Automatica 48 (2012) 1735–1740

Contents lists available at SciVerse ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

Spatial prediction with mobile sensor networks using Gaussian processes with built-in Gaussian Markov random fields☆

Yunfei Xu [a], Jongeun Choi [a,b,1]

[a] Department of Mechanical Engineering, Michigan State University, United States

[b] Department of Electrical and Computer Engineering, Michigan State University, United States

ARTICLE INFO

Article history:
Received 26 May 2011
Received in revised form
11 October 2011
Accepted 12 January 2012
Available online 15 June 2012

Keywords:
Mobile sensor networks
Gaussian processes
Gaussian Markov random fields (GMRF)

ABSTRACT

In this paper, a new class of Gaussian processes is proposed for resource-constrained mobile sensor networks. Such a Gaussian process builds on a GMRF with respect to a proximity graph over a surveillance region. The main advantages of using this class of Gaussian processes over standard Gaussian processes defined by mean and covariance functions are its numerical efficiency and scalability due to its built-in GMRF and its capability of representing a wide range of non-stationary physical processes. The formulas for predictive statistics such as predictive mean and variance are derived and a sequential field prediction algorithm is provided for sequentially sampled observations. For a special case using compactly supported weighting functions, we propose a distributed algorithm to implement field prediction by correctly fusing all observations. Simulation and experimental results illustrate the effectiveness of our approach.

1. Introduction

Gaussian processes (or Gaussian random fields) defined by mean and covariance functions over a continuum space (Rasmussen & Williams, 2006) have been frequently used for mobile sensor networks to statistically model physical phenomena such as harmful algal blooms, temperature etc. (Graham & Cortés, 2012; Krause, Singh, & Guestrin, 2008; Leonard et al., 2007; Xu & Choi, 2011; Xu, Choi, & Oh, 2011). Space and time indices can be considered as the input vector to the Gaussian process in order to model the spatio-temporal environmental process (Leonard et al., 2007; Xu et al., 2011).

However, Gaussian process regression, based on the standard mean and covariance functions, requires an inversion of a covariance matrix whose size grows as the number of observations increases. To overcome this increase in complexity, a number of approximation methods for Gaussian process regression have been proposed in the machine learning community. This complexity issue in the context of mobile sensor networks has been tackled in Oh, Xu, and Choi (2010) and Xu et al. (2011). In Xu et al. (2011), it has been proposed that spatio-temporal Gaussian process regression can be applied to truncated observations including only measurements near the target position and time of interest for robots with limited resources.

Recently, there have been efforts to find a way to fit a computationally efficient Gaussian Markov random field (GMRF) on a discrete lattice to a Gaussian random field on a continuum space (Cressie & Verzelen, 2008; Hartman & Hössjer, 2008; Rue & Tjelmeland, 2002). Such methods have been developed using a fitting with a weighted L_2-type distance (Rue & Tjelmeland, 2002), using a conditional-mean least-squares fitting (Cressie & Verzelen, 2008), and for dealing with large data by fast Kriging (Hartman & Hössjer, 2008). It has been demonstrated that GMRFs with small neighborhoods can approximate Gaussian fields surprisingly well (Rue & Tjelmeland, 2002). This approximated GMRF and its regression are very attractive for the resource-constrained mobile sensor networks due to its computational efficiency and scalability (Le Ny & Pappas, 2010) as compared to the standard Gaussian process and its regression, which is not scalable as the number of observations increases.

Mobile sensing agents form an ad-hoc wireless communication network in which each agent usually operates under a short communication range, with limited memory and computational power. For resource-constrained mobile sensor networks, developing distributed prediction algorithms for robotic sensors using only local information from local neighboring agents has been one of the most fundamental problems (Bertsekas & Tsitsiklis, 1999; Bullo, Cortés, & Martínez, 2009; Choi, Oh, & Horowitz, 2009; Cortés, 2009; Graham & Cortés, 2012; Olfati-Saber, Fax, & Murray, 2007).

☆ This work has been supported by the National Science Foundation through CAREER Award CMMI-0846547. This support is gratefully acknowledged. The material in this paper was partially presented at the 4th Annual ASME Dynamic Systems and Control Conference (DSCC), October 31–November 2, 2011, Arlington, Virginia, USA. This paper was recommended for publication in revised form by Associate Editor Wolfgang Scherrer, under the direction of Editor Torsten Söderström.

E-mail addresses: xuyunfei@egr.msu.edu (Y. Xu), jchoi@egr.msu.edu (J. Choi).

[1] Tel.: +1 517 432 3164; fax: +1 517 353 1750.

0005-1098/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.automatica.2012.05.029

The contribution of the paper is as follows. First, a new class of Gaussian processes is proposed for resource-constrained mobile sensor networks. Such a Gaussian process builds on a GMRF (Rue & Held, 2005) with respect to a proximity graph, e.g., the Delaunay graph of a set of vertices over a surveillance region. As a result, this class of Gaussian processes can represent a wide class of non-stationary Gaussian processes and provide computationally efficient and scalable regression analysis. We show that predictive statistics such as predictive mean and variance at any point can be computed by using an inversion of a fixed size matrix, regardless of the number of observations. Exploiting this result, we propose a sequential prediction algorithm which is scalable to deal with sequentially sampled observations (Section 2). We also develop a distributed and scalable statistical inference algorithm for a simple sampling scheme by applying the Jacobi over-relaxation and discrete-time average consensus algorithms (Section 3). Simulation and experimental study demonstrate the usefulness of the proposed model and algorithms (Section 4).

Standard notation will be used throughout the paper. Let $\mathbb{R}$, $\mathbb{R}_{>0}$, $\mathbb{Z}_{>0}$ denote, respectively, the sets of real numbers, positive real numbers, and positive integers. Let $(a)_i$ and $(A)_{ij}$ denote the i-th element in a vector a, and the i,j-th element in a matrix A. Given a matrix $A \in \mathbb{R}^{m\times n}$, let $\text{row}_i(A) \in \mathbb{R}^n$ denote the i-th row of A. The positive definiteness of a matrix A is denoted by $A \succ 0$. Let E, Var and Corr denote, respectively, the operators of expectation, variance and correlation. An undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a tuple consisting of a set of vertices $\mathcal{V} := \{1, \ldots, n\}$ and a set of edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$. The neighbors of $i \in \mathcal{V}$ in $\mathcal{G}$ are denoted by $\mathcal{N}_i := \{j \in \mathcal{V} \mid \{i,j\} \in \mathcal{E}\}$. Other notation will be explained in due course.

2. Spatial prediction

In this section, we first propose a new class of Gaussian random fields with built-in Gaussian Markov random fields (GMRF) (Rue & Held, 2005). Then we show how to compute the prediction at any point of interest based on Gaussian process regression, and provide a sequential field prediction algorithm for mobile sensor networks.

2.1. Spatial model based on GMRF

Let $\gamma := (\gamma(p_1), \ldots, \gamma(p_m))^T$ be a zero-mean GMRF (Rue & Held, 2005) with respect to an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the location of vertex i is denoted by p_i in the surveillance region $\mathcal{D}$. Such locations of vertices will be referred to as *generating points*. The inverse covariance matrix (precision matrix) $Q \succ 0$ has the property $(Q)_{ij} \neq 0 \Leftrightarrow \{i,j\} \in \mathcal{E}$. If the graph $\mathcal{G}$ has small cardinalities of the neighbor sets, its precision matrix Q becomes sparse with many zeros in its entries. This plays a key role in computation efficiency of a GMRF which can be greatly exploited by the resource-constrained mobile sensor network.

The spatial field is modeled by a Gaussian process with a built-in GMRF defined as

$$z(s) = \mu(s) + \sum_{j=1}^{m} \lambda(s, p_j)\gamma(p_j), \tag{1}$$

where $\lambda(\cdot,\cdot)$ is a weighting function. The new class of Gaussian processes is capable of representing a wide range of non-stationary Gaussian fields, by selecting (1) different number of generating points m, (2) different locations of generating points $\{p_j \mid j = 1, \ldots, m\}$ over $\mathcal{D}$, (3) a different structure of the precision matrix Q, and (4) different weighting functions $\{\lambda(\cdot, p_j) \mid j = 1, \ldots, m\}$.

Remark 1. The number of generating points could be determined by a model selection criterion such as the Akaike information criterion (Akaike, 1974). Similar to hyperparameter estimation in the standard Gaussian process regression, one can estimate all other parameters using maximum likelihood (ML) optimization (Rasmussen & Williams, 2006; Xu & Choi, 2011). This optimization is non-convex and so the initial conditions need to be chosen carefully to avoid local minima. In our approach, we use basic structures for weighting functions and the precision matrix, however, we make them as functions of the locations of generating points. Different spatial resolutions can be obtained by a suitable choice of locations of generating points. As an example shown in Fig. 1, higher resolution can be obtained by higher density of generating points (see lower left corner). In this way, we only need to determine the locations of generating points. This approach will be demonstrated with real-world data in Section 4.2.

2.2. Gaussian process regression

Suppose we have a collection of observations $y := (y_1, \ldots, y_n)^T$ whose entries are sampled at the corresponding points $s_1, \ldots, s_n$. The noise corrupted measurement $y_i \in \mathbb{R}$ is given by $y_i = z(s_i) + \epsilon_i$, where $\epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma_\epsilon^2)$ is an independent and identically distributed (i.i.d.) Gaussian white noise. We then have the following results.

Proposition 2. *Let $\Lambda \in \mathbb{R}^{n\times m}$ be a matrix obtained by $(\Lambda)_{ij} = \lambda(s_i, p_j)$ and let $\lambda \in \mathbb{R}^m$ be a vector obtained by $(\lambda)_i = \lambda(s_0, p_i)$, where s_0 is a point of interest. Then the covariance matrix of y and the covariance between y and $z(s_0)$ are given by*

$$C := \mathrm{E}[(y - \mathrm{E}\,y)(y - \mathrm{E}\,y)^T] = \Lambda Q^{-1}\Lambda^T + \sigma_\epsilon^2 I,$$

$$k := \mathrm{E}[(y - \mathrm{E}\,y)z(s_0)] = \Lambda Q^{-1}\lambda,$$

where $Q \in \mathbb{R}^{m\times m}$ is the precision matrix of the GMRF $\gamma \in \mathbb{R}^m$.

Proof. The proof follows by simple algebra. □

By Proposition 2, we can make a prediction at the point of interest s_0 using Gaussian process regression (Rasmussen & Williams, 2006). This is summarized by the following theorem.

Theorem 3. *For given y, the prediction of $z(s_0)$ at any location $s_0 \in \mathcal{D}$ is given by the conditional distribution $z(s_0)|y \sim \mathcal{N}(\hat{z}(s_0), \sigma^2(s_0))$, where the predictive mean and variance are obtained by*

$$\begin{aligned} \hat{z}(s_0) &= \mu(s_0) + \lambda^T \hat{Q}^{-1}\hat{y}, \\ \sigma^2(s_0) &= \lambda^T \hat{Q}^{-1}\lambda, \end{aligned} \tag{2}$$

with $\hat{Q} = Q + \sigma_\epsilon^{-2}\Lambda^T\Lambda \in \mathbb{R}^{m\times m}$ and $\hat{y} = \sigma_\epsilon^{-2}\Lambda^T(y - \mu) \in \mathbb{R}^m$.

Proof. The result follows by simple algebra based on Proposition 2. □

Remark 4. When the generating points $\{p_1, p_2, \ldots, p_m\}$ are not known a priori, they can be estimated by maximizing the likelihood function. Given n observations $y = (y_1, y_2, \ldots, y_n)^T$ sampled at $\{s_1, s_2, \ldots, s_n\}$, the maximum likelihood estimate of the generating points can be obtained via solving the following optimization problem

$$\hat{p}_{\mathrm{ML}} = \arg\max_p \log \pi(y), \tag{3}$$

where $\pi(y)$ is the likelihood function.

Remark 5. Note that the number of generating points m is fixed and the number of observations n may grow in time, and so in general we consider $m \ll n$. Theorem 3 shows that only the inversion of an $m \times m$ matrix $\hat{Q} = Q + \sigma_\epsilon^{-2}\Lambda^T\Lambda$ is required in order to compute the predictive distribution of the field at any point. The computational complexity grows linearly with the number of observations, i.e., $O(nm^2)$, as compared to the standard Gaussian

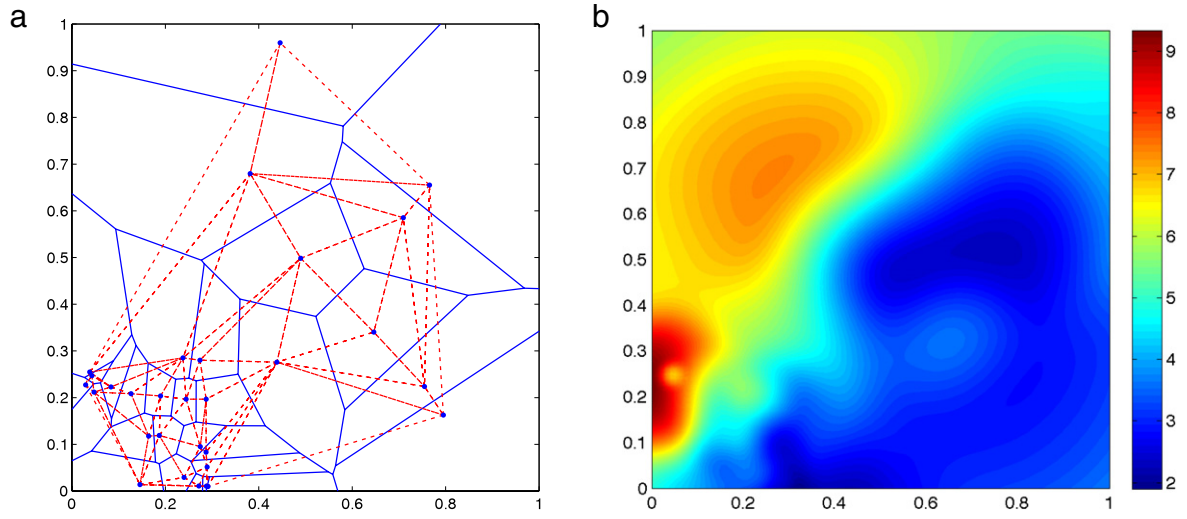


Fig. 1. (a) The generating points in blue dots and the associated Delaunay graph with edges in red dotted lines. The Voronoi partition is also shown in blue solid lines. (b) A Gaussian random field with a built-in GMRF with respect to the Delaunay graph in (a).

process regression which requires $O(n^3)$. To eliminate the linear growth of complexity $O(nm^2)$, as n increases, in what follows, we present a sequential field prediction algorithm for sequential observations by exploiting the results of Theorem 3. In particular, we show that the computational complexity will be fixed at $O(Nm^2)$, where N is the number of newly added observations in each iteration.

2.3. Sequential prediction algorithm

Consider a sensor network consisting of N mobile sensing agents distributed in the surveillance region $\mathcal{D}$. The index of the robotic sensors is denoted by $\mathcal{I} := \{1, \ldots, N\}$. The sensing agents sample the environmental field at time $t \in \mathbb{Z}_{>0}$ and send the observations to a central station which is in charge of the data fusion.

At time t, agent i makes an observation $y_i(t)$ at location $s_i(t)$. Denote the collection of observations at time t by $y_t := (y_1(t), \ldots, y_N(t))^T$. We have the following proposition.

Proposition 6. *At time $t \in \mathbb{Z}_{>0}$, the predictive mean and variance at any point of interest can be obtained via (2) with*

$$\hat{Q}_t = \hat{Q}_{t-1} + \sigma_\epsilon^{-2} \Lambda_t^T \Lambda_t, \qquad \hat{Q}_0 = Q$$
$$\hat{y}_t = \hat{y}_{t-1} + \sigma_\epsilon^{-2} \Lambda_t^T (y_t - \mu_t), \qquad \hat{y}_0 = 0,$$

where $(\Lambda_t)_{ij} = \lambda(s_i(t), s_j(t))$, and $(\mu_t)_i = \mu(s_i(t))$.

Proof. The result can be obtained easily by noting that $A^T A = A_1^T A_1 + A_2^T A_2$, where $A = (A_1^T, A_2^T)^T$. □

Based on Proposition 6, the prediction of $z(s_0)$ can be computed sequentially as time t increases.

3. Distributed spatial prediction

In this section, we propose a distributed approach, in which robotic sensors exchange only local information between neighbors, to implement the field prediction effectively fusing all observations collected by all sensors correctly. This distributed approach can be implemented for a class of weighting functions $\lambda(\cdot, \cdot)$ in (1) that have compact supports. In particular, we consider the weighting function defined by

$$\lambda(s, p_j) = \lambda(\|s - p_j\|/r), \tag{4}$$

where $\lambda(h) = (1 - h)\cos(\pi h) + \frac{1}{\pi}\sin(\pi h)$ if $h \leq 1$, and 0 otherwise. Notice that the weighting function $\lambda(\cdot, \cdot)$ in (4) has a compact support, i.e., $\lambda(s, p_j)$ is non-zero if and only if the distance $\|s - p_j\|$ is less than the support $r \in \mathbb{R}_{>0}$.

3.1. Distributed computation

We first briefly introduce distributed algorithms for solving linear systems and computing the averages. They will be used as major tools for distributed implementation of field prediction.

- **Jacobi over-relaxation method:** The Jacobi over-relaxation (JOR) (Bertsekas & Tsitsiklis, 1999) method provides an iterative solution of a linear system $Ax = b$, where $A \in \mathbb{R}^{N \times N}$ is a nonsingular matrix and $x, b \in \mathbb{R}^N$. If agent i knows the $\text{row}_i(A) \in \mathbb{R}^N$ and b_i, and $a_{ij} = (A)_{ij} = 0$ if agent i and agent j are not neighbors, then the recursion is given by

$$x_i^{(k+1)} = (1 - h)x_i^{(k)} + \frac{h}{a_{ii}} \left(b_i - \sum_{j \in \mathcal{N}_i} a_{ij} x_j^{(k)} \right). \tag{5}$$

  This JOR algorithm converges to the solution of $Ax = b$ from any initial condition if $h < 2/N$ (Cortés, 2009). At the end of the algorithm, agent i knows the i-th element of $x = A^{-1}b$.
- **Discrete-time average consensus:** The Discrete-time average consensus (DAC) provides a way to compute the arithmetic mean of elements in the a vector $c \in \mathbb{R}^N$. Assume the graph is connected. If agent i knows the i-th element of c, the network can compute the arithmetic mean via the following recursion (Olfati-Saber et al., 2007)

$$x_i^{(k+1)} = x_i^{(k)} + \epsilon \sum_{j \in \mathcal{N}_i} a_{ij} (x_j^{(k)} - x_i^{(k)}), \tag{6}$$

  with initial condition $x(0) = c$, where $a_{ij} = 1$ if $j \in \mathcal{N}_i$ and 0 otherwise, $0 < \epsilon < 1/\Delta$, and $\Delta = \max_i(\sum_{j \neq i} a_{ij})$ is the maximum degree of the network. After the algorithm converges, all nodes in the network know the average of c, i.e., $\sum_{i=1}^n c_i/N$.

3.2. Distributed prediction algorithm

Consider a GMRF with respect to a proximity graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ that generates a Gaussian random field in (1). The index of the generating points is denoted by $\mathcal{V} := \{1, \ldots, m\}$. The location of

the i-th generating point is p_i. The edges of the graph are considered to be $\mathcal{E} := \{\{i, j\} \mid \|p_i - p_j\| \leq R\}$, where R is a constant that ensures the graph is connected.

Consider a mobile sensor network consisting of N mobile sensing agents distributed in the surveillance region $\mathcal{D}$. For simplicity, we assume that the number of agents is equal to the number of generating points, i.e., $N = m$. The index of the robotic sensors is denoted by $\mathcal{I} = \mathcal{V}$. The location of agent i is denoted by s_i.

The assumptions made for the resource-constrained mobile sensor networks are listed as follows.

A.1 Agent i is in charge of sampling at point s_i within a r-disk centered at p_i, i.e., $\|s_i - p_i\| < r$.
A.2 r is the radius of the support of the weighting function in (4) and also satisfies that $0 < r < \frac{R}{2}$.
A.3 Agent i can only locally communicate with neighbors in $\mathcal{N}_i := \{j \in \mathcal{I} \mid \{i, j\} \in \mathcal{E}\}$ defined by the connected proximity graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$.
A.4 Agent i knows $\text{row}_i(Q)$, i.e., the i-th row of Q, where $(Q)_{ij} \neq 0$ if and only if $j \in \{i\} \cup \mathcal{N}_i$.

Remark 7. As in A.1, it is reasonable to have at least one agent collect measurements that are correlated with a random variable from a single generating point. This sampling rule may be modified such that a single agent dynamically samples for multiple generating points or more number of agents samples for a generating point depending on available resources. Since there is at least one agent in charge of a generating point by A.1, it is natural to have A.3 and A.4 taking advantage of the proximity graph for the GMRF. Notice that each agent only knows local information of Q as described in A.4.

From A.1 and A.2, since $R > 2r$, we have $\lambda(s_\ell, p_i) = 0$ if $\ell \notin \mathcal{N}_i$. Thus the matrix $\hat{Q} = Q + \sigma_\epsilon^{-2}\Lambda^T\Lambda \in \mathbb{R}^{m\times m}$ and the vector $\hat{y} = \sigma_\epsilon^{-2}\Lambda^T(y - \mu) \in \mathbb{R}^m$ can be obtained in the following form.

$$\begin{aligned}(\hat{Q})_{ij} &= (Q)_{ij} + \sigma_\epsilon^{-2} \sum_{\ell\in\{\{i\}\cup\mathcal{N}_i\}\cap\{\{j\}\cup\mathcal{N}_j\}} \lambda(s_\ell, p_i)\lambda(s_\ell, p_j),\\ (\hat{y})_i &= \sigma_\epsilon^{-2} \sum_{\ell\in\{\{i\}\cup\mathcal{N}_i} \lambda(s_\ell, p_i)(y_\ell - \mu_\ell).\end{aligned} \tag{7}$$

Notice that $\hat{Q}$ has the same sparsity as Q. From (7), A.3 and A.4, agent i can compute $\text{row}_i(\hat{Q})$ and $(\hat{y})_i$ by using only local information from neighbors. Using $\text{row}_i(\hat{Q})$ and $(\lambda)_i$, agent i can obtain the i-th element in the vector $\hat{Q}^{-1}\lambda = (Q + \sigma_\epsilon^{-2}\Lambda^T\Lambda)^{-1}\lambda$ via JOR by using only local information. Finally, using $(\hat{y})_i$ and $(\lambda)_i$ the prediction mean and variance can be obtained via the discrete-time average consensus algorithm. Notice that the sequential update of $\hat{Q}$ and $\hat{y}$ for sequential observations proposed in Section 2.3 can be also applied to the distributed algorithm. The distributed algorithm for sequential field prediction under assumptions A.1–4 is summarized in Table 1.

The number of robotic sensors and the sampling rule can be modified or optimized to maintain a better quality of the prediction and the corresponding distributed algorithm may be derived in a same way accordingly.

Table 1
Distributed algorithm for sequential field prediction.

Input:
(1) a set of target points $\mathcal{S}$
(2) the topology of sensor network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ in which $\mathcal{E} := \{\{i, j\} \mid \|p_i - p_j\| \leq R\}$
Output:
(1) prediction mean $\{\hat{z}(s_0) \mid s_0 \in \mathcal{S}\}$
(2) prediction error variance $\{\sigma^2(s_0) \mid s_0 \in \mathcal{S}\}$
Assumption:
(A1) agent $i \in \mathcal{I}$ is in charge of sampling at point s_i within a r-disk centered at p_i, i.e., $\|s_i - p_i\| < r$
(A2) the radius of the support of the weighting function satisfies $0 < r < \frac{R}{2}$
(A3) agent $i \in \mathcal{I}$ can only locally communicate with neighbors $\mathcal{N}_i := \{j \in \mathcal{I} \mid \{i, j\} \in \mathcal{E}\}$ defined by the connected graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$
(A4) agent $i \in \mathcal{I}$ initially has $\text{row}_i(\hat{Q}) \leftarrow \text{row}_i(Q)$, $(\hat{y})_i \leftarrow 0$

At time t, agent $i \in \mathcal{I}$ in the network does the following concurrently:

1: take measurement y_i from its current location s_i
2: update $\text{row}_i(\hat{Q}) \leftarrow \text{row}_i(\hat{Q}) + \text{row}_i(\sigma_\epsilon^{-2}\Lambda^T\Lambda)$ by exchanging information from neighbors $\mathcal{N}_i$
3: update $(\hat{y})_i \leftarrow (\hat{y})_i + (\sigma_\epsilon^{-2}\Lambda^T(y - \mu))_i$ by exchanging information from neighbors $\mathcal{N}_i$
4: **for** $s_0 \in \mathcal{S}$ **do**
5: compute $(\lambda)_i = \lambda(s_0, p_i)$
6: compute $(\hat{Q}^{-1}\lambda)_i$ via JOR
7: compute $\hat{z}(s_0) = \mu(s_0) + \lambda^T\hat{Q}^{-1}\hat{y}$ via DAC
8: compute $\sigma^2(s_0) = \lambda^T\hat{Q}^{-1}\lambda$ via DAC
9: **end for**

4. Simulation and experiment

In this section, we apply the proposed schemes to both simulation and experimental study.

4.1. Simulation

We first apply our proposed prediction algorithms to a numerically generated Gaussian random field $z(\cdot)$ based on a GMRF with respect to a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ defined in (1). The mean function $\mu(\cdot)$ is assumed to be constant and $\mu = 5$ is used in the simulation. We assume the generating points of the GMRF, indexed by $\mathcal{V} = \{1, \ldots, m\}$ where $m = 30$, are located at $\{p_1, \ldots, p_m\}$ in a 2-D unit area $\mathcal{D}$. The edges of the graph are assumed to be $\mathcal{E} := \{\{i, j\} \mid \|p_i - p_j\| \leq R\}$, where $R = 0.4$.

The GMRF $\gamma = (\gamma(p_1), \ldots, \gamma(p_m))^T$ has a zero-mean and the precision matrix Q is given by

$$(Q)_{ij} = \begin{cases} |\mathcal{N}_i| + c_0, & \text{if } j = i,\\ -1, & \text{if } j \in \mathcal{N}_i,\\ 0, & \text{otherwise,}\end{cases}$$

where $|\mathcal{N}_i|$ denotes the degree of node i, i.e., the number of connections it has to other nodes, $c_0 = 0.1$ is used to ensure Q is positive definite since a Hermitian diagonally dominant matrix with real non-negative diagonal entries is positive semi-definite (Rue & Held, 2005). We use compactly supported weighting functions defined in (4) for both centralized and distributed schemes with different support r. The sensor noise level is given by $\sigma_\epsilon = 0.5$. Since the optimal sampling is beyond the scope of this paper, in the simulation, we use a random sampling strategy in which robotic sensors sample at random locations at each time instance.

We consider a scenario in which prediction is implemented in a distributed fashion (Table 1) under assumptions A.1–4 for the resource-constrained mobile sensor network in Section 3.2. In particular, $N = 30$ robotic sensors are distributed according to the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, which is connected. Agent i is in charge of the sampling with in a r-disk centered at p_i, where the support $r = 0.2$ is used. Agent i has a fixed neighborhood, i.e., $\mathcal{N}_i = \{j \mid \{i, j\} \in \mathcal{E}\}$. In the simulation, $h = 0.02$ in (5) and $\epsilon = 0.02$ in (6) are chosen to ensure the convergence of the JOR algorithm and the DAC algorithm.

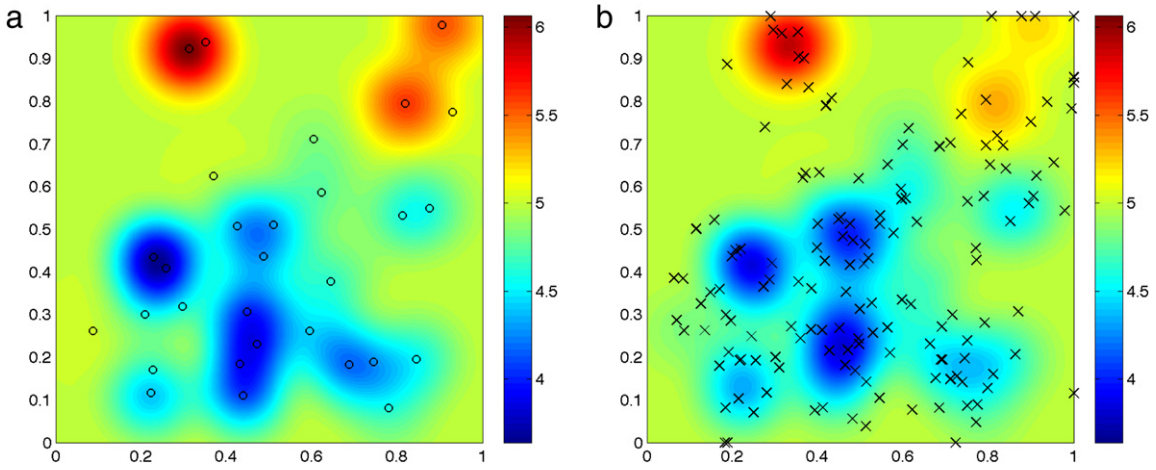


Fig. 2. Simulation results for the distributed scheme. (a) The true field, (b) the predicted field at time $t = 5$. The generating points are shown in circles, and the sampling locations are shown in crosses.

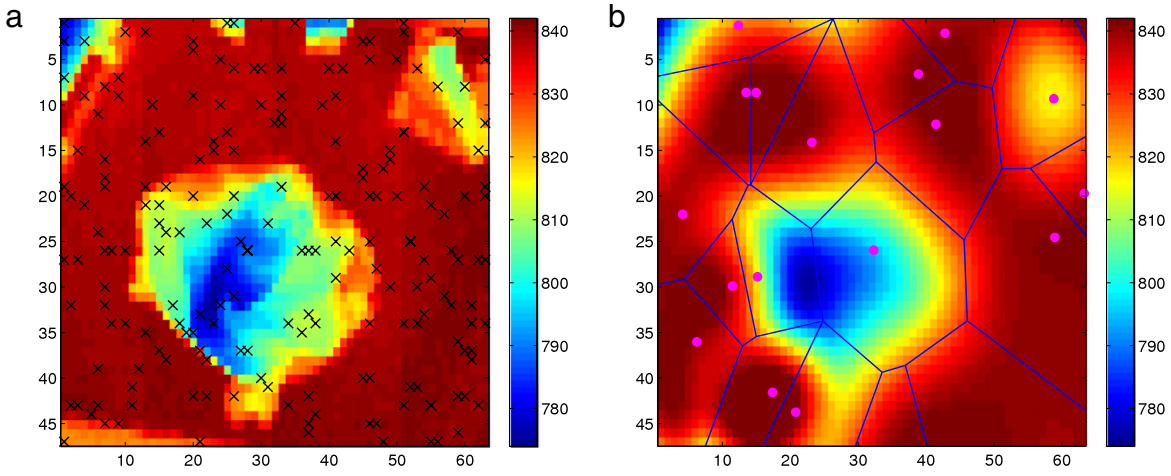


Fig. 3. (a) The true field on grid positions obtained by the Kinect sensor and randomly sampled positions indicated in black crosses. (b) The fitted Gaussian random field with a build-in GMRF with respect to the Delaunay graph.

The true and the predicted fields at time $t = 5$ are shown in Fig. 2-(a) and (b), respectively. The normalized RMS errors (RMS divided by the range of observed values) computed over about 10,000 grid points at $t = 1, 2, \ldots, 5$ are 10%, 8%, 5.5%, 4.5%, and 4%, respectively. The computational time at each time instance remains fixed due to the nice structure of the proposed Gaussian field in (1) and its consequent results from Theorem 3.

4.2. Experiment

In order to show the practical usefulness of the proposed approach, we apply the centralized scheme in Theorem 3 on experimentally obtained observations. We first measured depth values of a terrain on grid points by using a Microsoft Kinect sensor (Microsoft corporation, 0000) as shown in Fig. 3-(a). As pointed out in Remark 1, we make the structures of weighting functions and the precision matrix as functions of the locations of generating points. In particular, two generating points are neighbors if and only if their corresponding Voronoi cells intersect. The individual weighting function takes the same form as in (4) and its support size r_i is selected to be the largest distance between the generating point i and its neighbors. We then predict the field by our model with 20 estimated generating points given by the ML estimator in (3) using a subset of experimental observations, i.e., 200 randomly sampled observations denoted by crosses in Fig. 3-(a). The estimated positions of generating points along with the predicted field are shown in Fig. 3-(b). In this experiment, it is clear to see that our approach effectively produces the predicted field, which is very close to the true field for the case of unknown generating points.

5. Conclusion

In this paper, we introduced a new class of Gaussian processes with built-in GMRFs for modeling a wide range of environmental fields. The Gaussian process regression for the predictive statistics at any point of interest was provided and a sequential field prediction algorithm with fixed complexity was proposed to deal with sequentially sampled observations. For a special case with compactly supported weighting functions, we proposed a distributed field prediction algorithm in which the prediction can be computed via Jacobi over-relaxation algorithm and discrete-time average consensus. Optimal learning and sampling of the Gaussian process with built-in GMRF using mobile sensor networks will be investigated in the future.

Acknowledgments

The authors thank Mr. Justin Mrkva for his effort to obtain the experimental data used in Section 4.2. The authors would also like

to thank the associate editor and the anonymous reviewers for their valuable comments and suggestions.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *19*(6), 716–723.

Bertsekas, D. P., & Tsitsiklis, J. N. (1999). *Parallel and distributed computation: numerical methods*. Englewood Cliffs: Prentice Hall.

Bullo, F., Cortés, J., & Martínez, S. (2009). *Applied mathematics series*, *Distributed control of robotic networks*. Princeton University Press.

Choi, J., Oh, S., & Horowitz, R. (2009). Distributed learning and cooperative control for multi-agent systems. *Automatica*, *45*, 2802–2814.

Cortés, J. (2009). Distributed Kriged Kalman filter for spatial estimation. *IEEE Transactions on Automatic Control*, *54*(12), 2816–2827.

Cressie, N., & Verzelen, N. (2008). Conditional-mean least-squares fitting of Gaussian Markov random fields to Gaussian fields. *Computational Statistics & Data Analysis*, *52*(5), 2794–2807.

Graham, R., & Cortés, J. (2012). Cooperative adaptive sampling of random fields with partially known covariance. *International Journal of Robust and Nonlinear Control*, *22*(5), 504–534.

Hartman, L., & Hössjer, O. (2008). Fast kriging of large data sets with Gaussian Markov random fields. *Computational Statistics & Data Analysis*, *52*(5), 2331–2349.

Krause, A., Singh, A., & Guestrin, C. (2008). Near-optimal sensor placements in Gaussian processes: theory, efficient algorithms and empirical studies. *The Journal of Machine Learning Research*, *9*, 235–284.

Le Ny, J., & Pappas, G.J. (2010). On trajectory optimization for active sensing in Gaussian process models. In *Proceedings of the 48th IEEE conference on decision and control* (pp. 6286–6292).

Leonard, N. E., Paley, D. A., Lekien, F., Sepulchre, R., Fratantoni, D. M., & Davis, R. E. (2007). Collective motion, sensor networks, and ocean sampling. *Proceedings of the IEEE*, *95*(1), 48–74.

Microsoft corporation. Official website of Kinect for Xbox 360. http://www.xbox.com/en-US/kinect.

Oh, S., Xu, Y., & Choi, J. (2010). Explorative navigation of mobile sensor networks using sparse Gaussian processes. In *Proceedings of the 49th IEEE conference on decision and control (CDC)* (pp. 3851–3856).

Olfati-Saber, R., Fax, J. A., & Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, *95*(1), 215–233.

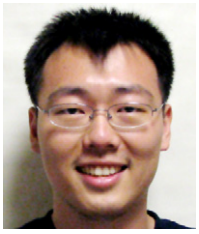
Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. Cambridge, Massachusetts, London, England: The MIT Press.

Rue, H., & Held, L. (2005). *Gaussian Markov random fields: theory and applications*. Chapman & Hall.

Rue, H., & Tjelmeland, H. (2002). Fitting Gaussian Markov random fields to Gaussian fields. *Scandinavian Journal of Statistics*, *29*(1), 31–49.

Xu, Y., & Choi, J. (2011). Adaptive sampling for learning Gaussian processes using mobile sensor networks. *Sensors*, *11*(3), 3051–3066.

Xu, Y., Choi, J., & Oh, S. (2011). Mobile sensor network navigation using Gaussian processes with truncated observations. *IEEE Transactions on Robotics*, *17*(6), 1118–1131.



Yunfei Xu received his Ph.D. in Mechanical Engineering from Michigan State University in 2011. He also received his M.S. and B.S. degrees in Automotive Engineering from Tsinghua University, Beijing, China, in 2004 and 2007, respectively. Currently, he is a Research Fellow with the Department of Mechanical Engineering at Michigan State University. His current research interests include environmental adaptive sampling algorithms, Gaussian processes, and statistical learning algorithms with applications to robotics and mobile sensor networks. His paper was a finalist for the Best Student Paper Award at the Dynamic System and Control Conference (DSCC) 2011. Dr. Xu is a member of IEEE and ASME.

Jongeun Choi received his Ph.D. and M.S. degrees in Mechanical Engineering from the University of California at Berkeley in 2006 and 2002 respectively. He also received a B.S. degree in Mechanical Design and Production Engineering from Yonsei University at Seoul, Republic of Korea in 1998. He is currently an Assistant Professor with the Department of Mechanical Engineering and the Department of Electrical and Computer Engineering at the Michigan State University. His research interests include adaptive, distributed and robust control and statistical learning algorithms, with applications to mobile robotic sensors, environmental adaptive sampling, engine control, and biomedical problems. He was a recipient of an NSF CAREER Award in 2009. His papers were finalists for the Best Student Paper Award at the 24th American Control Conference (ACC) 2005 and the Dynamic System and Control Conference (DSCC) 2011. Dr. Choi is a member of IEEE and ASME.