Sensor Placement Algorithm for Radio Environment Map Construction in Cognitive Radio Networks

H. Birkan Yilmaz, Chan-Byoung Chae, and Tuna Tugcu

* a School of Integrated Technology, Yonsei University, Korea
b NETLAB, Dept. of Computer Engineering, Bogazici University, Bebek, Istanbul, Turkey

E-mail: {birkan.yilmaz@yonsei.ac.kr, cbchae@yonsei.ac.kr, tugcu@boun.edu.tr}

Abstract—In cognitive radio, current trend is to utilize geolocation database for TV bands. Considering more dynamic bands in terms of primary user activity, however, necessitates the use of Radio Environment Map (REM), which is an advanced knowledge base that stores live multidomain information on the entities in the network and the environment. In Cognitive Radio Networks (CRNs), mobile nodes that are capable of measuring the energy of the frequency bands are less capable compared to dedicated sensing nodes in the network. Therefore, deployment algorithm of the dedicated sensor nodes is of great importance and affects the constructed REM interference map quality. We propose a novel deployment algorithm for CRNs that considers user distribution probabilities. Numerical results confirm that the proposed deployment algorithm significantly improves the REM performance. The proposed algorithm is compared with random deployments and it is applied on Kriging and LiVE REM construction techniques.

Index Terms—Cognitive radios, radio environment map.

I. INTRODUCTION

Research on cognitive radio (CR) has matured from early work on the core idea of CRs to current multi-dimensional proposals such as signal processing techniques, machine learning algorithms, and regulations. CR is the prospective solution for the long debated spectrum under-utilization problem. The rules demanded by the regulators, such as the detection of a primary signal at -114 dBm by FCC, are highly conservative [1]. Instead, accessing the centralized database, which stores available spectrum based on geographical coordinates or the properties of registered primary transmitters, has been accepted more promising for mitigating these challenges while meeting the regulatory obligations. Currently, database-based TV White Space (TVWS) operation is being considered by the regulators in the US and the UK.

Since TV bands are quasi-static, database-based approaches are appropriate and applicable. For more dynamic bands, however, these database-based solutions (geolocation database) must be improved. A step further from geolocation database lies Radio Environment Map (REM). REM was first defined as an abstraction of a real-world environment storing multidomain information [2]. Contrary to the geolocation DB that stores only the quasi-static Primary User (PU) related information, REM is an advanced knowledge base that stores live multi-domain information on the entities in the network as well as the environment historically [3]. Therefore, REM can act as an enabler for cognition in radios [4].

Essential functionality of a REM is constructing dynamic interference map for each frequency at each location of interest, which is volatile data. Because it is impractical to have measurements at each location in the CR operation area, REM fuses the available measurements to estimate the interference level at locations with no measurement data. Radio interference map interpolation is the most critical part of a REM construction method. On top of estimated interference levels, one can easily apply a thresholding for detecting the no-talk zone when there is an active PU [5].

REM data sources are not defined in the literature specifically. Collecting measurements for constructing the REM can be done via mobile CRs and dedicated sensors. However, mobile devices are less capable devices and must use their battery efficiently. Therefore, deploying dedicated sensors for increasing the quality of the REM must be considered. In [5], the effect of sensor geometries on PU and the environmental parameter estimation are studied. Distributed spectrum sensing for Cognitive Radio Networks (CRNs) by exploiting sparsity is proposed in [6]. Similar studies in sensor networks and concentrator location problems can be seen. To the best of our knowledge, however, the sensor placement analysis in REM domain is a virgin research area.

While placing the dedicated sensors, user distribution of the area must be considered and the areas where the mobile devices are sparse must be patched via more capable devices. In this paper, we propose a sensor placement algorithm for CRNs for constructing better REM interference maps. Main idea of the proposed algorithm is to patch the uncovered locations via dedicated sensors. K-means clustering algorithm is utilized while finding appropriate locations for dedicated sensor deployment. The proposed deployment algorithm improves the REM construction performance significantly by utilizing user distribution probabilities on the area for finding uncovered locations.

II. SYSTEM MODEL

We consider an environment in which the primary network reuses every frequency in distant cells to avoid interference between primary users. This approach allows us to consider every active transmission for a single channel in the area of interest at a specific time. Secondary Users (SUs) are the mobile cognitive users that are responsible for sensing and
reporting of measurements, where the PU is the licensed user of a specific bandwidth shared with the SUs.

CRs and the dedicated sensors send their measurements to REM manager to estimate the real situation in terms of Received Signal Strength (RSS). In Fig. 1, CRs and a dedicated sensor are represented as handheld devices and a pentagon, respectively.

REM manager collects each node’s location information and RSS value, i.e. \((x, y, \text{RSS})\), from the measuring nodes and constructs the radio interference map via estimating the RSS values at locations without any measurements. RSS values depend on the characteristics of the propagation environment. We consider the log-normal shadowing environment for slow fading channel analysis as assumed in the literature [7].

### A. RSS Measurements in a Slow Fading Channel

Long-term (slow) fading arises when the coherence time of the channel is large relative to the delay constraint of the channel. In this regime, the amplitude and phase change imposed by the channel can be considered roughly constant over the period of use. Slow fading can be caused by events such as shadowing, where a large obstruction such as a hill or large building obscures the main signal path between the transmitter and the receiver. The amplitude change caused by shadowing is often modeled using a log-normal distribution.

Under log-normal channel, RSS measurements in decibel (dB) obey normal interference map; hence, the formulations in decibels are more appropriate. The ideal RSS at the \(i\)th node, denoted by \(P_{rx}^{i}\), is expressed as

\[
P_{rx}^{i} = P_{tx}^{i} - P_{L_0} - 10 \log_{10} d_{(x,y)}^{2} + S_i + e_i
\]

where \(P_{L_0}\) and \(\alpha\) are reference path loss and path loss exponents, respectively. The distance between the active user and the \(i\)th secondary user or the dedicated sensor is denoted by \(d_{(x,y)}\). Parameters \(S_i\) and \(e_i\) are Gaussian random variables with mean zero and variance \(\sigma^2_s\) and \(\sigma^2_e\) for expressing the effect of log-normal shadowing and measurement error, respectively. Especially, \(e_i\) may change from device to device. For example, mobile CR nodes may have less sensitive sensing units (higher \(e_i\)) that results in less accurate measurements.

RSS values at each node have severe disturbances caused by the shadowing effect, so we measure raw RSS values and evaluate sample mean to reduce the shadowing effect. Mean received power, \(\overline{P_{rx}^i}\), formulation is given by

\[
\overline{P_{rx}^i} = \frac{\sum_{j=1}^{N_m} P_{rx}^{i,j}}{N_m},
\]

where \(P_{rx}^{i,j}\) is the \(j\)th measured RSS value (in terms of dBW) at the \(i\)th node and \(N_m\) is the total number of samples for unit measurement. Effective \(N_m\) may depend on the sensing node capabilities and the rate of mobility. For example, dedicated sensors may have higher \(N_m\) values for achieving more accurate measurements.

### B. Architecture and Topology

An active PU on a specific frequency, \(f_0\) and SUs existing in a rectangular region are sketched in Fig. 2. Some regions attract mobile users more than other regions due to the point of interest existing in that region. This information can easily be obtained by location servers that analyze the user locations in the network. The proposed algorithm is run at new deployment plans and the user distribution of the areas are queried from the network entities for using as an input to the algorithm. Fig. 2 depicts a scenario where the region on the left has more mobile CRs compared to other regions.

We consider a rectangular area, and due to the location of point of interests, we have heterogeneous distribution of mobile CRs. We model the area consisting of polygons with probabilities of containing the mobile users. We denote the user distribution as

\[
\mathcal{U}_d = \{(A_i, p_i)|i = 1...n\},
\]

where \(A_i, p_i\) are the ordered set of points describing a polygon and the probability of having mobile node in that polygon, respectively. Mobile CRs are distributed to the area according to the given \(\mathcal{U}_d\). Therefore, knowing \(\mathcal{U}_d\) enables us to cleverly select where to deploy new sensors in the region. In Fig. 2, maximum circle area is pinned and it is the biggest circular area lacking measurements that has potential to increase the performance significantly. Therefore, the unoccupied areas in...
terms of sensing nodes should be considered for placing a new dedicated sensor to collect information about the proximity.

C. Background Information on REM Construction

REM was first defined as an abstraction of a real-world environment storing multi-domain information [2]. In more general terms, it can also be considered as an intelligent network entity that can further process the gathered information, inspect the spatio-temporal characteristics, and derive a map of the RF environment [3].

REM construction is a broad concept, referring to creating a complete map of CRN coverage area. However, we will only focus on deriving interference level at each “pixel” of the CRN. In the literature, REM construction techniques can be broadly put into two classes: spatial statistics based methods and transmitter location determination based methods, which are also referred to as direct and indirect methods, respectively [8]. We consider Kriging and Location Estimation based (LiVe) REM construction techniques in this paper that are representatives of direct and indirect methods, respectively. Details about LiVe and Kriging can be found in [9].

Kriging basically weights the measurement points to find the estimated value for point \( p \), where weights are the solutions of a system of linear equations that minimize the variance of the error of prediction. The weights used in linear combination depend on spatial correlation derived from semivariogram model of the measurement data. Semivariogram determines the correlation between any two points in the considered system based on their distance separation [10]. Although Kriging requires more measurement points, it is the most commonly applied technique in the literature [11] due to its higher precision.

LiVe consists of two main steps; first the active transmitter location and transmit power are estimated, then the REM construction method is executed by utilizing these estimated values. After appropriate averaging, channel disturbance is minimized. If there is no disturbance in the measurements in (dB), then

$$10\alpha \log_{10} d(x, y) \approx P_{tx} - P_{L0} - P_{MCD}.$$  (4)

Hence, after some manipulations, we get

$$x_i^2 + y_i^2 \approx 2x_{pu}x_i + 2y_{pu}y_i + 10^{\frac{P_{tx} - P_{L0} - P_{MCD}}{10}} - R^2.$$  (5)

where \( R^2 = x_i^2 + y_i^2 \) for \( (x_i, y_i) \) and the \( (x_{pu}, y_{pu}) \) are the locations of the \( i \)th sensing node and active user, respectively. Hence, we can express (5) in the matrix form \( A\theta \approx b \) where

$$A = \begin{pmatrix}
2x_1 & 2y_1 & 10^{-\frac{P_{tx} - P_{L0} - P_{MCD}}{10}} & -1 \\
2x_2 & 2y_2 & 10^{-\frac{P_{tx} - P_{L0} - P_{MCD}}{10}} & -1 \\
. & . & . & . \\
2x_{N_MCD} & 2y_{N_MCD} & 10^{-\frac{P_{tx} - P_{MCD}}{10}} & -1
\end{pmatrix},$$

$$b = \begin{pmatrix}
x_1^2 + y_1^2 \\
x_2^2 + y_2^2 \\
. & . & . \\
x_{N_MCD}^2 + y_{N_MCD}^2
\end{pmatrix}.$$  

In a matrix notation the location and power estimation problem becomes a minimization problem. The estimation problem formulated in the matrix form can be solved easily using the least squares method [12]. Let \( \hat{\theta} \) be the estimated value for \( \theta \). Then, the solution is computed as

$$\hat{\theta} = \arg \min \| A\theta - b \| = (A^TA)^{-1}A^T b$$  (6)

where \( \| A\theta - b \|^2 = (A\theta - b)^T (A\theta - b) \). Hence we first estimate the location and the power of the active user.

After obtaining the estimated values for \( (x_{pu}, y_{pu}) \) and \( P_{tx} \). We proceed with the REM construction method by utilizing these estimated values \( \hat{(x_{pu}, y_{pu})} \) and \( \hat{P}_{tx} \). We evaluate the estimated power at any location in (7) by using the estimated data and channel information.

$$P_{tx}(x, y) = \hat{P}_{tx} - P_{L0} - 10\alpha \log_{10} \hat{d}(x, y)$$  (7)

where \( \hat{d}(x, y) = \sqrt{(x - x_{pu})^2 + (y - y_{pu})^2} \).

D. REM Performance Metrics

How accurate a REM describes the real operation environment is measured by REM quality metrics. These quality metrics can be evaluated through comparing the estimated REM with the true REM. Number of sensing nodes, their distribution in the network and sensing accuracies, dynamics of the propagation environment, and accuracy of the propagation modeling are the key parameters affecting the performance of the REM construction methods.

One of the ways used in the literature to quantify the similarity between the true and the estimated REM is average Root Mean Squared Error (RMSE) calculated over each grid point as the difference in RSS between the estimated REM and the true REM [13]. However, this metric falls short of identifying in which regions the spectrum opportunities are lost because of over-estimating the RSS and in which regions the interference levels are under-estimated that leads to CR collisions with on-going PU transmissions in the region. Hence, Correct Detection Zone Ratio and False Alarm Zone Ratio are introduced in [9].

The areas where the true REM and the estimated REM contours both determine forbidden or allowed zones correctly are called the Correct Detection Zone type-0 (Z_{CD}^0) and Correct Detection Zone type-1 (Z_{CD}^1), respectively. False Alarm Zone \( Z_{FA}^1 \) is the area in which transmission is forbidden due to estimation errors but the area is actually the allowed zone for transmission. Finally, Missed Detection Zone (Z_{MD}) is the area where transmission is not allowed but estimated REM infers the opposite. The regions are normalized according to true REM zones, and determine the Z_{CD}^1 and the Z_{FA}^1 ratios.
Algorithm 1: MAX-CIRCLE for deployment locations

Input: Deployment instance: $D_{i_d}$
Input: Number of sensors for deployment: $m$
Output: Set of deployment points: $S$

1. $S ← ∅$
2. for $k ← 1$ to $m$ do
   3. Find maximum circle center $(x_k, y_k)$ for $D_{i_d}$
   4. $S ← S ∪ \{(x_k, y_k)\}$
   5. $D_{i_d} ← D_{i_d} ∪ \{(x_k, y_k)\}$
6. return $S$

The main idea behind KARMA algorithm is to consider necessary amount of deployments obeying $U_d$ and place the dedicated sensors at maximum circle centers. These dedicated sensor points are clustered using k-means and the centroid of the clusters are good candidates for deployment locations. These centroid points are on the average good patches for fixing empty spaces.

KARMA algorithm utilizes the Max-Circle algorithm and finds the appropriate locations by the following pseudo-code:

Algorithm 1 acts on a given deployment. KARMA algorithm, however, just assumes the knowledge of user distribution $U_d$. KARMA algorithm averages out the clusters of possible deployment locations of Max-Circle algorithm on different random deployments. KARMA algorithm can be run offline for new deployment plans hence its runtime requirements are not so strict.

III. K-MEANS OF MAX-CIRCLE (KARMA) ALGORITHM

In Fig. 2, the maximum circle in the sparse area in terms of sensing nodes is pinned. This point and its proximity lack measurements. Therefore, sparse areas in terms of sensing nodes should be considered for new deployments. Measurements at these locations not only used for these locations but also improves the estimation made for other locations.

We denote an instance of a deployment obeying $U_d$ with $D_{i_d} = \{\text{Sensing node locations}\}$. For a given $D_{i_d}$, most appropriate place for sensor deployment is the center of the maximum circle not touching a sensing node. Therefore, we can give the Max-Circle algorithm as follows:

Algorithm 2: KARMA for deployment locations

Input: User distribution: $U_d$
Input: Number of sensors for deployment: $m$
Input: Number of deployments: $n_d$
Output: Set of deployment points: $S$

1. $S ← ∅$
2. $S_{temp} ← ∅$
3. for $k ← 1$ to $n_d$ do
   4. $D_{i_d}^k ← \text{new random deployment obeying } U_d$
   5. $S_{temp} ← \text{Run Algorithm 1 with } D_{i_d}^k, m$
   6. $S ← S ∪ S_{temp}$
7. $S ← \text{Run K-means clustering on } S \text{ with } m \text{ for centroids}$
8. return $S$

TABLE I

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area dimensions</td>
<td>$1000 \times 1000 \text{ m}^2$</td>
</tr>
<tr>
<td>Primary user transmission energy ($P^{tx}$)</td>
<td>0.1 W</td>
</tr>
<tr>
<td>Path loss exponent ($\alpha$)</td>
<td>3.5</td>
</tr>
<tr>
<td>Reference path loss ($P_{Lo}$)</td>
<td>38.4 dB</td>
</tr>
<tr>
<td>Log-normal shadowing standard deviation ($\sigma_s$)</td>
<td>8 dB</td>
</tr>
<tr>
<td>Number of mobile CR devices</td>
<td>32</td>
</tr>
<tr>
<td>$N_m$ of mobile CR device</td>
<td>128</td>
</tr>
<tr>
<td>Number of dedicated sensor devices</td>
<td>8</td>
</tr>
<tr>
<td>$N_m$ of dedicated sensor devices</td>
<td>256</td>
</tr>
<tr>
<td>Number of deployments</td>
<td>40</td>
</tr>
<tr>
<td>Number of replications for each deployment</td>
<td>32</td>
</tr>
</tbody>
</table>

IV. PERFORMANCE ANALYSIS

We consider an area of $1000m \times 1000m$ and $U_d$ has three subregions with probabilities of containing mobile nodes as listed below:

$U_d = \{(\text{Rectangle}[[0, 0), (200, 1000)]], 0.5),\ (\text{Rectangle}[(200, 0), (1000, 500)]], 0.2),\ (\text{Rectangle}[(200, 500), (1000, 1000)]], 0.3)\}$.

In each subregion mobile nodes are distributed according to $U_d$. The sub-area coordinates are similar to the ones depicted in Fig. 2 with probabilities 0.5, 0.2, and 0.3.

The performance of the deployment algorithms is evaluated by considering the RMSE, CDZR$_1$, and FAZR values for Kriging and LiVe REM construction techniques. Dedicated sensor placement is done via KARMA and random deployment scheme for understanding the improvement gained by KARMA algorithm. Simulations are conducted using a custom simulator in Matlab. Invariant parameters are listed in Table I and the other parameters are listed in the caption part of the performance figures. Throughout the simulations log-normal shadowing with urban parameters is assumed.

The performance of KARMA algorithm is compared by random deployment of dedicated sensors. There are eight new dedicated (more sensitive) sensors for deployment while there exist 32 mobile CR devices in the area. Kriging and LiVe
REM construction techniques are used for fair comparison of deployment improvement. In general, LiVE REM construction technique performs better than Kriging. Therefore, improvement provided to Kriging by KARMA is more than that provided to LiVE REM construction technique.

In Fig. 3, the x-axis and the y-axis represent the number of dedicated sensors and RMSE values, respectively. Dashed curves correspond to Kriging and the curves with diamond marker correspond to scenarios where the deployment is done via KARMA. In both sub-figures, increasing the number of dedicated sensors decreases the RMSE value and system performance increases. There is more room for improvement in the Kriging case, hence RMSE value is improved much more for the first few sensors compared to LiVE REM construction technique. Deployment of new sensors via KARMA algorithm performs better than random deployment in terms of RMSE of constructed REMs. For eight dedicated sensors case, KARMA algorithm results in more than 20% and 37% improvement compared to random deployment for Kriging and LiVE REM construction techniques, respectively.

In Figs. 3(a) and 3(b), capabilities of the mobile CRs are different (i.e. \( \sigma_e = 3 \) dB and 7 dB, respectively). For \( \sigma_e = 7 \) dB, RMSE values are higher due to more errors introduced by mobile CR nodes. More useful performance metrics such as CDZR_1 and FAZR provide a better understanding of the network. Better REM construction methods fit better on the true REM no-talk zone contour with predetermined threshold value (-120 dBW in our case). Having higher CDZR_1 means disturbing the active user is less probable. Having less FAZR means, white spaces can be utilized better. Even if there is a big difference in terms of RMSE values, this difference might be insignificant after thresholding and determining the no-talk zone.

In Fig. 4, number of dedicated sensors versus CDZR_1 for FAZR = 0.005 is depicted. KARMA algorithm performs better than random deployment and improves the performance of both Kriging and LiVE REM construction significantly. Kriging has more room for improvement at the first deployment. Hence, its improvement at the first deployment with KARMA is significant and improvement decreases with increasing the number of dedicated sensors.

V. CONCLUSIONS

REMs act as cognition engines by building long-term knowledge via processing spectrum measurements collected from sensors to estimate the state of locations. In this paper, we propose KARMA deployment algorithms for deploying new dedicated sensors when we know how mobile nodes are distributed in the environment. We compare KARMA with random deployments for Kriging and the LiVE REM construction techniques. The simulation results confirmed that KARMA improves the REM quality significantly. Utilizing KARMA algorithm decreases the performance gap between
Kriging and the LiVe REM construction in terms of RMSE and CDZR$_1$. For future work, we plan to compare the performance of KARMA with the optimal deployment and investigate the effect of $N_m$ and $\sigma_e$ in more detail.

VI. ACKNOWLEDGEMENT

This research was funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea, under the “IT Consilience Creative Program” (NIPA- 2014-H0201-14-1001) supervised by the NIPA (National IT Industry Promotion Agency), the State Planning Organization of Turkey (TAM - 2007K120610), Bogazici University Research Fund (BAP) under grant number 7437, and by COST IC0902 action.

REFERENCES