

# An Efficient Adaptive Sampling Approach for Mobile Robotic Sensor Networks using Proximal ADMM

Viet-Anh Le<sup>1</sup>, Linh Nguyen<sup>2</sup> and Truong X. Nghiem<sup>1</sup>

**Abstract**—Adaptive sampling in a resource-constrained mobile robotic sensor network for monitoring a spatial phenomenon is a fundamental but challenging problem. In applications where a Gaussian Process is employed to model a spatial field and then to predict the field at unobserved locations, the adaptive sampling problem can be formulated as minimizing the negative log determinant of a predicted covariance matrix, which is a non-convex and highly complex function. Consequently, this optimization problem is typically addressed in a grid-based discrete domain, although it is combinatorial NP-hard and only a near-optimal solution can be obtained. To overcome this challenge, we propose using a proximal alternating direction method of multipliers (Px-ADMM) technique to solve the adaptive sampling optimization problem in a continuous domain. Numerical simulations using a real-world dataset demonstrate that the proposed PxADMM-based method outperforms a commonly used grid-based greedy method in the final model accuracy.

## I. INTRODUCTION

A network of mobile robotic sensors is frequently employed in applications of monitoring environmental spatial fields, such as exploring ecosystems on land and in ocean, observing chemical concentration, and monitoring air pollutants and indoor climates [1]. Nevertheless, the fundamental problem of how to optimally drive the resource-constrained mobile sensors on sampling paths so that the information gained by the sensor measurements is maximal while the prediction uncertainty is minimal, known as *adaptive sampling*, is still practically challenging. In such a problem setting, while the mobile robots navigate through the environment, their on-board sensors take measurements of the spatial field and continually update a model of the field. The collective measurements and the corresponding model are then used to decide optimally where the mobile sensors should move at every moving step.

The model used to represent the environmental spatial field in adaptive sampling can be either parametric or non-parametric [2], [3]. A parametric model requires *a priori* assumptions about the model structure and initial conditions, which are not easily obtained in practice. In contrast, a non-parametric model can be learned purely based on the collected data, resulting in its wide adoption in the literature [2], [3]. One such non-parametric model is the Gaussian Process (GP) [4]. A GP model can effectively learn an unknown spatial

phenomenon from a limited number of sensor measurements and then predict it at any unobserved locations of interest.

Finding the optimal sampling locations for the robotic sensors at each time step can be formulated as minimizing the prediction uncertainty in the regions of interest. In the literature, various adaptive sampling optimization formulations have been considered. For instance, Xu *et al.* in [5] proposed to minimize the Fisher information matrix based objective function to find the optimal sampling locations for a MRSN. In [6], the authors considered a minimization problem where the cost function is derived from the average of the prediction variances over prespecified target points. In [7], the maximum a posterior estimation was exploited to design an adaptive sampling strategy for an MRSN by minimizing the prediction error variances. The adaptive sampling optimization problems in the abovementioned works [5]–[7] were solved by the gradient descent method. However, the robot dynamics and constraints on the physical mobility of the robots were not considered. With expectation of a trade-off between exploration and exploitation, Marchant *et al.* in [8] employed a Bayesian optimization approach to derive a sampling strategy for a MRSN, where the travelled distances of mobile robots were taken into account. More interestingly, some works used concepts from information theory to formulate the adaptive sampling optimization problem. For instance, in both [9] and [10], the authors utilized conditional entropy in a sampling optimality criterion to represent prediction uncertainty. Xu *et al.* in [10] aimed to minimize not only the prediction errors but also the hyperparameter uncertainty in the sampling optimization problem. In our previous works [11], [12], we employed both the conditional entropy and the posterior variances to design an adaptive sampling strategy for an MRSN. The dynamic constraints in the network along with a collision avoidance scheme were also incorporated into the optimization problem.

To the best of our knowledge, most of the works that employed continuous optimization algorithms to solve the adaptive sampling problem did not take into account the dynamics and the mobility constraints of the robots. On the other hand, the adaptive sampling optimization problems in the works that considered the robot dynamics and resource constraints were usually solved by grid-based search algorithms. However, these search algorithms suffer from the combinatorial NP-hard complexity of the grid-based approach and can only provide near-optimal solutions. To address the shortcomings of the above two approaches in the literature, in this work we propose using a proximal alternating direction method of multipliers (PxADMM) method [13]

<sup>1</sup>V-A. Le and T. Nghiem are with the School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86011, USA {vl385,truong.nghiem}@nau.edu

<sup>2</sup>L. Nguyen is with the School of Engineering, Information Technology and Physical Sciences, Federation University Australia, Churchill, VIC 3842, Australia l.nguyen@federation.edu.au

to solve the adaptive sampling optimization problem for MRSNs subjecting to constraints on the physical mobility of the robots in a continuous domain. In our adaptive sampling optimization formulation, the spatial field is modeled by a GP and the conditional entropy is employed as the cost function that is represented by the log determinant of the predicted covariance matrix. The PxADMM method overcomes the critical computational intractability issue of the resulting optimization problem by using a first-order approximation of the log determinant objective function. The proposed approach was evaluated in numerical simulations using a real-world dataset, which demonstrated that it could achieve a more accurate model of the spatial environmental field compared to a commonly used grid-based greedy method while maintaining a practical real-time computational performance.

The remaining of the paper is organized as follows. Section II introduces a MRSN for monitoring a spatial field where the adaptive sampling optimization problem based on conditional entropy is formulated. Section III presents our proposed PxADMM method to practically address the adaptive sampling optimization problem. The numerical evaluation of our approach is presented in Section IV, followed by the conclusions in Section V.

## II. MOBILE SENSOR NETWORKS FOR EFFICIENT SPATIAL PREDICTION

This section mathematically formalizes the adaptive sampling optimization problem of a resource-constrained network of mobile sensors for efficiently observing a phenomenon in an environment.

### A. Mobile Robotic Sensor Networks

Consider a network of  $M$  mobile robotic sensors, where each mobile robot carries a spatial-field sensor and all the sensors are assumed to be identical. The dynamics of each robot is described by the following discretized single-integrator model

$$\mathbf{s}_{i,t+1} = \mathbf{s}_{i,t} + \mathbf{u}_{i,t}\Delta T, \quad (1)$$

where  $\mathbf{s}_{i,t+1}$  and  $\mathbf{s}_{i,t}$  are the location vectors of robot  $i$  at time steps  $t+1$  and  $t$ , respectively,  $\mathbf{u}_{i,t}$  is the control input vector of robot  $i$  at time step  $t$ , and  $\Delta T$  is the duration between two time steps. It is assumed that the network is operated in a compact 2D Euclidean space of interest  $\mathcal{Q} \subset \mathbb{R}^2$ , i.e.,  $\mathbf{s}_{i,t} \in \mathcal{Q}$  for all  $i = 1, \dots, M$  and all  $t \geq 0$ . Moreover, due to the robot actuation constraint, we define a bound constraint on the control input  $\mathbf{u}_{i,t} \in \mathcal{U}_i$  where  $\mathcal{U}_i := \{\mathbf{u} \in \mathbb{R}^2 \mid \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max}\}$ . Let  $\mathbf{s}_t = [\mathbf{s}_{i,t}]_{i=1,\dots,M} \in \mathbb{R}^{2 \times M}$  and  $\mathbf{u}_t = [\mathbf{u}_{i,t}]_{i=1,\dots,M} \in \mathbb{R}^{2 \times M}$  denote the matrices of all mobile robotic sensor locations and inputs at time step  $t$ , while  $\mathbf{s}_{0:t} = [\mathbf{s}_k]_{k=0,\dots,t} \in \mathbb{R}^{2 \times (t+1)M}$  and  $\mathbf{u}_{0:t} = [\mathbf{u}_k]_{k=0,\dots,t} \in \mathbb{R}^{2 \times (t+1)M}$  are defined as their collective matrices.

At each time step, the mobile sensor  $i$  takes a noisy measurement of the spatial field of interest at its current location, which is modeled as

$$y_{i,t} = h(\mathbf{s}_{i,t}) + w_i, \quad (2)$$

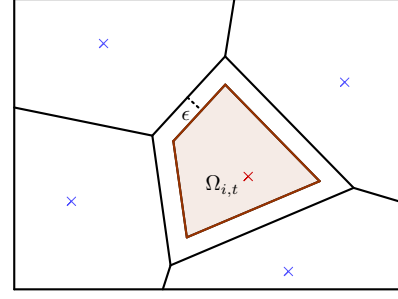


Fig. 1: A constrained region (shaded area) for a mobile sensor (red cross).

where  $h : \mathbb{R}^2 \rightarrow \mathbb{R}$  is a latent function representing the spatial phenomenon while  $w_i$  is a Gaussian zero-mean independent and identically distributed noise. All the sensor measurements at time step  $t$  are denoted by  $\mathbf{y}_t = [\mathbf{y}_{i,t}]_{i=1,\dots,M} \in \mathbb{R}^{2 \times M}$  and the collective measurements from time step 0 to time step  $t$  are denoted by  $\mathbf{y}_{0:t} = [\mathbf{y}_k]_{k=0,\dots,t} \in \mathbb{R}^{2 \times (t+1)M}$ .

As commonly used in the research literature, a Gaussian Process (GP) is employed to learn a model of the latent spatial field from the collective sensor measurements. At time step  $t$ , let  $\mathcal{G}_{h,t}$  denote the GP model trained on the data set  $\mathcal{D}_t = (\mathbf{s}_{0:t}, \mathbf{y}_{0:t})$ . Statistically,  $\mathcal{G}_{h,t}$  is fully specified by a mean function  $m(\mathbf{s}; \theta_t)$  and a covariance function  $\text{cov}(\mathbf{s}, \mathbf{s}'; \theta_t)$  that are parameterized by the hyperparameters  $\theta_t$ . For further details about GPs, interested readers are referred to [4].

**Inter-robot collision avoidance:** To avoid physical collision among the mobile robots during their navigation, we employ a technique based on the Voronoi partition. At time  $t$ , the Voronoi partition is  $\{V_{i,t}\}_{i=1,\dots,M}$ , where  $V_{i,t}$  defined as

$$V_{i,t} := \{\mathbf{q} \in \mathcal{Q} \mid \|\mathbf{q} - \mathbf{s}_{i,t}\| \leq \|\mathbf{q} - \mathbf{s}_{j,t}\|, \forall j \neq i\}$$

is the Voronoi cell of the mobile robot  $i$ . Each Voronoi cell  $V_{i,t}$  is shrunk by a small safety threshold  $\epsilon > 0$ , as illustrated in Fig. 1, to obtain the safe constraint set  $\Omega_{i,t}$  of the next location of robot  $i$  at the next time step  $t+1$ , i.e.,  $\mathbf{s}_{i,t+1} \in \Omega_{i,t}$ .

### B. Adaptive Sampling Optimization Problem

Given a limited number of mobile sensors and their limited measurements during the course of operation of an MRSN, in order to effectively monitor a spatial phenomenon, it is crucial to drive the mobile sensors on the *most informative sampling paths*. In our previous work [11], we showed that optimal sampling locations for the mobile sensors at the next time step can be determined by maximizing the conditional entropy at the next sensor locations given the observations collected up to the current time step.

Given the GP model  $\mathcal{G}_{h,t}$  trained on the collected observations  $\mathbf{y}_{0:t}$  up to time  $t$ , the measurements  $\hat{\mathbf{y}}_{t+1}$  at the next sensor locations  $\mathbf{s}_{t+1}$  can be predicted by a posterior mean vector and a posterior covariance as follows,

$$\mu_{\hat{\mathbf{y}}_{t+1}|\mathbf{y}_{0:t}} = \mathbf{m}(\mathbf{s}_{t+1}) + \Sigma_{t+1,0:t} \Sigma_{0:t}^{-1} (\mathbf{y}_{0:t} - \mathbf{m}(\mathbf{s}_{0:t})) \quad (3)$$

$$\Sigma_{\hat{\mathbf{y}}_{t+1}|\mathbf{y}_{0:t}} = \Sigma_{t+1,t+1} - \Sigma_{t+1,0:t} \Sigma_{0:t}^{-1} \Sigma_{0:t,t+1}^T.$$

Here,  $\mathbf{m}(\mathbf{s}_{0:t})$  and  $\mathbf{m}(\mathbf{s}_{t+1})$  are the mean vectors of the latent variables at  $\mathbf{s}_{0:t}$  and  $\mathbf{s}_{t+1}$ , respectively. The  $M \times M$  matrix  $\Sigma_{t+1,t+1}$  is the covariance matrix at  $\mathbf{s}_{t+1}$ ,  $\Sigma_{t+1,0:t}$

is the  $M \times (t+1)M$  cross-covariance matrix between  $\hat{\mathbf{y}}_{t+1}$  and  $\mathbf{y}_{0:t}$ , and  $\Sigma_{0:t}$  is the  $(t+1)M \times (t+1)M$  covariance matrix of  $\mathbf{y}_{0:t}$ . For simplicity, we denote the covariance matrix  $\Sigma_{\hat{\mathbf{y}}_{t+1}|\mathbf{y}_{0:t}}$  as  $\Sigma(\mathbf{s}_{t+1})$  henceforth. Therefore, the conditional entropy at the next sensor locations can be computed by the logarithm of determinant of the covariance matrix. In other words, the optimal sampling locations for the mobile sensors at the next time step can be obtained by solving the following optimization problem,

$$\mathbf{s}_{t+1}^* = \operatorname{argmax}_{\mathbf{s}_{t+1}} \log \det \Sigma(\mathbf{s}_{t+1}). \quad (4)$$

Given the dynamic constraints on the robot movement presented in Section II-A, the optimization problem for the adaptive sampling in an MRSN can be formulated as

$$\begin{aligned} & \underset{\{\mathbf{s}_{t+1}, \mathbf{u}_t\}}{\text{minimize}} && -\log \det \Sigma(\mathbf{s}_{t+1}) \\ & \text{subject to} && \\ & \mathbf{s}_{i,t+1} = \mathbf{s}_{i,t} + \mathbf{u}_{i,t} \Delta T, \quad \forall i = 1, \dots, M \\ & \mathbf{u}_{i,t} \in \mathcal{U}_i, \quad \mathbf{s}_{i,t+1} \in \Omega_{i,t}, \quad \forall i = 1, \dots, M. \end{aligned} \quad (5)$$

It can be seen that (5) is a highly nonconvex optimization problem, due to the objective function, that can lead to computational intractability. For example, in our case study presented in Section IV, the popular nonlinear programming solver Ipopt [14] failed to solve the adaptive sampling optimization problem.

### C. Grid-based Methods

In the literature, to overcome the computational difficulty of the adaptive sampling problem (5), it is usually solved in a discrete domain, where the region of interest is discretized into a grid and a solution is searched for by evaluating the objective function at a set of possible locations on the grid. Nevertheless, the grid-based optimization problem is combinatorial NP-hard [15], which can only be solved by a heuristic search algorithm, such as a greedy algorithm, resulting in a near-optimal solution. More importantly, the performance of the grid-based method is dependent on the size of the environment. Indeed, typically, a minimum resolution of the discretization grid must be met, hence the larger the environment is, the larger the grid size is in terms of the number of grid points, resulting in a longer search for the near-optimal solution. Consequently, in a large environment and/or a large-scale MRSN, grid-based methods may not be practical.

In this paper, we propose to use the Proximal ADMM (Px-ADMM) algorithm [13] to efficiently solve the optimization problem (5) in a continuous domain, where a stationary solution is guaranteed to be obtained. The PxADMM algorithm does not require discretization of the environment by a grid, thus is less influenced by the size of the environment than is a grid-based method. Details of the proposed approach will be presented in the next section.

## III. ADAPTIVE SAMPLING STRATEGY USING PROXIMAL ADMM

For non-convex consensus optimization problems, the PxADMM algorithm [13] overcomes a challenge of Classical ADMM [16] that exactly minimizing the augmented

Lagrangian function is expensive. Since the adaptive sampling optimization problem (5) is non-convex, we employ the PxADMM approach to solve it efficiently.

We first rewrite (5) in the following consensus form

$$\begin{aligned} & \underset{\{\mathbf{s}, \mathbf{z}, \mathbf{u}\}}{\text{minimize}} && f_G(\mathbf{z}) + I_C(\mathbf{s}, \mathbf{u}) \\ & \text{subject to} && \mathbf{s} = \mathbf{z}, \end{aligned} \quad (6)$$

where  $\mathbf{u} = \mathbf{u}_t$ ,  $\mathbf{s} = \mathbf{s}_{t+1}$ ,  $\mathbf{z}$  is a copy of  $\mathbf{s}$  and  $f_G(\mathbf{z}) = -\log \det \Sigma(\mathbf{z})$ . All the constraints in (5) are represented by  $(\mathbf{s}, \mathbf{u}) \in \mathcal{C}$  and  $I_C$  is the indicator function of  $\mathcal{C}$ . Hence, the augmented Lagrangian function of (6) is defined by

$$\mathcal{L}(\mathbf{z}, \mathbf{s}, \mathbf{u}, \mu) = f_G(\mathbf{z}) + I_C(\mathbf{s}, \mathbf{u}) + \mu^T(\mathbf{z} - \mathbf{s}) + \frac{\rho}{2} \|\mathbf{z} - \mathbf{s}\|_2^2, \quad (7)$$

where  $\mu$  is a vector of the associated dual variables and  $\rho$  is a regularization parameter.

To overcome the complexity of the log determinant of a covariance matrix (i.e., the  $f_G(\cdot)$  function), the PxADMM method [13] use the first-order approximation of  $f_G(\cdot)$ , which results in the following computation steps:

$$\mathbf{s}^{(k+1)}, \mathbf{u}^{(k+1)} = \underset{\mathbf{s}, \mathbf{u} \in \mathcal{C}}{\operatorname{argmin}} \left\| \mathbf{s} - (\mathbf{z}^k + (1/\rho)\mu^{(k)}) \right\|_2^2 \quad (8a)$$

$$\begin{aligned} \mathbf{z}^{(k+1)} = \underset{\mathbf{z}}{\operatorname{argmin}} & \nabla^T f_G(\mathbf{s}^{(k+1)})(\mathbf{z} - \mathbf{s}^{(k+1)}) + \\ & (\mu^{(k)})^T(\mathbf{z} - \mathbf{s}^{(k+1)}) + \frac{\rho + L}{2} \left\| \mathbf{z} - \mathbf{s}^{(k+1)} \right\|_2^2 \end{aligned} \quad (8b)$$

$$\mu^{(k+1)} = \mu^{(k)} + \rho(\mathbf{z}^{(k+1)} - \mathbf{s}^{(k+1)}), \quad (8c)$$

Here,  $L$  is a positive constant that satisfies the condition

$$\left\| \nabla^T f_G(\mathbf{x}) - \nabla^T f_G(\mathbf{x}') \right\| \leq L \|\mathbf{x} - \mathbf{x}'\| \quad (9)$$

for all  $\mathbf{x}, \mathbf{x}' \in \mathcal{Q}$ .

It is important to note that the  $z$ -minimization step (8b) in the PxADMM algorithm is a convex quadratic minimization that can be solved analytically as

$$\mathbf{z}^{(k+1)} = \mathbf{s}^{(k+1)} - \frac{1}{(\rho + L)} (\nabla f_G(\mathbf{s}^{(k+1)}) + \mu^{(k)}) \quad (10)$$

in which the derivative of  $f_G(\cdot)$  with respect to each element  $\nu_j$  of a vector  $\nu$  can be computed by

$$\frac{\partial f_G(\nu)}{\partial \nu_j} = -\operatorname{tr} \left( \Sigma^{-1}(\nu) \frac{\partial \Sigma(\nu)}{\partial \nu_j} \right),$$

where from (3) we have

$$\frac{\partial \Sigma(\nu)}{\partial \nu_j} = \frac{\partial \Sigma_{t+1,t+1}(\nu)}{\partial \nu_j} - 2\Sigma_{t+1,0:t} \Sigma_{0:t}^{-1} \left( \frac{\partial \Sigma_{t+1,0:t}(\nu)}{\partial \nu_j} \right)^T.$$

The PxADMM algorithm for solving the adaptive sampling optimization problem (5) thus can be detailed in Algorithm 1.

*Remark 1:* The PxADMM algorithm can allow the computation for solving the problem (5) to be distributed to the agents, which leads to reduced computation time in comparison with a centralized implementation. The benefits of the distributed computation is significant if the network size is large or the local constraints for each agent is more complex. However, in our case study in Section IV, since the number of sensing agents is small and the problem (8a) is simple, the centralized implementation is utilized.

We now state a result on the convergence of Algorithm 1. First, for the problem (6), the assumption A3 (on convexity and lower bound of  $I_{C_i}$ ) in [13] is satisfied, then we make the

---

**Algorithm 1** Proximal ADMM algorithm for solving (5)

---

**Require:**  $\mathbf{z}^{(0)}, \mu^{(0)}, \epsilon_{res}, k_{\max}, \rho, L$   
1: **for**  $k = 0, \dots, k_{\max}$  **do**  
2:   Compute  $\mathbf{s}^{(k+1)}$  and  $\mathbf{u}^{(k+1)}$  by (8a)  
3:   Compute  $\mathbf{z}^{(k+1)}$  by (10)  
4:   Update  $\mu^{(k+1)}$  by (8c)  
5:   **if**  $\|\mathbf{z}^{(k+1)} - \mathbf{s}^{(k+1)}\| < \epsilon_{res}$  **then**  
6:     Stop and return  $\mathbf{s}^{(k+1)}, \mathbf{u}^{(k+1)}$   
7: **return**  $\mathbf{s}^{(k_{\max})}, \mathbf{u}^{(k_{\max})}$

---

following assumption, which is equivalent to the assumption A1 in [13]:

*Assumption 1:* There exists a positive constant  $L$  so that the equation (9) holds.

*Lemma 1:* Algorithm 1 converges to a stationary solution of the problem (6) if the parameter  $\rho$  is chosen such that

$$\alpha = \frac{\rho - 7L}{2} - \left(\frac{4L}{\rho^2} + \frac{1}{\rho}\right)2L^2 > 0 \text{ and} \quad (11)$$
$$\beta = \frac{\rho}{2} - \left(\frac{4L}{\rho^2} + \frac{1}{\rho}\right)8L^2 > 0.$$

*Proof:* The lemma can be derived from Theorem 2.10 in [13]. ■

Note that we can always find a large enough value of  $\rho$  so that the above condition in Lemma 1 holds.

*Remark 2:* At each time step, because Algorithm 1 only converges to a stationary solution, it is restarted multiple times with different random initial conditions, and the final solution for the problem (6) corresponds to the lowest objective value obtained.

The proposed adaptive sampling strategy using PxADMM for a MRSN is summarized as follows. At each time step, each mobile sensor takes a measurement of the spatial phenomenon at its current location and sends both the observation and the location information to a central station. After receiving data from all the mobile sensors, the central station updates the training data set, retrain the GP model and executes Algorithm 1 to solve the adaptive sampling optimization problem (5). The control inputs obtained by Algorithm 1 are then sent back to the mobile sensors for their next navigation. The procedure is repeated until a predefined number of sampling steps is reached or the prediction uncertainty is lower than an expected threshold.

#### IV. EXPERIMENTAL RESULTS AND DISCUSSION

To demonstrate the effectiveness of the proposed approach, we conducted several numerical experiments using a real-world temperature data set obtained from the Intel Berkeley Research Lab [17]. The simulations were executed on a DELL computer with a 3.0 GHz Intel Core i5 CPU and 8 Gb RAM, where the algorithms were implemented in the Python programming language. In this case study, we used the constant mean and the squared exponential covariance function for all the Gaussian Process models.

We first created a ground truth of the spatial temperature phenomenon for the purposes of generating sensor measurements and verifying predictions. Given the 54 temperature

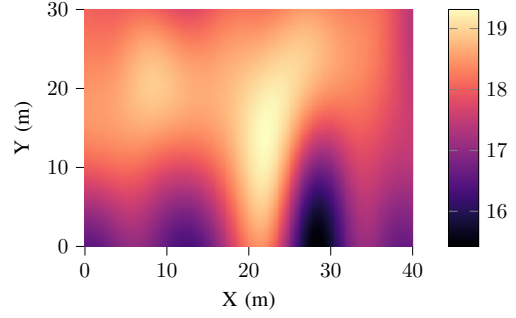


Fig. 2: A generated ground truth of the indoor temperature field.

observations collected by the 54 sensors deployed in the Intel Berkeley Research Lab, a ground-truth GP model was learned, which will be denoted  $\text{model}_{\text{GT}}$ . It was used to generate a ground truth of the indoor temperature field as shown in Figure 2. Whenever a mobile sensor takes a measurement of the temperature phenomenon at its current location, the measurement can be generated by  $\text{model}_{\text{GT}}$ . In the experiments, a network of 5 mobile robotic sensors was employed to monitor the indoor temperature phenomenon in an environment of size 40 m-by-30 m:  $\mathcal{Q} := \{(x, y) \mid 0 \text{ m} \leq x \leq 40 \text{ m}, 0 \text{ m} \leq y \leq 30 \text{ m}\}$ . The control input vector for each mobile sensor was bounded between  $\mathbf{u}_{\min} = [-1; -1]$  (m/s) and  $\mathbf{u}_{\max} = [1; 1]$  (m/s). The sampling time was  $\Delta T = 2 \text{ s}$ . The parameters of the PxADMM algorithm were selected as  $\rho = 0.3$ ,  $L = 0.03$ ,  $\epsilon_{res} = 10^{-5}$ , and  $k_{\max} = 100$ . As discussed in Section III, at each time step, the PxADMM algorithm was restarted five times to improve the local optimal solution. The initial solution guess for the algorithm at each run was chosen in uniformly random directions around the current sensor location. The maximum number of sampling steps was  $t_{\max} = 15$ , i.e., a total of 80 measurements would be made by the 5 robots.

To ensure a fair evaluation of our proposed approach, we conducted 1000 simulations, in each of which the mobile sensors were set to start at random positions. The results obtained by one particular experiment at three time steps 5, 10 and 15, respectively, are demonstrated in Figure 3 as an example, where the arbitrary starting locations of the robots are shown in the white circles. Figure 3 illustrates not only the sampling paths for the robotic sensors up to the time step considered but also the predicted variances (in log scale) in the entire environment, calculated at 90,601 positions on a grid of size  $301 \times 301$ , obtained at that time instant. It can be clearly seen that (1) at every time step the mobile sensors tended to move to the regions with the higher prediction uncertainty, and (2) the more the robots moved, the more the prediction uncertainty in the whole environment was reduced. Therefore, the proposed algorithm was successful at minimizing the prediction errors for adaptive sampling objective. The mean predictions of the temperature field in the entire environment for the example presented in Figure 3 are depicted in the heat maps in Figure 4 at time steps 5, 10 and 15, respectively. It can be seen that the more sensor measurements were gathered, the more accurately the MRSN

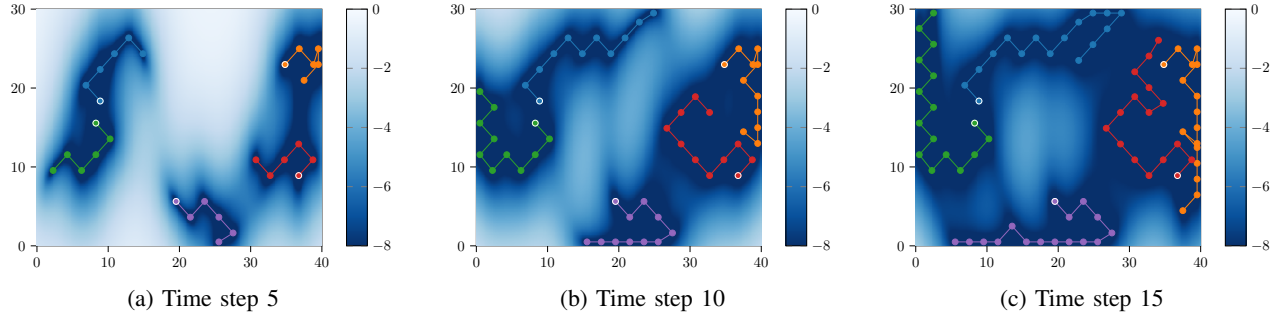


Fig. 3: An example of the predicted variances (in log scale) in the entire environment and the sampling trajectories of the mobile sensors obtained by the PxADMM based approach at some specific time instants. The starting locations of the mobile robots are shown in the white circles.

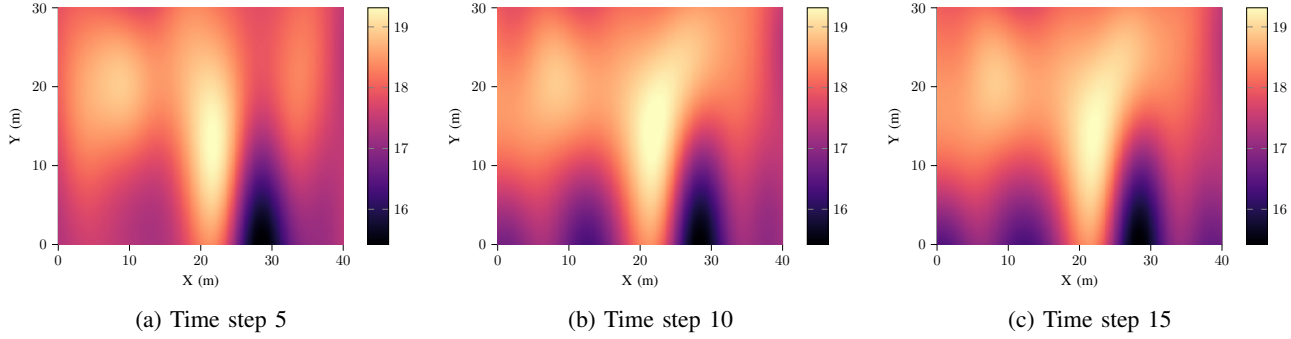


Fig. 4: An example of the predicted means in the entire environment obtained by the PxADMM based approach at some specific time instants.

predicted the spatial field. At time step 15, with 80 samples, the predicted temperature in the entire environment is highly comparable to the ground truth depicted in Figure 2.

#### A. Comparison with the grid-based methods

In this section, we investigate further how efficiently our algorithm leads to better predictions of the spatial field as compared with the grid-based greedy method at three different grid sizes of  $50 \times 50$ ,  $100 \times 100$  and  $150 \times 150$ . In this evaluation, three different validation metrics including the average log predicted variances (ALPVs), the root mean squared errors (RMSEs), and the maximum absolute errors (MAEs) were utilized. Both the RMSEs and MAEs were computed based on all the predicted temperatures and their corresponding ground truth as depicted in Figure 2. The evaluation results for both the proposed algorithm and the greedy algorithm are illustrated in Figure 5. It can be clearly seen that in the three validation metrics, after the fifth time step, the PxADMM based approach always outperforms the grid-based greedy method at all the grid sizes considered. Furthermore, after the 10<sup>th</sup> time step, the RMSE obtained by the proposed algorithm are close to 0 while those obtained by the grid-based greedy method tend to stay at about 0.15, as shown in Figure 5b.

We summarize the evaluation results in the three validation metrics at the final time step for the 1000 simulation experiments to reflect the effect of the starting locations of the mobile sensors on the predictions. The results are depicted in Figure 6 in the boxplot form. Overall, the

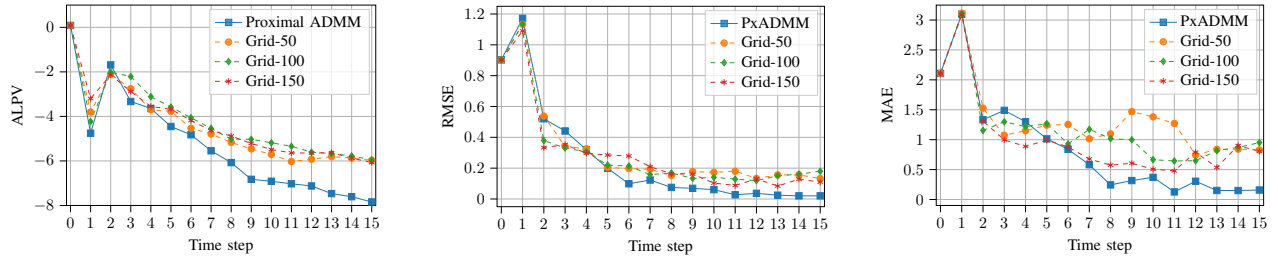
PxADMM based algorithm outperforms the grid-based greedy method. Statistically, in all the three validation metrics presented in Figure 6, the interquartile ranges obtained by the proposed approach are always smaller than the corresponding counterparts obtained by the grid-based greedy algorithm in all the three grid settings. And more interestingly, in about 75 percent of all the experiments, the ALPVs obtained by the PxADMM based method are better than all of those obtained by all the grid-based greedy tests as shown in Figure 6a. Likewise, the 75 percentile of the RMSEs boxplot in the PxADMM test is lower than the 25 percentiles of the three plots in the greedy experiments while a similar trend is closely followed by the MAEs validation metric, as depicted in Figures 6b and 6c, respectively.

To demonstrate whether the proposed algorithm can be employed in real-time systems, we summarize the average computation time of the different algorithms at each time step over the 1000 experiments in Figure 7. Requiring about 0.75s on average at each time step for computation, our proposed approach demonstrates its practicality in real-time systems while maintaining its efficacy in predicting the spatial field. Figure 7 also depicts the complexity issue of the greedy algorithm, where increasing the grid size causes significantly longer computation time.

## V. CONCLUSIONS

This paper discussed an efficient approach to address the adaptive sampling optimization problem in a MRSN for monitoring a spatial phenomenon by using the PxADMM



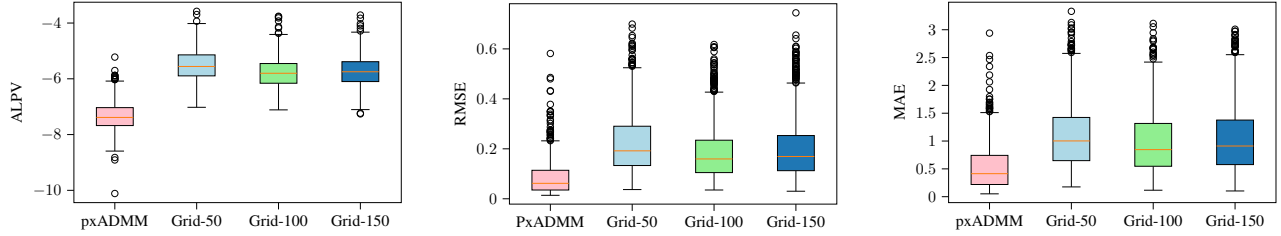


(a) The average log predicted variances

(b) The root mean squared errors

(c) The maximum absolute errors

Fig. 5: The validation metrics for an experiment.



(a) The average log predicted variances

(b) The root mean squared errors

(c) The maximum absolute errors

Fig. 6: Box plots of the validation metrics at the 15<sup>th</sup> time step for all 1000 experiments.

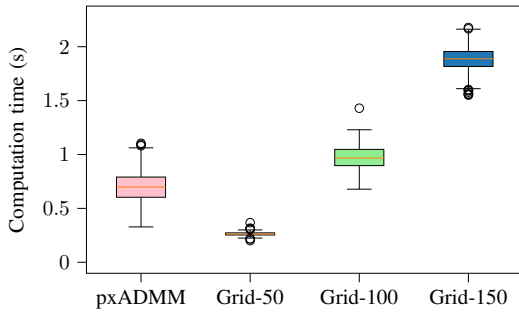


Fig. 7: The boxplot for average computation times in 1000 simulations.

algorithm. The adaptive sampling problem is formulated as minimizing the negative log determinant of a predicted covariance matrix obtained from a GP model used to represent the spatial field. The PxADMM based approach was proposed to handle the non-convexity and high complexity of the objective function. The results obtained in 1000 simulation experiments using a realistic temperature dataset demonstrated the advantages of our proposed approach as compared with the grid-based greedy method. Moreover, computing time of the algorithm in the implementation has shown its high practicality. Our future works will extend the proposed approach to the adaptive sampling problem for the MRSN with nonhomonic dynamics and multi-step predictions.

## REFERENCES

- [1] T. Arampatzis, J. Lygeros, and S. Manesis, "A survey of applications of wireless sensors and wireless sensor networks," in *Proceedings of the 2005 IEEE International Symposium on, Mediterrean Conference on Control and Automation Intelligent Control*, 2005. IEEE, 2005, pp. 719–724.
- [2] L. Nguyen, N. Ulapane, and J. V. Miro, "Adaptive sampling for spatial prediction in environmental monitoring using wireless sensor networks: A review," in *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2018, pp. 346–351.
- [3] D. A. Paley and A. Wolek, "Mobile Sensor Networks and Control: Adaptive Sampling of Spatiotemporal Processes," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, pp. 91–114, 2020, publisher: Annual Reviews.
- [4] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. MIT press Cambridge, MA, 2006, vol. 2, no. 3.
- [5] Y. Xu and J. Choi, "Adaptive sampling for learning Gaussian processes using mobile sensor networks," *Sensors*, vol. 11, no. 3, pp. 3051–3066, 2011, publisher: Molecular Diversity Preservation International.
- [6] Y. Xu, J. Choi, and S. Oh, "Mobile sensor network navigation using gaussian processes with truncated observations," *IEEE Transactions on Robotics*, vol. 27, no. 6, pp. 1118–1131, 2011, publisher: IEEE.
- [7] Y. Xu, J. Choi, S. Dass, and T. Maiti, "Sequential Bayesian prediction and adaptive sampling algorithms for mobile sensor networks," *IEEE Transactions on Automatic Control*, vol. 57, no. 8, pp. 2078–2084, 2011, publisher: IEEE.
- [8] R. Marchant and F. Ramos, "Bayesian optimisation for intelligent environmental monitoring," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 2242–2249.
- [9] S. Huang and J. Tan, "Adaptive sampling using mobile sensor networks," in *2012 IEEE International Conference on Robotics and Automation*. IEEE, 2012, pp. 657–662.
- [10] Y. Xu, J. Choi, S. Dass, and T. Maiti, "Efficient Bayesian spatial prediction with mobile sensor networks using Gaussian Markov random fields," *Automatica*, vol. 49, no. 12, pp. 3520–3530, 2013, publisher: Elsevier.
- [11] L. V. Nguyen, S. Kodagoda, R. Ranasinghe, and G. Dissanayake, "Information-driven adaptive sampling strategy for mobile robotic wireless sensor network," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 1, pp. 372–379, 2015, publisher: IEEE.
- [12] —, "Adaptive placement for mobile sensors in spatial prediction under locational errors," *IEEE Sensors Journal*, vol. 17, no. 3, pp. 794–802, 2016, publisher: IEEE.
- [13] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 337–364, 2016, publisher: SIAM.
- [14] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [15] C.-W. Ko, J. Lee, and M. Queyranne, "An exact algorithm for maximum entropy sampling," *Operations Research*, vol. 43, no. 4, pp. 684–691, 1995.
- [16] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.
- [17] "Intel Lab Data," <http://db.csail.mit.edu/labdata/labdata.html>, 2004.