Level Set Estimation with Dynamic Sparse Sensing

Jing Yang, Zuoen Wang, and Jingxian Wu
Department of Electrical Engineering, University of Arkansas, Fayetteville, AR 72701 USA

***Abstract*—In this paper, we study the level set estimation of a spatial-temporally correlated random field by using a small number of spatially distributed sensors. The level sets of a random field are defined as regions where data values exceed a certain threshold. We propose a new active sparse sensing and inference scheme, which can accurately extract level sets in a large random field with a small number of sensors strategically and sparsely placed in the random field. In the proposed active sparse sensing scheme, a central controller dynamically selects a small number of sensing locations according to the information revealed from past measurements, with the objective to minimize the expected level set estimation errors. The expected estimation error is explicitly expressed as a function of the sensing locations, and the results are used to formulate optimal and sub-optimal selection of sensing locations. Simulation results demonstrate that the proposed algorithms can achieve significant performance gains over baseline passive sensing algorithms that do not proactively select the sensing locations.**

I. Introduction

Level set estimation is the process of using observations of a function x defined on a Hilbert space $\mathcal{X}$ to estimate the region(s) in $\mathcal{X}$ where the function value exceeds some critical value γ; i.e. $\mathcal{S} := \{\mathbf{s} \in \mathcal{X} : x(\mathbf{s}) \geq \gamma\}$. Level set estimation is of paramount importance in many large-scale sensing applications, such as the detection of spectrum holes for cogntive radio networks [1], the accurate monitoring and tracking of traffic congestion [2] or air/water/noise pollution [3].

In these and many other applications, identifying level sets is the primary task, while estimating the value of the function (i.e. the power in spectrum sensing) is often secondary, if not irrelevant. Consequently, level set estimation can be equivalently considered as a mapping problem that draws the level contour or boundary in a random field. Intuitively, data that are further away from the boundary are usually quite distinct from the level of interests, thus there is less ambiguity in terms of level set identification in those regions. Therefore, it is desirable to collect more data samples or place more sensors at the locations where the boundary is likely to lie.

This paper describes a new dynamic sparse sensing and inference scheme for rapid and accurate extraction of level sets of a spatial-temporally correlated random field. In the system under consideration, a data fusion center (FC) performs level set estimation by using current and past noisy data observations from a small number of sensor nodes sparsely distributed in the random field. One of the main novelties of the proposed scheme is that it can dynamically adjust the sensing locations through active learning and adaptation of level set boundaries by analyzing past sensing data. Therefore, the proposed scheme can achieve accurate estimation of the level sets with only a small number of sensors strategically placed at critical locations of the random field. While many methods have been devised for level set estimation in a static setting [4]–[8], the temporally evolving nature of the random field requires a dynamic level set estimation, which makes the estimation problem different and challenging.

We introduce a Gaussian process (GP) prior model to capture the spatial-temporal correlations inherent in the random field [9]. In order to efficiently exploit previous measurements and use it to guide the active sensing process, we propose a two-step active sensing scheme. The first step is to obtain an initial estimation of the random field based on historical data samples, and the second step is to actively probe the field to refine the initial estimation. The expected estimation error is explicitly characterized as a function of the sensing locations, and the results are used to formulate the optimal sensing location selection problem as a combinatorial problem. A low complexity greedy algorithm is then proposed by developing lower bounds of the expected estimation error.

The problem formulation and methodology developed in this paper can benefit many large-scale sensing applications with “big data”. It can also be applied to perform “information distillation”, the process that extracts useful data from an ocean of data that have already been collected.

II. System Model

We consider a sensing system with multiple sensor nodes placed over a measurement field $\mathcal{X} \subset \mathbb{R}^2$. Define the three-dimensional space-time coordinate vector as $\mathbf{c} = [\mathbf{s}, t]^T \in \mathcal{X} \times \mathbb{R}_+$, where $\mathbf{s} = [s_1, s_2]^T$ is the space coordinate, and t is the time variable. Sensor nodes measure a spatial-temporally dependent physical quantity, $x(\mathbf{c})$, such as the temperature or power level of wireless signals, etc. The *prior* distribution of $\{x(\mathbf{c})\}$ is a zero-mean Gaussian process with covariance

$$k(\mathbf{c}_i, \mathbf{c}_j) = k_s(\|\mathbf{s}_i - \mathbf{s}_j\|) \cdot k_t(|t_i - t_j|), \tag{1}$$

where $k_s(\cdot), k_t(\cdot)$ are defined as the spatial and temporal covariance, respectively. The ℓ_2-norm $\|\mathbf{s}_i - \mathbf{s}_j\|$ measures the Euclidean distance between the two coordinates $\mathbf{s}_i, \mathbf{s}_j \in \mathcal{X}$.

The sensing samples observed at the FC can be modeled as the sum of the ground truth $x(\mathbf{c})$, and a noise term $z(\mathbf{c})$, i.e.,

$$y(\mathbf{c}) = x(\mathbf{c}) + z(\mathbf{c}).$$

where $z(\mathbf{c})$ is a zero-mean Gaussian random variable with variance σ^2, and it captures the distortions introduced during the sensing and transmission stages.

At time t, we are interested in identifying the γ-level set of $\{x(\mathbf{s}, t)\}$, which is defined as $\mathcal{S}(t) := \{\mathbf{s} \in \mathcal{X} : x(\mathbf{s}, t) > \gamma\}$. Without loss of generality, we assume $\gamma > 0$.

*This work was supported in part by the National Science Foundation under Grants ECCS-1202075 and ECCS-1405403.

We partition the measurement field $\mathcal{X}$ with equal-sized grid, and the time axis into discrete time slots. It is assumed that the grid edge length and time slot duration are small enough such that the signal remains approximately unchanged within one grid and in one slot. Index the coordinates of the grid as $1, 2, \ldots, n, \ldots, L$, and let $\mathbf{s}_i$ be the coordinate for the i-th grid point. Then, $\mathcal{X}$ can be slightly modified as $\mathcal{X} := \{\mathbf{s}_i : i = 1, \ldots, L\}$. The error metric at slot t_n is

$$e(\bar{\mathcal{S}}, t_n) := \frac{1}{L} \sum_i \mathbb{I}\left\{\mathbf{s}_i \in \Delta(\mathcal{S}_n, \bar{\mathcal{S}})\right\} \tag{2}$$

where $\mathcal{S}_n$ is the level set at slot t_n, $\bar{\mathcal{S}}$ is the estimated level set, $\Delta(\mathcal{S}_n, \bar{\mathcal{S}}) = (\mathcal{S}_n \cap \bar{\mathcal{S}}^c) \cup (\mathcal{S}_n^c \cap \bar{\mathcal{S}})$ denotes the symmetric difference, and $\mathcal{S}_n^c$ is the complement of $\mathcal{S}_n$, and $\mathbb{I}\{E\} = 1$ if event E is true and 0 otherwise.

During each slot, sensing samples are collected from a number of locations. Let $\mathcal{C}_1, \mathcal{C}_2, \ldots$ be the sets of spatial coordinates the FC has collected a sample from at time $t_1, t_2, \ldots$. Denote $\mathbf{x}_n$ and $\mathbf{y}_n$ be the true and observed data samples at $\mathcal{C}_n$, respectively. Define $\mathbf{x}^n := \{\mathbf{x}_i\}_{i=1}^n$, $\mathbf{y}^n := \{\mathbf{y}_i\}_{i=1}^n$, and $\mathcal{C}^n = \{\mathcal{C}_i\}_{i=1}^n$. Then, at each time slot t_n, the dynamic level set estimation problem is to obtain an estimate of $\mathcal{S}_n$ such that the expected estimation error $\mathbb{E}[e(\bar{\mathcal{S}}_n, t_n)|\mathcal{C}^n, \mathbf{y}^n]$ is minimized.

III. Optimal Level Set Estimation in GP

In this section we present the optimal level set estimation algorithm given $\mathbf{y}^n$ and $\mathcal{C}^n$. The optimal algorithm will be used to facilitate the development of the dynamic sensing algorithm in the next section.

Define $\mathbf{c}_{in} := [\mathbf{s}_i, t_n]$, $x_{\mathbf{c}_{in}} := x(\mathbf{s}_i, t_n)$, $\mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^n) := \mathbb{E}\left[x_{\mathbf{c}_{in}} (\mathbf{x}^n)^T\right]$ and $\mathbf{K}(\mathcal{C}^n, \mathcal{C}^n) := \mathbb{E}\left[\mathbf{x}^n (\mathbf{x}^n)^T\right]$. The GP regression based level set estimation algorithm is given in Algorithm 1.

Algorithm 1 GP regression based level set estimation

```
Input: $\mathcal{C}^n, \mathbf{y}^n$ at $t_n$.
Run GP regression for $\mathbf{c}_{in} = [\mathbf{s}_i, t_n], \forall \mathbf{s}_i \in \mathcal{X}$:
    $\bar{m}(\mathbf{c}_{in}) := \mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^n)[\mathbf{K}(\mathcal{C}^n, \mathcal{C}^n) + \sigma^2 \mathbf{I}_{|\mathcal{C}^n|}]^{-1} \mathbf{y}^n$
Threshold $\bar{m}(\mathbf{c}_{in})$:
    $\bar{\mathcal{S}}_n = \{\mathbf{s}_i \in \mathcal{X} : \bar{m}(\mathbf{c}_{in}) > \gamma\}$
Output $\bar{\mathcal{S}}_n$.
```

Theorem 1 *The GP regression based level set estimation minimizes the expected estimation error with given $(\mathcal{C}^n, \mathbf{y}^n)$, i.e., $\mathbb{E}[e(\bar{\mathcal{S}}, t_n)|\mathcal{C}^n, \mathbf{y}^n]$.*

Corollary 1 *The expected error given by Algorithm 1 with given $(\mathcal{C}^n, \mathbf{y}^n)$ is*

$$\mathbb{E}[e(\bar{\mathcal{S}}, t_n)|\mathcal{C}^n, \mathbf{y}^n] = \frac{1}{L} \sum_i Q\left(\frac{|\gamma - \bar{m}(\mathbf{c}_{in})|}{\sqrt{\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in})}}\right) \tag{3}$$

where $Q(x)$ is the Gaussian-Q function, $\bar{m}(\mathbf{c}_{in})$ and $\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in})$ are the posteriori mean and variance of $x_{\mathbf{c}_{in}}$ given $(\mathcal{C}^n, \mathbf{y}^n)$

$$\bar{m}(\mathbf{c}_{in}) = \mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^n)[\mathbf{K}(\mathcal{C}^n, \mathcal{C}^n) + \sigma^2 \mathbf{I}_{|\mathcal{C}^n|}]^{-1} \mathbf{y}^n \tag{4}$$

$$\begin{aligned} \bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) = k(\mathbf{c}_{in}, \mathbf{c}_{in}) - \mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^n)[\mathbf{K}(\mathcal{C}^n, \mathcal{C}^n) + \sigma^2 \mathbf{I}_{|\mathcal{C}^n|}]^{-1} \\ \cdot \mathbf{K}(\mathcal{C}^n, \mathbf{c}_{in}) \end{aligned} \tag{5}$$

and $\mathbf{I}_{|\mathcal{C}^n|}$ is an identity matrix of size $|\mathcal{C}^n|$.

The proofs are omitted here for brevity.

The optimality of Algorithm 1 is conditioned upon the fact that $\mathcal{C}_n$ and $\mathbf{y}_n$ is given. We will discuss how to actively select $\mathcal{C}_n$ based on sensing history $(\mathcal{C}^{n-1}, \mathbf{y}^{n-1})$ in the next section.

IV. Active Sensing for Level Set Estimation

Optimal and sub-optimal active sensing algorithms are presented in this section. With the information extracted from sensing history $(\mathcal{C}^{n-1}, \mathbf{y}^{t-1})$, at the beginning of time slot t_n, the FC is able to obtain updated information regarding the current function value $\mathbf{x}$, which is different from its prior distribution $\mathcal{N}(\mathbf{0}, \mathbf{K})$. With such information, the FC is able to adaptively select a subset of location $\mathcal{C}_n$ to sense, with the goal to minimize the expected level set estimation error. Based on this intuition, we propose a two-step active sensing scheme, as described in Algorithm 2.

Algorithm 2 Two-step active sensing scheme

```
Input: $\mathcal{C}^{n-1}, \mathbf{y}^{n-1}$ at the beginning of $t_n$.
STEP 1: Update the distribution of $\mathbf{x}$.
STEP 2: Decide $\mathcal{C}_n$: actively select $N$ locations $\mathbf{s}_i \in \mathcal{X}$
    for sensing.
Obtain measurements $\mathbf{y}_n$ from $\mathcal{C}_n$.
Run Algorithm 1 with $\mathcal{C}^n$, $\mathbf{y}^n$.
Output $\bar{\mathcal{S}}_n$.
```

The first step is to obtain an initial estimate of the distribution of $\mathbf{x}$, i.e., mean $\hat{\mathbf{m}}$, and covariance $\hat{\mathbf{K}}$, based on the sensing history up to to t_{n-1}, i.e., $(\mathcal{C}^{n-1}, \mathbf{y}^{t-1})$. The observation history up to time slot t_{n-1} thus provides a rough sketch of the level set. Then, the second step is to sample the sensing field $\mathcal{X}$ in t_n, i.e., to select a subset of locations to make observations, and refine the level set estimate based on Algorithm 1 once samples are collected from $\mathcal{C}_n$. Details of the two steps are provided in the following subsections.

A. Optimal Active Sensing Location Selection

The major difficulty of the active sensing scheme lies in the step of selecting $\mathcal{C}_n$. Assuming the sensing cost is proportional to the number of sensing actions performed by the sensor nodes, the optimization problem with a sensing budget constraint can be formulated as

$$\begin{aligned} \text{minimize}_{\mathcal{C}_n} \quad & \mathbb{E}_{\mathbf{y}_n}\left\{\mathbb{E}[e(\bar{\mathcal{S}}, t_n)|\mathcal{C}^n, \mathbf{y}^n]\right\} \\ \text{s.t.} \quad & |\mathcal{C}_n| \leq N \end{aligned} \tag{6}$$

where $\mathbb{E}[e(\bar{\mathcal{S}}, t_n)|\mathcal{C}^n, \mathbf{y}^n]$ is given in (3). The reason that we take another layer of expectation with respect to $\mathbf{y}_n$ in the objective function is due to the fact that $\mathbf{y}_n$ is unknown before the selection of $\mathcal{C}_n$.

We first decompose the estimation error in (3) as a function of $(\mathcal{C}_n, \mathbf{y}_n)$ and $(\mathcal{C}^{n-1}, \mathbf{y}^{n-1})$. Denote

$$\begin{aligned}\hat{m}(\mathbf{c}_{in}) &= \mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^{n-1})[\mathbf{K}(\mathcal{C}^{n-1}, \mathcal{C}^{n-1}) + \sigma^2 \mathbf{I}_{|\mathcal{C}^{n-1}|}]^{-1} \\ &\quad \cdot \mathbf{y}^{n-1} \qquad (7) \\ \hat{k}(\mathbf{c}_{in}, \mathbf{c}_{jn}) &= k(\mathbf{c}_{in}, \mathbf{c}_{jn}) - \mathbf{K}(\mathbf{c}_{in}, \mathcal{C}^{n-1})][\mathbf{K}(\mathcal{C}^{n-1}, \mathcal{C}^{n-1}) \\ &\quad + \sigma^2 \mathbf{I}_{|\mathcal{C}^{n-1}|}]^{-1} \mathbf{K}(\mathcal{C}^{n-1}, \mathbf{c}_{jn}) \qquad (8)\end{aligned}$$

which are the posterior mean and covariance of $x(\mathbf{c}_{in})$ and $x(\mathbf{c}_{jn})$ given $(\mathcal{C}^{n-1}, \mathbf{y}^{n-1})$. Besides, let

$$\begin{aligned}h(\mathbf{c}_{in}, \mathcal{C}_n) &:= \hat{\mathbf{K}}(\mathbf{c}_{in}, \mathcal{C}_n)[\hat{\mathbf{K}}(\mathcal{C}_n, \mathcal{C}_n) + \sigma^2 \mathbf{I}_{|\mathcal{C}_n|}]^{-1} \\ &\quad \cdot [\mathbf{y}_n - \hat{\mathbf{m}}(\mathcal{C}_n)] \\ \sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n) &:= \hat{\mathbf{K}}(\mathbf{c}_{in}, \mathcal{C}_n)[\hat{\mathbf{K}}(\mathcal{C}_n, \mathcal{C}_n) + \sigma^2 \mathbf{I}_{|\mathcal{C}_n|}]^{-1} \\ &\quad \cdot \hat{\mathbf{K}}(\mathcal{C}_n, \mathbf{c}_{in})\end{aligned}$$

where the elements of the posterior mean vector $\hat{\mathbf{m}}(\mathcal{C}_n) = \mathbb{E}(\mathbf{x}_n | \mathcal{C}^{n-1}, \mathbf{y}^{n-1})$ is defined in (7), and $\hat{\mathbf{K}}(\mathbf{c}_{in}, \mathcal{C}_n) = \mathbb{E}\left\{[x_{\mathbf{c}_{in}} - \hat{m}(\mathbf{c}_{in})][\mathbf{x}_n - \hat{\mathbf{m}}(\mathcal{C}_n)]^T | \mathcal{C}^{n-1}, \mathbf{y}^{n-1}\right\}$ and $\hat{\mathbf{K}}(\mathcal{C}_n, \mathcal{C}_n) = \mathbb{E}\left\{[\mathbf{x}_n - \hat{\mathbf{m}}(\mathcal{C}_n)][\mathbf{x}_n - \hat{\mathbf{m}}(\mathcal{C}_n)]^T | \mathcal{C}^{n-1}, \mathbf{y}^{n-1}\right\}$ are the posterior covariance vector and matrix with elements $\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{jn})$ defined in (8).

Then, we can decompose $\bar{m}(\mathbf{c}_{in})$ and $\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in})$ defined in (4) and (5) in the following form.

$$\bar{m}(\mathbf{c}_{in}) = \hat{m}(\mathbf{c}_{in}) + h(\mathbf{c}_{in}, \mathcal{C}_n) \qquad (9)$$

$$\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) = \hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) - \sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n) \qquad (10)$$

With those notations, the optimization problem in (6) is equivalent to

$$\begin{aligned}\min\nolimits_{\mathcal{C}_n} \ & \frac{1}{L} \sum_i \mathbb{E}_{h(\mathbf{c}_{in}, \mathcal{C}_n)} \left[Q\left(\frac{|\hat{m}(\mathbf{c}_{in}) + h(\mathbf{c}_{in}, \mathcal{C}_n) - \gamma|}{\sqrt{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) - \sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n)}} \right) \right] \\ \text{s.t. } & |\mathcal{C}_n| \leq N, \\ & h(\mathbf{c}_{in}, \mathcal{C}_n) \sim \mathcal{N}(0, \sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n)) \qquad (11)\end{aligned}$$

This is a combinatorial optimization problem, and it is NP-hard in general. In the following section, we propose greedy algorithms to approximately solve the problem.

B. Lower Bound Based Greedy Algorithms

In order to simplify the notation, we define

$$\gamma_i := \frac{|\hat{m}(\mathbf{c}_{in}) - \gamma|}{\sqrt{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})}}, \quad \sigma_i := \frac{\sigma_h(\mathbf{c}_{in}, \mathcal{C}_n)}{\sqrt{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})}} \qquad (12)$$

Theorem 2 *The expected error with fixed sensing location selection $\mathcal{C}_n$ in (11) is lower bounded by*

$$\begin{aligned}\mathbb{E}_{h(\mathbf{c}_{in}, \mathcal{C}_n)} \left[Q\left(\frac{|\hat{m}(\mathbf{c}_{in}) + h(\mathbf{c}_{in}, \mathcal{C}_n) - \gamma|}{\sqrt{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) - \sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n)}} \right) \right] \geq & \qquad (13) \\ \min\left\{ \frac{1}{\pi} \exp\left\{ -\frac{\gamma_i^2}{2} \right\}, Q(\gamma_i) \right\} \cdot \tau_i & \qquad (14)\end{aligned}$$

for every $\tau_i \in [0, 1]$, where $\tau_i = \sqrt{1 - \sigma_i^2}$.

The lower bound (13) is a linear function in τ_i and is easy to evaluate. Therefore, we propose to minimize the lower bound instead. Define

$$\alpha_i = \min\left\{ \frac{1}{\pi} \exp\left\{ -\frac{\gamma_i^2}{2} \right\}, Q(\gamma_i) \right\} \qquad (15)$$

Then, the optimization problem (11) is modified as

$$\begin{aligned}\min\nolimits_{\mathcal{C}_n} \quad & \frac{1}{L} \sum_i \alpha_i \sqrt{1 - \frac{\sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n)}{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})}} \\ \text{s.t.} \quad & |\mathcal{C}_n| \leq N \qquad (16)\end{aligned}$$

In (16), α_i and $\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})$ are independent of the choice of $\mathcal{C}_n$. Only $\sigma_h^2(\mathbf{c}_{in}, \mathcal{C}_n)$ depends on $\mathcal{C}_n$. The optimization problem (16) is still an NP-hard problem. We propose Algorithm 3 to solve it in a greedy fashion.

Algorithm 3 A greedy algorithm

```
Input: $\mathcal{C}^{n-1}$, $\mathbf{y}^{n-1}$, $\mathcal{C}_n = \emptyset$, $\mathcal{F} = \mathcal{X}$.
Update the distribution of $\mathbf{x}$, obtain $\hat{\mathbf{m}}, \hat{\mathbf{K}}$.
Assign $\hat{\mathbf{K}}$ to $\bar{\mathbf{K}}$.
Calculate $\gamma_i$, $\alpha_i$, for $i = 1, \cdots, L$.
for $k = 1, 2, \ldots, N$ do
    Calculate $\sigma_h^2(\mathbf{c}_{in}, \mathbf{s}_j) = \frac{\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{jn})^2}{\bar{k}(\mathbf{c}_{jn}, \mathbf{c}_{jn}) + \sigma^2}$ for $\mathbf{s}_j \in \mathcal{F}$
    $$l = \arg\min_j \frac{1}{L} \sum_i \frac{\alpha_i \sqrt{\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{in}) - \sigma_h^2(\mathbf{c}_{in}, \mathbf{s}_j)}}{\sqrt{\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})}} \qquad (17)$$
    $\mathcal{C}_n \leftarrow \mathcal{C}_n \cup l$, $\mathcal{F} \leftarrow \mathcal{F} \backslash l$.
    Update $\bar{\mathbf{K}}$: for $i, j = 1, 2, \ldots, L$,
    $$\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{jn}) \leftarrow \bar{k}(\mathbf{c}_{in}, \mathbf{c}_{jn}) - \frac{\bar{k}(\mathbf{c}_{in}, \mathbf{c}_{ln}) \bar{k}(\mathbf{c}_{ln}, \mathbf{c}_{jn})}{\bar{k}(\mathbf{c}_{ln}, \mathbf{c}_{ln}) + \sigma^2}$$
end for
Output $\mathcal{C}_n$.
```

At the beginning of each time slot, the system obtains an initial estimate of $\mathbf{x}$, characterized as $(\hat{\mathbf{m}}, \hat{\mathbf{K}})$. Intuitively, if the initially estimated mean $\hat{m}(\mathbf{c}_{in})$ deviates significantly from the threshold γ, even with a large variance $\hat{k}(\mathbf{c}_{in}, \mathbf{c}_{in})$, the probability of incorrectly classifying $\mathbf{s}_i$ in slot t_n is very small, and bringing in another sample from $\mathbf{s}_i$ will not help much; on the other hand, if $\hat{m}(\mathbf{c}_{in})$ is quite close to the threshold γ, sensing around $\mathbf{s}_i$ potentially can make the classification much more accurate. Therefore, minimizing the level set estimation error is not equivalent to minimizing the total posterior variance. Essentially, to estimate the level set is to search for the *boundary* of the level sets. For sensing locations far away from the boundary, their actual values do not have much impact on the level set estimation, thus more sensing resources should be allocated for locations around the boundary. Such an approach has the potential to significantly reduce the number of required sensing samples because the area of boundary is usually only a very small percentage of the total area, thus achieving sparse sampling.

The objective function in (16) coincides with our intuition. The weight α_i is a function of γ_i, which is defined in (12). We note that in its definition, the numerator $|\hat{m}(\mathbf{c}_{in}) - \gamma|$ measures

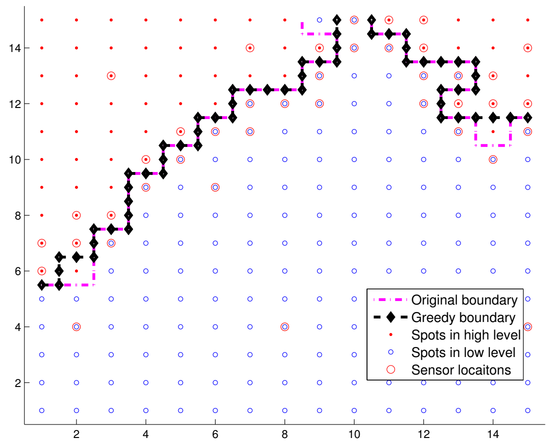

Fig. 1. Red circles represent 40 sensing locations selected by the sensing algorithms. The ground truth are represented as red solid dots in the high level sets and blue circles in the low level sets.

the deviation of $\hat{m}(\mathbf{c}_{in})$ from the threshold γ, which is then normalized by $\sqrt{\hat{k}(\mathbf{c}_{in},\mathbf{c}_{in})}$, the estimated standard deviation. The larger the value of γ_i, the less likely an classification error will happen at $\mathbf{s}_i$. This is reflected by α_i, since it is decreasing in γ_i. With a small weight α_i, the term $\alpha_i\tau_i$ plays a less important role in the optimization (16). The solution to (16) thus automatically allocates more resources to the locations with heavy weights α_is.

V. Simulation Results

We consider a sensor network in a 2-D squared area with $d \times d$ grids. The covariance functions is selected as $k(\mathbf{c}_i, \mathbf{c}_j) = \rho_s^{\|\mathbf{s}_i-\mathbf{s}_j\|} \cdot \rho_t^{|t_i-t_j|}$, where ρ_s and $\rho_t \in [0,1]$ are the spatial and temporal correlation coefficients, respectively. The signal to sensing noise ratio is 30 dB, and the level set threshold is $\gamma = 0.1$.

To illustrate the sensing decision of the proposed algorithm, we first consider a special scenario, where $\rho_t = 1$ and $N = 1$. This may correspond to a temporally slow-varying sensing field and the time interval between any two consecutive sensing actions is small and thus negligible. We set $\rho_s = 0.96$. The level set is estimated based on the first 40 locations selected by Algorithm 3 in Fig. 1. Most of the sensing locations of the greedy algorithm are around the boundary. As a result, the greedy algorithm gets an accurate estimation of the boundary. This matches with our optimization objective, as the accurate identification of the boundary plays a critical role for level set estimation. In this example, the average level set estimation error from the greedy algorithm is 0.0422, yet that from a baseline passive sensing scheme that randomly selects N samples in each slot is 0.0701.

To reduce the estimation complexity, we propose a sliding window scheme, which only keeps samples collected in the most recent T time slots for the regression in Algorithm 1. The average level set estimation error as a function of T is plotted in Fig. 2. In the simulation, we set $\rho_t = 0.9$, $\rho_s = 0.9$, $d = 13$ and fix the total number of samples collected in each slot as $N = 15$. The level set estimation error is normalized by scaling it with $1/Q(\gamma)$, as it is the maximum error given by Algorithm 1 for any sensing scheme. The normalized level set estimation error decreases as T increases, and the proposed greedy algorithm significantly outperforms the passive sensing algorithm. For both algorithms, the level set estimation error do not decrease significantly when $T \geq 6$.

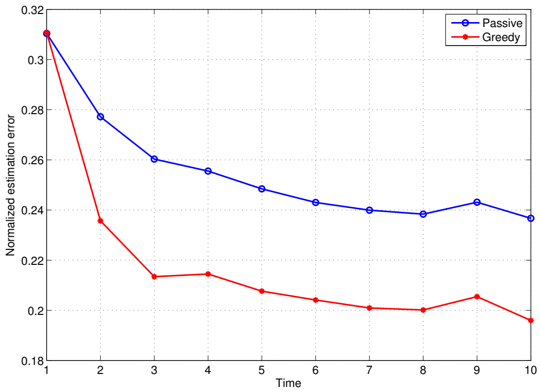

Fig. 2. Performance versus window size T

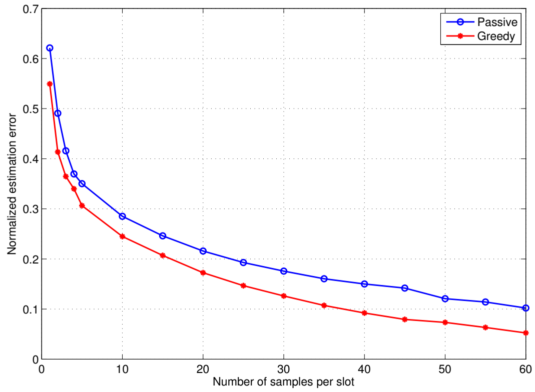

Fig. 3. Performance versus sensing set size N.

Fig. 3 demonstrates the effects of the number of selected sensing locations N on the performance of level set estimation. We set $\rho_t = 0.9$, $\rho_s = 0.9$, and fix the window size T to be 5. For the level set estimation error, the greedy algorithm is strictly better than the passive sensing algorithm. The performance gap between the active sensing schemes and the passive sensing remains almost a constant (around 0.05).

VI. Conclusions

We proposed a dynamic sparse sensing scheme for level set estimation in spatial-temporally correlated random field. The sparse sensing scheme can dynamically adjust the selection of sensing locations based on past sensing results, thus achieving the rapid and accurate extraction of level sets in a large random field with a small number of sensing samples. A greedy algorithm was proposed to achieve dynamic sparse sensing, and it achieved significant performance gains over passive sensing algorithms that do not proactively select the sensing locations.

References

[1] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Phys. Commun.*, vol. 4, no. 1, pp. 40–62, Mar. 2011.

[2] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Nericell: rich monitoring of road and traffic conditions using mobile smartphones," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, ser. SenSys '08. New York, NY, USA: ACM, 2008, pp. 323–336.

[3] N. Maisonneuve, M. Stevens, and B. Ochab, "Participatory noise pollution monitoring using mobile phones," *Info. Pol.*, vol. 15, no. 1,2, pp. 51–71, Apr. 2010.

[4] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[5] X. Han, C. Xu, and J. L. Prince, "A topology preserving level set method for geometric deformable models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 25, no. 6, pp. 755–768, 2003.

[6] A. Gotovos, N. Casati, G. Hitz, and A. Krause, "Active learning for level set estimation," in *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, ser. IJCAI'13, 2013, pp. 1344–1350.

[7] M. Hurtado, C. Muravchik, and A. Nehorai, "Enhanced sparse bayesian learning via statistical thresholding for signals in structured noise," *IEEE Trans. Sig. Proc.*, vol. 61, no. 21, pp. 5430–5443, Nov 2013.

[8] C. Scott and M. Davenport, "Regression level set estimation via cost-sensitive classification," *IEEE Trans. Sig. Proc.*, vol. 55, no. 6, pp. 2752–2757, June 2007.

[9] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.