# Distributed Environmental Modeling and Adaptive Sampling for Multi-Robot Sensor Coverage

Wenhao Luo
Robotics Institute, Carnegie Mellon University
Pittsburgh, USA
wenhao@cs.cmu.edu

Changjoo Nam
Center for Robotics Research, Korea Institute of Science and Technology
Seoul, Republic of Korea
cjnam@kist.re.kr

George Kantor
Robotics Institute, Carnegie Mellon University
Pittsburgh, USA
kantor@ri.cmu.edu

Katia Sycara
Robotics Institute, Carnegie Mellon University
Pittsburgh, USA
katia@cs.cmu.edu

## ABSTRACT

We consider the problem of online distributed environmental modeling and adaptive sampling for multi-robot sensor coverage, where a team of robots spread out over the workspace in order to optimize the sensing performance over environmental phenomena, whose distribution is often referred to as a density function. Unlike most existing works that either assume certain knowledge of the density function beforehand or centrally learn the density function assuming global knowledge of collected data from all the robots, we propose a *fully distributed* adaptive sampling approach to allow robots to efficiently learn the *unknown* density function online. In particular, we developed adaptive coverage controllers based on the learned density functions for minimizing the sensing cost. To capture significantly different components of the environmental phenomenon with *only locally collected data* for each robot when global knowledge is not available, we propose a *distributed mixture of Gaussian Processes* algorithm that enables robots to collaboratively learn the global density function by exchanging only model-related parameters. We empirically demonstrate the effectiveness of our algorithm via evaluation on real-world data gathered from agricultural field robot and indoor static sensors.

## KEYWORDS

Multi-Robot Systems; Mixture of Gaussian Processes; Adaptive Sampling; Distributed Robot Systems; Environmental Modeling

**ACM Reference Format:**
Wenhao Luo, Changjoo Nam, George Kantor, and Katia Sycara. 2019. Distributed Environmental Modeling and Adaptive Sampling for Multi-Robot Sensor Coverage. In *Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), Montreal, Canada, May 13–17, 2019,* IFAAMAS, 9 pages.

## 1 INTRODUCTION

Multi-robot systems are capable of doing complex tasks and have been widely used in applications such as environmental sampling [11], coverage [4], and others, in which the robots employ local communication or control laws to achieve some collective goals. In the *Multi-Robot Sensor Coverage* problem [3, 4, 20, 24], 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 [18]. In real-world scenario, we are motivated by the agricultural monitoring task as shown in Fig. 1 where multiple robots need to persistently monitor the crops and the preferred deployment locations are those close to high yield plots. However, the robots have no prior knowledge of the quality of each plot, namely the density function, and hence have to establish an estimate model first by taking samples to learn the density function, such as counting stalks or temperature over the field [17], which could be time-consuming. To that end, it is desired for the robots to take as few samples for efficient environmental modeling while approaching to its estimated optimal sensing locations.

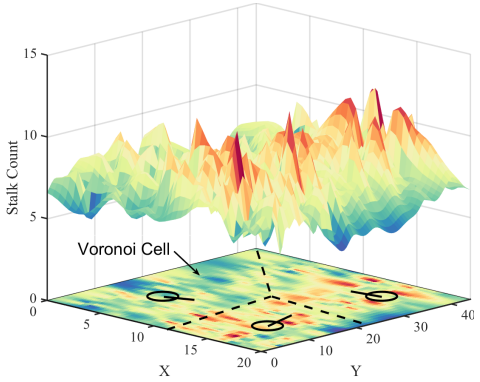



**Figure 1: An example scenario of three robots navigating in field to find best locations that maximize the sensing/coverage performance over environmental phenomenon such as stalk count of the agricultural crops. The upper layer represents the actual distribution of stalk count from the dataset collected by CMU agricultural robot platform, the Robotanist [17]. The lower layer represents the 2D field with projected heat map of the distribution (density function of stalk count).**

*Proc. of the 18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2019), N. Agmon, M. E. Taylor, E. Elkind, M. Veloso (eds.), May 13–17, 2019, Montreal, Canada.* 

The *Multi-Robot Sensor Coverage* problem [4] and its variants [3, 20] have been extensively studied with the optimal solutions of Centroidal Voronoi Tessellation (CVT) [5] and its gradient-based coverage controller, but 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. To interleave with the two tasks of density function learning and coverage control, recent works [23, 24] propose to couple the parameterized unknown density function with a class of known Gaussian-like basis function and robots can model the environment and perform sensor coverage by learning the weights of the basis functions from collected samples in a distributed manner. However, such parametric inference methods still 1) requires prior knowledge of the basis functions for every robot and cannot represent complicated distributions, for example the one in Fig. 1, and 2) assumes perfect observations from collected samples without noise. In our most recent work [13], we proposed to model the density function with a mixture of Gaussian Processes (GPs) from noisy observations that allows more general centralized distribution learning with non-parametric inference. But since the GPs is non-parametric, it requires knowledge of samples collected by all the robots, which may not be scalable and impractical in bandwidth-constrained environment where the global knowledge of the samples from all the robots is not available. 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 while allowing for distributed computation with only local data (samples) when global knowledge of all the collected data is not accessible.

In this paper, we propose a distributed adaptive coverage control strategy with online density function modeling through mixture of GPs and relaxes the assumption of global knowledge for learning the density function as assumed in our previous work [13]. In particular, we propose a distributed consensus learning algorithm for the mixture of GPs with local data only. In this work, at each round of sampling, each robot first employs the Gaussian Mixture Model (GMM) to classify its collected samples and extract the local GMM parameters. With the distributed consensus learning algorithm, the consensus on global GMM parameters is reached that best classifies the local data for each robot to locally fit a mixture of GPs and predict the density function on its own. In this way, only local GMM parameters are exchanged among robots whose size is independent from the number of collected samples, and hence we avoid the transmission of all local data from every robot. Then the locally learned density function is used to construct the decentralized adaptive coverage controller with the information-theoretic criterion for adaptive sampling that drives each robot towards its updated estimated optimal location, which could reduce the actual sensing cost (increase sensing performance) and also the model uncertainty for the learned density function with new samples collected on the location.

The main *contribution* of our novel approach is three-fold. First, we couple the adaptive sampling with information-theoretic criterion into the multi-robot coverage control framework for efficient distributed model learning and simultaneous locational optimization with a reduced number of samples in an initially unknown environment. Second, we present a fully distributed algorithm that allows for collaboratively learning the generalized non-parametric mixture of GPs model of density function with local data only. This could also be very useful for other decentralized data-driven multi-robot adaptive sampling and informative sampling tasks as most literature still assume the transmission of global data from all the robots for learning and evaluating the environment phenomenon, which is not scalable and may not be practical. Third, extensive empirical results are provided using real-world dataset including the agricultural field data collected by agricultural robot [17] and a public dataset [1] from Intel Berkeley Research Lab to demonstrate the superior performance of our approach.

## 2 RELATED WORK

In the multi-robot sensor coverage problem [3, 4], 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 [5] 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 distribution (often modelled as density function) [4, 20], or basis functions consisting of density function [23], which could be impractical in real-world application. To allow for online density function learning and adaptive coverage control, recent works [24] 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 [8], this approach could demand unnecessarily larger number of samples to take before reaching the optimal locations.

To improve sampling efficiency, GP-based adaptive sampling methods [7, 8, 19] with Bayesian optimization framework [9] have been studied for information gathering to maximize the total value (e.g. utility or informativeness) of sequentially collected samples. [10, 25] extends adaptive sampling in multi-robot systems where the robots make sequential decisions regarding the next best waypoint to sample and then perform the path planning. The sampling criterion is often determined by predicted utility using GP model or information-theoretic criterion such as mutual information gain [2, 14] to maximize the sampled utility or model uncertainty reduction respectively. Besides GP-based adaptive sampling approaches, ergodic control methods have been proposed in [15, 16] to track the unknown spatial distribution by using ergodicity metrics to optimize time averaged trajectory in accordance with the expected spatial distribution, with the final trajectory statistics matching to the initially unknown distribution. As in our problem the goal combines the environmental distribution modeling and sensor coverage control, we propose to use adaptive sampling to augment the multi-robot adaptive coverage controller. A recent work [8] proposed an efficient voronoi-based multi-robot informative adaptive sampling, where each robot only takes the best samples within its assigned partitioned region. [19] developed input-dependent model using the general approach of mixture of GPs [28] to accurately represent complex distributions with the linear combination of different GP

models learned on-line. Our recent work [13] proposed to extend mixture of GPs in adaptive multi-robot coverage control. However, these approaches still require global information of the collected samples by all the robots. [6] proposed a distributed EM algorithm for classification tasks with Gaussian mixture model. Inspired by this work, we propose to employ the distributed EM algorithm with consensus learning as a heuristic method to classify each robot's local data and embed it into our distributed computation of mixture of GPs to improve the local prediction accuracy, which leads to better multi-robot coverage performance.

## 3 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 [3, 4, 20].

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

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

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 sensing cost function of static multi-robot coverage can be formally defined as follows [3, 4].

$$\mathcal{H}(x_1, \ldots, x_n) = \sum_{i=1}^{n} \int_{q \in V_i} \|q - x_i\|^2 \phi(q) dq \tag{2}$$

Hence the lower $\mathcal{H}(x_1, \ldots, x_n)$ 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.

$$x_i^* = \arg\min \mathcal{H}(x_1, \ldots, x_n) = \frac{\int_{V_i} q\phi(q) dq}{\int_{V_i} \phi(q) dq} = C_{V_i} \tag{3}$$

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, 20]). The decentralized gradient-based move-to-centroid controller [4] 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 $C_{V_i}$ by using the predicted value of $\phi(q)$. Also note that in the distributed settings, each robot $i$ is tasked to learn $\phi(q)$ over the points $q \in V_i$ that are inside the robot's current assigned Voronoi cell $V_i$ defined in (1) with its own collected local data only.

## 4 MIXTURE OF GAUSSIAN PROCESSES IN ENVIRONMENT MODELING AND ADAPTIVE COVERAGE CONTROL LAW

In this section, we introduce the model of mixture of GPs used on each robot for predicting the density function $\phi(\cdot)$ with its locally sampled training data set and the resulting decentralized coverage law. The distributed algorithm to compute the mixture of GPs will be introduced in Section 5.

### 4.1 Gaussian Process Regression (GP)

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 [22]. Assume the target phenomenon satisfies a multivariate joint Gaussian distribution [9, 12], 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 $q$.

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 $\mathbf{y_i} = [y_1^i, \ldots, y_{N_i}^i]^T$ 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 $\phi(q_{test}) \sim \mathcal{N}(\mu_{q_{test}|\tilde{V}_i, \mathbf{y_i}}, \sigma^2_{q_{test}|\tilde{V}_i, \mathbf{y_i}})$.

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

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 $\sigma_f$ 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 $(\sigma_n, \sigma_f, l)$ are optimized from the local training data $\{\tilde{V}_i, \mathbf{y_i}\}$, which will be introduced in Section 4.2.

### 4.2 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 $\theta_i = \{\sigma_n, \sigma_f, l\}$. 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. 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 $\theta_i$ as described in [21].

### 4.3 Mixture of Gaussian Process Models and Adaptive Sampling Strategy

Although the uni-model GP introduced above provides good generalization of the density function to learn, it could fail to capture significantly different components of the function distribution as described in [13, 19, 27]. This necessities the use of mixture of GP models that better generalize the function as the one in Fig. 1. The mixture of GP models proposed in [28] is a linear combination of multiple GP models. Assuming the environmental phenomenon can be described by a set of GP models $\{\mathcal{GP}_1, \ldots, \mathcal{GP}_m\}$ with $m$ as the number of Gaussian components, and denote $P(z(q) = i_g)$ as the probability of any random point $q \in Q$ being best described by the $i_g$th GP model. Then for any robot $i$ we have the mixture of GPs defined by the local conditional posterior mean $\mu^*_{q_{test}|\tilde{V}_i,\mathbf{y}_i}$ and variance $\sigma^{*2}_{q_{test}|\tilde{V}_i,\mathbf{y}_i}$ for any query/testing location $q_{test} \in Q$ as follows.

$$\begin{aligned}\mu^*_{q_{test}|\tilde{V}_i,\mathbf{y}_i} &= \sum_{i_g=1}^{m} P(z(q_{test}) = i_g)\cdot \mu_{q_{test}|\tilde{V}_i^{i_g},\mathbf{y}_i^{i_g}} \\ \sigma^{*2}_{q_{test}|\tilde{V}_i,\mathbf{y}_i} &= \sum_{i_g=1}^{m} P(z(q_{test}) = i_g)\cdot \Big(\sigma^2_{q_{test}|\tilde{V}_i^{i_g},\mathbf{y}_i{}^{i_g}} + \\ &\qquad (\mu_{q_{test}|\tilde{V}_i^{i_g},\mathbf{y}_i^{i_g}} - \mu^*_{q_{test}|\tilde{V}_i,\mathbf{y}_i})^2\Big)\end{aligned} \tag{8}$$

where $\tilde{V}_i^{i_g} \subset \tilde{V}_i$ is the subset of collected data by robot $i$ that can be best described by the $i_g$th GP model. We will discuss how to compute the weight distribution $P(z(q_{test}) = i_g)$ and membership set $\tilde{V}_i^{i_g}$ in Section 5. For any point $q$, its actual value of phenomenon $\phi(q)$ is assumed to be sampled from the Gaussian distribution $\mathcal{N}(\mu^*_{q|\tilde{V}_i,\mathbf{y}_i}, \sigma^{*2}_{q|\tilde{V}_i,\mathbf{y}_i})$. And the common approach for efficient sampling and modeling is to navigate the robots to the point $q^* = \arg\max \mu^*_{q^*|\tilde{V}_i,\mathbf{y}_i}$ or $q^* = \arg\max \sigma^{*2}_{q|\tilde{V}_i,\mathbf{y}_i}$ to maximize the sampled value of phenomenon or minimize the prediction uncertainty.

In our problem, we want to simultaneously sample the area with a 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 $\phi(\cdot)$ to approximate to the actual $C_{V_i}$. Here we use the Gaussian Process Upper Confidence Bound (GP-UCB) [26], 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.

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

where $\beta$ is a parameter relates to the current sampling iteration number and regret bound [26]. When $\beta$ is specified by a much higher value, then our solution becomes similar to the informative sampling [24] 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 the unknown density function realization with the GP-UCB evaluation (9), which yields our adaptive sampling strategy for each robot $i$ as follows.

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

And the local adaptive 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 model-related information (not training data) from other robots for generalizing the overall regression model. To simplify our discussion, we assume the robots are always connected as in [2] and are able to share their model-related information by communicating with its direct Voronoi neighbors [24].

## 5 DISTRIBUTED LEARNING OF MIXTURE OF GAUSSIAN PROCESSES MODELS

In this section, we introduce the distributed consensus learning of mixture of GPs for every robot $i$ and the overall learning and adaptive coverage control algorithm that runs on each robot with the knowledge of only local data.

## 5.1 Local Training Data Classification with Distributed Expectation-Maximization (EM)

Recalling the mixture of $m$ GPs in (8) for each robot $i$ to predict $\mu^*_{q|\tilde{V}_i,\mathbf{y}_i}$ and $\sigma^{*2}_{q|\tilde{V}_i,\mathbf{y}_i}$ at any query point $q$, it requires the knowledge of 1) classified local data set $\tilde{V}_i = \{\tilde{V}_i^1, \ldots, \tilde{V}_i^m\}$ and 2) *predicted* weight distribution $P(z(q) = i_g)$ for $i_g = 1, \ldots, m$ where $q \notin \tilde{V}_i$, also known as the gating function. Such information can be computed using the well known EM algorithm [28] with the assumption that the global knowledge of all data collected from every robot is available during EM computation. The EM algorithm consists of two stages such as the estimation (E) stage for computing the weight distribution $P(z(q_j) = i_g)$ where $q_j \in \tilde{V}_i$ and the maximization (M) stage for updating the GP models, and it keeps looping until convergence under some threshold [28]. In order to relax the assumption of global data, we propose to first employ Gaussian Mixture Models (GMM) with distributed EM algorithm [6] via peer-to-peer inter-robot communication for local training data classification and computation of weight distribution $P(z(q_j) = i_g)$ for *collected* points $q_j \in \tilde{V}_i$. Then we use the corresponding labeled local data $\tilde{V}_i$ with weight distribution for training the mixture of GPs, which will be described in our distributed mixture of GPs algorithm in Section 5.2.

Although a GP model is an infinite dimension object, the real-world phenomenon can often be characterized by a finite number of Gaussian components [24, 27]. Here we assume the value of environmental phenomenon $\phi(q)$ is drawn from $m$ Gaussian components (corresponding to $m$ GP models) with each component described by a set of unknown model parameters $\Theta_{i_g} = \{\alpha_{i_g}, \mu_{i_g}, \Sigma_{i_g}, i_g = 1, \ldots, m\}$ where $\alpha_{i_g}$ is the probability of $\phi(q)$ drawn from the $i_g$th Gaussian component $\mathcal{N}_{i_g}(\mu_{i_g}, \Sigma_{i_g})$. Then we can rewrite the global summary quantities for each Gaussian component $\mathcal{N}_{i_g}$ as follows.

$$\begin{aligned} \alpha_{i_g} &= \frac{1}{n}\sum_{i=1}^{n} |\mathbf{y}_i|\alpha_{i,i_g}, \quad \lambda_{i_g} = \frac{1}{n}\sum_{i=1}^{n} |\mathbf{y}_i|\lambda_{i,i_g} \\ \gamma_{i_g} &= \frac{1}{n}\sum_{i=1}^{n} |\mathbf{y}_i|\gamma_{i,i_g}, \quad \mu_{i_g} = \frac{\lambda_{i_g}}{\alpha_{i_g}}, \quad \Sigma_{i_g} = \frac{\gamma_{i_g}}{\alpha_{i_g}} \end{aligned} \tag{12}$$

where $\{\alpha_{i,i_g}, \lambda_{i,i_g}, \gamma_{i,i_g}\}$ are the local summary quantities for each robot $i$ that can be computed locally as follows, given the information of its current *estimated* global model parameter $\tilde{\Theta}_{i,i_g} = \{\tilde{\alpha}_{i,i_g}, \tilde{\mu}_{i,i_g}, \tilde{\Sigma}_{i,i_g}\}$ for $i_g = 1, \ldots, m$ (obtained via dynamic consensus in (14)) and the observed value $\mathbf{y}_i = \{y_j\}, j = 1, \ldots, |\mathbf{y}_i|$ of local data set $\tilde{V}_i$.

$$\begin{aligned} p(z(q_j) = i_g|y_j, \tilde{\Theta}_{i,i_g}) &= \frac{\tilde{\alpha}_{i,i_g} \cdot p(y_j|\tilde{\mu}_{i,i_g}, \tilde{\Sigma}_{i,i_g})}{\sum_{h=1}^{m} \tilde{\alpha}_{i,h} \cdot p(y_j|\tilde{\mu}_{i,h}, \tilde{\Sigma}_{i,h})} \\ \alpha_{i,i_g} &= \frac{1}{|\mathbf{y}_i|}\sum_{j=1}^{|\mathbf{y}_i|} p(z(q_j) = i_g|y_j, \tilde{\Theta}_{i,i_g}) \\ \lambda_{i,i_g} &= \sum_{j=1}^{|\mathbf{y}_i|} y_j \cdot p(z(q_j) = i_g|y_j, \tilde{\Theta}_{i,i_g}) \end{aligned} \tag{13}$$

$$\gamma_{i,i_g} = \sum_{j=1}^{|\mathbf{y}_i|} p(z(q_j) = i_g|y_j, \tilde{\Theta}_{i,i_g}) \cdot (y_j - \tilde{\mu}_{i,i_g})(y_j - \tilde{\mu}_{i,i_g})^T$$

in which the probability $p(y_j|\tilde{\mu}_{i,i_g}, \tilde{\Sigma}_{i,i_g})$ can be directly computed from the Gaussian distribution defined by $\tilde{\mathcal{N}}_{i_g}(\tilde{\mu}_{i,i_g}, \tilde{\Theta}_{i,i_g})$. Then for each robot $i$, we define $\mathbf{x}_{i,i_g} \in \mathbb{R}^3$ as its local estimate of the global summary quantities $\alpha_{i_g}, \lambda_{i_g}, \gamma_{i_g}$ and define its own statistics $\mathbf{u}_{i,i_g} = [|\mathbf{y}_i|\alpha_{i,i_g}, \lambda_{i,i_g}, \gamma_{i,i_g}]^T$ as the local summary quantities. A consensus filter can be designed with convergence and stability guarantee [6], so that each robot $i$ will agree on the similar value of Gaussian components $\Theta_{i_g}$ via peer-to-peer communication through connected network graph.

$$\dot{\mathbf{x}}_{i,i_g} = \sum_{j \in \text{neighbor of } i} (\mathbf{x}_{j,i_g} - \mathbf{x}_{i,i_g}) + (\mathbf{u}_{i,i_g} - \mathbf{x}_{i,i_g}) \tag{14}$$

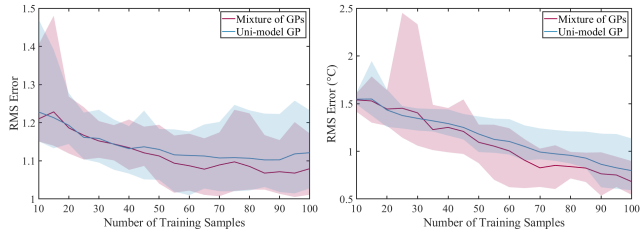
where the neighbors of robot $i$ are specified by all robots located spatially within a predefined distance to the robot $i$. It is noted that the above computation only relies on the robot's local data set $\mathbf{y}_i$ and their communication packets of model information $\mathbf{x}_{i,i_g}$ (related to $\tilde{\Theta}_{i,i_g}$) only. With the converged estimated Gaussian Mixture Model parameters $\Theta_{i_g}$ for all $i_g = 1, \ldots, m$, we are able to compute the weight distribution $P(z(q_j) = i_g) = p(z(q_j) = i_g|y_j, \Theta_{i_g})$ w.r.t. each Gaussian component $i_g$ for each training data $\{q_j; y_j\}$ of robot $i$ and obtained the training data classifications as follows.

$$\tilde{V}_i^{i_g}: \qquad \{q_j \in \mathbb{R}^2 | i_g = \arg\max P(z(q_j) = i_g)\}, \forall i_g = 1, \ldots, m \tag{15}$$

Intuitively, equations (12)-(15) indicate the process that the robots exchange model-related parameters to dynamically form an estimated and converged global statistics $\Theta_{i_g}$ of GMM, and hence to gradually readjust the classification of its local data in (15) as more samples are collected. As noted in [6], the standard EM with centralized computing has the complexity of communication in bytes as $O(n^{3/2})$ and $O(n^2)$ for the worst case, while with the distributed EM in (12)-(15), the complexity is $O(n)$ that is linear to the number of robots $n$. In the following Section 5.2, we will provide a complete distributed algorithm for computing the mixture of GPs with the classified local data.

## 5.2 Distributed Mixture of GPs in Adaptive Coverage Control

With the classified training dataset $\tilde{V}_i^{i_g}, \forall i_g = 1, \ldots, m$ from (15) for each robot $i$ and weight distribution $P(z(q_j) = i_g)$ of each collected data therein, then the robots only need the knowledge of the predicted weight distribution (gating function) $P(z(q) = i_g)$ for $i_g = 1, \ldots, m$ that defines the likelihood each query data $q$ belongs to the $i_g$th GP, so as to complete the modeling of mixture of GPs in (8). Such gating function mapping from $q$ to $P(z(q) = i_g)$ can be learned using another GP for each robot $i$ by considering the already obtained training data $\{q_j; P(z(q_j) = i_g)\}, \forall q_j \in \tilde{V}_i$ in the similar form as (5). To that end, for any new data $q$ sensed by robot $i$ (inside robot $i$'s Voronoi cell), it can compute the prediction (8) using (15) and the learned gating function $P(z(q) = i_g)$, which further yields the updated control law (10)-(11) to govern the motion of the robots.




**(a) Agricultural Data: Stalk Count** **(b) Office Data: Temperature**

**Figure 2: Prediction performance comparison between mixture of GPs and uni-model GP on two real-world dataset.**

Finally, our algorithm of distributed mixture of GPs in adaptive coverage control running on each robot $i$ is summarized as follows.

*Step 1:* Take one sample from robot' current location and update its local data set $\{\tilde{V}_i; \mathbf{y_i}\}$. Recompute its Voronoi region.

*Step 2:* Compute local GMM parameter $\mathbf{u}_{i,i_g} = [|\mathbf{y}_i|\alpha_{i,i_g}, \lambda_{i,i_g}, \gamma_{i,i_g}]^T$ using (13) and current belief of global GMM parameters $\tilde{\Theta}_{i,i_g}$ on current local data set $\{\tilde{V}_i; \mathbf{y_i}\}$.

*Step 3:* Exchange local GMM model information of $\mathbf{x}_{i,i_g}$ with neighbors and compute the consensus using (14) till convergence and get updated GMM parameters $\tilde{\Theta}_{i,i_g}$ from converged $\mathbf{x}_{i,i_g}$ for all $i_g = 1, \ldots, m$.

*Step 4:* Classify local data set $\tilde{V}_i^{i_g}$ and weight distribution $P(z(q_j) = i_g)$ of each training data using (15) with $\tilde{\Theta}_{i,i_g}$ for all $i_g = 1, \ldots, m$.

*Step 5:* Train gating function $P(z(q) = i_g)$ with training data $\{q_j; P(z(q_j) = i_g)\}$ and together with the classified local data set $\tilde{V}_i^{i_g}, \forall i_g = 1, \ldots, m$, locally fit mixture of GPs using (8).

*Step 6:* With the learned mixture of GPs, predict the posterior mean and variance of density function $\phi(q)$ over its Voronoi region and evaluate each $h(q)$ in the region using (9).

*Step 7:* Compute local adaptive coverage control law with (11), execute, and go back to *Step 1*.

## 6 EXPERIMENTAL RESULTS AND ANALYSIS

In this section, we present several empirical results on two real-world datasets from the agricultural robotic sampling application [17] and Intel Berkeley Lab [1] with MATLAB toolboxes: the GPML [21]. The agricultural dataset contains data of the number of stalks counted per grid over the $21 \times 45 = 945$ distinct grids collected in August 2017 over a sorghum field, and the Intel indoor dataset contains sensory temperature data collected from 54 sensors in an office area between Feb 28th and Apr 5th, 2004. In our particular tasks, we use the 2D location information with the stalk count and temperature readings (degrees in Celsius) respectively as the ground truth of the environmental phenomenon over map and compare our algorithm performance to other approaches.

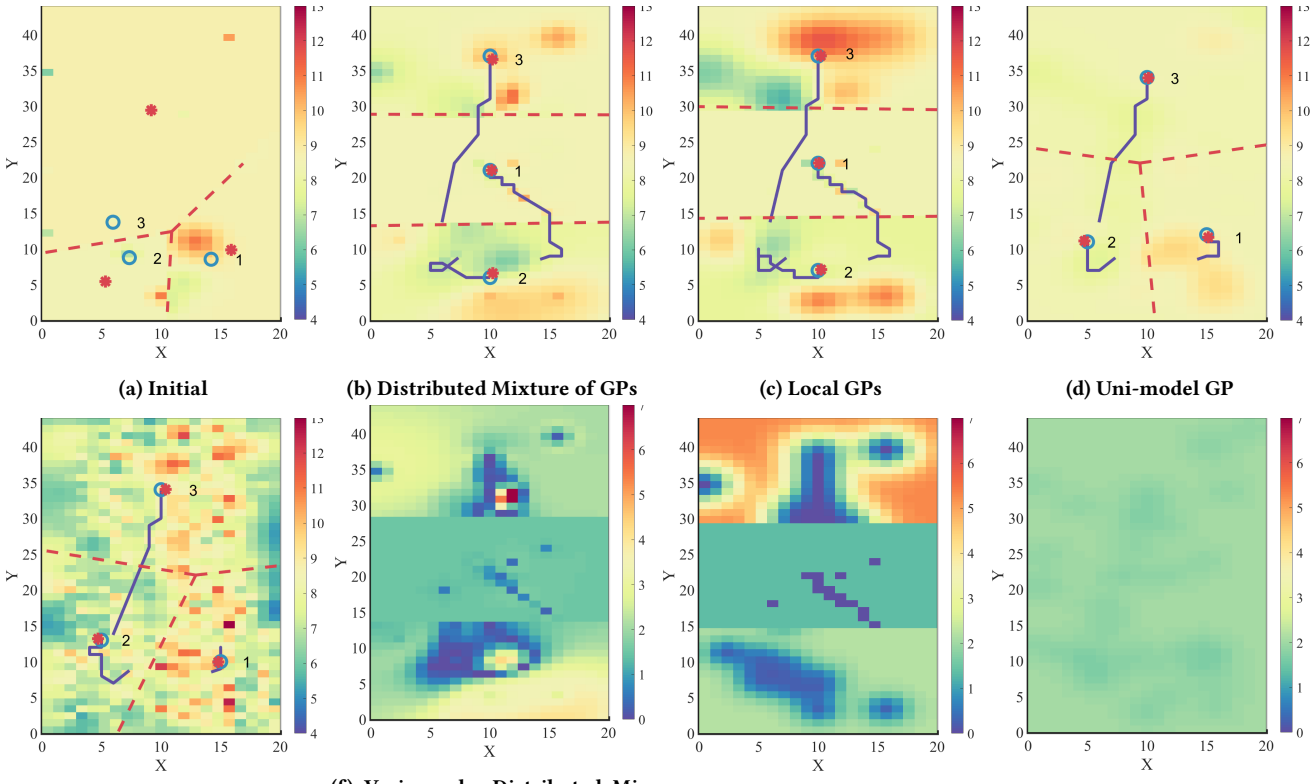
Before implementing our multi-robot sensor coverage task, we first provide the empirical results for static prediction of the two environmental phenomenon using centralized mixture of GPs ($m = 3$) and centralized uni-model GP respectively. For each dataset, we randomly select a growing number of data from 10 to 100 to serve as the training data and use the rest of the unselected data as testing data. After 10 random trials at each training sample setting, the prediction performance on the Root Mean Square (RMS) error are plotted in Fig. 2 with solid lines as the mean RMS error covered by the maximum-minimum error bar showing the maximum and minimum predictive RMS error in each 10 trials. The results suggest as the number of training samples grow, the mixture of GPs outperforms the uni-model GP in both of the tasks, indicating that for the considered real-world non-smooth data the mixture of GPs can better characterize the distribution of environmental phenomenon.

### 6.1 Simulation Example with Agricultural Dataset

For multi-robot sensor coverage task, we consider the example in Fig. 3 where we have 3 robots deployed from random starting points shown in Fig. 3(a) to find the optimal final configurations for stationary sensing as shown earlier in Fig. 1, where the distribution of stalk count has multiple peaks around the top right corners.

As shown in Fig. 3(a), the robots initially have little knowledge about the true distribution with only 13 randomly chosen prior training data points over the map. With our distributed mixture of GPs algorithm in Fig. 3(b), the robots are governed by our proposed adaptive coverage controller (11) with distributed learning of mixture of GPs ($k_p = 0.5, \beta = 1, m = 3$) to simultaneously learn the environmental model and try to approach the actual centroid of each Voronoi cell based on its own model inference. We assume any robot pairs sharing the same Voronoi edge could communicate to each other. The converged results of configurations under adaptive coverage controllers with other modeling approaches are shown in Fig. 3(c)-(e), where the local GPs (Fig. 3(c)) are trained by each robot with its own local collected data without communication and the uni-model GP (Fig. 3(d)) assumes global knowledge of all robots' collected data. The known model (Fig. 3(e)) is the ground truth controller with full knowledge of the density function as done in [4]. They have different modeling of $\phi(q)$ but use the same form of our adaptive coverage control laws (4). Quantitative prediction and coverage performance on the same map are also provided in Fig. 4 and evaluated by the metrics of 1) Root Mean Square (RMS) error between predicted stalk count and ground truth stalk count on all unvisited locations, and 2) the actual sensing cost computed by (2). Besides the mentioned comparison algorithms, in Fig. 4 we also introduce the result from centralized mixture of GPs modeling algorithm [13] with the same parameter settings ($k_p = 0.5, \beta = 1, m = 3$), but use the training data from all robots. It is noted from Fig. 4 that when assuming global knowledge of all robot's data, the centralized mixture of GPs has better prediction performance over uni-model GP. When the global information is not available, our proposed distributed mixture of GPs has a better prediction performance w.r.t. RMS error than local GPs and the best coverage performance level.

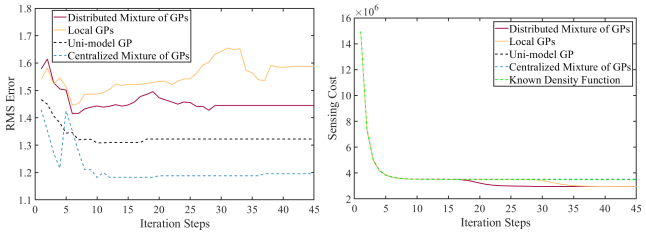
In particular, it is also noted from Fig. 3(b) that the robots with a distributed GP mixture can successfully identify the top right corner with 3 distinguished peak areas by exchanging model information, while in Fig. 3(c) robot 3 falsely predicts a larger peak area due to no information exchange. With the uni-model GP shown in Fig. 3(d), the robots fail to identify significantly different components due to its uni-model inference (although it performs better in prediction RMS error compared to the distributed algorithms due to



(a) Initial (b) Distributed Mixture of GPs (c) Local GPs (d) Uni-model GP

(e) Control with known model (f) Variance by Distributed Mixture of GPs (g) Variance by local GPs (h) Variance by uni-model GP

**Figure 3: An simulation example of 3 robots covering an agricultural field and the modeling results using our distributed mixture of GP model in comparison to 1) local GPs with local data and 2) uni-model GP with global knowledge of collected data from all robots. Robots' current positions are marked by blue circles with dark blue history footprints in (a)-(e). The background heatmap in (a)-(d) indicates the predicted stalk count distribution based on the collected data and (e) the true distribution. The background heatmap in (f)-(h) represents the predicted variance over the map. Edge of Voronoi cells and the ground-truth optimal locations for that particular configurations are represented by red dashed lines and red stars, respectively.**



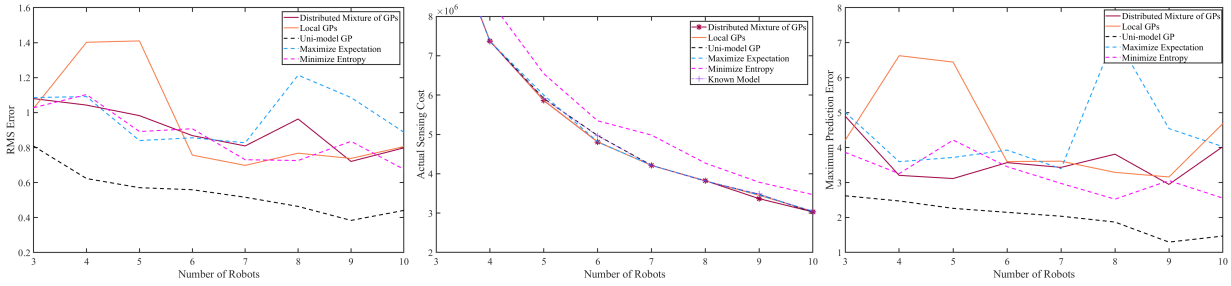

(a) RMS error (b) Sensing cost

**Figure 4: Prediction and coverage performance comparison.**

its global knowledge of all the collected data). Results of prediction variance are shown in Fig. 3(f)-(h) and our distributed mixture of GPs has smoother prediction due to the consensus of environment model but are still able to capture the prediction uncertainty differences over the area. Local GP method in Fig. 3(g) has much larger prediction variance among different robots as they are sampling in different places with significantly distinct components without information exchanging. Using uni-model GP in Fig. 3(h) ignores the local features and hence the prediction variance is almost the same over any unvisited areas. It is noted that even with only a few samples collected from the map, the converged configurations are very close to the optimal ones from the actual temperature distribution due to our adaptive sampling criterion and algorithm that trades off between uncertainty reduction and centroid approaching.

## 6.2 Quantitative Results

To further compare our algorithms performance in other environmental phenomenon, we run 40 trials on 5 different sets of data from Intel Berkeley dataset [1] with 3 to 10 robots respectively in Fig. 5. Note that as the complexity of communication for our distributed mixture of GPs is linear to the number of robots and each robot only computes on the local data only, our algorithm is also scalable to a larger number of robots. The other modeling algorithms we are comparing includes the mentioned local GPs, uni-model GP, known model coverage control, as well as the Expectation maximization ($\beta = 0$) and Entropy minimization algorithm ($\beta > 20$) coupled with the same form of coverage controller in (4).




**(a) RMS error** **(b) Sensing cost** **(c) Maximum prediction error**

**Figure 5: Comparisons of RMS error, sensing cost, and maximum prediction error occurred v.s. different number of robots for different algorithms.**

Expectation maximization algorithm seeks to find the area with highest value of predicted environmental phenomenon while the Entropy minimization algorithm seeks to find the area with highest prediction variance to reduce model uncertainty. The centralized uni-model GP has the best performance w.r.t. RMS error and prediction error due to its global knowledge of data. In particular, the Entropy minimization has the best prediction performance w.r.t. the RMS error and maximum prediction error, but the worst coverage performance since it only prefers high uncertainty area to the place with higher density value, and thus lead to inferior suboptimal configurations due to local minima nature of the move-to-centroid controller (finding global optimal config is NP hard). Our distributed Mixture of GPs has the best coverage performance even with only local data (considering the scale of the cost) and in general the second best performance in prediction (following Entropy minimization) due to the GP-UCB criteria in our algorithm that trades off between uncertainty reduction and prediction maximization. As the number of robots increases, our algorithm outperforms the local GPs w.r.t. RMS error and maximum prediction error as our distributed Mixture of GPs is able to approximate global statistics via the consensus algorithm.

## 7 CONCLUSION

In this paper, we present an adaptive sampling algorithm with distributed environmental learning in multi-robot sensor coverage problem using the 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. A distributed consensus learning algorithm is employed so that no global knowledge of training data is required and only model-based parameters are exchanged between robots that are independent from the number of collected data. Simulation results have shown the effectiveness of our algorithm compared to other approaches. In the future, we will investigate heuristic algorithms for choosing the proper number of GP components that are currently predetermined in our distributed computation of mixture of GPs models. We will also research on other information gathering model such as ergodicity metric and to derive theoretical guarantee on the modeled density function and coverage performance.

## ACKNOWLEDGMENTS

This work was supported by U.S. Department of Agriculture (DOA) award 2017-6700-726152, DARPA Cooperative Agreement No.: HR00111820051 and AFOSR awards FA9550-15-1-0442.

## REFERENCES

[1] 2004. Intel Lab Data. (2004). http://db.csail.mit.edu/labdata/labdata.html
[2] Micah Corah and Nathan Michael. 2017. Efficient Online Multi-robot Exploration via Distributed Sequential Greedy Assignment. In *Proceedings of Robotics: Science and Systems*. Cambridge, Massachusetts. https://doi.org/10.15607/RSS.2017.XIII.070
[3] Jorge Cortés. 2010. Coverage optimization and spatial load balancing by robotic sensor networks. *IEEE Trans. Automat. Control* 55, 3 (2010), 749–754.
[4] Jorge Cortes, Sonia Martinez, Timur Karatas, and Francesco Bullo. 2004. Coverage control for mobile sensing networks. *IEEE Transactions on robotics and Automation* 20, 2 (2004), 243–255.
[5] Qiang Du, Maria Emelianenko, and Lili Ju. 2006. Convergence of the Lloyd algorithm for computing centroidal Voronoi tessellations. *SIAM journal on numerical analysis* 44, 1 (2006), 102–119.
[6] Dongbing Gu. 2008. Distributed EM algorithm for Gaussian mixtures in sensor networks. *IEEE Transactions on Neural Networks* 19, 7 (2008), 1154–1166.
[7] Geoffrey A Hollinger and Gaurav S Sukhatme. 2014. Sampling-based robotic information gathering algorithms. *The International Journal of Robotics Research* 33, 9 (2014), 1271–1287.
[8] Stephanie Kemna, John G Rogers, Carlos Nieto-Granda, Stuart Young, and Gaurav S Sukhatme. 2017. Multi-robot coordination through dynamic Voronoi partitioning for informative adaptive sampling in communication-constrained environments. In *ICRA*. IEEE, 2124–2130.
[9] Andreas Krause, Ajit Singh, and Carlos Guestrin. 2008. Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *Journal of Machine Learning Research (JMLR)* 9 (February 2008), 235–284.
[10] Kian Hsiang Low, John M Dolan, and Pradeep Khosla. 2008. Adaptive multi-robot wide-area exploration and mapping. In *Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems-Volume 1*. International Foundation for Autonomous Agents and Multiagent Systems, 23–30.
[11] Wenhao Luo, Shehzaman S Khatib, Sasanka Nagavalli, Nilanjan Chakraborty, and Katia Sycara. 2016. Distributed knowledge leader selection for multi-robot environmental sampling under bandwidth constraints. In *IROS*. IEEE, 5751–5757.
[12] W. Luo, C. Nam, and K. Sycara. 2017. 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)*. 118–123. https://doi.org/10.1109/MFI.2017.8170416
[13] Wenhao Luo and Katia Sycara. 2018. Adaptive Sampling and Online Learning in Multi-Robot Sensor Coverage with Mixture of Gaussian Processes. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 6359–6364.

[14] Kai-Chieh Ma, Lantao Liu, and Gaurav S Sukhatme. 2017. Informative planning and online learning with sparse gaussian processes. In *ICRA*. IEEE, 4292–4298.

[15] Lauren M Miller and Todd D Murphey. 2013. Trajectory optimization for continuous ergodic exploration. In *2013 American Control Conference*. IEEE, 4196–4201.

[16] Lauren M Miller, Yonatan Silverman, Malcolm A MacIver, and Todd D Murphey. 2016. Ergodic exploration of distributed information. *IEEE Transactions on Robotics* 32, 1 (2016), 36–52.

[17] Tim Mueller-Sim, Merritt Jenkins, Justin Abel, and George Kantor. 2017. The Robotanist: A ground-based agricultural robot for high-throughput crop phenotyping. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 3634–3639.

[18] Atsuyuki Okabe and Atsuo Suzuki. 1997. Locational optimization problems solved through Voronoi diagrams. *European Journal of Operational Research* 98, 3 (1997), 445–456.

[19] Ruofei Ouyang, Kian Hsiang Low, Jie Chen, and Patrick Jaillet. 2014. Multi-robot active sensing of non-stationary gaussian process-based environmental phenomena. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. International Foundation for Autonomous Agents and Multiagent Systems, 573–580.

[20] Alyssa Pierson, Lucas C Figueiredo, Luciano CA Pimenta, and Mac Schwager. 2017. Adapting to sensing and actuation variations in multi-robot coverage. *The International Journal of Robotics Research* 36, 3 (2017), 337–354.

[21] Carl Edward Rasmussen and Hannes Nickisch. 2010. Gaussian processes for machine learning (GPML) toolbox. *Journal of Machine Learning Research* 11, Nov (2010), 3011–3015.

[22] Carl Edward Rasmussen and Christopher KI Williams. 2006. *Gaussian processes for machine learning*. Vol. 1. MIT press Cambridge.

[23] Mac Schwager, Daniela Rus, and Jean-Jacques Slotine. 2009. Decentralized, adaptive coverage control for networked robots. *The International Journal of Robotics Research* 28, 3 (2009), 357–375.

[24] Mac Schwager, Michael P Vitus, Samantha Powers, Daniela Rus, and Claire J Tomlin. 2017. Robust adaptive coverage control for robotic sensor networks. *IEEE Transactions on Control of Network Systems* 4, 3 (2017), 462–476.

[25] Amarjeet Singh, Andreas Krause, and William J Kaiser. 2009. Nonmyopic Adaptive Informative Path Planning for Multiple Robots.. In *IJCAI*, Vol. 3. 2.

[26] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias W Seeger. 2012. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory* 58, 5 (2012), 3250–3265.

[27] Cyrill Stachniss, Christian Plagemann, and Achim J Lilienthal. 2009. Learning gas distribution models using sparse Gaussian process mixtures. *Autonomous Robots* 26, 2 (2009), 187–202.

[28] Volker Tresp. 2001. Mixtures of Gaussian processes. In *Advances in neural information processing systems*. 654–660.