Adaptive Sampling and Online Learning in Multi-Robot Sensor Coverage with Mixture of Gaussian Processes

Wenhao Luo, *Student Member, IEEE*, Katia Sycara, *Fellow, IEEE*

***Abstract*— We consider the problem of online environmental sampling and modeling for multi-robot sensor coverage, where a team of robots spread out over the workspace in order to optimize the overall sensing performance. In contrast to most existing works on multi-robot coverage control that assume prior knowledge of the distribution of environmental phenomenon, also known as density function, we relax this assumption and enable the robot team to efficiently learn the model of the unknown density function on-line using adaptive sampling and non-parametric inference such as Gaussian Process (GP). To capture significantly different components of the environmental phenomenon, we propose a new approach with mixture of locally learned Gaussian Processes for collective model learning and an information-theoretic criterion for simultaneous adaptive sampling in multi-robot coverage. Our approach demonstrates a better generalization of the environment modeling and thus the improved performance of coverage without assuming the density function is known a priori. We demonstrate the effectiveness of our algorithm via simulations of information gathering from indoor static sensors.**

I. Introduction

Multi-robot systems are capable of doing complex tasks and have been widely used in applications such as environmental sampling [1], coverage [2], and others, in which the robots employ local communication or control laws to achieve some collective goals. In this paper we are interested in the *Multi-Robot Sensor Coverage* problem [2], [3], [4], [5], where a group of robots are deployed in an environment from given starting configurations and then seek for the final optimal placements such that the overall sensing performance over the environmental phenomenon from those particular locations is maximized, which is also known as the *Locational Optimization* problem [6]. Although the *Multi-Robot Sensor Coverage* problem [2] and its variants [3], [5] have been extensively studied with the optimal solutions of Centroidal Voronoi tessellation (CVT) [8], the results are often based on the assumption that the density function is known beforehand, which may not be applicable in real-world situations where the robots operate in unknown environments. This motivates the need for taking samples along with the coverage control law so as to efficiently learn the distribution of environmental phenomenon via statistical models such as GP, while at the same time allowing each robot to estimate its current belief of the optimal sensing locations. The problem of navigating robots to collect the samples that best describe the environment model is referred to as the *informative sampling* [9], or *adaptive sampling* [10] if the sampling strategy updates accordingly based on the on-line observed sample value. On the other hand, due to the distributed nature of multi-robot systems and spatially neighboring correlations of locations nearby, each robot could have a different GP model that best describes the density function for its local data segment. In contrast, modeling with uni-model GP [11] may have poor prediction performance when the density function is significantly different at various locations. It is challenging to 1) efficiently learn the density function on-line while optimizing the coverage performance, and 2) mix various GPs from all robots for an input-dependent model. In this paper, we present a novel approach to tackle these challenges by combining the *Multi-Robot Sensor Coverage Control* with *adaptive sampling* and *online learning* using *mixture of GP models*.

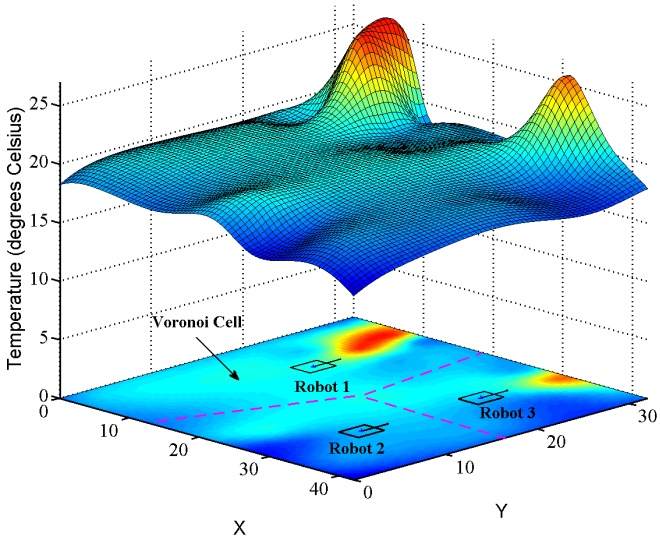


Fig. 1: Three robots are deployed and navigate to locations that maximize the sensing/coverage performance over environmental phenomenon such as temperature in the map. The upper layer represents the actual temperature distribution interpolated from 54 deployed stationary sensors in the Intel Berkeley Lab [7]. The lower layer represents the 2D multi-robot sensor coverage scenario with projected heat map of the temperature distribution.

We consider the example scenario shown in Fig. 1, where a group of robots move in an environment to find the best stationary placements with maximum sensing performance for monitoring the environmental phenomenon such as temperature over the map. It is desired for the robots to spread out while keep as close as possible to the centroid of each assigned sub-region determined by 1) the Euclidean distance between the robots and all points in the region (Voronoi

This work was supported in part by U.S. Department of Agriculture (DOA) award 2017-6700-726152.

*The authors are with the Robotics Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. Email: {wenhao, katia}@cs.cmu.edu.

cell) and 2) the temperature distribution [2], [5]. Since we do not assume the temperature distribution as a prior, the robots need to learn the temperature distribution on-line by continuously taking observations. For efficient sampling and modeling, the robots are desired to move towards the places that have best predicted sensing performance and are most informative to reduce further prediction uncertainty based on the learned non-parametric model. This differentiates our paper from other recent works [4], [12] that still assumes some form of the parameterized density function. We also notice from Fig. 1 that the temperature distribution consists of relatively smooth background and two peaks reflecting higher temperature observations, which can be locally captured by the corresponding robot close by but difficult to be accurately represented using uni-model GP as mentioned in [13].

This paper presents two main contributions of our novel approach. First, we couple the adaptive sampling with information-theoretic criterion into the multi-robot coverage control framework for efficient model learning and simultaneous locational optimization with reduced number of samples in an initially unknown environment. Second, we present an algorithm that allows for collaboratively learning the generalized model of density function using Mixture of GPs with hyper-parameters learned locally from each robot. The resulting GP mixture model provides improved prediction accuracy and reduced model uncertainty for complex distributions with significantly distinctive components, and hence increases the multi-robot coverage performance. We provide experimental results using a public dataset [7] from Intel Berkeley Research Lab demonstrating the superior performance of our approach.

II. Related Work

In the multi-robot sensor coverage problem [2], [3], the sensing performance to optimize is determined by the distance between each robot and its assigned point to sense assuming negative correlation as well as the density function of the points. Solutions of such a locational optimization problem are known as the centroid of the Voronoi tessellation [8] and the algorithm is often referred to as the move-to-centroids controller navigating the robots towards the centroids of their Voronoi cells. However, most of them assume the prior knowledge of either the environmental phenomenon (often modelled as density function) [2], [5], or basis functions of density function [12], which could be impractical in real-world application. To allow for online model learning, recent works [4] have proposed to use two-stage decoupled processes that embed an on-line sampling process to first obtain an estimate of the density function and then follow the move-to-centroid control law in performing the multi-robot coverage. As mentioned in [11], this approach could demand unnecessarily larger number of samples to take before reaching the optimal locations.

To improve sampling efficiency, a very recent work [11] proposed to use informative adaptive sampling within dynamic Voronoi partitions for the robots, so that each robot will only need to take the best samples close by. However, in [11] the uni-model GP for prediction was built on aggregated samples collected from all robots, which may not provide accurate prediction regarding distinctive distribution components as shown in Fig. 1. To develop an input-dependent model, the general approach of mixture of GPs was proposed in [14] and has been applied to single robot environment modeling [13] that is able to accurately represent complex distributions with the linear combination of different GP models. Inspired by these works, we propose to employ the adaptive sampling with information-criterion for efficient modeling of the unknown environmental phenomenon and extend the mixture of GPs to multi-robot model learning with locally optimized hyper-parameters to improve the environment modeling efficiency and accuracy in solving the mutli-robot sensor coverage problem.

III. Problem Statement

Consider a set of n robots moving in a bounded environment $Q \subset \mathbb{R}^2$ and assume the environment can be discretized into a set of point $q \in Q$, with the position of each robot $i \in \{1, 2, \ldots, n\}$ denoted by $x_i \in Q$. We assume the environment is free of obstacles and can be partitioned into n Voronoi cells, as done in most multi-robot sensor coverage algorithm [2], [3], [5].

$$V_i = \{q \in Q | \|q - x_i\| \leq \|q - x_j\|, \forall j \neq i\} \tag{1}$$

where $\|\cdot\|$ is the l^2-norm. Each Voronoi cell V_i corresponds to its generator robot x_i who will be responsible for sensing the points inside the cell $q \in V_i$.

Regarding the distribution of environmental phenomenon on each point of interest q, there exists an unknown density function $\phi(\cdot) : Q \to \mathbb{R}_+$ that maps the location information q to the scalar value of the phenomenon $\phi(q)$. Intuitively, in environmental monitoring task we want each robot to stay close to the area with higher phenomenon value $\phi(\cdot)$ since the sensing performance usually degrades as the distance between the robot and the point to sense increases. As (see (1)) each point is assigned to one robot, the cost function of static multi-robot coverage can be formally defined as follows [2], [3].

$$\mathcal{H}(x_i, \ldots, x_n) = \sum_{i=1}^{n} \int_{q \in V_i} \|q - x_i\|^2 \phi(q) dq \tag{2}$$

Hence the lower $\mathcal{H}(x_i, \ldots, x_n)$ the better. Then by taking the gradient of (2), we have the local optimal solutions for minimizing $\mathcal{H}(\cdot)$ for all $i \in \{1, \ldots, n\}$ as follows.

$$x_i^* = \arg\min \mathcal{H}(x_i, \ldots, x_n) = \frac{\int_{V_i} q\phi(q) dq}{\int_{V_i} \phi(q) dq} = C_{V_i} \tag{3}$$

where $C_{V_i} \in \mathbb{R}^2$ is also referred to as the centroid of each Voronoi cell V_i. Although this critical point of $\mathcal{H}$ is a local minimum, due to the intractable solution (NP-hard) to the global optimum $\mathcal{H}$ the local optimal solution x_i^* is often considered optimal (see [3], [5]). The decentralized gradient-based move-to-centroid controller [2] has been proven to navigate the robots to the local optimal locations.

$$\dot{x_i} = k_p(C_{V_i} - x_i) \tag{4}$$

where k_p is a user-defined control gain. Note that the realization of $\phi(q)$ will not be available to the robots unless $q = x_i$ and without loss of generality we ignore the intermediate visited points between consecutive waypoints by the robots. To that end, the objective is to drive the robot towards the locations with high predicted value of the phenomenon and informativeness so as to efficiently learn the distribution $\phi(\cdot)$ while simultaneously optimizing $\mathcal{H}(\cdot)$ with (4). In other words, we will use the optimal controller with the same form as in (4), but with a different specification of C_{V_i}.

IV. Gaussian Process regression for single robot environment modeling

In this section, we introduce the modeling of density function by a single robot with its locally sampled training data set.

A. Gaussian Process Regression

A common approach for modeling spatial phenomena is GP regression. Such a natural non-parametric generalization of linear regression allows for modeling the hidden mapping from training data to the target phenomenon with consideration of uncertainty [15]. Assume the target phenomenon, such as temperature in our case, satisfies a multivariate joint Gaussian distribution [16], [17]. The learned GP model from training data outputs the Gaussian probability distribution of the phenomenon $\phi(q)$ specified by mean function $\mu(q) = \mathbb{E}[\phi(q)]$ and covariance function $k(q, q') = \mathbb{E}[(\phi(q) - \mu(q))^T(\phi(q') - \mu(q'))]$ for any query data.

Formally, let $\tilde{V}_i = [q_1^i, \ldots, q_{N_i}^i]^T$ be the set of N_i collected samples associated with observed noisy values of temperature $\mathbf{y_i} = [y_1^i, \ldots, y_{N_i}^i]^T$ by robot i. Each observation is noisy $y = \phi(q) + \epsilon$ with $\epsilon \sim N(0, \sigma_n^2)$ assuming the mean function to be zero without loss of generality. To that end, given a testing location $q_{test} \in Q$, we have the conditional posterior mean $\mu_{q_{test}|\tilde{V}_i,\mathbf{y_i}}$ and variance $\sigma^2_{q_{test}|\tilde{V}_i,\mathbf{y_i}}$ as follows from the learned GP model describing the Gaussian distribution of $\phi(q_{test}) \sim \mathcal{N}(\mu_{q_{test}|\tilde{V}_i,\mathbf{y_i}}, \sigma^2_{q_{test}|\tilde{V}_i,\mathbf{y_i}})$.

$$\begin{aligned}\mu_{q_{test}|\tilde{V}_i,\mathbf{y_i}} &= \mathbf{k}(q_{test})^T(\mathbf{K}_{\tilde{V}_i} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y_i} \\ \sigma^2_{q_{test}|\tilde{V}_i,\mathbf{y_i}} &= k(q_{test}, q_{test}) - \mathbf{k}(q_{test})^T(\mathbf{K}_{\tilde{V}_i} + \sigma_n^2\mathbf{I})^{-1} \\ &\quad \cdot \mathbf{k}(q_{test})\end{aligned} \tag{5}$$

where $\mathbf{k}(q_{test}) = [k(q_1^i, q_{test}), \ldots, k(q_{N_i}^i, q_{test})]^T$ with the covariance (kernel) function $k(q, q')$ that captures the correlation between q and q'. $\mathbf{K}_{\tilde{V}_i}$ is the positive definite symmetric kernel matrix $[k(q, q')]_{q,q' \in \tilde{V}_i \cup q_{test}}$. In particular, we use the following squared-exponential kernel function to specify the inter-sample correlation.

$$k(q, q') = \sigma_f^2 e^{-\frac{(q-q')^T(q-q')}{2l^2}} \tag{6}$$

where the hyper-parameters l and σ_f are length-scale and scale factor, respectively. Hence, each robot i maintains its own GP model learned from local samples $\{\tilde{V}_i, \mathbf{y_i}\}$ and the hyper-parameters of $(\sigma_n^2, \sigma_f^2, l)$ are optimized from the local training data $\{\tilde{V}_i, \mathbf{y_i}\}$, which will be introduced in Section IV-B.

B. Estimation of Hyper-Parameters

The GP model of each robot i is determined by its local training data set $\{\tilde{V}_i, \mathbf{y_i}\}$ and local hyper-parameters denoted by $\theta_i = \{\sigma_n^2, \sigma_f^2, l\}$. In particular, the hyper-parameters are desired to be the optimizer such that the kernel function can accurately describe the underlying phenomena. In order to improve computation efficiency, we assume the hyper-parameters for each robot are optimized using the local training data of the robot itself, regardless of the GP mixture process which we discuss in Section V-A. One common approach for learning the hyper-parameters in a Bayesian framework is to maximize the log of the marginal likelihood as follows.

$$\begin{aligned}\theta_i^* &= \arg\max_{\theta_i} \log p(\mathbf{y_i}|\tilde{V}_i, \theta_i) \\ &= -\frac{1}{2}\mathbf{y_i}^T\tilde{\mathbf{K}}_{\tilde{V}_i}^{-1}\mathbf{y_i} - \frac{1}{2}\log|\tilde{\mathbf{K}}_{\tilde{V}_i}| - \frac{N_i}{2}\log 2\pi\end{aligned} \tag{7}$$

where $\tilde{\mathbf{K}}_{\tilde{V}_i} = \mathbf{K}_{\tilde{V}_i} + \sigma_n^2\mathbf{I}$. The maximizer of (7) can be computed by taking the partial derivatives of the marginal likelihood $p(\mathbf{y_i}|\tilde{V}_i, \theta_i)$ w.r.t. the hyper-parameters θ_i as described in [18].

V. Mixture of Gaussian Processes Models in Multi-Robot Coverage

Given the local GP model learned by each robot, in this section, we introduce the centralized learning step of mixture of GPs for every robot, assuming knowledge of all the robots' data available through inter-robot communication, and then compute the corresponding decentralized control and sampling strategy built on the GP mixture model.

A. Mixture of Gaussian Process Models and Adaptive Sampling Strategy

The mixture of GP models proposed in [14] is a linear combination of multiple GP models. From the Section IV, we have a set of locally learned GP models $\{\mathcal{GP}_1, \ldots, \mathcal{GP}_n\}$ from all n robots as aforementioned and denote $P(z(q) = i)$ as the probability of any random point $q \in Q$ being best described by the ith GP model from robot i. Then we have the GP mixture model defined by the conditional posterior mean $\mu^*_{q|\tilde{V},\mathbf{Y}}$ and variance $\sigma^{*2}_{q|\tilde{V},\mathbf{Y}}$ for any location $q \in Q$ as follows.

$$\begin{aligned}\mu^*_{q|\tilde{V},\mathbf{Y}} &= \sum_{i=1}^{n} P(z(q) = i) \cdot \mu_{q|\tilde{V}_i,\mathbf{y}_i} \\ \sigma^{*2}_{q|\tilde{V},\mathbf{Y}} &= \sum_{i=1}^{n} P(z(q) = i) \cdot (\sigma^2_{q|\tilde{V}_i,\mathbf{y_i}} + (\mu_{q|\tilde{V}_i,\mathbf{y}_i} - \mu^*_{q|\tilde{V},\mathbf{Y}})^2)\end{aligned} \tag{8}$$

where $\{\tilde{V}, \mathbf{Y}\}$ represents the set of collected samples by all the robots with $\tilde{V} = \{\tilde{V}_1, \ldots, \tilde{V}_n\}$ and $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$. To that end, for any point q its actual temperature $\phi(q)$ is assumed to be sampled from the Gaussian distribution $\mathcal{N}(\mu^*_{q|\tilde{V},\mathbf{Y}}, \sigma^{*2}_{q|\tilde{V},\mathbf{Y}})$. And the common approach for efficient sampling and modeling is to navigate the robots to the point $q^* = \arg\max \mu^*_{q^*|\tilde{V},\mathbf{Y}}$ or $q^* = \arg\max \sigma^{*2}_{q|\tilde{V},\mathbf{Y}}$ to maximize the sampled value of phenomenon or minimize the prediction uncertainty.

In our problem, we want to simultaneously sample the area with high value of phenomenon to get closer towards the Voronoi centroid C_{V_i} while reducing the uncertainty for the learned model of the density function $\phi(\cdot)$. Here we use the Gaussian Process Upper Confidence Bound (GP-UCB) [19], a sequential stochastic optimization strategy that trades off between exploration (reduce prediction uncertainty) and exploitation (maximize sampled value). Each location q is evaluated with the information-theoretic criterion defined as follows.

$$h(q) = \mu^*_{q|\tilde{V},\mathbf{Y}} + \beta {\sigma^*}^2_{q|\tilde{V},\mathbf{Y}} \tag{9}$$

where β is a parameter relates to the current sampling iteration number and regret bound [19]. When β is specified by a much higher value, then our solution becomes similar to the informative sampling [4] in which we want to reduce the model uncertainty before switching to the static coverage optimization. The GP-UCB strategy works by sequentially sampling point q that maximizes (9) and immediately update the GP model accordingly, such that we will be able to reach a balance by such an adaptive sampling strategy between reducing future GP model uncertainty and maximizing sampled value. However, our primary goal is to minimize the sensing cost function $\mathcal{H}(\cdot)$ in (2) by approaching unknown centroid of Voronoi cell C_{V_i} for each robot i. Thus, we modify the optimal solution in (4) by replacing unknown density function realization with the GP-UCB evaluation (9), which yields our adaptive sampling strategy for each robot i as follows.

$$q^*_i = \frac{\int_{V_i} q h(q) dq}{\int_{V_i} h(q) dq} = \tilde{C}_{V_i} \tag{10}$$

And the local coverage control law for each robot i becomes

$$\dot{x_i} = k_p(\tilde{C}_{V_i} - x_i) \tag{11}$$

In this case, the robots are able to simultaneously consider density function learning and sensing performance optimization. To solve for the feedback control law (11), it boils down to optimize the mixture of GP model by 1) finding the appropriate weight distribution $P(z(q) = \cdot)$, and 2) modifying local GP model with training data from other robots for generalizing the overall regression model. To simplify our discussion, we assume the robots are always connected as in [20] and are able to share their sampled data by communicating with its direct Voronoi neighbors [4].

B. GP Mixture Model Learning with Expectation-Maximization (EM) for Prediction

The EM algorithm [13], [14] has been widely used for estimating hidden and observable variables, such as the weight distribution of Gaussian Mixture Models for unsupervised learning. It consists of two stages such as the estimation (E) stage and the maximization (M) stage and it keeps looping until convergence under some threshold [14]. In our problem, we initialize the probability of weight distribution for any given query data point q_j by setting

$$P(z(q_j) = i) \approx \begin{cases} 1 & \text{if} \quad q_j \in \tilde{V}_i \\ 0 & \text{Otherwise} \end{cases} \quad \forall\, i = 1, \ldots, n \tag{12}$$

Then in the E-stage, the algorithm updates the probability $P(z(q_j) = i)$ by computing the marginal likelihood of each data q_j for all GP models. To simplify the notation we use $\mathcal{N}_i(q_j)$ to define the probability of observation of q_j regarding the local GP model $\mathcal{GP}_i$. Then we have the $P(z(q_j) = i)$ update rule over the previous one as follows.

$$P(z(q_j) = i) := \frac{P(z(q_j) = i) \cdot \mathcal{N}_i(q_j)}{\sum_{k=1}^{n} P(z(q_j) = k) \cdot \mathcal{N}_k(q_j)} \tag{13}$$

Then in the M-stage, the local GP models will be modified by embedding the updated probability of each query point q_j to the GP model updates steps (5). Here we present the main result for updating model $\mathcal{GP}_i$ from [13], [14] as follows.

$$\begin{aligned} \mu_{q_{test}|\tilde{V}_i,\mathbf{y_i}} &= \mathbf{k}(q_{test})^T (\mathbf{K}_{\tilde{V}_i} + \Psi^i \mathbf{I})^{-1} \mathbf{y_i} \\ \sigma^2_{q_{test}|\tilde{V}_i,\mathbf{y_i}} &= k(q_{test}, q_{test}) - \mathbf{k}(q_{test})^T (\mathbf{K}_{\tilde{V}_i} + \Psi^i \mathbf{I})^{-1} \\ &\quad \cdot \mathbf{k}(q_{test}) \end{aligned} \tag{14}$$

where

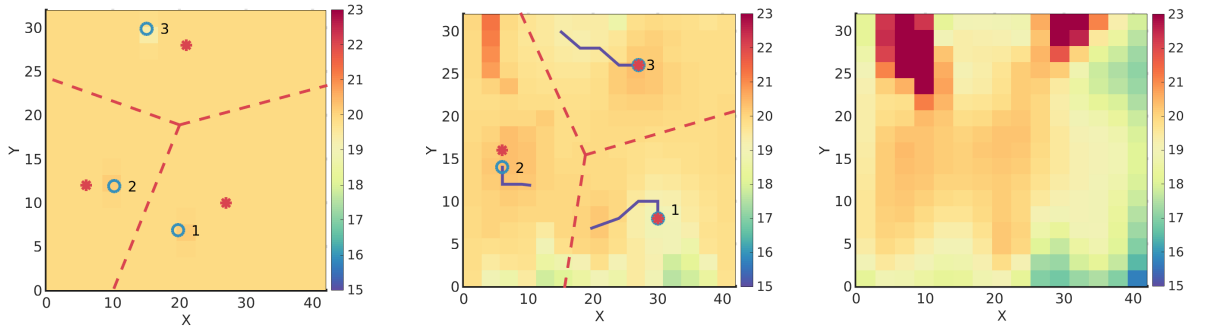
$$\Psi^i_{jj} = \frac{\sigma^2_n}{P(z(q_j) = i)} \tag{15}$$

It is noted that by modifying the value of diagonal hyper-parameter Ψ^i_{jj} from local value of σ^2_n the effects of each training data to the local GP models are adjusted so as to account for the observations for the points outside the local traning data set. Once the EM algorithm converges, we will have the new training data set consisting of $\{q_j\}$ and the corresponding weight distribution $P(z(q_j) = i)$ for each updated GP model i. With such training data set and the updated GP model, for any new query data q^*_j, we can predict its corresponding weight distribution $P(z(q^*_j) = i)$ as well as the expected value from local GP models (14), and then feed into the GP mixture model (8) to further yield the updated control law (10)-(11) to govern the motion of the robots.

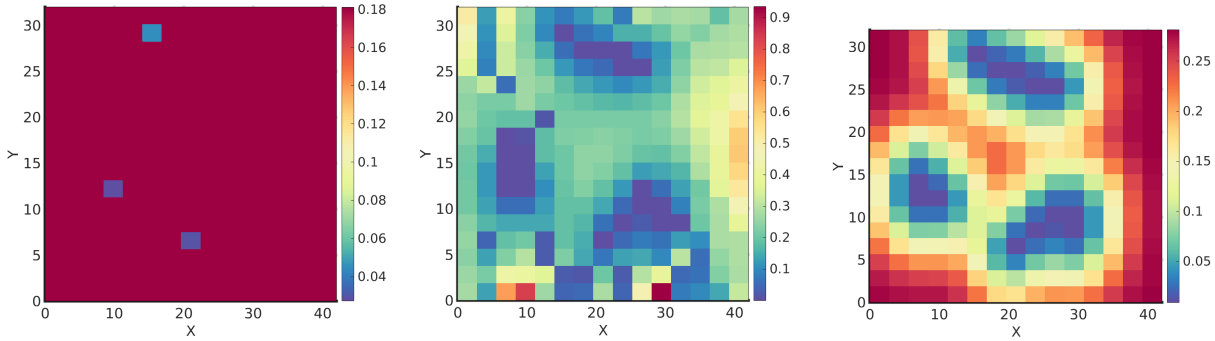
VI. RESULTS

In this section, we present several simulation results on the benchmark real-world dataset from Intel Berkeley Lab [7] with MATLAB toolboxes: the GPML [18]. The dataset contains sensory data collected from 54 sensors in an office area between Feb 28th and Apr 5th, 2004. The data includes the time-stamped readings such as sensor 2D locations, temperature, humidity, light, and voltage. In our particular tasks, we use the 2D location information (meters) of each sensor with the temperature readings (degrees Celsius) as the ground truth of the environmental phenomenon over map and compare our algorithm performance to other approaches.

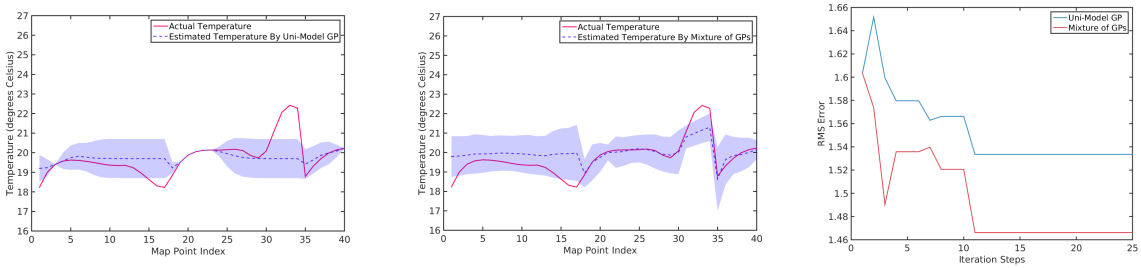
First we consider an example where we have 3 robots deployed from random starting points $(19.78, 6.84)$, $(10.22, 11.89)$ and $(15.09, 29.90)$ to find the optimal final configurations for stationary sensing as shown earlier in Fig. 1, where the temperature distribution has two peaks around the top corners. Once deployed the robots are governed by our adaptive coverage controller (11) with mixture of GPs ($k_p = 0.5, \beta = 10$) to simultaneously learn the environmental model and try to approach the actual centroid of each Voronoi cell based on its own model inference. As shown in Fig. 2(a) and (d) the robots first assume a



(a) Initial configurations (b) Converged final configurations (c) Actual temperature distribution

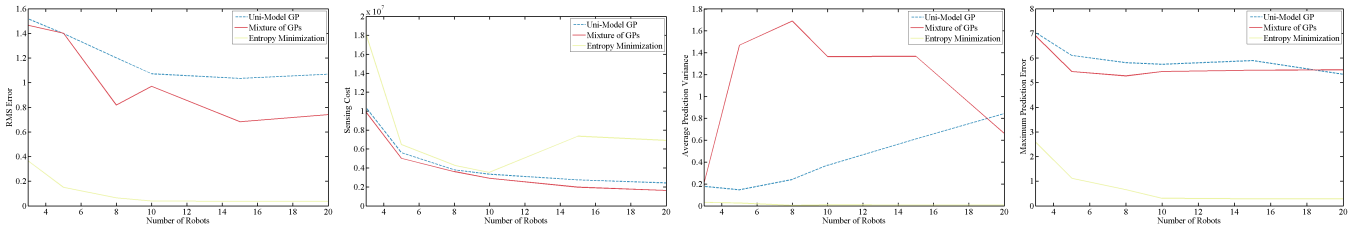


(d) Initial prediction variance by GP mixture model (e) Final prediction variance by GP mixture model (f) Final prediction variance by uni-model GP



(g) Prediction from uni-model GP (h) Prediction from GP mixture model (i) RMS error comparison

Fig. 2: An example of the multi-robot sensor coverage and environment modeling results by using GP mixture model with comparison to uni-model GP. (a) Initial and (b) final configurations of the robots (marked by blue circles) with history footprints by controller using GP mixture model. The background heatmap indicates the predicted temperature distribution based on the sampled data. Edge of Voronoi cells and the optimal locations (centroids of Voronoi cells from actual temperature distribution) are represented by red dashed lines and red stars, respectively. (c) Actual temperature distribution over map. (d) Initial and (e) final predicted variance distribution by GP mixture model. (f) Converged prediction variance from uni-model GP. (g)-(h) Temperature prediction comparison with standard deviation from (g) uni-model GP and (h) GP mixture model. (i) RMS error comparison.



(a) (b) (c) (d)

Fig. 3: Comparisons of RMS error, sensing cost, average prediction variance and maximum prediction error occurred v.s. different number of robots.

uniform distribution of temperature over the map based on the data collected from the initial configurations with little uncertainties. Note that the temperature from each discrete point can only be acquired when the robot chooses it as the next point to visit (sample) except for the initial configuration. The converged results are shown in Fig. 2(b) in comparison with the actual temperature distribution in Fig. 2(c) and the prediction variance is also given in Fig. 2(e). It is noted that the converged configuration is very close to the optimal one from the actual temperature distribution due to our adaptive sampling algorithm that trades off between uncertainty reduction and centroid approaching. Moreover, although none of the robots actually visited the top corner areas with much higher temperature, the mixture of GPs is able to identify the prediction differences among the robots over similar areas and adjust the mixture accordingly to best fit the various local features, by predicting a higher temperature with higher uncertainties over those areas in Fig. 2(e). In contrast, using uni-model GP could ignore the local features and hence the prediction variance is almost the same over any unvisited areas as shown in Fig. 2(f).

To better understand the performance, we provide the comparisons on local temperature prediction as shown in Fig. 2(g)-(h). Note that although the uni-model GP can have accurate prediction over places close to the sampled points, it fails to recognize the peak temperature in areas surrounding the robots, which can be identified by the mixture of GPs. We also compare the root-mean-square error (RMS) in Fig. 2(i), where the mixture of GPs shows a better performance with lower prediction error. To further compare our algorithms with other sampling approaches, we run the simulations with different number of robots under different algorithms, including the aforementioned uni-model GP and the Entropy minimization algorithm [11] that seeks to find the point in Voronoi cell which best reduces the model prediction uncertainty. The results are shown in Fig. 3 and our Mixture of GPs algorithm always outperforms the uni-model GP algorithm. Although the algorithm in [11] has a better sampling performance w.r.t. model uncertainty, the resultant sensing cost is much higher in Fig. 3(b) and hence it cannot be directly applied to our problem. The reason lies in that it prefers areas with higher uncertainty to the ones with higher predicted value, and due to the greedy based coverage control law it could converge to the locations that are further away from the areas with peak temperature.

VII. Conclusion

In this paper, we present an adaptive sampling algorithm for learning the density function in multi-robot sensor coverage problem using Mixture of Gaussian Processes models. By using the information-theoretic sampling criterion we are able to modify the traditional coverage control law to consider the uncertainty as well as the potential environmental phenomenon inferred from the environmental model learned on-line. Besides, considering significantly different components that may exist in the real-world environmental phenomenon, we propose to employ the mixture of GP models to capture local features for the global distribution by optimizing the linear combination of GP models locally learned by the robots. Simulation results have shown the effectiveness of our algorithm compared to other approaches. Despite the decentralized coverage control law, the learning of mixture of GPs in this work is still centralized and requires data from all the robots. In the future, we will investigate fully distributed GP mixture model learning algorithm.

References

[1] W. Luo, S. S. Khatib, S. Nagavalli, N. Chakraborty, and K. Sycara, "Distributed knowledge leader selection for multi-robot environmental sampling under bandwidth constraints," in *IROS*. IEEE, 2016, pp. 5751–5757.

[2] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, "Coverage control for mobile sensing networks," *IEEE Transactions on robotics and Automation*, vol. 20, no. 2, pp. 243–255, 2004.

[3] J. Cortés, "Coverage optimization and spatial load balancing by robotic sensor networks," *IEEE Transactions on Automatic Control*, vol. 55, no. 3, pp. 749–754, 2010.

[4] M. Schwager, M. P. Vitus, S. Powers, D. Rus, and C. J. Tomlin, "Robust adaptive coverage control for robotic sensor networks," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 3, pp. 462–476, 2017.

[5] A. Pierson, L. C. Figueiredo, L. C. Pimenta, and M. Schwager, "Adapting to sensing and actuation variations in multi-robot coverage," *The International Journal of Robotics Research*, vol. 36, no. 3, pp. 337–354, 2017.

[6] A. Okabe and A. Suzuki, "Locational optimization problems solved through voronoi diagrams," *European Journal of Operational Research*, vol. 98, no. 3, pp. 445–456, 1997.

[7] (2004) Intel lab data. [Online]. Available: http://db.csail.mit.edu/labdata/labdata.html

[8] Q. Du, M. Emelianenko, and L. Ju, "Convergence of the lloyd algorithm for computing centroidal voronoi tessellations," *SIAM journal on numerical analysis*, vol. 44, no. 1, pp. 102–119, 2006.

[9] K.-C. Ma, L. Liu, and G. S. Sukhatme, "Informative planning and online learning with sparse gaussian processes," in *ICRA*. IEEE, 2017, pp. 4292–4298.

[10] A. Singh, A. Krause, and W. J. Kaiser, "Nonmyopic adaptive informative path planning for multiple robots." in *IJCAI*, vol. 3, 2009, p. 2.

[11] S. Kemna, J. G. Rogers, C. Nieto-Granda, S. Young, and G. S. Sukhatme, "Multi-robot coordination through dynamic voronoi partitioning for informative adaptive sampling in communication-constrained environments," in *ICRA*. IEEE, 2017, pp. 2124–2130.

[12] M. Schwager, D. Rus, and J.-J. Slotine, "Decentralized, adaptive coverage control for networked robots," *The International Journal of Robotics Research*, vol. 28, no. 3, pp. 357–375, 2009.

[13] C. Stachniss, C. Plagemann, and A. J. Lilienthal, "Learning gas distribution models using sparse gaussian process mixtures," *Autonomous Robots*, vol. 26, no. 2, pp. 187–202, 2009.

[14] V. Tresp, "Mixtures of gaussian processes," in *Advances in neural information processing systems*, 2001, pp. 654–660.

[15] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, 2006, vol. 1.

[16] A. Krause, A. Singh, and C. Guestrin, "Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies," *Journal of Machine Learning Research (JMLR)*, vol. 9, pp. 235–284, February 2008.

[17] W. Luo, C. Nam, and K. Sycara, "Online decision making for stream-based robotic sampling via submodular optimization," in *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, Nov 2017, pp. 118–123.

[18] C. E. Rasmussen and H. Nickisch, "Gaussian processes for machine learning (gpml) toolbox," *Journal of Machine Learning Research*, vol. 11, no. Nov, pp. 3011–3015, 2010.

[19] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for gaussian process optimization in the bandit setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[20] M. Corah and N. Michael, "Efficient online multi-robot exploration via distributed sequential greedy assignment," in *Proceedings of Robotics: Science and Systems*, Cambridge, Massachusetts, July 2017.