster head. The threshold γ in the boundary cluster decision rule in Section III-A is set to be 0.6. As p_h increases, the performance of the SRG and ERD algorithms remain constant as these algorithms do not use clustering. The performance of our proposed DBE

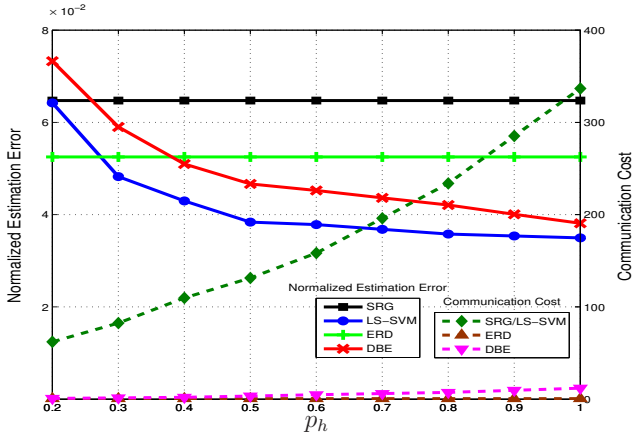

Fig. 2. Normalized estimation errors and total communication costs for different values of p_h when $\gamma = 0.6$.

algorithm on the other hand, becomes better than the SRG and the ERD algorithms, but still underperforms the centralized LS-SVM algorithm. Centralized methods like LS-SVM and SRG however have higher communication costs than the ERD and DBE algorithms as only short range communications are required for the latter algorithms. We see that the DBE algorithm achieves arguably the best trade-off between the estimation error and communication cost if p_h is chosen to be sufficiently large.

We now set the probability $p_h = 0.8$ and vary the threshold γ in the boundary cluster decision rule, with a higher value of γ corresponding to more clusters being chosen as the boundary clusters. We see from Figure 3 that although the estimation error decreases with increasing γ, the rate of decrease is not very significant. This is because most of the actual boundary clusters have already been included for reasonable values of γ. This implies that in practice, a reasonably small value of γ can be chosen to reduce the communication costs incurred during boundary estimation.

We next fix $p_h = 0.8$ and $\gamma = 0.6$, and vary the SU density from 5 to 160 SU per km^2. In Figure 4, we use simulation to compute the sum of the communication cost and estimation error (2) weighted by β. For comparison, we also plot the cost function $C(\lambda)$ in (15). Although $C(\lambda)$ does not include the estimation errors incurred in clusters far away from the boundary of $\mathcal{R}$, it is seen that it still serves as an upper bound to the simulated cost as estimation errors in clusters located far from the boundary are very small. We see from Figure 4 that the simulated and theoretical optimal SU densities are nearly identical to each other.

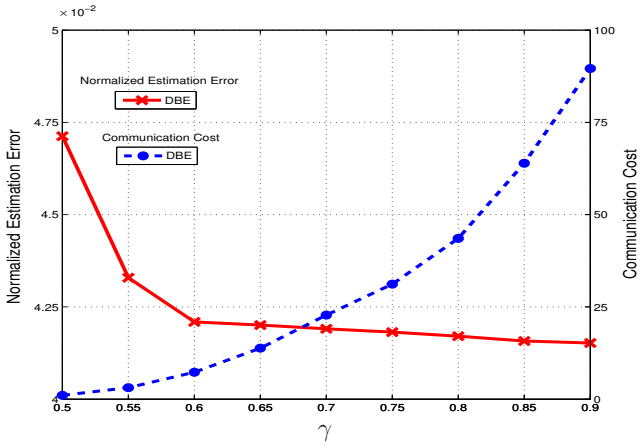

Fig. 3. Normalized estimation errors and total communication costs when $p_h = 0.8$, and the threshold γ is varied.

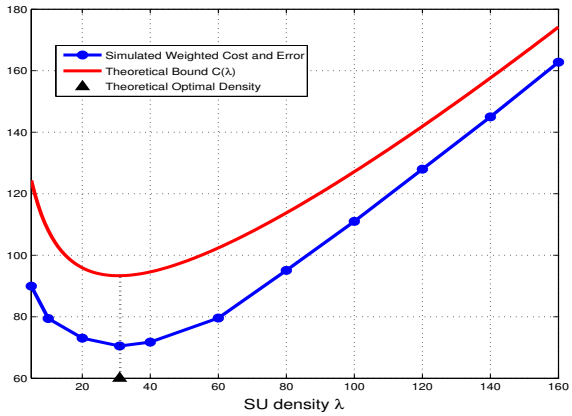

Fig. 4. Weighted sum of communication cost and estimation error when $\gamma = 0.6$ and $p_h = 0.8$, with varying SU density.

In Figure 5, we show the convergence of the normalized estimation error with respect to the number of inter-cluster communications for a particular simulation run with 70 boundary clusters. We see that for the DBE algorithm the estimation error converges within 200 messages (or equivalently about 3 passes over all boundary clusters). This shows that in practice, $t_{\max}$ in the DBE algorithm can be chosen to be a reasonably small value. The randomized DBE algorithm on the other hand requires a much larger number of inter-cluster communications, but less coordination amongst the boundary cluster heads.

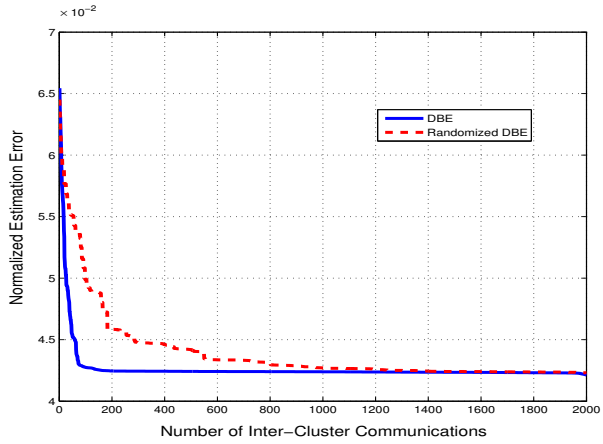

Fig. 5. Normalized estimation error versus number of inter-cluster communications.

B. Throughput

In this subsection, we present numerical results for the ROCs and throughputs of the FC and DBE-SS methods after boundary estimation with $p_h = 0.8$ and $\gamma = 0.6$. Recall that the fusion center has no knowledge of the ROC of individual SUs, and a simple k-out-of-N fusion rule is utilized in place of optimal fusion. Figure 6 shows the ROC curves of the two methods. It is seen that the DBE-SS method has a higher detection probability for each false alarm probability because only information from SUs in $\mathcal{R}$ are utilized, leading to less errors.

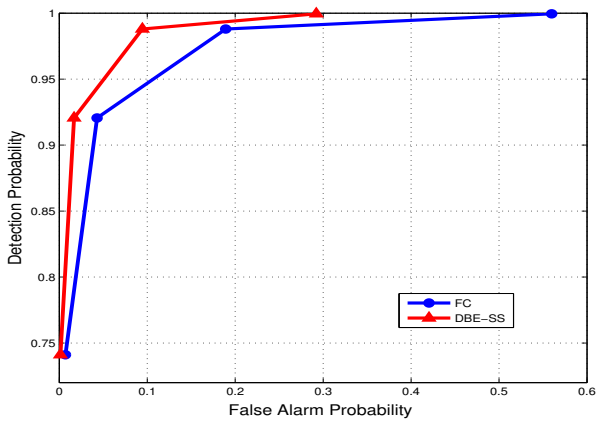

Fig. 6. ROC curves under FC and DBE-SS.

In Figure 7, we vary the detection probability and plot the PU throughput versus the throughput per SU for both DBE-SS and FC methods. The throughput per SU for the DBE-SS method is

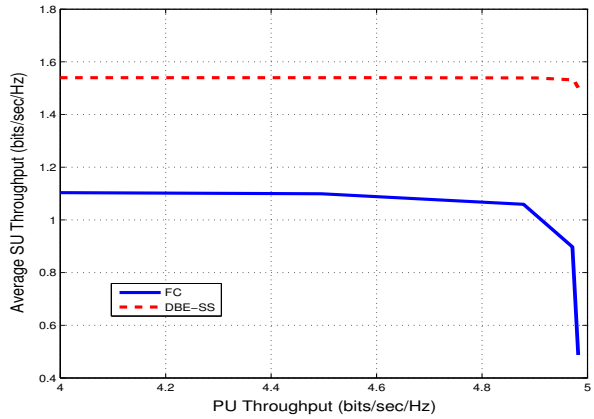

Fig. 7. PU throughput and throughput per SU under FC and DBE-SS.

relatively flat over all PU throughputs as SUs outside $\hat{\mathcal{R}}$ can transmit regardless of whether the PU is present or not. We also see that the SU throughput is higher than that for the FC method. Figure 8 shows the average SU throughput when the PU throughput is fixed at 4 bits/sec/Hz, and the volume of A is decreased. We see that the DBE-SS method should only be used if A is more than 10% larger than $\mathcal{R}$.

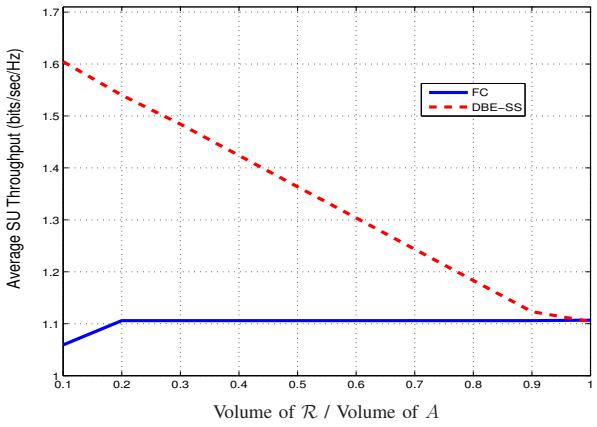

Fig. 8. Average throughput per SU as volume of A changes.

C. Robustness

We now compare the robustness of the various boundary estimation algorithms. We fix $p_h = 0.8$ and $\gamma = 0.6$. To simulate SU sensing errors, a boundary cluster is randomly chosen with

probability ς, and then a random subset of the SU sensing decisions in the chosen cluster is changed from -1 to 1, while an equal number of SU sensing decisions is changed from 1 to -1. We plot the average normalized estimation error in Figure 9. Figure 9 shows that our proposed DBE algorithm is more robust than the other benchmark boundary estimation methods, except for the centralized LS-SVM method. We also compare with a modified version of the DBE algorithm in which we set $\eta_j = 0$ for all $j = 1, \ldots, M$ so that the smoothness constraint (6) no longer applies. We see that including the smoothing constraint improves the robustness of our algorithm as neighboring boundary clusters moderate their local classifiers to avoid an abrupt change in the average classification function value within their clusters.

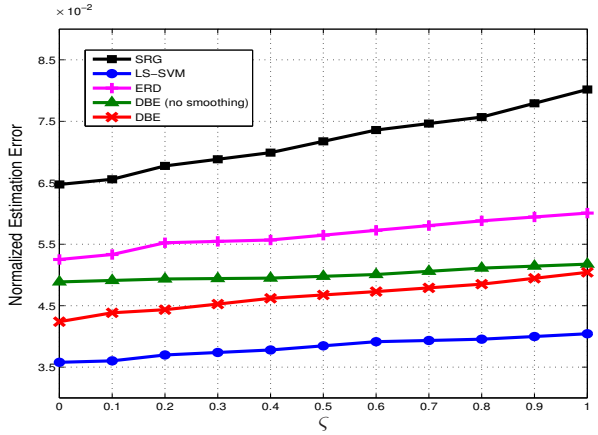

Fig. 9. Robustness comparison of boundary estimation algorithms after flipping the observations.

Next, we compare the estimation error of the DBE algorithm with and without the smoothness constraint (6) when the SUs are no longer distributed as a homogeneous Poisson point process. With probability ω, a boundary cluster is independently populated with 20 SUs uniformly distributed inside the cluster. With probability $1 - \omega$, a boundary cluster is divided into four quadrants, and a quadrant is chosen randomly. The chosen quadrant is then populated with 20 SUs uniformly. In Figure 10, we see that the smoothing constraint results in a lower estimation error. Figure 11 shows a portion of the estimated boundaries.

VI. Conclusion

We have developed a distributed boundary estimation algorithm for estimating the no-talk region of a PU in a cognitive radio network, and analyzed the trade-offs between the communi-

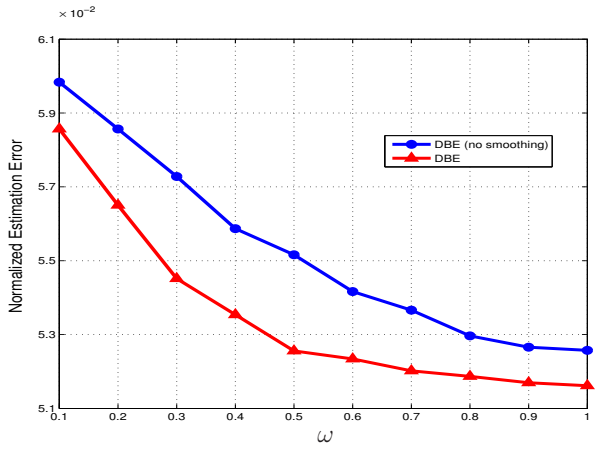

Fig. 10. Robustness comparison of DBE algorithm with and without smoothing constraints after changing the density.

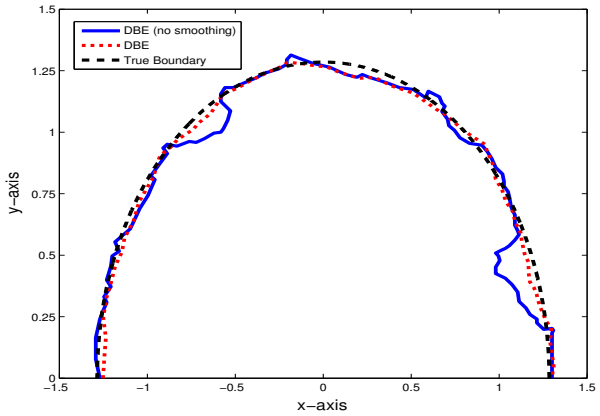

Fig. 11. Estimated boundaries with and without smoothing constraints.

cation cost and estimation error of our proposed method. We derive approximate upper bounds for the communication cost and estimation error, and provide a method to compute the optimal SU density. Simulation results suggest that our proposed algorithm have lower estimation errors and better robustness compared to various other methods.

We have made various simplifying and heuristic assumptions in deriving the optimal SU density. Our simulations however shows that despite these assumptions, the theoretical optimal SU density found is not significantly different from the actual one. Bounding the estimation error more precisely remains a difficult open problem. In this work, we have also assumed that the PU is static and the received interference of SUs at the PU do not vary over time. Future work includes developing boundary estimation techniques for the cases where PUs are mobile

and where communication channels are time variant.

REFERENCES

[1] J. Unnikrishnan and V. V. Veeravalli, "Cooperative sensing for primary detection in cognitive radio," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 18–27, 2008.

[2] Z. Li, F. R. Yu, and M. Huang, "A cooperative spectrum sensing consensus scheme in cognitive radios," in *Proc. IEEE 28th Annual Joint Conf. of the IEEE Computer and Communications Societies*, 2009, pp. 2546–2550.

[3] W. Zhang and K. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, 2008.

[4] R. Tandra, S. M. Mishra, and A. Sahai, "What is a spectrum hole and what does it take to recognize one?" *Proc. IEEE*, vol. 97, no. 5, pp. 824–848, 2009.

[5] R. Tandra, A. Sahai, and V. Veeravalli, "Unified space-time metrics to evaluate spectrum sensing," *IEEE Commun. Mag.*, vol. 49, no. 3, pp. 54–61, 2011.

[6] M. Vu, N. Devroye, and V. Tarokh, "On the primary exclusive region of cognitive networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 7, pp. 3380–3385, 2009.

[7] Z. Wei, Z. Feng, Q. Zhang, and W. Li, "Three regions for space-time spectrum sensing and access in cognitive radio networks," in *Proc. IEEE Global Communications Conf.*, 2012, pp. 1283–1288.

[8] Q. Li, Z. Feng, W. Li, and T. A. Gulliver, "Joint temporal and spatial spectrum sharing in cognitive radio networks: A region-based approach with cooperative spectrum sensing," in *Proc. IEEE Wireless Communications and Networking Conf.*, 2013, pp. 620–625.

[9] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Proc. 1st Int. Workshop on Algorithms for Sensor Systems*, 2004, pp. 123–136.

[10] R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," in *Proc. 4th Int. Symp. Information Processing in Sensor Networks*, 2005, pp. 254–260.

[11] Y. Wang, J. Gao, and J. S. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proc. 12th Annual Int. Conf. on Mobile Computing and Networking*, 2006, pp. 122–133.

[12] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[13] Y. Yang, Y. Liu, Q. Zhang, and L. Ni, "Cooperative boundary detection for spectrum sensing using dedicated wireless sensor networks," in *Proc. IEEE 29th Annual Joint Conf. of the IEEE Computer and Communications Societies*, 2010, pp. 1–9.

[14] K. K. Chintalapudi and R. Govindan, "Localized edge detection in sensor fields," in *Proc. IEEE 1st Int. Workshop on Sensor Network Protocols and Applications*, 2003, pp. 59–70.

[15] B. Krishnamachari and S. Iyengar, "Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 241–250, 2004.

[16] Q. Chen, K.-Y. Lam, and P. Fan, "Comments on "Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks"," *IEEE Trans. Comput.*, vol. 54, no. 9, pp. 1182–1183, 2005.

[17] R. Nowak and U. Mitra, "Boundary estimation in sensor networks: Theory and methods," in *Proc. Information Processing in Sensor Networks*, 2003, pp. 80–95.

[18] S. Duttagupta, K. Ramamritham, and P. Kulkarni, "Tracking dynamic boundaries using sensor network," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 10, pp. 1766–1774, 2011.

[19] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "A collaborative training algorithm for distributed learning," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1856–1871, 2009.

[20] S. Tozlu, M. Senel, W. Mao, and A. Keshavarzian, "Wi-fi enabled sensors for internet of things: A practical approach," *IEEE Commun. Mag.*, vol. 50, no. 6, pp. 134–143, 2012.

[21] D. Cabric, I. O'Donnell, M.-W. Chen, and R. Brodersen, "Spectrum sharing radios," *IEEE Circuits Syst. Mag.*, vol. 6, no. 2, pp. 30–45, 2006.

[22] R. Tandra and A. Sahai, "SNR walls for signal detection," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 4–17, 2008.

[23] R. Durrett, *Probability: Theory and Examples*, 3rd ed. Belmont, CA: Duxbury, 2004.

[24] V. N. Vapnik, *Statistical learning theory*. Wiley, New York, 1998.

[25] R. Rom and M. Sidi, *Multiple Access Protocols: Performance and Analysis*. Springer London, 2011.

[26] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4635–4643, 2006.

[27] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*. New York: Oxford University Press, 1997.

[28] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized Representer Theorem," in *Proc. 14th Annual Conf. on Computational Learning Theory and 5th European Conf. on Computational Learning Theory*, 2001, pp. 416–426.

[29] J. J. Caffery, *Wireless Location in CDMA Cellular Radio Systems*. Boston : Kluwer Academic, 2000.

[30] C. Badea, S. Grivaux, and V. Mï£¡ller, "The rate of convergence in the method of alternating projections," *St. Petersburg Math. J*, vol. 23, pp. 413–434, 2012.

[31] L. Debnath and P. Mikusinski, *Introduction to Hilbert spaces with applications*, 2nd ed. San Diego, CA: Academic Press, 1999.

[32] M. Z. Win, P. C. Pinto, and L. A. Shepp, "A mathematical theory of network interference and its applications," *Proc. IEEE*, vol. 97, no. 2, pp. 205–230, 2009.

[33] C. han Lee and M. Haenggi, "Interference and outage in poisson cognitive networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1392–1401, 2012.

[34] D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic Geometry and its Applications*. John Wiley & Sons, Chichester, 1995.

[35] O. Bousquet and A. Elisseeff, "Stability and generalization," *Journal of Machine Learning Research*, vol. 2, pp. 499–526, 2002.

[36] Y.-C. Liang, Y. Zeng, E. C. Y. Peh, and A. T. Hoang, "Sensing-throughput tradeoff for cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1326–1337, 2008.

[37] W. C. Y. Lee, "Lee's model [cellular radio path loss prediction]," in *Proc. IEEE 42nd Vehicular Technology Conf.*, 1992, pp. 343–348.

[38] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and M. Walid G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1454–1466, 2001.

Distributed Boundary Estimation for Spectrum Sensing in Cognitive Radio Networks

Yi Zhang, *Student Member, IEEE*, Wee Peng Tay, *Member, IEEE*, Kwok Hung Li, *Senior Member, IEEE*, Dominique Gaïti, *Member, IEEE*

***Abstract*—In a cognitive radio network, a primary user (PU) shares its spectrum with secondary users (SUs) temporally and spatially, while allowing for some interference. We consider the problem of estimating the no-talk region of the PU, i.e., the region outside which SUs may utilize the PU's spectrum regardless of whether the PU is transmitting or not. We propose a distributed boundary estimation algorithm that allows SUs to estimate the boundary of the no-talk region collaboratively through message passing between SUs, and analyze the trade-offs between estimation error, communication cost, setup complexity, throughput and robustness. Simulations suggest that our proposed scheme has better estimation performance and communication cost trade-off compared to several other alternative benchmark methods, and is more robust to SU sensing errors, except when compared to the least squares support vector machine approach, which however incurs a much higher communication cost.**

***Index Terms*—Cognitive radio, boundary estimation, spectrum sensing.**

I. Introduction

A cognitive radio (CR) network improves radio spectrum utilization by permitting unlicensed secondary users (SUs) to access the same spectrum when the licensed primary users (PUs) are not using it, or when SU transmissions do not interfere significantly with the PUs. Various spectrum sensing methods have been proposed, including centralized [1], distributed [2] and relay-assisted cooperative detection schemes [3].

In this paper, we consider the *spatial* usage diversity of the PU by letting the PU fix an interference temperature limit that allows for interference from SUs in its licensed spectrum below a threshold [4], [5]. This translates to a no-talk region around the PU, in which SUs opportunistically transmit only if the PU is not transmitting [4], [6]. SUs outside of this no-talk region can transmit regardless of whether the PU is active or not. Our main goal is to develop a distributed algorithm that allows SUs to cooperatively determine the boundary of the no-talk region.

In [6]–[8], different spectrum sharing regions, including a primary exclusive region and the no-talk region, are defined. However, all these works assume that the propagation path loss between the PU and SUs are isotropic, and all regions are assumed to be circular. Bounds on the radius of each region are given based on interference and outage considerations, which are characterized in terms of propagation parameters like path loss exponents. In practice, the propagation environment may be very difficult to model quantitatively, and the no-talk region is unlikely to be circular. Therefore, in this work, we develop boundary estimation methods for the no-talk region without relying on extensive assumptions about the shape of the region.

Boundary estimation is widely used in different sensor networking applications, and has been extensively studied. Methods based on node degrees [9], connectivity information [10], and topology information [11] have been proposed to estimate the coverage region of a sensor network. Although the definition of a no-talk region was comprehensively addressed in [4], little work has been done to estimate the boundary of this region. A classifier-based cooperative boundary detection algorithm for estimating the no-talk region using support vector machines (SVM) [12] has been proposed in [13]. A computational geometry method based on convex hulls has also been utilized for boundary estimation in [13]. All these works however assume that sensors send their local information to a fusion center, and boundary estimation is performed at the fusion center. In a cognitive radio network, constraints on energy and bandwidth usually restrict SUs from communicating with a single fusion center effectively. Localized edge detection algorithms based on statistical, image processing and classification methods have been proposed in [14] to allow a sensor to locally decide whether or not it is located on or near a boundary. A distributed Bayesian algorithm has also been proposed to determine event regions [15], [16]. These methods however do not make use of cooperation between sensors. A hierarchical tree-based estimation method using recursive dyadic partitioning [17] and a dynamic boundary tracking algorithm that combines spatial and temporal estimation techniques [18] have been proposed for boundary estimation in ad-hoc networks. However, this method is again centralized, and does not consider the smoothness of the estimated boundary.

In this paper, we consider the cooperative estimation of the PU's no-talk region by exploiting local communications amongst SUs. Our main contributions are the following:

1) We propose a distributed boundary estimation method based on the distributed learning framework of [19], and with additional smoothness constraints. Sensors outside the estimated no-talk region are allowed to transmit even if the PU is transmitting.
2) We provide approximate theoretical bounds for the communication cost incurred by our proposed method and the expected estimation error, so that the approximate optimal SU density can be inferred. This is useful for randomly

This research was supported in part by the MOE AcRF Tier 1 Grant RG25/10. Y. Zhang, W. P. Tay, and K. H. Li, are with the Nanyang Technological University, Singapore. E-mail: yzhang29@e.ntu.edu.sg, {wptay,ekhli}@ntu.edu.sg. Dominique Gaïti is with University of Technology of Troyes, France. E-mail: dominique.gaiti@utt.fr.

allocating SUs to estimate the no-talk regions of multiple PUs transmitting over different frequency bands. We note that our theoretical performance analysis is not considered in [19], and to the best of our knowledge, is new.

3) We derive order bounds for the setup complexity of our proposed method, and expressions for the throughput achievable by the PU and SUs.
4) Simulations suggest that our proposed boundary estimation algorithm have better trade-offs in the throughput and setup communication cost than various other boundary estimation algorithms in the literature, and is more robust to SU sensing errors except when compared to the centralized least squares SVM (LS-SVM) method, which however incurs a much higher communication cost.

Our method allows better spatial usage of the spectrum and improves the overall system throughput, albeit at the cost of estimating the boundary. For a stationary PU, this is a fixed cost that does not contribute significantly to the overall operational energy cost. An example is the use of CR systems in the Internet-of-Things framework [20], where devices like electrical appliances are fixed and CRs in the devices allow opportunistic use of the cluttered spectrum.

The rest of this paper is organized as follows. In Section II, we introduce our system model and problem formulation. In Section III, we propose a distributed boundary estimation algorithm for estimating the boundary of $\mathcal{R}$. In Section IV, we analyze the trade-offs between communication cost and estimation error of our boundary estimation method for a Poisson field of SUs, and determine its setup complexity and throughput. Simulation results are provided in Section V. Finally, we conclude in Section VI.

II. System Model

Suppose that there is one PU and N SUs in a bounded region $A \subset \mathbb{R}^d$.[1] We say that the PU is active if it is transmitting in its licensed spectrum. Suppose that the PU is located at x_0. We assume that all wireless channels are symmetric, and define the no-talk region [4] of the PU to be the set $\mathcal{R} = \{x \in \mathbb{R}^d : P_0 - L(x, x_0) > \theta_0\}$, where P_0 is the transmit power of the PU, $L(x, x_0)$ is the average propagation loss function between the PU and a SU located at x, and θ_0 is a fixed threshold. The average propagation loss can be modeled as $L(x, x_0) = l(\|x - x_0\|) + S(x, x_0) + F(x, x_0)$,[2] where $l(\|x - x_0\|)$ is the power attenuation due to the distance $\|x - x_0\|$ between a SU at location x and the PU at location x_0, $S(x, x_0)$ represents the average shadowing effect, and $F(x, x_0)$ is the average power loss due to multipath fading. We suppose that the PU can tolerate an average interference below the fixed threshold θ_0 so that SUs outside of $\mathcal{R}$ can utilize the PU spectrum regardless of whether it is active or not. SUs within the no-talk region $\mathcal{R}$ are required to refrain from using the PU spectrum if the PU is transmitting. Note that the threshold θ_0 is chosen to include a safety margin or budget for the propagation loss due to shadowing and fading, and other parameters like the average density of SUs. The reader is referred to [4] for a detailed discussion of the different considerations involved in defining the no-talk region of a PU.

In this paper, we aim to estimate the no-talk region $\mathcal{R}$, or equivalently the boundary of $\mathcal{R}$, in order to facilitate spatial spectrum sharing between the PU and SUs. The average propagation loss $L(x, x_0)$ for a SU at position x depends on various factors including the terrain, the type and number of reflectors and attenuators between the PU and SU, and other ambient environmental factors. The propagation loss function is thus difficult to determine to good accuracy in practice, and therefore we assume that $L(x, x_0)$ is unknown, and adopt a learning approach to estimate the region $\mathcal{R}$ solely based on the received power at the SUs. We make the following assumptions.

Assumption 1:

(a) Communications by SUs are over relatively shorter distances than the PU, and hence the transmit power of each SU is at most P_0. Multiple SUs can share the PU's spectrum spatially (see Figure 1).[3]
(b) The region $\mathcal{R}$ is compact, and has a smooth[4] boundary.
(c) Time can be discretized into intervals and the PU is active in each interval with known probability $\pi \in [0, 1]$, independently across intervals.[5]

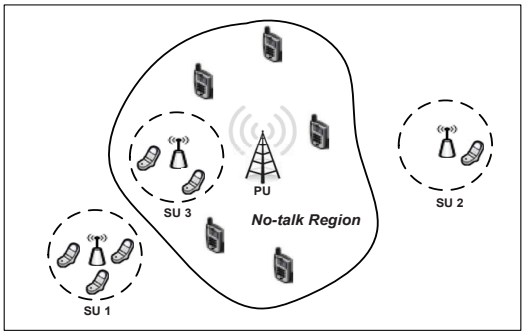

Fig. 1. Spatial spectrum sharing between PU and multiple SUs. SU 1 and 2 can use the licensed spectrum of the PU without spectrum sensing. SU 3 can only utilize the spectrum when the PU is inactive.

We let each SU sample the PU licensed spectrum over a sufficiently long calibration phase in order to perform boundary estimation of $\mathcal{R}$. We assume that $\mathcal{R}$ has a smooth boundary in Assumption 1(b) to avoid the case where a temporary degradation in the channel between the PU and a SU during the calibration phase may incur a large estimation error. This assumption is also valid in most practical situations, except when there are strong attenuators close to the boundary, in which case our solution leads to an estimated no-talk region larger than the actual one. In our problem formulation (cf. Section III-B), we will not impose Assumption 1(b) strictly, but rather adopt a simpler constraint to approximate it. In

[1] In most applications, $d = 2$ or 3, which corresponds to SUs scattered over a geographical region or in a building respectively.

[2] All power quantities are expressed in dB.

[3] Various spectrum sharing schemes have been described in [21].

[4] Formally, this means that the boundary is parameterizable and differentiable in that parameter.

[5] Although we restrict our analysis to the case where the PU active probability π is the same across intervals, our analysis can be easily generalized to the case where each interval has a different PU active probability.

addition, for practicality, we require that the estimation algorithm is distributed, with each SU having access only to local information (its own observations and information from its neighbors).

Suppose that SU i is located at x_i, for $i = 1, \ldots, N$, and suppose that the calibration period for SU i is divided into J observation intervals, where the PU is active with probability π during each interval independently (cf. Assumption 1(c)). Let $Y_i[j]$ be the signal sample obtained by SU i in interval j, where $j = 1, \ldots, J$. If the PU is inactive during interval j, we have

$$Y_i[j] = W_i[j],$$

where the noise variables $W_i[j]$ are independent zero mean Gaussian random variables with variance σ_W^2. If the PU is active during interval j, we have

$$Y_i[j] = X_i[j] + W_i[j],$$

where $X_i[j]$ is the PU signal, which is assumed to be a Gaussian random variable with zero mean, and independent of $W_i[j]$. The variance or power of $X_i[j]$ is then given by $\mu_i = P_0 - L(x_i, x_0)$.

Let $\mathcal{H}_0 : \mu_i \leq \theta_0$ and $\mathcal{H}_1 : \mu_i > \theta_0$ be the hypotheses that SU i is outside and within the no-talk region $\mathcal{R}$, respectively. For the sake of generality, we assume that the PU signal modulation scheme is unknown to the SUs. Therefore, SUs are constrained to use energy detection methods [4], [22] in order to perform the hypothesis test. For this purpose, SU i forms the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |Y_i[j]|^2$ and uses the following threshold rule to determine the hypothesis:

$$T_i \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\lessgtr}} \theta, \tag{1}$$

where θ is chosen so that the false alarm probability is below a predefined threshold $\alpha \in (0, 1)$. Since μ_i is unknown a priori, we need to make further approximations in order to determine θ. The mean of T_i is given by

$$m(\mu_i) = \pi\mu_i + \sigma_W^2,$$

and applying Wald's identity [23], we obtain after some algebra, the variance of T_i is

$$\eta(\mu_i) = \frac{1}{J}\left(\mu_i^2\pi(3-\pi) + 2\sigma_W^2(\sigma_W^2 + 2\pi)\right).$$

Assuming that J is sufficiently large, the central limit theorem [23] allows us to approximate the distribution of T_i as a Gaussian distribution with mean $m(\mu_i)$ and variance $\eta(\mu_i)$. Since both $m(\cdot)$ and $\eta(\cdot)$ are increasing functions, the threshold θ can now be chosen to ensure that the false alarm constraint in the worst case situation is satisfied, by setting

$$\mathcal{Q}\left(\frac{\theta - m(\theta_0)}{\sqrt{\eta(\theta_0)}}\right) = \alpha,$$

where $\mathcal{Q}(\cdot)$ is the complementary cumulative distribution function for the standard normal distribution. We note that the test (1) and the choice of θ do not require knowledge of the PU transmit power P_0 or the locations of the PU and SUs.

When there are multiple PUs transmitting in the same spectrum, the no-talk region is the union of all PUs' no-talk regions. Suppose there are $N' > 1$ PUs, and that all PU signals are uncorrelated. Let $Y_{ip}[j]$ be the signal sample received by SU i from the PU p in the interval j. For a sufficiently large J and $p \neq p'$, we have $\frac{1}{J}\sum_{j=1}^{J} Y_{ip}[j]Y_{ip'}[j] \approx 0$, and the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |\sum_{p=1}^{N'} Y_{ip}[j]|^2$ can be approximated as $\frac{1}{J}\sum_{p=1}^{N'}\sum_{j=1}^{J} |Y_{ip}[j]|^2$. A threshold for T_i similar to that in (1) can be found to determine if SU i is within the no-talk region of at least one PU.

Let u_i be the decision of SU i for the test (1), where $u_i = -1$ if it decides in favor of $\mathcal{H}_0$, and $u_i = 1$ otherwise. Recall that x_i is the position of SU i, and let $y_i = (x_i, u_i)$, for $i = 1, \ldots, N$. Our aim is to learn a function $f : \mathbb{R}^d \mapsto \mathbb{R}$ based on the collection of pairs $y = \{y_i : i = 1, \ldots, N\}$, so that $f(x) \geq 0$ or $f(x) < 0$ if a SU location x is inside or outside the region $\mathcal{R}$ respectively, and with $\{x : f(x) = 0\}$ corresponding to the boundary of $\mathcal{R}$. In the same spirit as statistical learning theory [24], we can regard each y_i as being drawn independently from the same joint distribution $p(x, u)$ (which is unknown because the path loss $L(x, x_0)$ is unknown). Then, in estimating the boundary of $\mathcal{R}$, we hope to obtain a function f with small generalization error

$$\mathcal{E} = \mathbb{E}[(f(X) - U)^2], \tag{2}$$

where (X, U) has joint distribution $p(x, u)$. If $\mathcal{E}$ is large, the throughput at the PU deteriorates because of interference from SUs that wrongly believe themselves to be in $\mathcal{R}^c$.[6] Therefore, we are interested to study the trade-offs in throughputs with $\mathcal{E}$ and the communication cost of performing the boundary estimation.

In a centralized estimation algorithm, the data y is sent to a fusion center, which trains a global function. Such centralized algorithms suffer from several disadvantages, including the need to select a site for the fusion center, the susceptibility of the whole network to a single point of failure at the fusion center, the need for significant processing power and memory storage at the fusion center, and the use of long range communications as the area A becomes large. In this paper, we consider distributed algorithms, in which each SU communicates only with neighboring SUs to collaboratively estimate the boundary of $\mathcal{R}$.

For the convenience of the reader, we list some commonly used notations in Table I. Some of these notations have been defined in this section, while the remaining ones will be defined formally in the sequel where they first appear. In addition, we adopt the following definitions. For SU i and cluster C, we use $i \in C$ to mean $x_i \in C$. The number of SUs in y belonging to C is given by $|C|$. The indicator function $\mathbf{1}\{S\}$ equals to 1 if the statement S is true and 0 otherwise.

III. Distributed Boundary Estimation

In this section, we propose a distributed boundary estimation algorithm that determines the boundary of the set $\mathcal{R}$ based

[6] We use $\mathcal{R}^c = A \backslash \mathcal{R}$ to denote the complement of $\mathcal{R}$ in the region of interest A.

TABLE I
SUMMARY OF NOTATIONS USED

Symbol	Definition
A	region of interest containing the PU and all SUs
$\mathcal{R}$	no-talk region of PU
N	number of SUs in region of interest A
δ	broadcast range of a SU
p_h	probability of a SU to become a cluster head
γ	threshold to determine if a cluster is a boundary cluster
M	number of boundary clusters
C_j	the jth boundary cluster, $j = 1, \ldots, M$
$\mathcal{B}$	set of boundary clusters $\{C_1, \ldots, C_M\}$
$\mathcal{N}(C_j)$	set of neighboring clusters of C_j
f_{C_j}	local boundary estimation function of cluster C_j

on message passing between SUs. The SUs are grouped into clusters, and most communications are over relatively short ranges within clusters. Each cluster has a SU that serves as the cluster head. The cluster head communicates with SUs inside its cluster, performs most of the necessary computations required for distributed estimation of the boundary, and communicates with other cluster heads. Cluster heads thus expend more energy than typical SUs inside the cluster. Incentives can be designed to compensate cluster heads; an example being given higher priority to access the spectrum. Such incentive mechanisms are out of the scope of our current work, and will not be discussed here.

Our distributed boundary estimation procedure consists of the following steps.

(i) *Formation of clusters.* Each SU independently nominates itself to be a cluster head with probability p_h. A cluster head broadcasts a message over a control channel to all SUs within a distance δ to inform them of their inclusion into the cluster. To avoid collisions amongst cluster heads, a carrier sense multiple access protocol [25] is used. Note that a cluster head can also belong to another cluster, and a SU can belong to multiple clusters.

(ii) *Boundary cluster identification.* We design a metric to identify those clusters that lie close to the boundary of the set $\mathcal{R}$. We call these the boundary clusters.

(iii) *Distributed boundary estimation.* Messages are exchanged between members of a boundary cluster and its cluster head. In addition, messages are exchanged between cluster heads of neighboring boundary clusters to collaboratively estimate the boundary of $\mathcal{R}$.

In the following subsections, we describe steps (ii) and (iii) in detail.

A. Boundary Cluster Identification

Let C be a cluster, and $\mathcal{U}^- = \frac{1}{|C|}\sum_{i\in C} \mathbf{1}\{u_i = -1\}$ to be the fraction of SUs in cluster C with $u_i = -1$. The clusters within $\mathcal{R}$ have a higher probability of $1 - \mathcal{U}^-$ being much larger than $\mathcal{U}^-$, while the reverse is true for clusters that are far from the PU. To identify those clusters that are close to the boundary of $\mathcal{R}$, we let $\mathcal{S} = \max(\mathcal{U}^-, 1 - \mathcal{U}^-)$, and say that C is a boundary cluster if and only if $\mathcal{S} \leq \gamma$, where γ to be a fixed threshold. If C is not a boundary cluster, the cluster head declares it to be within $\mathcal{R}$ if $\mathcal{U}^- < 1/2$, and outside vice versa. We call those clusters in the former class *inside* clusters, and those in the latter class *outside* clusters.

B. Distributed Boundary Estimation

To learn a function f that can be used to determine if a new SU location x (not necessarily belonging to the training data y) is within or without $\mathcal{R}$, we consider the following approach. A SU at location x queries its cluster head to check the types of cluster it belongs to (recall that a SU may belong to multiple clusters). If it belongs to an inside cluster, we let $f(x) = 1$ and declare that it belongs to $\mathcal{R}$. If it does not belong to any inside clusters, and it belongs to a boundary cluster, it uses a local function, which we describe below, to determine its location status. Finally, if it is not within any inside or boundary clusters, we let $f(x) = -1$, and declare that the SU is in $\mathcal{R}^c$.

Let $\mathcal{B} = \{C_1, \ldots, C_M\}$ be the set of boundary clusters. The boundary clusters collaboratively estimate the boundary of $\mathcal{R}$ based on local information and message exchanges between cluster heads. We use the reproducing kernel Hilbert space (RKHS) [19], [26] formulation to obtain a function that distinguishes a location x to be inside or outside $\mathcal{R}$. However, since we do not assume that there is a central authority to perform the estimation, we consider instead finding a collection $\{f_{C_j}\}$ of local functions, each corresponding to a boundary cluster. If x is not within an inside cluster and it belongs to a boundary cluster, we let the estimation function $f(x)$ take the value $f_C(x)$ where C is chosen randomly from the set of boundary clusters containing x.

Let H_K be a RKHS corresponding to a kernel $K(\cdot,\cdot)$ that serves as a similarity measure between two SU locations. We restrict to kernels that are radial basis functions (RBF), i.e., those kernels that can be expressed as functions of the Euclidean distance between two SUs. For each $C_j \in \mathcal{B}$, let $\mathcal{N}(C_j)$ be the set of indices k with $j \neq k$ and $|C_k \cap C_j| \neq 0$. We call those clusters in $\mathcal{N}(C_j)$ the neighboring clusters of C_j. Our goal is to

$$\min \sum_{i\in\cup_{j=1}^{M} C_j} (z_i - u_i)^2 + \sum_{m=1}^{M} \nu_m \|f_{C_m}\|^2_{H_K} + \sum_{m=1}^{M} \sum_{k\in\mathcal{N}(C_m)} \eta_m \epsilon^2_{m,k} \tag{3}$$

subject to

$$f_{C_m} \in H_K, \ \forall\ C_m \in \mathcal{B}, \tag{4}$$

$$z_i = f_{C_m}(x_i), \ \forall\ i \in C_m, C_m \in \mathcal{B}, \tag{5}$$

$$\epsilon_{m,k} = \frac{1}{|C_m|}\sum_{i\in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|}\sum_{i\in C_k} f_{C_k}(x_i), \quad \forall\ k \in \mathcal{N}(C_j), C_j \in \mathcal{B}, \tag{6}$$

where $\|\cdot\|_{H_K}$ is the norm of H_K, and ν_m, η_m, for $j = 1, \ldots, M$, are positive constants. The minimization in (3) is over all variables z_i, f_{C_m} and $\epsilon_{m,k}$. The constraints (4) require that the local classifier f_{C_m} from each boundary cluster C_m is chosen from the RKHS H_K. The constraints (5) ensure that if a SU belongs to multiple boundary clusters, the classification result remains the same regardless of the local classifier used. Finally, the constraints (6) ensure that the estimated boundary is smooth (cf. Assumption 1(b)).

The kernel least squares minimization problem (3) is similar to that proposed in [19], which considers a general distributed learning framework, but without additional constraints like (6). The reference [19] also provides a distributed method to iteratively obtain the optimizers $\{f_{C_m}\}$ by message exchanges between cluster heads. In the following, we show that their distributed algorithm can be adapted to our minimization problem (3). Our argument is similar to that in [19], and treats the minimization in (3) as projections onto closed convex subspaces of a Hilbert space. This can be done because of the successive orthogonal projection (SOP) theorem [27], which we state below without proof.

Theorem 1: Let $\{\Lambda_m\}_{m=1}^M$ be a set of closed, convex, and affine subsets of a RKHS H, and whose intersection $\Lambda = \bigcap_{m=1}^M \Lambda_m$ is nonempty. For any $v_0 \in H$, let v^* be the orthogonal projection of v_0 onto Λ, and for all $n \geq 1$, let v_n be the orthogonal projection of v_{n-1} onto $\Lambda_{(n \mod M)}$. Then, $\lim_{n\to\infty} \|v_n - v^*\| = 0$.

Suppose that $S = \sum_{m=1}^M |\mathcal{N}(C_m)|$ is the total number of variables $\epsilon_{m,k}$ where $m = 1, \ldots, M$ and $k \in \mathcal{N}(C_m)$. Let $H = \mathbb{R}^N \times H_K^M \times \mathbb{R}^S$. In the sequel, to avoid cluttered notations, we let $v = ((z_i), (f_m), (\epsilon_{m,k})) \in H$ to denote an element from H with the understanding that the index i runs from 1 to N, the index m runs from 1 to M, and the index $k \in \mathcal{N}(C_m)$ for each m. We let H be a Hilbert space by letting the squared norm of v be

$$\|v\|_H^2 = \sum_{i=1}^N |z_i|^2 + \sum_{m=1}^M \nu_m \|f_{C_m}\|_{H_K}^2 + \sum_{m=1}^M \sum_{k\in\mathcal{N}(C_m)} \eta_m \epsilon_{m,k}^2.$$

For each $j = 1, \ldots, M$, let

$$\begin{aligned}\Lambda_j = \Big\{ &((z_i), (f_m), (\epsilon_{m,k})) \in H : z_i = f_{C_m}(x_i), \forall\ i \in C_m, \\ &\text{and} \\ &\epsilon_{m,k} = \frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|} \sum_{i\in C_k} f_{C_k}(x_i), \\ &\forall\ k \in \mathcal{N}(C_m) \Big\}.\end{aligned}$$

It can be shown that Λ_j is a closed subspace of H. Then, the minimization problem (3) is equivalent to finding the projection of $(u_1, \ldots, u_N, 0, \ldots, 0)$ onto the closed and convex set $\Lambda = \cap_{j=1}^M \Lambda_j$.

As pointed out in [19], instead of directly finding the projection onto Λ, Theorem 1 allows us to iteratively project onto each Λ_m, for $m = 1, \ldots, M$. The SOP algorithm first finds the projection v_1 of $(u_1, \ldots, u_N, 0, \ldots, 0)$ onto Λ_1, then finds the projection of v_1 onto Λ_2, and so on. Projections are performed over all Λ_m, $m = 1, \ldots, M$, with multiple iterations over the indices m. Suppose that at an iteration, we seek to project $v = ((\tilde{z}_i), (\tilde{f}_j), (\tilde{\epsilon}_{j,k}))$ onto Λ_m. This is equivalent to

$$\begin{aligned}\min\quad &\sum_{i\in C_m} (f_{C_m}(x_i) - \tilde{z}_i)^2 + \nu_m \left\| f_{C_m} - \tilde{f}_{C_m} \right\|_{H_K}^2 \\ &+ \eta_m \sum_{k\in\mathcal{N}(C_m)} (\epsilon_{m,k} - \tilde{\epsilon}_{m,k})^2 \qquad (7)\end{aligned}$$

subject to

$$\begin{aligned}&f_{C_m} \in H_K, \\ &\epsilon_{m,k} = \frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i) - \frac{1}{|C_k|} \sum_{i\in C_k} f_{C_k}(x_i), \\ &\forall\ k \in \mathcal{N}(C_m).\end{aligned}$$

The minimization in (7) is over f_{C_m} and $\epsilon_{m,k}$, and involves only data from C_m and its neighboring clusters. It is thus a *local* optimization problem. Suppose that $(f_{C_m}^*, (\epsilon_{m,k}^*)_{k\in\mathcal{N}(C_m)})$ is the optimizer for (7). The projected point is then given by $v^* = ((z_i^*), (f_j^*), (\epsilon_{j,k}^*))$, where

$$\begin{aligned}&z_i^* = \tilde{z}_i \text{ if } i \notin C_m, \text{ and } z_i^* = f_{C_m}^*(x_i) \text{ if } i \in C_m, \\ &f_{C_j}^* = \tilde{f}_{C_j} \text{ and } \epsilon_{j,k}^* = \tilde{\epsilon}_{j,k} \text{ if } j \neq m.\end{aligned}$$

The messages that cluster C_m passes to a neighboring cluster C_k are $\{z_i^* : i \in C_m \cap C_k\}$ and $\frac{1}{|C_m|} \sum_{i\in C_m} f_{C_m}(x_i)$, where the first message represents its current best estimate of $\{u_i : i \in C_m \cap C_k\}$ subject to the constraints (4)-(6), and serves as the "training labels" [19] for the SUs in both clusters. The second message encodes the average value achieved by $f_{C_m}^*$, and allows C_k to adjust its own classifier to improve the smoothness of the estimated boundary. The following result is a direct consequence of the Representer Theorem [28], and its proof is omitted. It characterizes the form of the optimal solution $f_{C_m}^*$ for (7).

Proposition 1: For each $C_m \in \mathcal{B}$, the optimal solution to the minimization problem (7) is given by

$$f_{C_m}^*(x) = \sum_{i\in C_m} \beta_{m,i} K(x, x_i).$$

Furthermore, if the kernel $K(\cdot,\cdot)$ is a radial basis function, the computation of $f_{C_m}(x)$ requires only knowledge of $\|x - x_i\|$, for all $i \in C_m$.

From (7) and Proposition 1, to train the classifier for a cluster $C_m \in \mathcal{B}$, we require the cluster head to know $\|x_i - x_j\|$, for all $i, j \in C_m$. This can be obtained using various ranging techniques. Examples include methods in which each SU i broadcasts a pilot signal with known transmit power, or exchange messages with timestamps [29]. Our distributed boundary estimation algorithm is formally stated in Algorithm 1, which we call the DBE algorithm. The following proposition shows that the classifiers in the DBE algorithm converges.

Proposition 2: For each C_m, where $m = 1, \ldots, M$, the sequence $(f_{C_m}^t)$ in line 9 of the DBE algorithm converges as number of iterations $t \to \infty$.

Proof: Since each Λ_m is a closed subspace of H, and their intersection $\Lambda = \cap_m \Lambda_m$ is nonempty, the result follows from Theorem 1. ■

The DBE algorithm presented in Algorithm 1 assumes that boundary cluster heads are synchronized so that local projections can be performed sequentially. We note however that it is still possible to achieve convergence if after a boundary cluster head has performed its local projection, it randomly chooses a neighboring boundary cluster head to pass information to. The chosen neighboring cluster head then repeats the same procedure. We call this the randomized DBE algorithm. Let $\mathcal{G}$ be the graph with vertex set $\mathcal{B}$, which has an

edge between C_i and C_j if they are neighboring clusters. We have the following convergence result.

Proposition 3: Suppose that $K(u,u) \leq \kappa^2$ for all $u \in H_K$, and $\mathcal{G}$ is connected. The estimation error in the randomized DBE algorithm converges to $\mathbb{E}[(f^*(X)-U)^2]$ where f^* is an optimal solution to (3).

Proof: Let f_n be the estimation function at the n-th projection of the randomized DBE algorithm. Since $\mathcal{G}$ is connected, the random sequence of chosen cluster heads is an irreducible and recurrent Markov chain so that every cluster head appears infinitely often in the random sequence. From [30], the sequence f_n is weakly convergent to an optimal estimation function f^*. Since weakly convergent sequences are bounded [31], we have $|f_n(x)| \leq \|f_n\|_{H_K}\sqrt{K(x,x)} \leq \kappa\|f_n\|_{H_K}$ is bounded. From the dominated convergence theorem [23] and the reproducing property of H_K, we obtain

$$\begin{aligned}\lim_{n\to\infty}\mathbb{E}[(f_n(X)-U)^2] &= \mathbb{E}[\lim_{n\to\infty}(\langle f_n, K(\cdot,X)\rangle_{H_K}-U)^2] \\ &= \mathbb{E}[\lim_{n\to\infty}(f^*(X)-U)^2],\end{aligned}$$

where $\langle\cdot,\cdot\rangle_{H_K}$ is the inner product of H_K, and the proof is now complete. ∎

Algorithm 1 Distributed Boundary Estimation (DBE)

1: **Initialization:**
- $\tilde{z}_i = u_i$, for $i = 1,\ldots,N$,
- $f^0_{C_j} = 0$, $m_{j,k} = 0$ and $\tilde{\epsilon}_{j,k} = 0$, for all $C_j \in \mathcal{B}$, $k \in \mathcal{N}(C_j)$.
- $t_{\max}$ = maximum number of iterations

2: **for** each $C \in \mathcal{B}$ **do**
3: **for** each $i \in C$ **do**
4: Compute $K(x_i,x_j)$ by measuring $\|x_i - x_j\|$ for all $j \in C$. Send computed values to the cluster head.
5: **end for**
6: **end for**
7: **for** $t = 1,\ldots,t_{\max}$ **do**
8: **for** $j = 1,\ldots,M$ **do**
9: Solve (7) by setting $f_{C_j}(x) = \sum_{i\in C_j}\beta_{j,i}K(x,x_i)$, and minimizing over $(\{\beta_{j,i} : i \in C_j\}, \epsilon_{j,k})$. Let $f^t_{C_j}$ be the optimal solution for f_{C_j}.
10: Update
- $\tilde{z}_i = f^t_{C_j}(x_i)$, and send $\tilde{z}_i$ to all $k \in \mathcal{N}(C_j)$.
- $\tilde{\epsilon}_{j,k} = \epsilon_{j,k}$,
- $m_{j,k} = \frac{1}{|C_j|}\sum_{i\in C_j} f^t_{C_j}(x_i)$, and send $m_{j,k}$ to all $k \in \mathcal{N}(C_j)$.

11: **end for**
12: **end for**

IV. Performance Analysis

In this section, we first analyze the trade-off between communication cost and estimation error in the DBE algorithm. Then, we propose a two-step approach to spatial spectrum sensing based on the DBE algorithm, and compare its setup complexity and throughput with that of the traditional fusion center (FC) approach.

A. Communication Cost and Estimation Error

We let the SU locations be distributed as a homogeneous Poisson point process Π in $\mathbb{R}^d$ with rate λ, and assume that the region of interest A has unit d-dimensional volume. Since we do not have any prior knowledge of the SU locations, it is reasonable to assume that SUs are located independently and randomly. The homogeneous Poisson point process captures this assumption and has been widely adopted in the literature to model the distribution of ad hoc communicating devices [32], [33]. The Poisson point process also makes the mathematical analysis tractable, which provides insights into the system performance in practical scenarios. In Section V-C, we present simulation results for a specific case when SUs are not distributed according to a homogeneous Poisson point process.

We consider the trade-off between communication cost and the estimation error resulting from the boundary estimation as the rate λ varies, and we determine an approximate optimal density for the SUs that minimizes a weighted sum of the communication cost and estimation error. Finding the optimal density is useful in the case where there are multiple PUs, and random subsets of SUs may be chosen to estimate the boundary of each PU. Intuitively, as SUs become more dense, the expected communication cost increases because the number of SUs in each cluster and the number of boundary clusters increase, but the expected estimation error decreases due to the availability of more training examples. In the following, because of technical difficulties, we present *heuristic* approximations to both the expected communication cost and estimation error, and determine the optimal density by minimizing a weighted sum of these approximations. We present simulation results in Section V to verify that the approximate optimal density found is close to the true optimal one.

For simplicity, we assume that the boundary cluster heads all come from a fixed region $\mathcal{D}$ with volume $b > 0$, that this region contains the boundary of $\mathcal{R}$, and that it is sufficiently small so that certain approximations, which we describe below, hold. In finding the optimal density, we will see that the region $\mathcal{D}$ need not be known in advance. We summarize some of the notations introduced in this section in Table II for ease of reference.

TABLE II
SUMMARY OF NOTATIONS USED

Symbol	Definition
λ	rate of SU location Poisson point process
$\mathcal{D}$	approximate region in A containing all boundary clusters
b	volume of the region $\mathcal{D}$
$B_x(\delta)$	disk of radius δ centered at x
v_d	volume of $B_0(1)$ in $\mathbb{R}^d$
$g(r)$	communication cost function between two SUs distance r apart
p_B	approximate probability a cluster centered in $\mathcal{D}$ is a boundary cluster (see (9))
κ	$K(u,u) \leq \kappa^2$ for all $u \in H_K$

1) Communication Cost: Suppose that the cost of sending a message from a SU at position x to another at position x' is given by a non-negative function $g(\|x-x'\|)$ with $g(0) = 0$. In many wireless applications, this cost is modeled by the power required to achieve a given signal to noise ratio at the receiver, and $g(r)$ is a function of the form cr^ζ, where $c > 0$ and $\zeta \in [2,5]$. Let a disk of radius δ centered at x be denoted as $B_x(\delta)$, and let v_d be the volume of a unit disk in $\mathbb{R}^d$. The expected communication cost can be found by

considering the intra-cluster communication cost and the inter-cluster communication cost separately. The intra-cluster cost is incurred when SUs within a cluster communicate with their cluster head. Let the cluster head of cluster C_j be $\bar{x}_j$. The intra-cluster cost is then given by

$$\begin{aligned}&\mathbb{E}\left[\sum_{j=1}^{M}\sum_{i\in C_j} g(\|x_i-\bar{x}_j\|)\right]\\&=\mathbb{E}[M]\ \mathbb{E}[|C_1|]\ \mathbb{E}\left[g(\|x\|)\mathbf{1}\{x\in B_0(\delta)\}\right]\\&=\lambda^3 b p_h v_d \delta^d\ G(\delta),\end{aligned}\tag{8}$$

where the first equality follows from Wald's identity [23], the expected number of boundary clusters is given by $\mathbb{E}[M] = p_h\lambda b$, and

$$G(\delta)=\int_{B_0(\delta)} g(\|x\|)\ \mathrm{d}x.$$

The inter-cluster communication cost is incurred when boundary cluster heads exchange messages during the execution of the DBE algorithm. Cluster heads form a marked Poisson process with rate $p_h\lambda$. Let $p_B(x)$ be the probability that a cluster C with cluster head at $x\in\mathcal{D}$, is a boundary cluster. We make the following approximations in order to compute $p_B(x)$: (i) we assume that the boundary cluster test in Section III-A does not include the observation at the cluster head; (ii) we replace the number of SUs $|C|$ in one cluster by $\mathbb{E}[|C|] = v_d\delta^d$; and (iii) we assume that every SU in a cluster has the same probability $\bar{\alpha} = 1-\alpha$ of declaring itself to be in $\mathcal{R}^c$ (this assumption is exact for those SUs in $\mathcal{R}^c$, and approximately true for all SUs in a boundary cluster if the cluster radius δ is sufficiently small). It can be shown that declaring a cluster C to be a boundary cluster is equivalent to requiring that $|C|(1-\gamma)\le\mathcal{U}^-\le|C|\gamma$. Under the above approximations, we then have for $x\in\mathcal{D}$,

$$p_B(x)\approx\sum_{(1-\gamma)v_d\delta^d\le k\le\gamma v_d\delta^d}\frac{\bar{\alpha}^k e^{\bar{\alpha}}}{k!}\triangleq p_B.\tag{9}$$

The expected inter-cluster communication cost is then given by

$$\begin{aligned}&t_{\max}\mathbb{E}\Big[\sum_{x,x'\in\mathcal{D}} g(\|x-x'\|)\\&\qquad\mathbf{1}\{B_x(\delta),B_{x'}(\delta)\in\mathcal{B},\|x-x'\|\le 2\delta\}\Big]\end{aligned}\tag{10}$$

$$\begin{aligned}&\le t_{\max}\mathbb{E}\left[\sum_{x,x'\in\mathcal{D}} g(\|x-x'\|)\mathbf{1}\{\|x-x'\|\le 2\delta\}p_B\right]\\&=p_B p_h^2\lambda^2 t_{\max}\int_{\mathcal{D}}\int_{B_x(2\delta)} g(\|x-x'\|)\mathrm{d}x'\mathrm{d}x\\&=\lambda^2 b p_h p_B t_{\max} G(2\delta),\end{aligned}\tag{11}$$

where the penultimate equality follows from two applications of the Slivnyak-Mecke Theorem [34].

From (8) and (11), the total expected communication cost per SU in $\mathcal{D}$ is then upper bounded by

$$C(\lambda)=\lambda^2 p_h v_d\delta^d G(\delta)+\lambda p_h p_B t_{\max}G(2\delta).\tag{12}$$

2) Estimation Error: To evaluate the estimation error $\mathcal{E}$ in (2), we consider

$$\mathbb{E}[(f^*(X)-U)^2\mathbf{1}\{X\in\mathcal{D}\}]=b\mathcal{E}_{\mathcal{D}},\tag{13}$$

where

$$\mathcal{E}_{\mathcal{D}}=\mathbb{E}[(f^*(X)-U)^2\mid X\in\mathcal{D}],\tag{14}$$

and f^* is the solution to (3) given the data $y=\{(x_i,u_i): i=1,\dots,N\}$. Compared to $\mathcal{E}$ in (2), we have ignored the estimation errors incurred in clusters close to the PU or far away from the no-talk region boundary. This is because for sufficiently large rate λ, these errors are largely dependent on the detection threshold instead of the rate.

Unfortunately, to the best of our knowledge, finding high probability bounds for the generalization error of learning problems like (3) is an open problem, because of correlations in the loss functions for the clusters due to constraints (5) and (6). We therefore make a simplification by dropping these constraints in our analysis, and assume the boundary clusters perform their learning *independently* of each other. Furthermore, for a boundary cluster C_j, let $f^*_{C_j}$ be the local estimation function corresponding to f^*, and we approximate (14) using

$$\begin{aligned}\tilde{\mathcal{E}}_{\mathcal{D}}&=\mathbb{E}\left[\frac{1}{M}\sum_{j=1}^{M}(f^*_{C_j}(X)-U)^2\mathbf{1}\{X\in C_j\}\mid X\in\mathcal{D}\right]\\&=\frac{v_d\delta^d}{bM}\sum_{j=1}^{M}R_j,\end{aligned}$$

where $R_j=\mathbb{E}[(f^*_{C_j}(X)-U)^2\mid X\in C_j]$.

We assume that the kernel K satisfies the bound $K(u,u)\le\kappa^2$ for all $u\in H_K$, and for some constant $\kappa>0$. We also assume that $\nu_j=\nu|C_j|$ for all $j=1,\dots,M$, and some positive constant ν. We first state two lemmas, the first of which follows from the Chernoff bound, and the second from Lemma 23, Theorems 12 and 22 of [35]. We omit their proofs here.

Lemma 1: For any measurable set C, let $N(C)$ and $\mu(C)$ be the count function and mean measure of the Poisson point process Π, respectively. For any $\varepsilon>0$, we have

$$\mathbb{P}(|N(C)-\mu(C)|\ge\varepsilon)\le 2e^{-\frac{1}{4}\varepsilon^2\mu(C)}.$$

Lemma 2: Suppose that $K(u,u)\le\kappa^2$ for all $u\in H_K$. Then, for any $j=1,\dots,M$, and any $\varepsilon>0$, with probability at least $1-\varepsilon$ over the random draw of the data y, we have

$$\begin{aligned}R_j\le\frac{1}{|C_j|}\sum_{i\in C_j}(f^*_{C_j}(x_i)-u_i)^2+\frac{4\kappa^2}{\nu|C_j|}\Big(\frac{\kappa}{\sqrt{\nu}}+1\Big)^2\\+\Big(\frac{8\kappa^2}{\nu}+1\Big)\Big(\frac{\kappa}{\sqrt{\nu}}+1\Big)^2\sqrt{\frac{\ln(1/\varepsilon)}{2|C_j|}}.\end{aligned}$$

For simplicity, we approximate $M\approx p_h\lambda b$. From Lemma 1, if λ is sufficiently large, we have for any region C, $(1-\varepsilon)\mu(C)\le N(C)\le(1+\varepsilon)\mu(C)$ with high probability.

Therefore, by choosing λ to be large enough, with probability at least $1-\varepsilon$, where $\varepsilon \in (0,1)$, we have for all $j = 1, \ldots, M$,

$$R_j \leq \frac{1}{\lambda(1-\varepsilon)v_d\delta^d} \sum_{i \in C_j} (f^*_{C_j}(x_i) - u_i)^2 + \frac{4\kappa^2}{\nu\lambda(1-\varepsilon)v_d\delta^d}(\frac{\kappa}{\sqrt{\nu}}+1)^2 + (\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{\ln(p_h\lambda b/\varepsilon)}{2\lambda(1-\varepsilon)v_d\delta^d}}.$$

Using the probability union bound, we have with probability at least $1-\varepsilon$,

$$\tilde{\mathcal{E}}_{\mathcal{D}} \leq \frac{1}{\lambda(1-\varepsilon)bM}\sum_i (f^*(x_i)-u_i)^2 + \frac{4\kappa^2}{\nu\lambda(1-\varepsilon)b}(\frac{\kappa}{\sqrt{\nu}}+1)^2 + \frac{1}{b}(\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{v_d\delta^d\ln(p_h\lambda b/\varepsilon)}{2\lambda(1-\varepsilon)}}. \quad (15)$$

Furthermore, Lemma 23 of [35] yields

$$(f^*_{C_j}(x)-u)^2 \leq (\frac{\kappa}{\sqrt{\nu}}+1)^2,$$

for all $j = 1, \ldots, M$, $x \in \mathcal{D}$, and $u \in \{-1, 1\}$. This implies that with probability one, $\tilde{\mathcal{E}}_{\mathcal{D}}$ is upper bounded by the right hand side of (15) plus $\varepsilon(\kappa/\sqrt{\nu}+1)^2$.

We aim to find $\lambda > 0$ that minimizes a weighted sum of the communication cost upper bound (12) and the estimation error upper bound given by (13) and (15). The objective function to be minimized is given by

$$C(\lambda) + \beta\left(\frac{F_3}{\lambda} + F_4\sqrt{\frac{\ln\lambda}{\lambda}}\right) = F_1\lambda^2 + F_2\lambda + \beta\left(\frac{F_3}{\lambda} + F_4\sqrt{\frac{\ln\lambda}{\lambda}}\right), \quad (16)$$

where $\beta > 0$ is a constant, and

$$F_1 = p_h v_d \delta^d G(\delta),$$
$$F_2 = p_h p_B t_{\max} G(2\delta),$$
$$F_3 = \frac{4\kappa^2}{\nu}(\frac{\kappa}{\sqrt{\nu}}+1)^2,$$
$$F_4 = (\frac{8\kappa^2}{\nu}+1)(\frac{\kappa}{\sqrt{\nu}}+1)^2\sqrt{\frac{(1-\varepsilon)v_d\delta^d}{2}}.$$

We have made the approximation that $\ln(p_h b/\varepsilon)/\lambda \approx 0$ when λ is sufficiently large so that the value of b need not be known a priori. The optimal rate can be found by setting the derivative with respect to λ of (16) to zero (it is clear that there is a positive minimizer) to obtain

$$4F_1\lambda^{\frac{5}{2}} + 2F_2\lambda^{\frac{3}{2}} + \beta(-2F_3\lambda^{-\frac{1}{2}} + F_4((\ln\lambda)^{-\frac{1}{2}} - (\ln\lambda)^{\frac{1}{2}})) = 0,$$

the solution of which can be computed numerically. To find the optimal SU density that minimizes the communication cost subject to the constraint that the estimation error is below a given level is equivalent to (16), where β is a Lagrange multiplier.

B. Setup Complexity and Throughput

Let $\hat{\mathcal{R}}$ be the estimator for $\mathcal{R}$ produced by the DBE algorithm. Those SUs outside of $\hat{\mathcal{R}}$ can utilize the spectrum without performing spectrum sensing, while SUs inside of $\hat{\mathcal{R}}$ perform collaborative spectrum sensing by sending their local sensing decisions to a fusion center. For convenience, we call our two-step approach the DBE-spectrum sensing (DBE-SS) method. We analyze the complexity and throughput of the DBE-SS method and the traditional FC method, where all SUs send their local sensing decisions to a fusion center. In the FC method, SUs do not know the boundary of $\mathcal{R}$, therefore the spectrum is utilized by the SUs only if the fusion center declares that the PU is inactive.

We first consider the complexity of performing boundary estimation using the DBE algorithm. Recall that each SU nominates itself to be a cluster head with probability p_h, and each cluster is covered by a disk of radius δ. Therefore, there are on average $O(\delta^d)$ SUs in a cluster and line 4 in the DBE algorithm has complexity $O(\delta^{2d})$. The optimization problem (7) can be viewed as a convex quadratic program with $O(\delta^d)$ constraints, with complexity $O(\delta^d\delta^{3d}) = O(\delta^{4d})$. The expected number of boundary clusters is bounded by $O(p_h N)$, therefore the overall expected complexity of the DBE algorithm is $O(p_h N\delta^{4d})$. On the other hand, in the FC approach, SUs route their local decisions to a fusion center using a minimum spanning tree (MST). If we assume that the underlying communication network is formed by joining any two SUs that are within distance δ of each other, then the complexity of setting up a MST (with global knowledge of the whole network topology) is $O(N\delta^d)$. Clearly, the DBE algorithm has higher complexity than the fusion center setup if $p_h > \delta^{-3d}$.

We now compare the throughput of the DBE-SS method with that achieved by the FC approach. We make several assumptions to simplify the analysis. Suppose that all SUs transmit at the same power $P_s < P_0$, and that in any given area, at most a fraction q of the SUs can share the spectrum. We assume additive white Gaussian noise channels with noise power N_0. We also assume that interference between the PU and the SUs outside of $\mathcal{R}$ is negligible, while the throughput for SUs in $\mathcal{R}$ when the PU is active is negligible. Then, the throughput of a SU in the absence of the PU is [36] $\bar{R}_0 = \log_2\left(1+\frac{P_s}{N_0}\right)$. We further assume that each SU uses a periodic frame structure of duration T, which includes a sensing duration of τ. Suppose that the PU is active with probability π. Then, the average throughput per user under the FC method is

$$R_s^{\text{FC}} = (1-\pi)q\bar{R}_0(1-\frac{\tau}{T})(1-Q_f^{\text{FC}}),$$

where Q_f^{FC} is the false alarm rate at the fusion center. As the fusion center does not know the receiver operating characteristic (ROC) of each sensor, it uses a simple k-out-of-N rule to fuse the SUs' local decisions. The false alarm is given by[7]

$$Q_f^{\text{FC}} = \sum_{i=1}^{k}\binom{qN}{i}\alpha^i(1-\alpha)^{qN-i}. \quad (17)$$

[7]To avoid cluttered expressions, we assume that qN is always an integer.

The value of k is chosen so that the probability of detection $Q_d^{\text{FC}} \geq \bar{\beta}$, for some fix threshold $\bar{\beta}$.

On the other hand, the average throughput per user for the SUs under DBE-SS is

$$R_s = (1-\pi)q\bar{R}_0\left(\frac{|\hat{\mathcal{R}}|}{N}(1-\frac{\tau}{T})(1-Q_f)+1-\frac{|\hat{\mathcal{R}}|}{N}\right) + \pi q\bar{R}_0\left(1-\frac{|\mathcal{R}\cup\hat{\mathcal{R}}|}{N}\right),$$

where Q_f is the false alarm rate for the SUs inside $\hat{\mathcal{R}}$, with the same minimum probability of detection $\bar{\beta}$, and can be computed in a similar manner as (17). It can be shown that if $\hat{\mathcal{R}} \approx \mathcal{R}$ and $|\mathcal{R}|/N$ is sufficiently small, then $R_s > R_s^{\text{FC}}$. This is intuitively clear as there is little benefit to perform boundary estimation if $\mathcal{R}$ covers almost all the SUs.

Next, we derive the throughput for the PU. In the FC approach, the PU average throughput is

$$R_p^{\text{FC}} = \pi\left(Q_d^{\text{FC}}\log_2\left(1+\frac{P_0}{N_0}\right) + (1-Q_d^{\text{FC}})\log_2\left(1+\frac{P_0}{N_0+qNP_s}\right)\right),$$

where Q_d^{FC} is the detection probability at the fusion center. Because of estimation errors in the DBE-SS scheme, $\mathcal{R}\backslash\hat{\mathcal{R}}$ may be non-empty. Interference from SUs in this set is bounded by $N_1 = q|\mathcal{R}\backslash\hat{\mathcal{R}}|P_s$. The PU average throughput under the DBE-SS scheme is then given by

$$R_p = \pi\left(Q_d\log_2\left(1+\frac{P_0}{N_0+N_1}\right) + (1-Q_d)\log_2\left(1+\frac{P_0}{N_0+q|\mathcal{R}|P_s}\right)\right),$$

where $Q_d \geq \bar{\beta}$ is the detection probability for the SUs inside $\hat{\mathcal{R}}$. In Section V, we present simulation results to compare the throughput R_s under the DBE-SS scheme with R_s^{FC} under the FC method, when $R_p = R_p^{\text{FC}}$.

V. Simulation Results

In this section, we present simulation results to verify the performance of the DBE algorithm and the DBE-SS method. In each simulation run, 1000 sensors are uniformly distributed in a region A of size 5×5 km^2, with the PU (e.g., a TV transmitter) located at the center of the region. We use the standard CCIR model [37] for the path loss. For each data point, we perform 1000 simulation runs using the parameters in Table III.

TABLE III
Simulation Parameters

Parameters	Value
PU transmit power P_0	40 dBm
PU transmitter antenna height	30 m
PU transmit antenna gain	6 dBd
PU active probability π	0.3
SU transmit power P_s	10 dBm
SU transmitter antenna height	3 m
SU transmit antenna gain	0 dBd
SU coverage radius δ	0.25 km
SU false alarm probability α	0.05
interference threshold θ_0	-75 dBm
shadow fading standard deviation σ	6 dB
frame duration T	20 ms
sensing time τ	1 ms
learning parameters $\nu_1 = \ldots = \nu_M$	1
learning parameters $\eta_1 = \ldots = \eta_M$	10
learning kernel $K(x,x')$	$e^{-0.5\|x-x'\|^2}$
communication cost function $g(r)$	r^2
noise power N_0	-10 dBm
estimation error parameter ε	0.01
trade-off weight β	500

A. Estimation Error and Communication Cost

We compare the communication cost incurred and the estimation performance of the DBE algorithm with that of various benchmark algorithms, including the following:

1) Centralized boundary estimation algorithm based on LS-SVM [12]: a global classifier is trained based on information from all SUs in the boundary clusters.
2) Centralized image processing based seeded region growing (SRG) algorithm [38]: we regard the decision u_i of each SU i as a pixel gray level in a binary image and segment the image by growing a region from a seed point using an intensity mean measure.
3) Distributed Bayesian event region detection (ERD) algorithm [15], [16]: a threshold decision scheme is applied to correct the errors of local SU decisions. We refer the reader to [15] for details.

The estimation performance is evaluated according to (2), normalized by four times the area of $\mathcal{R}$. Since the estimation function f takes values close to 1 or -1, the normalized estimation error is approximately the area in which misclassification occurs, expressed as a fraction of the area of $\mathcal{R}$. The communication cost is computed by assuming that each message passed between two SUs a distance r apart incurs a cost of $g(r) = r^2$.

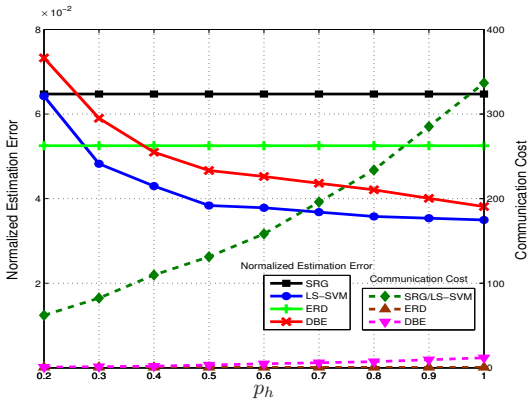

Fig. 2. Normalized estimation errors and total communication costs for different values of p_h when $\gamma = 0.6$.

Figure 2 shows the normalized estimation error and communication cost for each algorithm when choosing different values for p_h, which is the probability that each SU independently nominates itself to be a cluster head. The threshold γ in the boundary cluster decision rule in Section III-A is set to be 0.6. As p_h increases, the performance of the SRG and ERD algorithms remain constant as these algorithms do

not use clustering. The performance of our proposed DBE algorithm on the other hand, becomes better than the SRG and the ERD algorithms, but still underperforms the centralized LS-SVM algorithm. Centralized methods like LS-SVM and SRG however have higher communication costs than the ERD and DBE algorithms as only short range communications are required for the latter algorithms. We see that the DBE algorithm achieves arguably the best trade-off between the estimation error and communication cost if p_h is chosen to be sufficiently large.

We now set the probability $p_h = 0.8$ and vary the threshold γ in the boundary cluster decision rule, with a higher value of γ corresponding to more clusters being chosen as the boundary clusters. We see from Figure 3 that although the estimation error decreases with increasing γ, the rate of decrease is not very significant. This is because most of the actual boundary clusters have already been included for reasonable values of γ. This implies that in practice, a reasonably small value of γ can be chosen to reduce the communication costs incurred during boundary estimation.

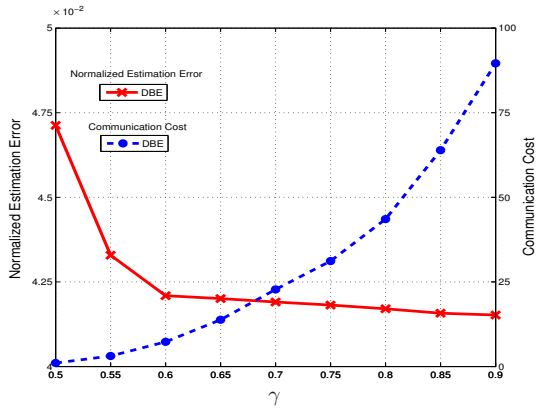

Fig. 3. Normalized estimation errors and total communication costs when $p_h = 0.8$, and the threshold γ is varied.

We next fix $p_h = 0.8$ and $\gamma = 0.6$, and vary the SU density from 5 to 160 SU per km^2. In Figure 4, we use simulation to compute the sum of the communication cost and estimation error (2) weighted by β. For comparison, we also plot the cost function $C(\lambda)$ in (16). Although $C(\lambda)$ does not include the estimation errors incurred in clusters far away from the boundary of $\mathcal{R}$, it is seen that it still serves as an upper bound to the simulated cost as estimation errors in clusters located far from the boundary are very small. We see from Figure 4 that the simulated and theoretical optimal SU densities are nearly identical to each other.

In Figure 5, we show the convergence of the normalized estimation error with respect to the number of inter-cluster communications for a particular simulation run with 70 boundary clusters. We see that for the DBE algorithm the estimation error converges within 200 messages (or equivalently about 3 passes over all boundary clusters). This shows that in practice, $t_{\max}$ in the DBE algorithm can be chosen to be a reasonably small value. The randomized DBE algorithm on the other hand requires a much larger number of inter-cluster communications, but less coordination amongst the boundary cluster heads.

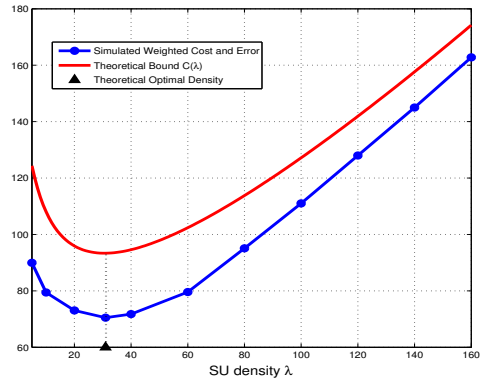

Fig. 4. Weighted sum of communication cost and estimation error when $\gamma = 0.6$ and $p_h = 0.8$, with varying SU density.

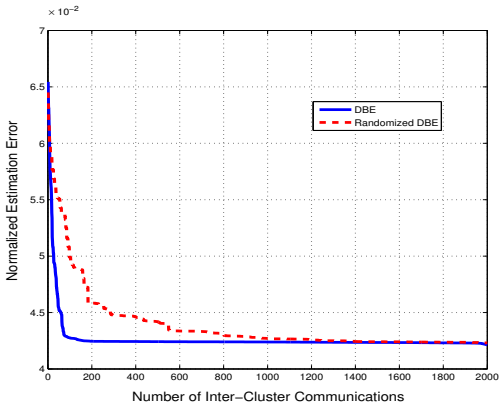

Fig. 5. Normalized estimation error versus number of inter-cluster communications.

B. Throughput

In this subsection, we present numerical results for the ROCs and throughputs of the FC and DBE-SS methods after boundary estimation with $p_h = 0.8$ and $\gamma = 0.6$. Recall that the fusion center has no knowledge of the ROC of individual SUs, and a simple k-out-of-N fusion rule is utilized in place of optimal fusion. Figure 6 shows the ROC curves of the two methods. It is seen that the DBE-SS method has a higher detection probability for each false alarm probability because only information from SUs in $\mathcal{R}$ are utilized, leading to less errors.

In Figure 7, we vary the detection probability and plot the PU throughput versus the throughput per SU for both DBE-SS and FC methods. The throughput per SU for the DBE-SS method is relatively flat over all PU throughputs as SUs outside $\hat{\mathcal{R}}$ can transmit regardless of whether the PU is present or not. We also see that the SU throughput is higher than that for the FC method. Figure 8 shows the average SU throughput when the PU throughput is fixed at 4 bits/sec/Hz, and the volume of A is decreased. We see that the DBE-SS method should only be used if A is more than 10% larger than $\mathcal{R}$.

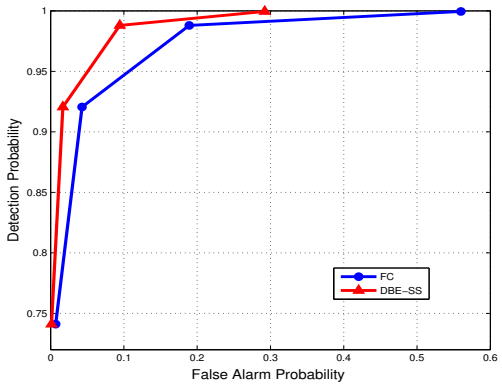

Fig. 6. ROC curves under FC and DBE-SS.

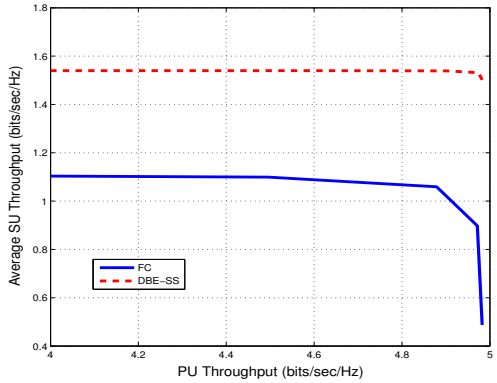

Fig. 7. PU throughput and throughput per SU under FC and DBE-SS.

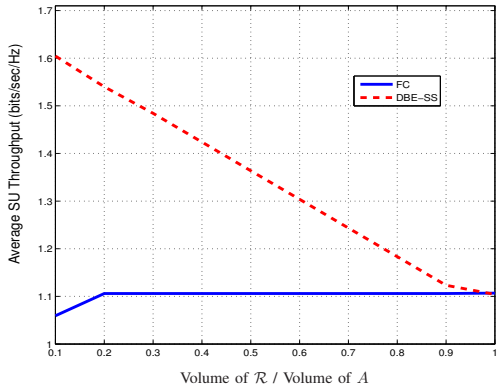

Fig. 8. Average throughput per SU as volume of A changes.

C. Robustness

We now compare the robustness of the various boundary estimation algorithms. We fix $p_h = 0.8$ and $\gamma = 0.6$. To simulate SU sensing errors, a boundary cluster is randomly chosen with probability ς, and then a random subset of the SU sensing decisions in the chosen cluster is changed from -1 to 1, while an equal number of SU sensing decisions is changed from 1 to -1. We plot the average normalized estimation error in Figure 9. Figure 9 shows that our proposed DBE algorithm is more robust than the other benchmark boundary estimation methods, except for the centralized LS-SVM method. We also compare with a modified version of the DBE algorithm in which we set $\eta_j = 0$ for all $j = 1, \ldots, M$ so that the smoothness constraint (6) no longer applies. We see that including the smoothing constraint improves the robustness of our algorithm as neighboring boundary clusters moderate their local classifiers to avoid an abrupt change in the average classification function value within their clusters.

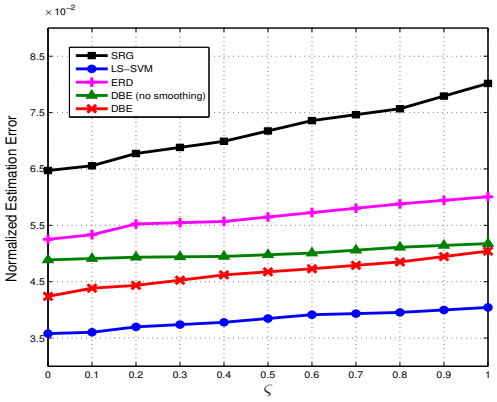

Fig. 9. Robustness comparison of boundary estimation algorithms after flipping the observations.

Next, we compare the estimation error of the DBE algorithm with and without the smoothness constraint (6) when the SUs are no longer distributed as a homogeneous Poisson point process. With probability ω, a boundary cluster is independently populated with 20 SUs uniformly distributed inside the cluster. With probability $1-\omega$, a boundary cluster is divided into four quadrants, and a quadrant is chosen randomly. The chosen quadrant is then populated with 20 SUs uniformly. In Figure 10, we see that the smoothing constraint results in a lower estimation error. Figure 11 shows a portion of the estimated boundaries.

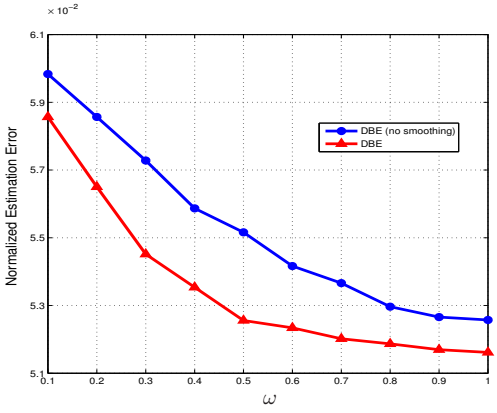

Fig. 10. Robustness comparison of DBE algorithm with and without smoothing constraints after changing the density.

VI. Conclusion

We have developed a distributed boundary estimation algorithm for estimating the no-talk region of a PU in a cognitive radio network, and analyzed the trade-offs between the communication cost and estimation error of our proposed method. We derive approximate upper bounds for the communication

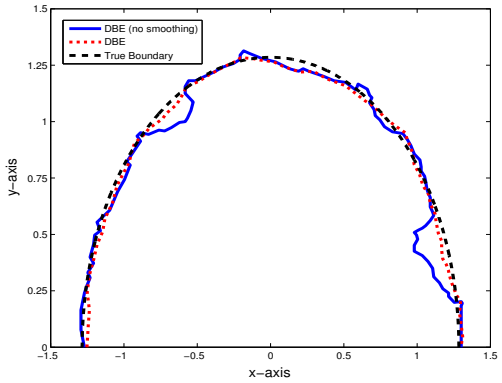

Fig. 11. Estimated boundaries with and without smoothing constraints.

cost and estimation error, and provide a method to compute the optimal SU density. Simulation results suggest that our proposed algorithm have lower estimation errors and better robustness compared to various other methods.

We have made various simplifying and heuristic assumptions in deriving the optimal SU density. Our simulations however shows that despite these assumptions, the theoretical optimal SU density found is not significantly different from the actual one. Bounding the estimation error more precisely remains a difficult open problem. In this work, we have also assumed that the PU is static and the received interference of SUs at the PU do not vary over time. Future work includes developing boundary estimation techniques for the cases where PUs are mobile and where communication channels are time variant.

REFERENCES

[1] J. Unnikrishnan and V. V. Veeravalli, "Cooperative sensing for primary detection in cognitive radio," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 18–27, 2008.

[2] Z. Li, F. R. Yu, and M. Huang, "A cooperative spectrum sensing consensus scheme in cognitive radios," in *Proc. IEEE 28th Annual Joint Conf. of the IEEE Computer and Communications Societies*, 2009, pp. 2546–2550.

[3] W. Zhang and K. Letaief, "Cooperative spectrum sensing with transmit and relay diversity in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4761–4766, 2008.

[4] R. Tandra, S. M. Mishra, and A. Sahai, "What is a spectrum hole and what does it take to recognize one?" *Proc. IEEE*, vol. 97, no. 5, pp. 824–848, 2009.

[5] R. Tandra, A. Sahai, and V. Veeravalli, "Unified space-time metrics to evaluate spectrum sensing," *IEEE Commun. Mag.*, vol. 49, no. 3, pp. 54–61, 2011.

[6] M. Vu, N. Devroye, and V. Tarokh, "On the primary exclusive region of cognitive networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 7, pp. 3380–3385, 2009.

[7] Z. Wei, Z. Feng, Q. Zhang, and W. Li, "Three regions for space-time spectrum sensing and access in cognitive radio networks," in *Proc. IEEE Global Communications Conf.*, 2012, pp. 1283–1288.

[8] Q. Li, Z. Feng, W. Li, and T. A. Gulliver, "Joint temporal and spatial spectrum sharing in cognitive radio networks: A region-based approach with cooperative spectrum sensing," in *Proc. IEEE Wireless Communications and Networking Conf.*, 2013, pp. 620–625.

[9] S. P. Fekete, A. Kröller, D. Pfisterer, S. Fischer, and C. Buschmann, "Neighborhood-based topology recognition in sensor networks," in *Proc. 1st Int. Workshop on Algorithms for Sensor Systems*, 2004, pp. 123–136.

[10] R. Ghrist and A. Muhammad, "Coverage and hole-detection in sensor networks via homology," in *Proc. 4th Int. Symp. Information Processing in Sensor Networks*, 2005, pp. 254–260.

[11] Y. Wang, J. Gao, and J. S. Mitchell, "Boundary recognition in sensor networks by topological methods," in *Proc. 12th Annual Int. Conf. on Mobile Computing and Networking*, 2006, pp. 122–133.

[12] J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Processing Letters*, vol. 9, no. 3, pp. 293–300, 1999.

[13] Y. Yang, Y. Liu, Q. Zhang, and L. Ni, "Cooperative boundary detection for spectrum sensing using dedicated wireless sensor networks," in *Proc. IEEE 29th Annual Joint Conf. of the IEEE Computer and Communications Societies*, 2010, pp. 1–9.

[14] K. K. Chintalapudi and R. Govindan, "Localized edge detection in sensor fields," in *Proc. IEEE 1st Int. Workshop on Sensor Network Protocols and Applications*, 2003, pp. 59–70.

[15] B. Krishnamachari and S. Iyengar, "Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks," *IEEE Trans. Comput.*, vol. 53, no. 3, pp. 241–250, 2004.

[16] Q. Chen, K.-Y. Lam, and P. Fan, "Comments on "Distributed Bayesian algorithms for fault-tolerant event region detection in wireless sensor networks"," *IEEE Trans. Comput.*, vol. 54, no. 9, pp. 1182–1183, 2005.

[17] R. Nowak and U. Mitra, "Boundary estimation in sensor networks: Theory and methods," in *Proc. Information Processing in Sensor Networks*, 2003, pp. 80–95.

[18] S. Duttagupta, K. Ramamritham, and P. Kulkarni, "Tracking dynamic boundaries using sensor network," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 10, pp. 1766–1774, 2011.

[19] J. B. Predd, S. R. Kulkarni, and H. V. Poor, "A collaborative training algorithm for distributed learning," *IEEE Trans. Inf. Theory*, vol. 55, no. 4, pp. 1856–1871, 2009.

[20] S. Tozlu, M. Senel, W. Mao, and A. Keshavarzian, "Wi-fi enabled sensors for internet of things: A practical approach," *IEEE Commun. Mag.*, vol. 50, no. 6, pp. 134–143, 2012.

[21] D. Cabric, I. O'Donnell, M.-W. Chen, and R. Brodersen, "Spectrum sharing radios," *IEEE Circuits Syst. Mag.*, vol. 6, no. 2, pp. 30–45, 2006.

[22] R. Tandra and A. Sahai, "SNR walls for signal detection," *IEEE J. Sel. Topics Signal Process.*, vol. 2, no. 1, pp. 4–17, 2008.

[23] R. Durrett, *Probability: Theory and Examples*, 3rd ed. Belmont, CA: Duxbury, 2004.

[24] V. N. Vapnik, *Statistical learning theory*. Wiley, New York, 1998.

[25] R. Rom and M. Sidi, *Multiple Access Protocols: Performance and Analysis*. Springer London, 2011.

[26] I. Steinwart, D. Hush, and C. Scovel, "An explicit description of the reproducing kernel Hilbert spaces of Gaussian RBF kernels," *IEEE Trans. Inf. Theory*, vol. 52, no. 10, pp. 4635–4643, 2006.

[27] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms, and Applications*. New York: Oxford University Press, 1997.

[28] B. Schölkopf, R. Herbrich, and A. J. Smola, "A generalized Representer Theorem," in *Proc. 14th Annual Conf. on Computational Learning Theory and 5th European Conf. on Computational Learning Theory*, 2001, pp. 416–426.

[29] J. J. Caffery, *Wireless Location in CDMA Cellular Radio Systems*. Boston : Kluwer Academic, 2000.

[30] C. Badea, S. Grivaux, and V. Mï£¡ller, "The rate of convergence in the method of alternating projections," *St. Petersburg Math. J*, vol. 23, pp. 413–434, 2012.

[31] L. Debnath and P. Mikusinski, *Introduction to Hilbert spaces with applications*, 2nd ed. San Diego, CA: Academic Press, 1999.

[32] M. Z. Win, P. C. Pinto, and L. A. Shepp, "A mathematical theory of network interference and its applications," *Proc. IEEE*, vol. 97, no. 2, pp. 205–230, 2009.

[33] C. han Lee and M. Haenggi, "Interference and outage in poisson cognitive networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 4, pp. 1392–1401, 2012.

[34] D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic Geometry and its Applications*. John Wiley & Sons, Chichester, 1995.

[35] O. Bousquet and A. Elisseeff, "Stability and generalization," *Journal of Machine Learning Research*, vol. 2, pp. 499–526, 2002.

[36] Y.-C. Liang, Y. Zeng, E. C. Y. Peh, and A. T. Hoang, "Sensing-throughput tradeoff for cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 4, pp. 1326–1337, 2008.

[37] W. C. Y. Lee, "Lee's model [cellular radio path loss prediction]," in *Proc. IEEE 42nd Vehicular Technology Conf.*, 1992, pp. 343–348.

[38] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and M. Walid G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE Trans. Image Process.*, vol. 10, no. 10, pp. 1454–1466, 2001.

Response to Reviewers' Comments

Reply to the reviewers' comments on "Distributed Boundary Estimation for Spectrum Sensing in Cognitive Radio Networks" (Manuscript I.D. 1569812379).

We would like to thank all the reviewers for their insightful, constructive and professional comments. We have revised the manuscript according to the reviewers' comments and we believe that this current revision contains enough contributions to be considered in JSAC-CRS. Attached below please find our responses to the suggestions and questions from the reviewers.

Response to Editor Review 1

Recommendation: Accept with minor revision

Comments: *The paper has improved after the revision and most of the issues raised by reviewers are resolved. However, there are two remaining comments that need to be addressed:*

Comment 1. *How does your approach deal with the case of multiple PUs which are uncorrelated but operate in the same frequency band. How do you distinguish them by using only energy detection?*

Response 1. Thank you for your comment. We have added a brief discussion on the case where there are multiple PUs in Section II, which we reproduce below for your convenience. When there are multiple PUs, we are interested in finding the union of all PUs' no-talk regions since as long as a SU is within the no-talk region of at least one PU, it has to perform temporal spectrum sensing in order to opportunistically utilize the PU spectrum. There is no need to distinguish the PUs explicitly.

" When there are multiple PUs transmitting in the same spectrum, the no-talk region is the union of all PUs' no-talk regions. Suppose there are $N' > 1$ PUs, and that all PU signals are uncorrelated. Let $Y_{ip}[j]$ be the signal sample received by SU i from the PU p in the interval j. For a sufficiently large J and $p \neq p'$, we have $\frac{1}{J}\sum_{j=1}^{J} Y_{ip}[j]Y_{ip'}[j] \approx 0$, and the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |\sum_{p=1}^{N'} Y_{ip}[j]|^2$ can be approximated as $\frac{1}{J}\sum_{p=1}^{N'}\sum_{j=1}^{J} |Y_{ip}[j]|^2$. A threshold for T_i similar to that in (1) can be found to determine if SU i is within the no-talk region of at least one PU. "

Comment 2. *Please provide related literature survey and discuss your contribution in comparison with other related papers.*

Response 2. Thank you for your suggestions. We have added some references on exclusion zone (EZ) estimation for spectrum utilization in Section I:

" In [6]-[8], different spectrum sharing regions, including a primary exclusive region and the no-talk region, are defined. However, all these works assume that the propagation path loss between the PU and SUs are isotropic, and all regions are assumed to be circular. Bounds on the radius of each region are given based on interference and outage considerations, which are characterized in terms of propagation parameters like path loss exponents. In practice, the propagation environment may be very difficult to model quantitatively, and the no-talk region is unlikely to be circular. Therefore, in this work, we develop boundary estimation methods for the no-talk region without relying on extensive assumptions about the shape of the region. "

Response to Reviewer 2

Recommendation: Accept

Comments: *The manuscript has been carefully revised by taking into consideration of the reviewer's previous concern and comments. The readability is improved significantly. The reviewer has no more further revision comments.*

Response 1. Thank you for your positive comments.

Response to Reviewer 3

Recommendation: Accept with minor revision

Comments: *I would like to thank authors for considering all of my comments and revising their paper substantially. My main concern regarding the excessive primary user side information requirements of the proposed technique is alleviated. I think the paper has improved a lot technically and deserves publication. My suggestion is to accept the paper with a minor revision as follows:*

Comment 1. *Your proposed scheme seems related to the exclusion zone (EZ) estimation approaches. There are a couple of major papers related to the EZ in cognitive radio systems, I think it shall improve your literature survey section, if you discuss some of the related works in EZ and describe what you have done more.*

Response 1. Thank you for your suggestions. We have added some references on exclusion zone (EZ) estimation for spectrum utilization in Section I:

" In [6]-[8], different spectrum sharing regions, including a primary exclusive region and the no-talk region, are defined. However, all these works assume that the propagation path loss between the PU and SUs are isotropic, and all regions are assumed to be circular. Bounds on the radius of each region are given based on interference and outage considerations, which are characterized in terms of propagation parameters like path loss exponents. In practice, the propagation environment may be very difficult to model quantitatively, and the no-talk region is unlikely to be circular. Therefore, in this work, we develop boundary estimation methods for the no-talk region without relying on extensive assumptions about the shape of the region. "

Response to Reviewer 4

Recommendation: Accept

Comments: *This paper considers the estimation of the no-talk region boundary of secondary users in cognitive radio networks. It is well-presented and the contribution is clear. Authors have addressed the comments of my last review. In this revision, more simulations and descriptions of the validity of the considered system models are provided and make this paper clearer.*

Comment 1. *One more question: if you consider the multiple PUs which are uncorrelated but operating in the same frequency band, how do you distinguish them by only energy detection?*

Response 1. Thank you for your comment. We have added a brief discussion on the case where there are multiple PUs in Section II, which we reproduce below for your convenience. When there are multiple PUs, we are interested in finding the union of all PUs' no-talk regions since as long as a SU is within the no-talk region of at least one PU, it has to perform temporal spectrum sensing in order to opportunistically utilize the PU spectrum. There is no need to distinguish the PUs explicitly.

" When there are multiple PUs transmitting in the same spectrum, the no-talk region is the union of all PUs' no-talk regions. Suppose there are $N' > 1$ PUs, and that all PU signals are uncorrelated. Let $Y_{ip}[j]$ be the signal sample received by SU i from the PU p in the interval j. For a sufficiently large J and $p \neq p'$, we have $\frac{1}{J}\sum_{j=1}^{J} Y_{ip}[j]Y_{ip'}[j] \approx 0$, and the test statistic $T_i = \frac{1}{J}\sum_{j=1}^{J} |\sum_{p=1}^{N'} Y_{ip}[j]|^2$ can be approximated as $\frac{1}{J}\sum_{p=1}^{N'}\sum_{j=1}^{J} |Y_{ip}[j]|^2$. A threshold for T_i similar to that in (1) can be found to determine if SU i is within the no-talk region of at least one PU. "