2014 American Control Conference (ACC)
June 4-6, 2014. Portland, Oregon, USA

Distributed Estimation and Tracking for Radio Environment Mapping

Ruofan Kong, Wenlin Zhang and Yi Guo

Abstract— **We study distributed estimation and tracking for radio environment mapping (REM). Comparing to existing REM using centralized methods, we provide a distributed solution eliminating the central station for map construction. Based on the random field model of the REM with shadow fading effects, we adopt consensus-based Kalman filter to estimate and track the temporal dynamic REM variation. The unknown parameters of REM temporal dynamics are estimated by a distributed Expectation Maximization algorithm that is incorporated with Kalman filtering. Our approach features distributed Kalman filtering with unknown system dynamics, and achieves dynamic REM recovery without localizing the transmitter. Simulation results show satisfactory performances of the proposed method where spatial correlated shadowing effects are successfully recovered.**

I. INTRODUCTION

Decentralized environmental modeling [1] and exploration [2] are active research topics inspired by the fast-developing consensus algorithms [3] or distributed filters [4]. With the aid of consensus-based information fusion, distributed sensors cooperatively detect and estimate the environmental parameters without a centralized processor. In this paper, we study consensus-based estimation and tracking of Radio Environment Mapping (REM), which is a typical application of decentralized environmental modeling in the context of cognitive radio networks.

Radio environment mapping [5] mainly refers to an integrated database that provides multi-domain environmental information and prior knowledge for cognitive radios, such as the geographical features, available services and networks, locations and activities of neighboring radios. Among those, one of the fundamental problems is the estimation and dynamic tracking of the radio signal propagation map, such as power spectral density map estimation [6] [7], or as an alternative, the channel gain estimation [8] and tracking [9]. The radio environment mapping studied in this paper aims at using distributed sensors to recover dynamic radio signal spatial propagation and spectral energy distribution in a given frequency range. Recovered signal propagation maps facilitate dynamic spectrum sharing, and help communicate the radio spectrum knowledge among common users of cognitive radio networks.

The REM estimation and tracking is challenging, as localizing transmitters may not be allowed by legislations. Without prior-known behaviors of the transmitters, the REM can be modeled as an uncertain dynamical system, such as spatial-temporal Gaussian random fields [10]. Existing REM results adopting this model include the radio tomographic imaging [11] [12] based on the medium scale correlated shadow fading characterization, passive localization [13] and intruder detection [14] by analyzing the interaction between the signal propagation and the environmental geometry. However, most of the work on REM uses centralized methods, where a central data collection and processing machine is available to generate the global radio map. Those methods suffer the dependency of reporting channels, bandwidth constraints, and scalability issues [15]. There has been limited work on distributed solutions to the REM problem without a central station.

From the decentralized control and sensing perspective, distributed Kalman filters play an important role in sensor networks to estimate and recover uncertain dynamic fields, such as temperature over an area in the ocean [1]. By local neighboring communications, distributed sensors compute centralized global information in each Kalman filter iteration using consensus algorithm or distributed filter without a centralized fusion center. Based on this scheme, consensus-based distributed Kalman filters are developed for decentralized environment modeling in various applications. Lynch et al. developed the PI-consensus filter [16], which adopts Kalman filter to model environments using mobile sensor networks [1]. To improve estimation accuracy in the environment represented by random fields, Cortes [17] developed distributed Kriged Kalman filter using PI-consensus filter for spatial estimation. To reduce the computational cost introduced by the PI-consensus filter, Saber proposed a sub-optimal Kalman filter tracking algorithm [18] to facilitate algorithm implementation. However, existing work on distributed Kalman filter assumes system parameters are either known or have been prior-estimated, which may not be feasible as prior knowledge is sometimes hard to estimate in practical systems, such as the REM problem discussed in this paper.

For unknown system estimation, two main techniques commonly used in the literature, the maximum likelihood estimation [19] and the Expectation Maximization (EM) [20], are both centralized methods. Inspired by the fact that there are no distributed solutions for environmental estimation and modeling with unknown system models, we study distributed consensus-based estimation and tracking of an uncertain field, and apply the method to solve the dynamic REM problem in the presence of uncertain system parameters.

In this paper, we consider distributed consensus-based

The work was supported in part by the National Science Foundation under Grants IIS-1218155 and CNS-1318748.

R. Kong and Y. Guo are with the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07307, U.S. rkong1@stevens.edu, yguo1@stevens.edu

W. Zhang is with the Western Digital Corporation, Irvine, CA 92612, U.S. wzhang5@hotmail.com. This work was done during W. Zhang's Ph.D. study at Stevens Institute of Technology.

Kalman filtering to estimate and track dynamic REM with correlated shadowing. We adopt random fields to model the REM as an uncertain dynamical system, of which, the spatial and temporal dynamics can be decoupled using function expansion. We develop consensus-based Kalman filtering with distributed EM algorithm to estimate the temporal dynamics as well as the system parameters that are used to recover and track the REM. Simulation results show the effectiveness of recovering the shadowing phenomenon and the dynamic tracking capability of REM. Our main contribution includes a distributed solution to the REM problem, and distributed Kalman filtering with unknown system parameters.

II. Problem Formulation

A. System Configuration and Distributed Sensor Placement

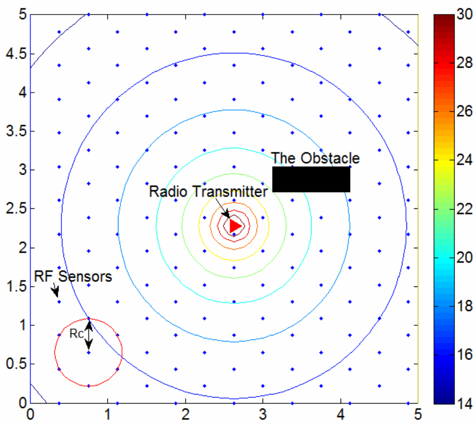

Fig. 1. The REM system setup. The blue dots denote the RF sensors. The black rectangle represents the stationary obstacle. The solid triangle is the RF transmitter. The red circle represents the RF sensor communication range with the radius R_c. The color bar indicates the radio signal strength measurement value.

As shown in Fig. 1, we consider a dynamic radio transmitter moving in a bounded area with the size of x_w in length and y_w in width where obstacles exist. To explore the radio signal distribution in this area, RF sensors are deployed for radio signal strength detection, and they can communicate with their neighboring sensors within their communication range R_c, using the communication topology indicated in Fig. 2. In this paper, we assume:

1) **The radio signal:** The radio transmitter moves in a bounded area, and the radio signal propagation model is represented in (30).
2) **The environment:** There are stationary obstacles with unknown positions.
3) **RF sensors and placement:** The sensors are uniformly deployed and they can communicate with their neighboring sensors within their communication range R_c. We assume each RF sensor can only detect its current position, signal strength measurement value, and the corresponding covariance of the measurement noise. They can also receive information from their one-hop neighboring sensors using the communication topology defined in Fig. 2. In addition to the sensing capability, we assume the sensors have onboard computational power and can process measured data real time.

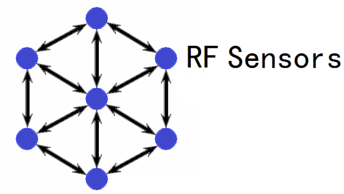

Fig. 2. Sensor communication topology. The blue solid circles denote the RF sensors.

In this paper, we define the global coordinates as: the positive x-axis pointing to the right horizontal direction, and the positive y-axis pointing to the up vertical direction. As shown in Fig. 1, the sensor placement scheme is described as follows. We place l columns parallel to the y-axis, and each column contains m sensors. The coordinate of the sensor locating at the eth row ($1 \leq e \leq m$) and fth column ($1 \leq f \leq l$) is at (x_c^{ef}, y_c^{ef}):

$$(x_c^{ef}, y_c^{ef}) = \begin{cases} \left(\frac{\sqrt{3}}{2}R_c + \frac{\sqrt{3}}{2}(f-1)R_c, (e-1)R_c\right), \\ \quad \text{if } f \text{ is an odd integer;} \\ \left(\frac{\sqrt{3}}{2}R_c + \frac{\sqrt{3}}{2}(f-1)R_c, \frac{1}{2}R_c + (e-1)R_c\right), \\ \quad \text{if } f \text{ is an even integer.} \end{cases}$$

The number of sensors in the odd column m_{odd} and even column m_{even} are expressed as:

$$\begin{cases} m_{odd} = m_{even} + 1 = Int(\frac{y_w}{R_c}) + 1, \\ \quad \text{if } Rem(\frac{y_w}{R_c}) < \frac{R_c}{2}; \\ m_{odd} = m_{even} = Int(\frac{y_w}{R_c}) + 1, \\ \quad \text{if } Rem(\frac{y_w}{R_c}) \geq \frac{R_c}{2}. \end{cases}$$

where $Int(x)$ is the integer part of x, and $Rem(x) = x - Int(x)$. Also, the number of column l is: $l = Int(\frac{x_w}{\frac{\sqrt{3}}{2}R_c})$.

Remark 1: Since we assume the sensor communication range as a circle, we adopt the sensor placement scheme as in [21]. For a more detailed analysis of coverage sensor placement, interested readers may refer to [22].

We are interested in distributed estimation and recovery of the REM caused by the dynamical RF signal. We formally define our problem in the next subsection.

B. The Model and Problem Statement

The radio signal propagation model is described in Appendix I, where the signal strength for each location cell is explicitly computed according to (30). However, it is not practical to use (30) in constructing REM because it needs the prior-known trajectory of the transmitter to determine the temporal dynamics P_0 and L_a in (30), while localizing radio transmitter is not allowed by legislations. Also, without knowing the environment geometry and behaviors of the radio transmitter, the spatial-temporal dynamics of shadow fading, the S_a term in (30), cannot be determined. This is the fundamental reason that the radio propagation process is usually modeled in a different way.

Since the interaction between the signal and the environmental geometry evolves as uncertain spatio-temporal dynamics, it is common to model the radio signal propagation as random fields [8]–[11]. Specifically, dropping the subscript a in (30) for convenience, the radio signal y at any cell a and time k is regarded as a random field $y(k, s)$, where $s \in \mathbb{R}^2$ is the space coordinate and $k \in \mathbb{R}_+$ is the time index. We have the same assumption [8]–[11] that any pair of $y(k_1, s_1)$ and $y(k_2, s_2)$, $s_1, s_2 \in s$, $k_1, k_2 \in k$, only correlated in space not in time, i.e.,

$$\mathrm{Cov}(y(k_1, s_1), y(k_2, s_2)) = C(\|s_1 - s_2\|^2)\delta(k_1 - k_2) \quad (1)$$

where $C(\|s_1 - s_2\|^2)$ is a positive symmetric function, and $\delta(k_1 - k_2)$ is the Dirac delta function. Neglecting the spatial correlated noise, the random field $y(k, s)$ from the Mercer's theorem has the function basis expansion as

$$y(k, s) = \sum_{j=1}^{\infty} \xi_j(k)\phi_j(s), \quad (2)$$

where $\xi(k)$ describes a temporal process consisting of infinite components, and $\phi(s)$ is the corresponding function basis to describe the spatial variation. For the consideration of computational cost, (2) usually takes the finite expansion as

$$y(k, s) = \sum_{j=1}^{M} \xi_j(k)\phi_j(s). \quad (3)$$

In the compact form, we have

$$y(k, s) = \phi(s)\xi(k), \quad (4)$$

where $\phi(s) = [\phi_1(s), ...\phi_M(s)]$, and $\xi(k) = [\xi_1(k), ..., \xi_M(k)]^T$.

The major convenience from the expansion (2) is the decoupling of temporal dynamics and spatial variation of $y(k, s)$. For temporal dynamics, $\xi(k)$ is assumed to evolve as [23],

$$\xi(k) = A\xi(k-1) + \nu(k), \quad (5)$$

where A is the system matrix of temporal dynamics of the REM behaviors, $\nu(k)$ denotes the Gaussian noise with zero mean and covariance matrix $Q(k)$.

For a network consisting of n RF sensors, the measurement of the i^{th} RF sensor at the corresponding cell is given by

$$Y_i(k, s) = y(k, s) + \zeta(k, s), \quad (6)$$

where $\zeta(k, s)$ denotes the Gaussian measurement noise with zero mean and covariance $R(k)$, which assumes to be not spatial correlated in this paper for simplicity.

Consider the system model (4) (5) and (6), the Kalman filter technique can be used to estimate the system state ξ for recovering the signal strength y, which requires the system matrix A in (5) prior-known. However, due to uncertainties in practical systems, the system matrix A, is often time-varying and unknown so that the accurate estimation for ξ becomes challenge. Therefore, based on the model, the **distributed REM tracking problem** is defined as to *estimate and track $y(k, s), k \in R_+, s \in R^2$ using distributed sensor measurements (6) in the random Gaussian field with the unknown matrix A in the temporal dynamics (5).*

Remark 2: The model (4) - (5) presented in this section is similar to the model in [1]. The major difference is that we assume that the system matrix A in (5) is unknown, while A is known or prior-estimated in [1]. A more detailed model for distributed tracking considering additive spatial correlated noise in a random field is presented in [17] using distributed Kriged Kalman filtering techniques. We will further extend the result to distributed Kriged Kalman filter for unknown dynamical systems in our future work.

III. The Algorithm

In this paper, our objective is to develop a distributed REM tracking algorithm so that the distributed sensors cooperatively estimate the system model parameters, and simultaneously track the dynamic REM. To solve this problem, we propose a distributed Kalman-EM filter based algorithm.

A. Kalman-EM Filter based Estimation and Tracking

Applying Kalman filter to track the state parameters ξ in the system (4) (5) and (6), the system matrix A in (5) needs to be estimated. To solve the problem, a learning method, Expectation Maximization (EM) is chosen to embed into the Kalman filter, for the purpose of learning the system matrix A. This learning method is inspired from the Bayesian perspective as it maximizes the expectation of parameter log-likelihood, so that both the observation Y and the system state ξ get the maximum estimation probability. We call it the Kalman-EM filter. In this subsection, we introduce the estimation process of the proposed Kalman-EM filter.

1) System Matrix Estimation: The system matrix estimation utilizes the Bayesian method, and the EM algorithm as the prediction tool given the observations. This algorithm includes two steps: $i)$ Maximize the expectation likelihood in terms of the system state using Kalman filter (E-step); $ii)$ Maximize the expectation likelihood in terms of the system matrix A (M-step). In our problem, for the system (4) (5) and (6), the estimated state $\xi(k)$ and observation $Y(k)$ follow the Gaussian process: $\xi(k) \sim N(A\xi(k-1),\ Q)$ and $Y(k) \sim N(\phi\xi(k),\ R)$, in which $(\cdot)(k)$ denotes the value of the variable at time k. The joint likelihood is:

$$logP\left(\{\xi(1:\hat{k})\}, \{Y(1:\hat{k})\}|A(\hat{k})\right) = log\left[P(\xi(1))\prod_{k=2}^{\hat{k}} P(\xi(k)|\xi(k-1))\prod_{k=1}^{\hat{k}} P(Y(k)|\xi(k))\right] \quad (7)$$

where

$$P(Y(k)|\xi(k)) = \exp\{-\frac{1}{2}[Y(k) - \phi\xi(k)]^T R^{-1} \cdot [Y(k) - \phi\xi(k)]\}(2\pi)^{-n/2}|R|^{-1/2},$$
$$P(\xi(k)|\xi(k-1)) = \exp\{-\frac{1}{2}[\xi(k) - A\xi(k-1)]^T Q^{-1} \cdot [\xi(k) - A\xi(k-1)]\}(2\pi)^{-n/2}|Q|^{-1/2}$$

with observations $Y(1:\hat{k}) = (Y(1),...,Y(\hat{k}))$ and states $\xi(1:\hat{k}) = (\xi(1),...,\xi(\hat{k}))$.

At time $\hat{k}$, we perform the E-step of EM to compute the expected log-likelihood E_L, taking unobserved state ξ into account [24]:

$$E_L(\hat{k}) = E[\log P(\xi(1:\hat{k}), Y(1:\hat{k}))|Y(1:\hat{k}); A(\hat{k})] \quad (8)$$

in which it calculates:

$$\begin{aligned} \hat{\xi}(k) &= E[\xi(k)|Y(1:k); A(k)] && (9)\\ P(k) &= E[\xi(k)\xi^T(k)|Y(1:k); A(k)] && (10)\\ P((k),(k-1)) &= E[\xi(k)\xi^T(k-1)|Y(1:k); A(k)] && (11) \end{aligned}$$

To choose an appropriate system matrix A that enables (8) to be maximized at each time instant, the M-step is performed, so we obtain:

$$\frac{\partial E_L(\hat{k})}{\partial A(\hat{k})} = -\sum_{k=1}^{\hat{k}} P((k),(k-1)) + \sum_{k=1}^{\hat{k}} Q^{-1}A(k)P(k-1) = 0 \quad (12)$$

which yields

$$A(\hat{k}) = \left(\sum_{k=1}^{\hat{k}} P((k),(k-1))\right)\left(\sum_{k=1}^{\hat{k}} P(k-1)\right)^{-1}. \quad (13)$$

2) Kalman-EM Filter Estimation and Tracking: Let $\tilde{\xi}$ represent the estimate of ξ, and V be the $M \times M$ covariance matrix correlated with estimate errors. Define the information matrix $\sigma(k) = V^{-1}(k)$, the information vector $\delta(k) = V^{-1}(k)\xi(k)$ and $M \times M$ scalar matrix $K_{M\times M}$. Based on initializations $A(0)$, $\sigma(0)$ and $\delta(0)$, both information vector τ and matrix σ at time k is obtained by the following steps of Kalman filter:

Prediction:

$$\begin{aligned} \tilde{\sigma}(k) &= (A(k-1)\sigma^{-1}(k-1)A^T(k-1)+Q)^{-1} && (14)\\ \tilde{\delta}(k) &= \tilde{\sigma}(k)A(k-1)\sigma^{-1}(k-1)\delta(k-1) && (15) \end{aligned}$$

Correction:

$$\begin{aligned} \sigma(k) &= \tilde{\sigma}(k) + \phi^T R^{-1}\phi && (16)\\ \delta(k) &= \tilde{\delta}(k) + \phi^T R^{-1}Y(k) && (17) \end{aligned}$$

Scalar matrix update:

$$K(k) = (\tilde{\sigma}^{-1}(k) - \sigma^{-1}(k))\tilde{\sigma}(k) \quad (18)$$

Also, (10) and (11) have the solutions for (8) as follows:

$$\begin{aligned} P(k) &= \hat{V}(k-1) + \hat{\xi}(k-1)\hat{\xi}^T(k-1) && (19)\\ P((k),(k-1)) &= \hat{V}((k),(k-1)) + \hat{\xi}(k)\hat{\xi}^T(k-1) && (20) \end{aligned}$$

where $\hat{V}(k-1)$, $\hat{\xi}(k-1)$ and $\hat{V}((k),(k-1))$ update all previous expectations of $\tilde{V}(k)$, $\tilde{\xi}(k)$ and $\tilde{V}((k),(k-1))$, respectively, for accurately estimating $A(k)$ once a new estimated state is generated, and $\tilde{V}((k),(k-1))$ denotes the cross covariance matrix between two consecutive time instants. Specifically, this is the E-step of Kalman-EM filter that performs backward recursions for $k = \hat{k},...,1$ on the equations:

$$J(k-1) = \sigma^{-1}(k-1)A^T(k-1)\tilde{\sigma}(k-1) \quad (21)$$

$$\hat{V}(k-1) = \sigma^{-1}(k-1) + J(k-1) \cdot\left(\hat{V}(k) - \tilde{\sigma}^{-1}(k-1)\right)J^T(k-1) \quad (22)$$

$$\hat{\xi}(k-1) = \sigma^{-1}(k-1)\delta(k-1) + J(k-1) \cdot\left(\hat{\xi}(k) - A(k-1)\sigma^{-1}(k-1)\delta(k-1)\right) \quad (23)$$

which is initialized as $\hat{V}(k) = \sigma^{-1}(k)$ and $\hat{\xi}(k) = \sigma^{-1}(k)\delta(k)$.

For $k = \hat{k},...,2$ on the equation, we have:

$$\hat{V}((k-1),(k-2)) = \sigma^{-1}(k-1)J^T(k-2) + J(k-1)\left(\hat{V}((k),(k-1)) - A(k-1)\sigma^{-1}(k-1)\right)J^T(k-2) \quad (24)$$

which is initialized as $\hat{V}((k),(k-1)) = (I - K(k))A(k-1)\sigma^{-1}(k-1)$, where I is an identity matrix of dimension $M \times M$.

In the M-step, (10) and (11), which are obtained from the above backforward iteration, are then combined with (13), to obtain the updated system matrix $A(k)$, which serves as the outcome of the differentiation of (8).

B. PI Consensus Filter for Distributed Estimation and Tracking

Considering (16) and (17), dropping the parameter of k, let

$$\begin{aligned} \phi^T R^{-1}\phi &= \sum_{i=1}^{n} \phi(s_i)^T R_i^{-1}\phi(s_i), && (25)\\ \phi^T R^{-1}Y(k) &= \sum_{i=1}^{n} \phi(s_i)^T R_i^{-1}Y_i(k), && (26) \end{aligned}$$

where n is the number of RF sensors in the area.

In order to calculate $\phi^T R^{-1}\phi$ and $\phi^T R^{-1}Y(k)$ above, we need the global information for $i = 1,\ldots,n$. However, in our system setup, each sensor only knows its own parameters R_i and $Y_i(k)$ using its own position and measurement values. Fortunately, each sensor can communicate with its neighbors and eventually get the global information through information propagation. To achieve this goal, we have each sensor implement a PI consensus filter [16] [1]. The discrete-time consensus estimator (running in a faster time scale) is given as

$$
\begin{aligned}
&v_i(\tau) = v_i(\tau-1) + \beta \Big\{ \gamma[u_i - v_i(\tau-1)] \\
&-K_P \sum_{j\in N_i} [v_i(\tau-1) - v_j(\tau-1)] \\
&+K_I \sum_{j\in N_i} [\eta_i(\tau-1) - \eta_j(\tau-1)] \Big\}, \\
&\eta_i(\tau) = \eta_i(\tau-1) - \beta \\
&\cdot \Big\{ K_I \sum_{j\in N_i} [v_i(\tau-1) - v_j(\tau-1)] \Big\}
\end{aligned} \tag{27}
$$

where $j \in N_i$ denotes that sensor j is in the one-hop neighbor set, N_i, of sensor i, u_i is the sensor i's vector input to the sum, η_i and v_i are the sensor i's internal state, and the estimate of the average of all agents' inputs, respectively. $\gamma > 0$ is the parameter governing the rate at which new information replaces old information in the dynamic averaging process, and β is the step size for the PI consensus filter. K_p and K_I are estimator gains. For convenience, (27) can also be written in the following compact form:

$$
\begin{aligned}
\begin{bmatrix} v(\tau) \\ \eta(\tau) \end{bmatrix} &= \left(\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} + \beta \begin{bmatrix} -\gamma I - L_P & {L_I}^T \\ -L_I & 0 \end{bmatrix} \right) \\
&\cdot \begin{bmatrix} v(\tau-1) \\ \eta(\tau-1) \end{bmatrix} + \begin{bmatrix} \gamma I \\ 0 \end{bmatrix} u
\end{aligned} \tag{28}
$$

where I is the identity matrix, $L_P = K_p L$, $L_I = K_I L$, and L is the graph $Laplacian$ defined in Appendix I.

Suppose the total sensor number n is known to all sensors, let $u_i = n\phi(s_i)^T R_i^{-1} \phi(s_i)$, running the above protocol, $v_i, i = 1, \ldots, n$, converges to $\phi^T R^{-1} \phi$ as time elapses. Also, when $u_i = n\phi(s_i)^T R_i^{-1} Y_i(k)$, $v_i, i = 1, \ldots, n$, converges to $\phi^T R^{-1} Y(k)$. Therefore, *distributed* estimation is achieved using the PI consensus filter scheme shown above.

The procedure of our proposed algorithm is described in Algorithm 1. The ith sensor collects its radio energy measurement Y_i, position s_i (Line 1), calculates its local quantity $\phi(s_i)^T R_i^{-1} \phi(s_i)$ and $\phi(s_i)^T R_i^{-1} Y_i(k)$. (Line 3). After that, each of the sensors performs the PI consensus filtering towards calculating the global quantities $\phi^T R^{-1} \phi$ and $\phi^T R^{-1} Y(k)$ by their neighbor communication (Line 4 - Line 10). Then they implement Kalman filter to estimate the system states (Line 12), run E-step backward recursion (Line 13), and estimate the parameter matrix A for M-step (Line 14), then update the dynamic REM by calculating (4) (Line 15). Thus, dynamic REM is recovered on each sensor real time.

Algorithm 1 Distributed REM tracking

Require:
Radio energy measurement Y, sensor position $\mathbf{s}$ =($\mathbf{m}$,$\mathbf{n}$). Initialization: information vector $\delta(0)$, information matrix $\sigma(0)$, system matrix $A(0)$, state covariance matrix Q, and measurement noise covariance matrix R.

Ensure:
To enable each sensor to reach agreement on REM estimation, and track the dynamical REM.

1: **for** time step k = 1 to $\hat{k}$ **do**
2: Each sensor i gets its corresponding position $s_i = (m, n)$ and sensor measurement of radio energy Y_i, then calculates its basis function $\phi(s_i)$.
3: Each sensor calculates $\phi(s_i)^T R_i^{-1} \phi(s_i)$ and $\phi(s_i)^T R_i^{-1} Y_i(k)$.
4: **for** $\tau = 1$ to p **do**
5: Each sensor sets $u_i = n\phi(s_i)^T R_i^{-1} \phi(s_i)$,
6: Each sensor gets its neighboring sensors' information $v_j, \eta_j, j \in N_i$,
7: Each sensor runs PI consensus filter in (27) to get v_i and η_i;
8: Each sensor sets $u_i = n\phi(s_i)^T R_i^{-1} Y_i(k)$,
9: Each sensor repeats line 6 and 7,
10: **end for**
11: Each sensor returns consensused global quantities $v_i \to \phi^T R^{-1} \phi$ and $\to \phi^T R^{-1} Y(k)$.
12: Each sensor runs Kalman filter line by line to calculate the priori, $\tilde{\sigma}(k)$ and $\tilde{\delta}(k)$, posteri $\sigma(k)$ and $\delta(k)$ and scalar matrix $K(k)$ using equations (14)-(18).
13: Each sensor runs E-step backward recursion line by line to calculate $J(k-1)$, $\hat{V}(k-1)$, $\hat{\xi}(k-1)$ and $\hat{V}((k-1),(k-2))$ using equations of (21)-(24) .
14: Each sensor substitutes $\hat{V}(k-1)$, $\hat{\xi}(k-1)$ and $\hat{V}((k-1),(k-2))$ into (19) and (20) to calculate $P((k),(k-1))$ and $P(k-1)$, and updates System Matrix $A(k)$ using equations of (13).
15: Each sensor calculates the energy data $y(k, s)$ at all the points in the global map and recovers REM using (4).
16: Next step k + 1 $\leftarrow k$.
17: **end for**

IV. Simulations

We generate the dynamical REM based on the system setup of Section II-A, in which, the bounded area consists of 47×41 regular cells, and the RF transmitter is moving around the obstacle. In the simulation, we denote the RF transmitter power in the form of the channel gain (dB). According to the radio propagation model (30), we set: $P_0 = 20dB$; the attenuation of the b^{th} cell $x_b = 5$ if the cell is occupied by an obstacle, while $x_b = 0.1$ if the cell is in free space; the environmental noise $\iota_a(k)$ for each cell follows the Gaussian distribution: $\iota_a(k) \sim N(0,\ 1)$; the parameter $\Gamma = 0.1$; we ignore the multi-path and interfere effect so that $F_a(k) = 0$. Fig. 3 shows the energy distribution of RF transmitter at different time instants, in which, the reddish area having the highest signal strength corresponds to the location of the radio transmitter, and the black rectangle denotes the obstacle.

We place $n = 156$ sensors with the communication range $R_c = 0.433$ over the bounded area as shown in Fig. 1, our goal is to estimate and track the REM using **Algorithm 1**. To implement our proposed algorithm, each sensor sets

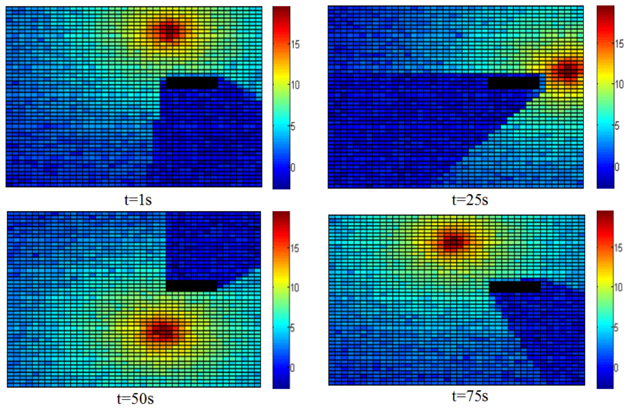

Fig. 3. Dynamical REM of the moving radio transmitter generated using (30).

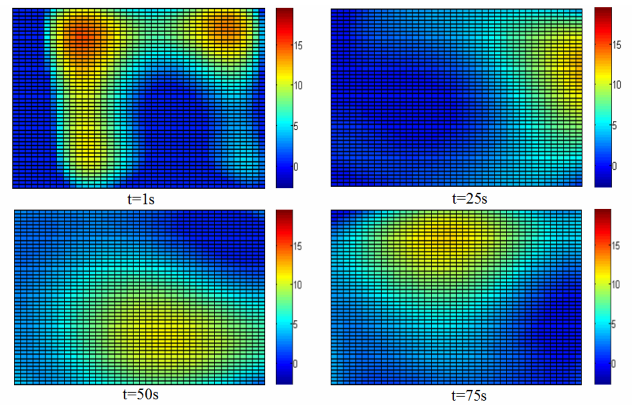

Fig. 4. Distributed tracking with estimated system matrix A for REM of the moving radio transmitter.

the parameters as following: The function basis subset in (4) contains $M = 15$ component sets as $\phi(s_x, s_y) = [1, s_x, s_y, s_x^2, s_y^2, s_x s_y, s_x^3, s_y^3, s_x s_y^2, s_x^2 s_y, s_x^4, s_y^4, s_x s_y^3, s_x^2 s_y^2, s_x^3 s_y]$ with s_x and s_y the horizontal and vertical positions, respectively; the covariance matrix of the temporal dynamic noise in (5) is $Q = 0.01 I_{15\times15}$; the measurement covariance matrix for each sensor in (6) is $R = 0.1 I_{15\times15}$; each sensor initializes the Kalman filter parameters in (14) and (15) as: $\tau_{(0)} = 0_{15\times1}$and $\sigma_{(0)} = 0.0001 I_{15\times15}$; the parameters of PI consensus filter (27) are: $\gamma = 1.6$, $\beta = 0.5$, $K_P = 0.1$ and $K_I = 0.01$; the initializations of PI consensus filter are: $v_i(0) = 0_{15\times15}$ and $\eta_i(0) = 0_{15\times15}$ for C estimation, and $v_i(0) = 0_{15\times1}$ and $\eta_i(0) = 0_{15\times1}$ for D estimation, and we randomly initialize the system matrix A.

Fig. 4 is the recovered REM generated by our algorithm. Comparing to the true REM shown in Fig. 3, it can be seen that the initial REM is poorly recovered at $t = 1s$, and the radio transmitter cannot be localized. However, as the distributed Kalman filter converges, the dynamically changing REM is tracked successfully, and the radio transmitter source can be approximately localized with the highest signal strength. Specifically, at time t=25s, 50s and 75s, it can be seen from the subfigures that the locations of the dynamic radio source is recovered (*i.e.*, they are approximately at the same place in the global coordinate), and the shadowing effect caused by the obstacle is also successfully recovered as displayed in dark areas of each subfigure.

We compare the REM tracking performance of our proposed algorithm with the existing method, which uses distributed Kalman filter together with the PI consensus filter and a randomly chosen system matrix A [1]. We utilize the Mean Absolute Error (MAE) as an evaluation criteria for REM recovery performance. The MAE is defined as:

$$MAE = \frac{1}{\mathcal{N}} \sum_{a=1}^{\mathcal{N}} |\hat{y}_a(s) - y_a(s)| \tag{29}$$

where $\mathcal{N}$ is the total cell number in the REM, and $\hat{y}_a(s)$ is the estimation of $y_a(s)$ in (30). Fig. 5 shows the comparison results: The high initial MAEs for both algorithms result from the random system initializations. The MAEs of both algorithms then decrease due to the convergence of distributed Kalman filter. It is easy to see that our proposed algorithm, which online estimates the system matrix A, has lower estimation MAE, and outperforms the existing algorithm [1] that uses the random system matrix A.

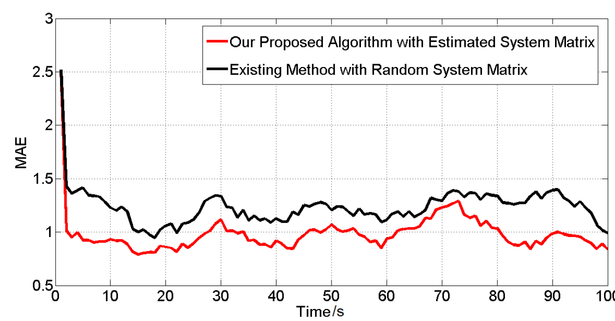

Fig. 5. MAE comparison of our proposed algorithm with the distributed method for REM of the moving transmitter.

V. Conclusions and Future work

In this paper, we studied distributed estimation and tracking of dynamic REM, which is modeled by radio signal propagation model as uncertain Gaussian random fields. Without localizing the radio transmitter, we developed distributed consensus-based Kalman-EM filtering techniques to estimate the unknown system parameters and to generate the REM heat map. Simulation results showed satisfactory performance of the proposed method, where spatial correlated shadowing effects are clearly recovered. Future work includes experimental validation of the proposed tracking algorithm using mobile robotic sensors with optimized data collecting schemes.

Appendix I
Preliminaries

Graph Theory: We mainly explain the consensus related graph notation. Given an index set $\mathcal{I} = \{1, 2, ..., n\}$, an undirected graph $\mathcal{G}$ consists of a triple $(\mathcal{V}, \mathcal{E}, \mathcal{A})$. $\mathcal{V} = \{\mathcal{V}_i | i \in \mathcal{I}\}$ is a finite nonempty set of nodes. The edge set

$\mathcal{E} = \{\mathcal{E}_{ij} = (\mathcal{V}_i, \mathcal{V}_j) | i, j \in \mathcal{I}\}$. We refer to $\mathcal{V}_i$ and $\mathcal{V}_j$ as the tail and head of the edge $(\mathcal{V}_i, \mathcal{V}_j)$. The weighted adjacency matrix $\mathcal{A} = \{\mathcal{A}_{ij} | \mathcal{A}_{ij} \neq 0 \Leftrightarrow \mathcal{E}_{ij} \in \mathcal{E}, \mathcal{A}_{ij} = 0 \Leftrightarrow \mathcal{E}_{ij} \notin \mathcal{E}\}$. For simplicity, we assume $\mathcal{A}_{ii} = 0$ and $\forall \mathcal{A}_{ij} \geq 0, i \neq j$. The set of neighbors of node i is denoted by $N_i = \{j : \mathcal{E}_{ij} \in \mathcal{E}\}$. The graph *Laplacian* associated with the graph $\mathcal{G}$ is defined as $\mathcal{L}(\mathcal{G}) = L = \Delta - \mathcal{A}$. The diagonal matrix $\Delta = [\Delta_{ij}]$ where $\Delta_{ij} = 0$ for all $i \neq j$ and $\Delta_{ii} = \deg_{\text{out}}(\mathcal{V}_i)$. $\deg_{\text{in}}(\mathcal{V}_i) = \sum_{j=1}^{n} \mathcal{A}_{ji}$ and $\deg_{\text{out}}(\mathcal{V}_i) = \sum_{j=1}^{n} \mathcal{A}_{ij}$ are called in-degree and out-degree of nodes $\mathcal{V}_i$, respectively. The Laplacian matrix always has a zero eigenvalue with the right eigenvector of one, which denotes as $\lambda_1 = 0, w_r = \mathbf{1} = [1, 1, \ldots, 1]^T$.

Radio Signal Propagation Model: We represent the radio energy distribution in regular cells with each cell having a signal strength value. The radio signal strength of the a^{th} cell with position s at time k is described as ([14]),

$$y_a(k,s) = P_0(k) - L_a(k) - S_a(k,s) - F_a(k,s) - \iota_a(k) \quad (30)$$

where

1) $P_0(k)$ is the transmission power of the Radio Frequency (RF) transmitters in dB,
2) $L_a(k)$ is the free space propagation loss,
3) $S_a(k,s)$ is the median scale shadow fading loss,
4) $F_a(k,s)$ is the small scale fading, e.g. multi-path or interference,
5) $\iota_a(k)$ is the environmental noise.

Specifically, assuming the transmitter trajectories are known, the free space propagation loss $L_a(k)$ is commonly modeled as

$$L_a(k)[dB] = L(d_0) + 10\Gamma \log(d_a(k)/d_0), \quad (31)$$

where d_0 is the reference distance and $d_a(k)$ is the distance between the RF transmitter and the a^{th} cell. The small scale fading $F_a(k,s)$ is usually modeled as log-norm distribution and can be neglected for simplicity. According to [11] [12] [14], the shadow fading $S_a(k,s)$ for a propagation link from the RF transmitter to the a^{th} cell, can be modeled as

$$S_a(k,s) = \sum_{b=1}^{n_c} \omega_{ab}(k) x_b(s) + \epsilon_a(k,s), \quad (32)$$

where $x_b(s), b = 1, ..., n_c$, is the attenuation occurring in cell b of the propagation link at time k, which is predetermined by the environmental geometry (assuming the environment geometry is temporally fixed). n_c is the number of cells covered by the propagation link. $\omega_{ab}(k)$ is the weighting ratio of the cell b. $\epsilon_a(k,s)$ is the model error.

References

[1] K. M. Lynch, I. B. Schwartz, P. Yang, and R. A. Freeman, "Decentralized environmental modeling by mobile sensor networks," *IEEE Transactions on Robotics*, vol. 24, pp. 710 –724, June 2008.

[2] W. Wu and F. Zhang, "Robust cooperative exploration with a switching strategy," *IEEE Transactions on Robotics*, vol. 28, pp. 828 –839, August 2012.

[3] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1520 – 1533, September 2004.

[4] R. Olfati-Saber, "Distributed Kalman filter with embedded consensus filters," in *Proceedings of the 44th IEEE Conference on Decision and Control and 2005 European Control Conference*, pp. 8179 – 8184, December 2005.

[5] Y. Zhao, *Enabling Cognitive Radios through Radio Environment Maps*. PhD thesis, Virginia Polytechnic Institute and State University, 2007.

[6] J. Bazerque and G. Giannakis, "Distributed spectrum sensing for cognitive radio networks by exploiting sparsity," *IEEE Transactions on Signal Processing*, vol. 58, pp. 1847 –1862, March 2010.

[7] W. Zhang, Z. Wang, Y. Guo, H. Liu, Y. Chen, and J. Mitola-III, "Distributed cooperative spectrum sensing based on weighted average consensus," in *Proceedings of IEEE Global Communications Conference 2011*, pp. 1 –6, December 2011.

[8] S.-J. Kim, E. Dall'Anese, and G. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, pp. 24 –36, February 2011.

[9] E. Dall'Anese, S.-J. Kim, and G. Giannakis, "Channel gain map tracking via distributed Kriging," *IEEE Transactions on Vehicular Technology*, vol. 60, pp. 1205 –1211, March 2011.

[10] J. Riihijarvi, P. Mahonen, M. Wellens, and M. Gordziel, "Characterization and modelling of spectrum for dynamic spectrum access with spatial statistics and random fields," in *Proceedings of the 19th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1–6, September 2008.

[11] N. Patwari and P. Agrawal, "Effects of correlated shadowing: Connectivity, localization, and RF tomography," in *Proceedings of International Conference on Information Processing in Sensor Networks*, pp. 82 –93, April 2008.

[12] P. Agrawal and N. Patwari, "Correlated link shadow fading in multi-hop wireless networks," *IEEE Transactions on Wireless Communications*, vol. 8, pp. 4024 –4036, August 2009.

[13] B. Wagner, N. Patwari, and D. Timmermann, "Passive RFID tomographic imaging for device-free user localization," in *Proceedings of the 9th Workshop on Positioning Navigation and Communication*, pp. 120 –125, March 2012.

[14] J. Wilson and N. Patwari, "Radio tomographic imaging with wireless networks," *IEEE Transactions on Mobile Computing*, vol. 9, pp. 621 –632, May 2010.

[15] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication*, vol. 4, pp. 40 – 62, March 2011.

[16] R. Freeman, P. Yang, and K. Lynch, "Stability and convergence properties of dynamic average consensus estimators," in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 338 –343, December 2006.

[17] J. Cortes, "Distributed Kriged Kalman filter for spatial estimation," *IEEE Transactions on Automatic Control*, vol. 54, pp. 2816 –2827, December 2009.

[18] R. Olfati-Saber, "Kalman-consensus filter : Optimality, stability, and performance," in *Proceedings of the 48th IEEE Conference on Decision and Control*, December 2009.

[19] N. Gupta and R. Mehra, "Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations," *IEEE Transactions on Automatic Control*, vol. 19, pp. 774 – 783, December 1974.

[20] R. Shumway and D. Stoffer, "An approach to time series smoothing and forecasting using the EM algorithm," *Journal of Time Series Analysis*, vol. 3, June 1982.

[21] Y. Guo and M. Balakrishnan, "Complete coverage control for non-holonomic mobile robots in dynamic environments," in *Proceedings of IEEE International Conference on Robotics and Automation*, p-p. 1704–1709, May 2006.

[22] X. Bai, D. Xuan, Z. Yun, T. H. Lai, and W. Jia, "Complete optimal deployment patterns for full-coverage and k-connectivity (k≤6) wireless sensor networks," in *Proceedings of the 9th ACM international symposium on Mobile ad hoc networking and computing*, pp. 401–410, May 2008.

[23] K. Mardia, C. Goodall, E. Redfern, and F. Alonso, "The Kriged Kalman filter," *Test*, vol. 7, pp. 217–285, December 1998.

[24] Z. Ghahramani and G. Hinton, "Parameter estimation for linear dynamical systems," *Technical Report CRG-TR-96-2, University of Toronto*, February 1996.