Recent Advances in Scaling up Gaussian Process Predictive Models for Large Spatiotemporal Data

Kian Hsiang Low[1], Jie Chen[2], Trong Nghia Hoang[1], Nuo Xu[1], and Patrick Jaillet[3]

[1] National University of Singapore {lowkh,nghiaht,xunuo}@comp.nus.edu.sg
[2] Singapore-MIT Alliance for Research and Technology chenjie@smart.mit.edu
[3] Massachusetts Institute of Technology jaillet@mit.edu

Abstract. The expressive power of *Gaussian process* (GP) models comes at a cost of poor scalability in the size of the data. To improve their scalability, this paper presents an overview of our recent progress in scaling up GP models for large spatiotemporally correlated data through parallelization on clusters of machines, online learning, and nonmyopic active sensing/learning.

1 Introduction

Gaussian process (GP) models are a rich class of Bayesian non-parametric models that can perform probabilistic regression by providing Gaussian predictive distributions with formal measures of predictive uncertainty. Unfortunately, the expressive power of a *full-rank GP* (FGP) model comes at a cost of poor scalability (i.e., cubic time) in the data size, which hinders its practical use for large data generated from environmental sensing and monitoring applications. To boost its scalability, two research trends are prevalent:

Model approximation. To improve the time efficiency of training with all the given data, structural assumptions have been imposed on the FGP model to yield two different classes of sparse GP approximation methods: (a) Low-rank approximate representations [7,27,31] of the FGP model are especially suitable for modeling smoothly-varying environmental phenomena with high spatiotemporal correlation (i.e., long length-scales) and they utilize all the data for predictions like FGP; and (b) localized regression and covariance tapering methods (e.g., local GPs [6,25] and compactly supported covariance functions [11]) are capable of modeling highly-varying phenomena with low correlation (i.e., short length-scales) but they use only local data for predictions, hence predicting poorly in areas with sparse data. Recent sparse GP approximation methods [3,28,30] have attempted to unify the best of both worlds.

Data/information gathering. Alternatively, the GP model can be trained with considerably less but highly informative data that are actively (as opposed to passively) gathered by optimizing some active sensing/learning[4] criterion defined using mean-square error, entropy, or mutual information [1,16,18,19]. This is particularly desirable in environmental sensing applications and tasks constrained by some sampling budget.

This paper presents an overview of our recent progress in scaling up GP models for large spatiotemporally correlated data along the two research directions discussed above. The specific contributions of our three recent works [3,13,32] include:

[4] Active sensing/learning in machine learning is also known as adaptive sampling in oceanography and control [17].

Parallel GP models. Though existing sparse GP approximation methods utilizing low-rank representations (i.e., including the unified approaches) [7,27,28,30,31] have improved the scalability of GP models to linear time in the data size, they remain computationally impractical for performing real-time predictions necessary in many time-critical environmental sensing and monitoring applications and decision support systems (e.g., precision agriculture [19], sensing and monitoring of ocean, freshwater, and traffic phenomena [1,2,4,5,10,17,18,20,21,26], GIS) that need to process and analyze huge quantities of data collected over short time durations (e.g., in traffic, meteorology, surveillance). To resolve this, our first work considers exploiting clusters of parallel machines to achieve efficient predictions in real time. The local GPs method [6] appears most straightforward to be "embarrassingly" parallelized but they suffer from discontinuities in predictions on the boundaries of different local GPs. The work of [25] rectifies this problem by imposing continuity constraints along the boundaries in a centralized manner. But, its use is restricted strictly to data with 1D and 2D input features.

Different from the parallel local GPs method, our proposed parallel GP models [3] (Section 3), which exploit low-rank approximate representations for distributing the computational load among parallel machines to achieve time efficiency and scalability, do not suffer from boundary effects, work with multi-dimensional input features, and exploit all the data for predictions but do not incur the cubic time cost of FGP model. We theoretically guarantee the predictive performances of our parallel GP models to be equivalent to that of some centralized sparse GP approximation methods and implement them using the *message passing interface* (MPI) framework to run in a cluster of 20 computing nodes for empirically evaluating their predictive performances, time efficiency, scalability, and speedups on a dataset featuring a real-world traffic phenomenon. Interestingly, our parallel GP models can be adapted to GP-based decentralized data fusion algorithms to be run on a network of mobile sensors for cooperative perception of spatiotemporally varying environmental phenomena, as detailed in [4,5].

Online GP model. When the data is expected to be streaming in over a (possibly indefinitely) long period of time, it is computationally impractical to repeatedly use existing offline sparse GP approximation methods [7,27,28,30,31] or online FGP model [9] for training at each time step because they incur, respectively, linear and quadratic time in the data size per time step. Our next work proposes a novel online sparse GP approximation method [32,22] (Section 4) that, in contrast to existing works mentioned above, is capable of achieving *constant* time and memory (i.e., independent of data size) per time step. We provide a theoretical guarantee on its predictive performance to be equivalent to that of the offline sparse *partially independent training conditional* (PITC) approximation method. Our proposed method [32] generalizes the sparse online GP model of [9] by relaxing its conditional independence assumption significantly, hence potentially improving the predictive performance. We empirically demonstrate the practical feasibility of using our generalized online sparse GP model through a real-world persistent mobile robot localization experiment.

Nonmyopic active sensing/learning. Its objective is to derive an optimal sequential policy that plans the most informative locations to be observed for minimizing the predictive uncertainty of the unobserved areas of a spatially varying environmental phenomenon given a sampling budget (e.g., number of deployed sensors, energy consump-

tion). To achieve this, many existing active sensing algorithms [1,4,5,16,18,19,20] have assumed the spatial correlation structure of the phenomenon modeled by GP (specifically, the parameters defining it) to be known, which is often violated in real-world applications. The predictive performance of the GP model in fact depends on how informative the gathered observations are for both parameter estimation and spatial prediction given the true parameters. Interestingly, as revealed in [23], policies that are efficient for parameter estimation are not necessarily efficient for spatial prediction with respect to the true model parameters. Thus, active sensing/learning involves a potential trade-off between sampling the most informative locations for spatial prediction given the current, possibly incomplete knowledge of the parameters (i.e., exploitation) vs. observing locations that gain more information about the parameters (i.e., exploration). To address this trade-off, one principled approach is to frame active sensing as a sequential decision problem that jointly optimizes the above exploration-exploitation trade-off while maintaining a Bayesian belief over the model parameters. Solving this problem then results in an induced policy that is guaranteed to be optimal in the expected active sensing performance [13]. Unfortunately, such a nonmyopic *Bayes-optimal active learning* (BAL) policy cannot be derived exactly due to an uncountable set of candidate observations and unknown model parameters. As a result, existing works advocate using greedy policies [24] or performing exploration and exploitation separately [15] to sidestep the difficulty of solving for the exact BAL policy. But, these algorithms are sub-optimal in the presence of budget constraints due to their imbalance between exploration and exploitation [13].

Our final work proposes a novel nonmyopic active sensing/learning algorithm [13,12] (Section 5) that can still preserve and exploit the principled Bayesian sequential decision problem framework for jointly optimizing the exploration-exploitation trade-off and hence does not incur the limitations of existing works. In particular, although the exact BAL policy cannot be derived, we show that it is in fact possible to solve for a nonmyopic ϵ-*Bayes-optimal active learning* (ϵ-BAL) policy given an arbitrary loss bound ϵ. To meet real-time requirement in time-critical applications, we then propose an asymptotically ϵ-optimal anytime algorithm based on ϵ-BAL with performance guarantee. We empirically demonstrate using a dataset featuring a real-world traffic phenomenon that, with limited budget, our approach outperforms state-of-the-art algorithms.

2 Modeling Environmental Phenomena with Gaussian Processes

The GP[5] can be used to model an environmental phenomenon as follows: The phenomenon is defined to vary as a realization of a GP. Let $\mathcal{X}$ be a set of sampling units representing the domain of the phenomenon such that each sampling unit $x \in \mathcal{X}$ denotes a d-dimensional feature vector and is associated with a realized (random) measurement z_x (Z_x) if x is observed (unobserved). Let $\{Z_x\}_{x\in\mathcal{X}}$ denote a GP, that is, every finite subset of $\{Z_x\}_{x\in\mathcal{X}}$ has a multivariate Gaussian distribution. The GP is fully specified by its *prior* mean $\mu_x \triangleq \mathbb{E}[Z_x]$ and covariance $\sigma_{xx'|\lambda} \triangleq \mathrm{cov}[Z_x, Z_{x'}|\lambda]$ for all locations

[5] GP regression in machine learning is equivalent to the data assimilation scheme called objective analysis or optimal interpolation or 3DVAR in oceanography and meteorology [2,17] when the domain is reduced to a finite set of grid points and all observations are at the grid points. It is also equivalent to kriging in geostatistics [8].

$x, x' \in \mathcal{X}$, the latter of which characterizes the spatial correlation structure of the phenomenon and can be defined using a covariance function parameterized by λ. When λ is known and a set $z_\mathcal{D}$ of realized measurements is observed for some set $\mathcal{D} \subset \mathcal{X}$ of sampling units, the FGP model can exploit these observations to predict the unobserved measurement for any sampling unit $x \in \mathcal{X} \setminus \mathcal{D}$ as well as provide its predictive uncertainty using a Gaussian predictive distribution $p(z_x|x, \mathcal{D}, z_\mathcal{D}, \lambda) = \mathcal{N}(\mu_{x|\mathcal{D},\lambda}, \sigma_{xx|\mathcal{D},\lambda})$ with the following *posterior* mean and variance, respectively:

$$\mu_{x|\mathcal{D},\lambda} \triangleq \mu_x + \Sigma_{x\mathcal{D}|\lambda}\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda}\left(z_\mathcal{D} - \mu_\mathcal{D}\right) \quad \text{and} \quad \sigma_{xx|\mathcal{D},\lambda} \triangleq \sigma_{xx|\lambda} - \Sigma_{x\mathcal{D}|\lambda}\Sigma^{-1}_{\mathcal{D}\mathcal{D}|\lambda}\Sigma_{\mathcal{D}x|\lambda} \tag{1}$$

where, with a slight abuse of notation, $z_\mathcal{D}$ is to be perceived as a column vector, $\mu_\mathcal{D}$ is a column vector with prior mean components $\mu_{x'}$ for all $x' \in \mathcal{D}$, $\Sigma_{x\mathcal{D}|\lambda}$ is a row vector with prior covariance components $\sigma_{xx'|\lambda}$ for all $x' \in \mathcal{D}$, $\Sigma_{\mathcal{D}x|\lambda}$ is the transpose of $\Sigma_{x\mathcal{D}|\lambda}$, and $\Sigma_{\mathcal{D}\mathcal{D}|\lambda}$ is a matrix with components $\sigma_{x'x''|\lambda}$ for all $x', x'' \in \mathcal{D}$. When λ is not known, a probabilistic belief $b_\mathcal{D}(\lambda) \triangleq p(\lambda|z_\mathcal{D})$ is maintained over all possible λ and updated using Bayes' rule to the posterior belief $b_{\mathcal{D}\cup\{x\}}(\lambda) \propto p(z_x|x, \mathcal{D}, z_\mathcal{D}, \lambda)\, b_\mathcal{D}(\lambda)$ given a new measurement z_x. Then, using belief $b_\mathcal{D}$, the predictive distribution is obtained by marginalizing out λ: $p(z_x|x, \mathcal{D}, z_\mathcal{D}) = \sum_{\lambda\in\Lambda} p(z_x|x, \mathcal{D}, z_\mathcal{D}, \lambda)\, b_\mathcal{D}(\lambda)$.

3 Parallel GP Models

In this section, we will present a class of parallel GP models (pPITC and pPIC) that distributes the computational load among parallel machines to achieve efficient and scalable approximate GP prediction by exploiting the notion of a support set. The key idea of the *parallel partially independent training conditional* (pPITC) approximation of FGP model is as follows: After distributing the data evenly among N machines (Step 1), each machine encapsulates its local data, based on a common prior support set $\mathcal{S} \subset \mathcal{X}$ where $|\mathcal{S}| \ll |\mathcal{D}|$, into a local summary that is communicated to the master[6] (Step 2). The master assimilates the local summaries into a global summary (Step 3), which is then sent back to the N machines to be used for predictions distributed among them (Step 4). These steps are detailed below. For simplicity, we omit the use of the known GP model parameters λ in our notations.

STEP 1: DISTRIBUTE DATA AMONG N MACHINES.

The data $(\mathcal{D}, y_\mathcal{D})$ is partitioned evenly into N blocks, each of which is assigned to a machine, as defined below:

Definition 1 (Local Data). *The local data of machine n is defined as a tuple $(\mathcal{D}_n, y_{\mathcal{D}_n})$ where $\mathcal{D}_n \subseteq \mathcal{D}$, $\mathcal{D}_n \bigcap \mathcal{D}_i = \emptyset$ and $|\mathcal{D}_n| = |\mathcal{D}_i| = |\mathcal{D}|/N$ for $i \neq n$.*

STEP 2: EACH MACHINE CONSTRUCTS AND SENDS LOCAL SUMMARY TO MASTER.

Definition 2 (Local Summary). *Given a common support set $\mathcal{S} \subset \mathcal{X}$ known to all N machines and the local data $(\mathcal{D}_n, y_{\mathcal{D}_n})$, the local summary of machine n is defined as a tuple $(\dot{y}^n_\mathcal{S}, \dot{\Sigma}^n_{\mathcal{S}\mathcal{S}})$ where $\dot{y}^n_\mathcal{S} \triangleq \Sigma_{\mathcal{S}\mathcal{D}_n}\Sigma^{-1}_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}\left(y_{\mathcal{D}_n} - \mu_{\mathcal{D}_n}\right)$ and $\dot{\Sigma}^n_{\mathcal{S}\mathcal{S}} \triangleq \Sigma_{\mathcal{S}\mathcal{D}_n}\Sigma^{-1}_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}\Sigma_{\mathcal{D}_n\mathcal{S}}$ such that $\mu_{\mathcal{D}_n}$ is defined in a similar manner as $\mu_\mathcal{D}$ in* (1) *and $\Sigma_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}$ is a matrix with posterior covariance components $\sigma_{xx'|\mathcal{S}}$ for all $x, x' \in \mathcal{D}_n$, each of which is defined in a similar way as* (1).

[6] One of the N machines can be assigned to be the master.

Remark. Since the local summary is independent of the outputs $y_{\mathcal{S}}$, they need not be observed. So, the support set $\mathcal{S}$ does not have to be a subset of $\mathcal{D}$ and can be selected prior to data collection. Predictive performances of pPITC and pPIC are sensitive to the selection of $\mathcal{S}$. An informative support set $\mathcal{S}$ can be selected from domain $\mathcal{X}$ using an iterative greedy active selection procedure [16] prior to observing data.

STEP 3: MASTER CONSTRUCTS AND SENDS GLOBAL SUMMARY TO N MACHINES.

Definition 3 (Global Summary). *Given a common support set $\mathcal{S} \subset \mathcal{X}$ known to all N machines and the local summary $(\dot{y}^n_{\mathcal{S}}, \dot{\Sigma}^n_{\mathcal{SS}})$ of every machine $n = 1, \ldots, N$, the global summary is defined as a tuple $(\ddot{y}_{\mathcal{S}}, \ddot{\Sigma}_{\mathcal{SS}})$ where $\ddot{y}_{\mathcal{S}} \triangleq \sum_{n=1}^{N} \dot{y}^n_{\mathcal{S}}$ and $\ddot{\Sigma}_{\mathcal{SS}} \triangleq \Sigma_{\mathcal{SS}} + \sum_{n=1}^{N} \dot{\Sigma}^n_{\mathcal{SS}}$.*

STEP 4: DISTRIBUTE PREDICTIONS AMONG N MACHINES.

To predict the unobserved measurement for any set $\mathcal{U}$ of sampling units, $\mathcal{U}$ is partitioned evenly into disjoint subsets $\mathcal{U}_1, \ldots, \mathcal{U}_N$ to be assigned to the respective machines $1, \ldots, N$. So, $|\mathcal{U}_n| = |\mathcal{U}|/N$ for $n = 1, \ldots, N$.

Definition 4 (pPITC). *Given a common support set $\mathcal{S} \subset \mathcal{X}$ known to all N machines and the global summary $(\ddot{y}_{\mathcal{S}}, \ddot{\Sigma}_{\mathcal{SS}})$, each machine m computes a predictive Gaussian distribution $\mathcal{N}(\widehat{\mu}_x, \widehat{\sigma}_{xx})$ of the unobserved measurement for all sampling units $x \in \mathcal{U}_n$ where $\widehat{\mu}_x \triangleq \mu_x + \Sigma_{x\mathcal{S}} \ddot{\Sigma}^{-1}_{\mathcal{SS}} \ddot{y}_{\mathcal{S}}$ and $\widehat{\sigma}_{xx} \triangleq \sigma_{xx} - \Sigma_{x\mathcal{S}} \left(\Sigma^{-1}_{\mathcal{SS}} - \ddot{\Sigma}^{-1}_{\mathcal{SS}} \right) \Sigma_{\mathcal{S}x}$.*

Theorem 1. *Let a common support set $\mathcal{S} \subset \mathcal{X}$ be known to all N machines. Let $\mathcal{N}(\mu^{\text{PITC}}_{x|\mathcal{D}}, \sigma^{\text{PITC}}_{xx|\mathcal{D}})$ be the predictive Gaussian distribution computed by the centralized PITC approximation of FGP model [27] for all sampling units $x \in \mathcal{U}$ where*

$$\mu^{\text{PITC}}_{x|\mathcal{D}} \triangleq \mu_x + \Gamma_{x\mathcal{D}} \left(\Gamma_{\mathcal{DD}} + \Lambda\right)^{-1} \left(y_{\mathcal{D}} - \mu_{\mathcal{D}}\right) \text{ and } \sigma^{\text{PITC}}_{xx|\mathcal{D}} \triangleq \sigma_{xx} - \Gamma_{x\mathcal{D}} \left(\Gamma_{\mathcal{DD}} + \Lambda\right)^{-1} \Gamma_{\mathcal{D}x} \tag{2}$$

such that $\Gamma_{\mathcal{BB}'} \triangleq \Sigma_{\mathcal{BS}} \Sigma^{-1}_{\mathcal{SS}} \Sigma_{\mathcal{SB}'}$ for all $\mathcal{B}, \mathcal{B}' \subset \mathcal{X}$ and Λ is a block-diagonal matrix constructed from the N diagonal blocks of $\Sigma_{\mathcal{DD}|\mathcal{S}}$, each of which is a matrix $\Sigma_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}$ for $n = 1, \ldots, N$ where $\mathcal{D} = \bigcup_{n=1}^{N} \mathcal{D}_n$. Then, $\widehat{\mu}_x = \mu^{\text{PITC}}_{x|\mathcal{D}}$ and $\widehat{\sigma}_{xx} = \sigma^{\text{PITC}}_{xx|\mathcal{D}}$.

Remark. Since PITC generalizes the Bayesian Committee Machine (BCM) of [29], pPITC generalizes parallel BCM [14], the latter of which assumes the support set $\mathcal{S}$ to be $\mathcal{U}$ [27]. As a result, parallel BCM does not scale well with large $\mathcal{U}$. Similarly, since PITC reduces to the *fully independent training conditional* (FITC) approximation method when Λ is a diagonal matrix constructed from $\sigma_{x'x'|\mathcal{S}}$ for all $x' \in \mathcal{D}$ (i.e., $N = |\mathcal{D}|$), pPITC generalizes parallel FITC.

Though pPITC scales very well with large data [3], it can predict poorly due to (a) loss of information caused by summarizing the realized measurements and correlation structure of the original data; and (b) sparse coverage of $\mathcal{U}$ by the support set. We propose a novel *parallel partially independent conditional* (pPIC) approximation of FGP model that combines the best of both worlds, that is, the predictive power of FGP and time efficiency of pPITC. pPIC is based on the following intuition: A machine can exploit its local data to improve the predictions of unobserved measurements that are highly correlated with its data. At the same time, pPIC can preserve the time efficiency of pPITC by exploiting its idea of encapsulating information into local and global summaries. The predictive Gaussian distribution computed by pPIC on each machine is (a)

more complicated mathematically because, to avoid exploiting the local data twice, its contribution to the summary information has to be removed, and (b) proven to be equivalent to that of the centralized PIC approximation of FGP model [30]. Interested readers are referred to [3] for more details.

Remark 1. The above equivalence results imply that the computational load of the centralized PITC and PIC approximations of FGP can be distributed among N parallel machines, hence improving the time efficiency and scalability of approximate GP prediction. Supposing $|\mathcal{U}| < |\mathcal{D}|$ and $|\mathcal{S}| < |\mathcal{D}|$ for simplicity, the $\mathcal{O}(|\mathcal{S}|^2|\mathcal{D}| + |\mathcal{D}|(|\mathcal{D}|/N)^2)$ time incurred by PITC and $\mathcal{O}(|\mathcal{S}|^2|\mathcal{D}| + |\mathcal{D}|(|\mathcal{D}|/N)^2 + N|\mathcal{D}|)$ time incurred by PIC can, respectively, be reduced to $\mathcal{O}(|\mathcal{S}|^2(|\mathcal{S}| + N + |\mathcal{U}|/N) + (|\mathcal{D}|/N)^3)$ incurred by pPITC and $\mathcal{O}(|\mathcal{S}|^2(|\mathcal{S}| + N + |\mathcal{U}|/N) + (|\mathcal{D}|/N)^3 + |\mathcal{D}|)$ time incurred by pPIC, the latter of which scale better with increasing data size $|\mathcal{D}|$. The speedups of pPITC and pPIC over their centralized counterparts (a) deviate further from ideal speedup with more machines N due to their additional $\mathcal{O}(|\mathcal{S}|^2N)$ time, and (b) grow with increasing data size $|\mathcal{D}|$ because, unlike the additional $\mathcal{O}(|\mathcal{S}|^2|\mathcal{D}|)$ time of PITC and PIC that increase with more data, they do not have corresponding $\mathcal{O}(|\mathcal{S}|^2|\mathcal{D}|/M)$ terms.

Remark 2. The equivalence results also shed some light on the underlying properties of pPITC and pPIC based on the structural assumptions of PITC and PIC, respectively: pPITC assumes that $Y_{\mathcal{D}_1}, \ldots, Y_{\mathcal{D}_M}, Y_{\mathcal{U}_1}, \ldots, Y_{\mathcal{U}_M}$ are conditionally independent given $Y_{\mathcal{S}}$. In contrast, pPIC can predict the unobserved measurements $Y_{\mathcal{U}}$ better since it imposes a less restrictive assumption of conditional independence between $Y_{\mathcal{D}_1 \bigcup \mathcal{U}_1}, \ldots, Y_{\mathcal{D}_M \bigcup \mathcal{U}_M}$ given $Y_{\mathcal{S}}$. Experimental results on two real-world datasets [3] show that pPIC achieves predictive accuracy comparable to FGP and significantly better than pPITC, thus justifying the practicality of such an assumption.

Remark 3. Predictive performances of pPITC and pPIC are improved by increasing size of $\mathcal{S}$ at the expense of greater time, space, and communication complexity [3].

Experiments and Discussion. This section empirically evaluates the predictive performances, time efficiency, scalability, and speedups of our proposed parallel GPs against their centralized counterparts and FGP on a dataset of size $|\mathcal{D}| = 41850$ featuring a real-world traffic phenomenon, which contains traffic speeds (km/h) along 775 road segments of an urban road network (including highways, arterials, slip roads, etc.) during the morning peak hours on April 20, 2011. The traffic speeds are the measurements. The mean speed is 49.5 km/h and the standard deviation is 21.7 km/h. Each sampling unit (i.e., road segment) is specified by a 5-dimensional vector of features: length, number of lanes, speed limit, direction, and time. The time dimension comprises 54 five-minute time slots. This spatiotemporal traffic phenomenon is modeled using a relational GP (previously developed in [5]) whose correlation structure can exploit both the road segment features and road network topology information. 10% of the data is randomly selected as test data for predictions (i.e., as $\mathcal{U}$). Our experimental platform is a cluster of 20 computing nodes connected via gigabit links: Each node runs a Linux system with Intel® Xeon® CPU E5520 at 2.27 GHz and 20 GB memory. More details of our experimental setup can be found in [3].

Fig. 1 shows that, with $N = 20$ machines and data size $|\mathcal{D}| = 32000$, pPITC and pPIC incur 2-4 orders of magnitude less time than FGP while achieving comparable predictive performances (respectively, *root mean square error* (RMSE) differences of

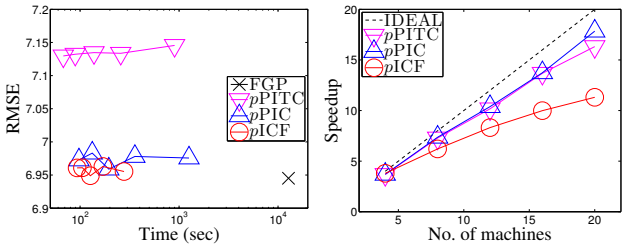


Fig. 1. Performance of parallel GP models with varying number $N = 4, 8, 12, 16, 20$ of machines, data size $|\mathcal{D}| = 32000$, and support set size $|\mathcal{S}| = 2048$. The *ideal* speedup of a parallel algorithm is defined to be the number N of machines running it.

less than 0.2 km/h and 0.05 km/h). Specifically, pPITC and pPIC incur only 1-2 minutes while FGP incurs more than 3.5 hours. Also, the speedups of pPITC and pPIC over their centralized counterparts deviate further from ideal speedup with more machines, as explained earlier. We have in fact proposed another parallel GP model in [3] called pICF that exploits parallel incomplete Cholesky factorization. For implementation details of pICF and more extensive experimental results, interested readers are referred to [3].

4 Generalized Online Sparse GP (GOSGP) Approximation

The key idea of our GOSGP approximation method [32] is to summarize the newly gathered data at regular time intervals/slices, assimilate the summary information of the new data with that of all the previously gathered data/observations, and then exploit the resulting assimilated summary information to compute a Gaussian predictive distribution of the unobserved measurement for any sampling unit. For simplicity, we omit the use of the known GP model parameters λ in our notations. Let $x_{1:t-1} \triangleq \{x_1, \ldots, x_{t-1}\}$ denote a set of sampling units from time steps 1 to $t-1$, each time slice n span time steps $(n-1)\tau + 1$ to $n\tau$ for some user-defined slice size $\tau \in \mathbb{Z}^+$, and the number of time slices available thus far up until time step t be denoted by N (i.e., $N\tau < t$).

Definition 5 (Slice Summary). *Given a support set $\mathcal{S} \subset \mathcal{X}$, a subset $\mathcal{D}_n \triangleq x_{(n-1)\tau+1:n\tau} \in x_{1:t-1}$ of sampling units associated with time slice n, and the column vector $z_{\mathcal{D}_n} = z_{(n-1)\tau+1:n\tau}$ of corresponding realized measurements, the slice summary of time slice n is defined as a tuple $(\mu^n_{\circledS}, \Sigma^n_{\circledS})$ for $n = 1, \ldots, N$ where $\mu^n_{\circledS} \triangleq \Sigma_{\mathcal{S}\mathcal{D}_n}\Sigma^{-1}_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}(z_{\mathcal{D}_n} - \mu_{\mathcal{D}_n})$ and $\Sigma^n_{\circledS} \triangleq \Sigma_{\mathcal{S}\mathcal{D}_n}\Sigma^{-1}_{\mathcal{D}_n\mathcal{D}_n|\mathcal{S}}\Sigma_{\mathcal{D}_n\mathcal{S}}$.*

Definition 6 (Assimilated Summary). *Given $(\mu^n_{\circledS}, \Sigma^n_{\circledS})$, the assimilated summary $(\mu^n_{@}, \Sigma^n_{@})$ of time slices 1 to n is updated from the assimilated summary $(\mu^{n-1}_{@}, \Sigma^{n-1}_{@})$ of time slices 1 to $n-1$ using $\mu^n_{@} \triangleq \mu^{n-1}_{@} + \mu^n_{\circledS}$ and $\Sigma^n_{@} \triangleq \Sigma^{n-1}_{@} + \Sigma^n_{\circledS}$ for $n = 1, \ldots, N$ where $\mu^0_{@} \triangleq 0$ and $\Sigma^0_{@} \triangleq \Sigma_{\mathcal{S}\mathcal{S}}$.*

Remark 1. After constructing and assimilating $(\mu^n_{\circledS}, \Sigma^n_{\circledS})$ with $(\mu^{n-1}_{@}, \Sigma^{n-1}_{@})$ to form $(\mu^n_{@}, \Sigma^n_{@})$, $\mathcal{D}_n = x_{(n-1)\tau+1:n\tau}$, $z_{\mathcal{D}_n} = z_{(n-1)\tau+1:n\tau}$, and $(\mu^n_{\circledS}, \Sigma^n_{\circledS})$ (Definition 5) are no longer needed and can be removed from memory. As a result, at time step t where $N\tau + 1 \leq t \leq (N+1)\tau$, only $(\mu^N_{@}, \Sigma^N_{@})$, $x_{N\tau+1:t-1}$, and $z_{N\tau+1:t-1}$ have to be kept in memory, thus requiring only constant memory (i.e., independent of t).

Remark 2. The slice summaries are constructed and assimilated at a regular time interval of τ, specifically, at time steps $N\tau + 1$ for $N \in \mathbb{Z}^+$.

Theorem 2. *Given $\mathcal{S} \subset \mathcal{X}$ and $(\mu_{@}^{N}, \Sigma_{@}^{N})$, our GOSGP approximation method computes a Gaussian predictive distribution $p(z_t|x_t, \mu_{@}^{N}, \Sigma_{@}^{N}) = \mathcal{N}(\widetilde{\mu}_{x_t}, \widetilde{\sigma}_{x_t x_t})$ of the measurement for any $x_t \in \mathcal{X}$ at time step t (i.e., $N\tau + 1 \leq t \leq (N+1)\tau$) where $\widetilde{\mu}_{x_t} \triangleq \mu_{x_t} + \Sigma_{x_t\mathcal{S}} \left(\Sigma_{@}^{N}\right)^{-1} \mu_{@}^{N}$ and $\widetilde{\sigma}_{x_t x_t} \triangleq \sigma_{x_t x_t} - \Sigma_{x_t\mathcal{S}} \left(\Sigma_{\mathcal{S}\mathcal{S}}^{-1} - \left(\Sigma_{@}^{N}\right)^{-1}\right) \Sigma_{\mathcal{S}x_t}$. If $t = N\tau + 1$, $\widetilde{\mu}_{x_t} = \mu^{\text{PITC}}_{x_t|x_{1:t-1}}$ and $\widetilde{\sigma}_{x_t x_t} = \sigma^{\text{PITC}}_{x_t x_t|x_{1:t-1}}$.* (3)

Remark 1. Theorem 2 implies that our GOSGP approximation method [32] is in fact equivalent to an online learning formulation/variant of the offline PITC [27]. Supposing $\tau < |\mathcal{S}|$, the $\mathcal{O}(t|\mathcal{S}|^2)$ time incurred by offline PITC can then be reduced to $\mathcal{O}(\tau|\mathcal{S}|^2)$ time (i.e., time independent of t) incurred by GOSGP [32] at time steps $t = N\tau + 1$ for $N \in \mathbb{Z}^+$ when slice summaries are constructed and assimilated. Otherwise, GOSGP [32] only incurs $\mathcal{O}(|\mathcal{S}|^2)$ time per time step.

Remark 2. The above equivalence result allows the structural property of GOSGP [32] to be elucidated using that of offline PITC: The measurements $Z_{\mathcal{D}_1}, \ldots, Z_{\mathcal{D}_N}, Z_{x_t}$ between different time slices are assumed to be conditionally independent given $Z_{\mathcal{S}}$. Such an assumption enables the data gathered during each time slice to be summarized independently of that in other time slices. Increasing slice size τ (i.e., less frequent assimilations of larger slice summaries) relaxes this conditional independence assumption (hence, potentially improving the predictive performance), but incurs more time at time steps when slice summaries are constructed and assimilated (see Remark 1).

Remark 3. Since offline PITC generalizes offline FITC, our GOSGP approximation method [32] generalizes the online learning variant of FITC (i.e., $\tau = 1$) [9].

When $N\tau+1 < t \leq (N+1)\tau$ (i.e., before the next slice summary of time slice $N+1$ is constructed and assimilated), the most recent observations (i.e., $\mathcal{D}' \triangleq x_{N\tau+1:t-1}$ and $z_{\mathcal{D}'} = z_{N\tau+1:t-1}$), which are often highly informative, are not used to update $\widetilde{\mu}_{x_t}$ and $\widetilde{\sigma}_{x_t x_t}$ (3). This may hurt the predictive performance when τ is large. To resolve this, we exploit incremental update formulas of Gaussian posterior mean and variance [32] to update $\widetilde{\mu}_{x_t}$ and $\widetilde{\sigma}_{x_t x_t}$ with the most recent observations, thereby yielding a Gaussian predictive distribution $p(z_t|x_t, \mu_{@}^{N}, \Sigma_{@}^{N}, \mathcal{D}', z_{\mathcal{D}'}) = \mathcal{N}(\widetilde{\mu}_{x_t|\mathcal{D}'}, \widetilde{\sigma}_{x_t x_t|\mathcal{D}'})$ where

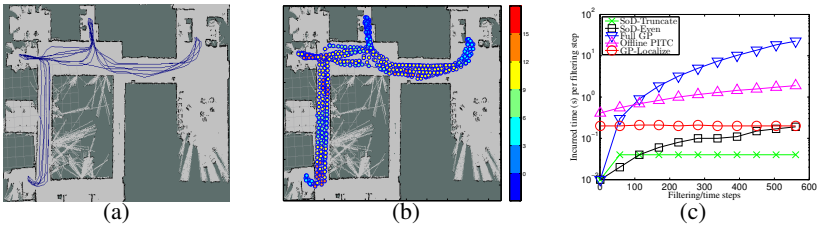
$$\widetilde{\mu}_{x_t|\mathcal{D}'} \triangleq \widetilde{\mu}_{x_t} + \widetilde{\Sigma}_{x_t\mathcal{D}'}\widetilde{\Sigma}^{-1}_{\mathcal{D}'\mathcal{D}'}\left(z_{\mathcal{D}'} - \widetilde{\mu}_{\mathcal{D}'}\right) \text{ and } \widetilde{\sigma}_{x_t x_t|\mathcal{D}'} \triangleq \widetilde{\sigma}_{x_t x_t} - \widetilde{\Sigma}_{x_t\mathcal{D}'}\widetilde{\Sigma}^{-1}_{\mathcal{D}'\mathcal{D}'}\widetilde{\Sigma}_{\mathcal{D}'x_t} \tag{4}$$

such that $\widetilde{\mu}_{\mathcal{D}'}$ is a column vector with mean components $\widetilde{\mu}_x$ (i.e., defined similarly to (3)) for all $x \in \mathcal{D}'$, $\widetilde{\Sigma}_{x_t\mathcal{D}'}$ is a row vector with covariance components $\widetilde{\sigma}_{x_t x}$ (i.e., defined similarly to (3)) for all $x \in \mathcal{D}'$, $\widetilde{\Sigma}_{\mathcal{D}'x_t}$ is the transpose of $\widetilde{\Sigma}_{x_t\mathcal{D}'}$, and $\widetilde{\Sigma}_{\mathcal{D}'\mathcal{D}'}$ is a matrix with covariance components $\widetilde{\sigma}_{xx'}$ (i.e., defined similarly to (3)) for all $x, x' \in \mathcal{D}'$.

Theorem 3. *Computing* (4) *incurs $\mathcal{O}(\tau|\mathcal{S}|^2)$ time at time steps $t = N\tau+1$ for $N \in \mathbb{Z}^+$ and $\mathcal{O}(|\mathcal{S}|^2)$ time otherwise. It requires $\mathcal{O}(|\mathcal{S}|^2)$ memory at each time step.*

So, GOSGP [32] incurs constant time and memory (i.e., independent of t) per time step.

Experiments and Discussion. In contrast to existing localization algorithms that train the GP observation model of a Bayes filter offline, GOSGP [32] is used to learn it *online* for persistent robot localization and the resulting algorithm is called *GP-Localize* [32]. The *adaptive Monte Carlo localization* (AMCL) package in ROS is run on a Pioneer 3-DX mobile robot mounted with a SICK LMS200 laser rangefinder to determine its trajectory (Fig. 2a) and the 561 locations at which the relative light measurements are



(a) (b) (c)

Fig. 2. (a) Pioneer 3-DX mobile robot trajectory of about 280 m in SMART FM IRG office/lab generated by AMCL package in ROS, along which (b) 561 relative light (%) observations/data are gathered at locations denoted by small colored circles. (c) Graphs of incurred time (s) per time step vs. number of time steps comparing different GP localization algorithms.

taken using a weather board (Fig. 2b); these locations are assumed to be ground truth. For empirical evaluation of GP-Localize with other real-world datasets, refer to [32].

The localization error (i.e., distance between the robot's estimated and true locations) and scalability of GP-Localize are compared to that of two sparse GP localization algorithms [32]: (a) The *Subset of Data (SoD)-Truncate* method uses $|\mathcal{S}| = 10$ most recent observations (i.e., compared to $|\mathcal{D}'| < \tau = 10$ most recent observations considered by GOSGP [32] besides the assimilated summary) as training data at each time step while (b) the *SoD-Even* method uses $|\mathcal{S}| = 40$ observations (i.e., compared to the support set of $|\mathcal{S}| = 40$ possibly unobserved locations selected *prior* to localization and exploited by GOSGP [32]) evenly distributed over the time of localization. The scalability of GP-Localize is further compared to that of GP localization algorithms employing full GP (FGP) and offline PITC. GP-Localize, SoD-Truncate, and SoD-Even achieve, respectively, localization errors of 2.1 m, 5.4 m, and 4.6 m averaged over all 561 time steps and 3 runs. Fig. 2c shows the time incurred by GP-Localize, SoD-Truncate, SoD-Even, FGP, and offline PITC at each time step. GP-Localize is clearly much more scalable (i.e., constant time) than FGP and offline PITC. Though it incurs slightly more time than SoD-Truncate and SoD-Even, it can localize significantly better.

5 Nonmyopic ϵ-Bayes-Optimal Active Sensing/Learning

Problem Formulation. To cast active sensing as a Bayesian sequential decision problem, we define a sequential active sensing policy $\pi \triangleq \{\pi_n\}_{n=1}^{N}$ that is structured to sequentially decide the next location $\pi_n(z_{\mathcal{D}}) \in \mathcal{X} \setminus \mathcal{D}$ to be observed at each stage n based on the current observations $z_{\mathcal{D}}$ over a finite planning horizon of N stages (i.e., sampling budget). To measure the predictive uncertainty over unobserved areas of the phenomenon, we use the entropy criterion and define the value under a policy π to be the joint entropy of its selected observations when starting with some prior observations $z_{\mathcal{D}_0}$ and following π thereafter [13]. The work of [19] has established that minimizing the posterior joint entropy (i.e., predictive uncertainty) remaining in unobserved locations of the phenomenon is equivalent to maximizing the joint entropy of π. Thus, solving the active sensing problem entails choosing a sequential BAL policy $\pi_n^*(z_{\mathcal{D}}) = \arg\max_{x\in\mathcal{X}\setminus\mathcal{D}} Q_n^*(z_{\mathcal{D}}, x)$ induced from the following N-stage Bellman equations, as formally derived in [13]:

$$\begin{array}{rcl} V_n^*(z_{\mathcal{D}}) & \triangleq & \max\limits_{x\in\mathcal{X}\setminus\mathcal{D}} Q_n^*(z_{\mathcal{D}}, x) \\ Q_n^*(z_{\mathcal{D}}, x) & \triangleq & \mathbb{E}\left[-\log p(Z_x|x,\mathcal{D},z_{\mathcal{D}})\right] + \mathbb{E}\left[V_{n+1}^*(z_{\mathcal{D}}\cup\{Z_x\})\,|x,\mathcal{D},z_{\mathcal{D}}\right] \end{array} \quad (5)$$

for stage $n = 1, \ldots, N$ where $p(z_x|x, \mathcal{D}, z_\mathcal{D})$ is defined in Section 2 and the second expectation term is omitted from right-hand side expression of Q_N^* at stage N. Unfortunately, since the BAL policy π^* cannot be derived exactly, we instead consider solving for an ϵ-BAL policy π^ϵ whose joint entropy approximates that of π^* within $\epsilon > 0$.

ϵ-BAL Policy. The key idea of our nonmyopic ϵ-BAL policy π^ϵ is to approximate the expectation terms in (5) at every stage using truncated sampling. Specifically, given realized measurements $z_\mathcal{D}$, a finite set of τ-truncated, i.i.d. observations $\{z_x^i\}_{i=1}^S$ [13] is generated and exploited for approximating V_n^* (5) through the following Bellman equations:

$$\begin{aligned} V_n^\epsilon(z_\mathcal{D}) &\triangleq \max_{x \in \mathcal{X} \setminus \mathcal{D}} Q_n^\epsilon(z_\mathcal{D}, x) \\ Q_n^\epsilon(z_\mathcal{D}, x) &\triangleq \frac{1}{S} \sum_{i=1}^{S} -\log p\left(z_x^i | x, \mathcal{D}, z_\mathcal{D}\right) + V_{n+1}^\epsilon\left(z_\mathcal{D} \cup \left\{z_x^i\right\}\right) \end{aligned} \tag{6}$$

for stage $n = 1, \ldots, N$. The use of truncation is motivated by a technical necessity for theoretically guaranteeing the *expected* active sensing performance (specifically, ϵ-Bayes-optimality) of π^ϵ relative to that of π^* [13].

Anytime ϵ-BAL ($\langle\alpha, \epsilon\rangle$-BAL) Algorithm. Although π^ϵ can be derived exactly, the cost of deriving it is exponential in the length N of planning horizon since it has to compute the values $V_n^\epsilon(z_\mathcal{D})$ (6) for all $(S|\mathcal{X}|)^N$ possible states $(n, z_\mathcal{D})$. To ease this computational burden, we propose an anytime algorithm based on ϵ-BAL that can produce a good policy fast and improve its approximation quality over time. The key intuition behind our *anytime* ϵ-BAL *algorithm* ($\langle\alpha, \epsilon\rangle$-BAL) is to focus the simulation of greedy exploration paths through the most uncertain regions of the state space (i.e., in terms of the values $V_n^\epsilon(z_\mathcal{D})$) instead of evaluating the entire state space like π^ϵ. Interested readers are referred to [13] for more details.

Experiments and Discussion. This section evaluates the active sensing performance and time efficiency of our $\langle\alpha, \epsilon\rangle$-BAL policy $\pi^{\langle\alpha,\epsilon\rangle}$ empirically under using a real-world dataset of a large-scale traffic phenomenon (i.e., speeds of road segments) over an urban road network; refer to [13] for additional experimental results on a simulated spatial phenomenon. Fig. 3a shows the urban road network $\mathcal{X}$ comprising 775 road segments in Tampines area, Singapore during lunch hours on June 20, 2011. Each road segment $x \in \mathcal{X}$ is specified by a 4D vector of features: length, number of lanes, speed limit, and direction. More details of our experimental setup can be found in [13].

The performance of our $\langle\alpha, \epsilon\rangle$-BAL policies with planning horizon length $N' = 3, 4, 5$ are compared to that of APGD and IE policies [15] by running each of them on a mobile robotic probe to direct its active sensing along a path of adjacent road segments according to the road network topology. Fig. 3 shows results of the tested policies averaged over 5 independent runs: It can be observed from Fig. 3b that our $\langle\alpha, \epsilon\rangle$-BAL policies outperform APGD and IE policies due to their nonmyopic exploration behavior. Fig. 3c shows that $\langle\alpha, \epsilon\rangle$-BAL incurs < 4.5 hours given a budget of $N = 240$ road segments, which can be afforded by modern computing power. To illustrate the behavior of each policy, Figs. 3d-f show, respectively, the road segments observed (shaded in black) by the mobile probe running APGD, IE, and $\langle\alpha, \epsilon\rangle$-BAL policies with $N' = 5$ given a budget of $N = 60$. Interestingly, Figs. 3d-e show that both APGD and IE cause the probe to move away from the slip roads and highways to low-speed segments

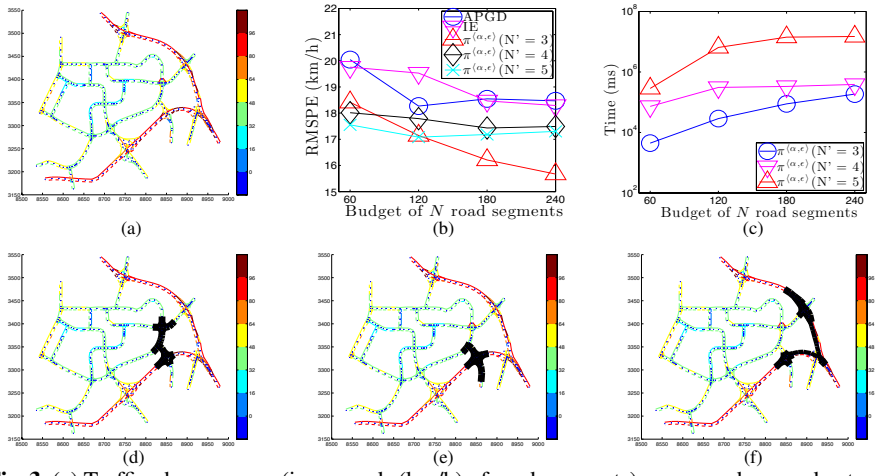


Fig. 3. (a) Traffic phenomenon (i.e., speeds (km/h) of road segments) over an urban road network, graphs of (b) *root mean square prediction error* (RMSPE) of APGD, IE, and $\langle\alpha, \epsilon\rangle$-BAL policies with horizon length $N' = 3, 4, 5$ and (c) total online processing cost of $\langle\alpha, \epsilon\rangle$-BAL policies with $N' = 3, 4, 5$ vs. budget of N segments, and (d-f) road segments observed (shaded in black) by respective APGD, IE, and $\langle\alpha, \epsilon\rangle$-BAL policies ($N' = 5$) with $N = 60$.

whose measurements vary much more smoothly; this is expected due to their myopic exploration behavior. In contrast, $\langle\alpha, \epsilon\rangle$-BAL nonmyopically plans the probe's path and direct it to observe the more informative slip roads and highways with highly varying traffic measurements (Fig. 3f) to achieve better performance.

Acknowledgments. This work was supported by the Singapore-MIT Alliance for Research & Technology Subaward Agreements No. 41 and No. 52.

References

1. Cao, N., Low, K.H., Dolan, J.M.: Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms. In: Proc. AAMAS. pp. 7–14 (2013)
2. Chao, Y., Li, Z., Farrara, J.D., Hung, P.: Blending sea surface temperatures from multiple satellites and in situ observations for coastal oceans. J. Atmos. Oceanic Technol. 26(7), 1415–1426 (2009)
3. Chen, J., Cao, N., Low, K.H., Ouyang, R., Tan, C.K.Y., Jaillet, P.: Parallel Gaussian process regression with low-rank covariance matrix approximations. In: Proc. UAI (2013)
4. Chen, J., Low, K.H., Tan, C.K.Y.: Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In: Proc. RSS (2013)
5. Chen, J., Low, K.H., Tan, C.K.Y., Oran, A., Jaillet, P., Dolan, J.M., Sukhatme, G.S.: Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In: Proc. UAI. pp. 163–173 (2012)
6. Choudhury, A., Nair, P.B., Keane, A.J.: A data parallel approach for large-scale Gaussian process modeling. In: Proc. SDM. pp. 95–111 (2002)
7. Cressie, N., Johannesson, G.: Fixed rank kriging for very large spatial data sets. J. R. Statist. Soc. B 70(1), 209–226 (2008)
8. Cressie, N., Wikle, C.K.: Statistics for Spatio-Temporal Data. Wiley (2011)
9. Csató, L., Opper, M.: Sparse online Gaussian processes. Neural Comput. 14, 641–669 (2002)
10. Dolan, J.M., Podnar, G., Stancliff, S., Low, K.H., Elfes, A., Higinbotham, J., Hosler, J.C., Moisan, T.A., Moisan, J.: Cooperative aquatic sensing using the telesupervised adaptive ocean sensor fleet. In: Proc. SPIE Conference on Remote Sensing of the Ocean, Sea Ice, and Large Water Regions. vol. 7473 (2009)

11. Furrer, R., Genton, M.G., Nychka, D.: Covariance tapering for interpolation of large spatial datasets. JCGS 15(3), 502–523 (2006)
12. Hoang, T.N., Low, K.H., Jaillet, P., Kankanhalli, M.: Active learning is planning: Nonmyopic ϵ-Bayes-optimal active learning of Gaussian processes. In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) Proc. ECML/PKDD Nectar Track. pp. 494–498. LNCS 8726, Springer, Heidelberg (2014)
13. Hoang, T.N., Low, K.H., Jaillet, P., Kankanhalli, M.: Nonmyopic ϵ-Bayes-optimal active learning of Gaussian processes. In: Proc. ICML. pp. 739–747 (2014)
14. Ingram, B., Cornford, D.: Parallel geostatistics for sparse and dense datasets. In: Atkinson, P.M., Lloyd, C.D. (eds.) Proc. geoENV VII. pp. 371–381. Quantitative Geology and Geostatistics Volume 16, Springer, Netherlands (2010)
15. Krause, A., Guestrin, C.: Nonmyopic active learning of Gaussian processes: An exploration-exploitation approach. In: Proc. ICML. pp. 449–456 (2007)
16. Krause, A., Singh, A., Guestrin, C.: Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. JMLR 9, 235–284 (2008)
17. Leonard, N.E., Palley, D.A., Lekien, F., Sepulchre, R., Fratantoni, D.M., Davis, R.E.: Collective motion, sensor networks, and ocean sampling. Proc. IEEE 95(1), 48–74 (2007)
18. Low, K.H., Dolan, J.M., Khosla, P.: Adaptive multi-robot wide-area exploration and mapping. In: Proc. AAMAS. pp. 23–30 (2008)
19. Low, K.H., Dolan, J.M., Khosla, P.: Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing. In: Proc. ICAPS. pp. 233–240 (2009)
20. Low, K.H., Dolan, J.M., Khosla, P.: Active Markov information-theoretic path planning for robotic environmental sensing. In: Proc. AAMAS. pp. 753–760 (2011)
21. Low, K.H., Podnar, G., Stancliff, S., Dolan, J.M., Elfes, A.: Robot boats as a mobile aquatic sensor network. In: Proc. IPSN-09 Workshop on Sensor Networks for Earth and Space Science Applications (2009)
22. Low, K.H., Xu, N., Chen, J., Lim, K.K., Özgül, E.B.: Generalized online sparse Gaussian processes with application to persistent mobile robot localization. In: Calders, T., Esposito, F., Hüllermeier, E., Meo, R. (eds.) Proc. ECML/PKDD Nectar Track. pp. 499–503. LNCS 8726, Springer, Heidelberg (2014)
23. Martin, R.J.: Comparing and contrasting some environmental and experimental design problems. Environmetrics 12(3), 303–317 (2001)
24. Ouyang, R., Low, K.H., Chen, J., Jaillet, P.: Multi-robot active sensing of non-stationary Gaussian process-based environmental phenomena. In: Proc. AAMAS. pp. 573–580 (2014)
25. Park, C., Huang, J.Z., Ding, Y.: Domain decomposition approach for fast Gaussian process regression of large spatial data sets. JMLR 12, 1697–1728 (2011)
26. Podnar, G., Dolan, J.M., Low, K.H., Elfes, A.: Telesupervised remote surface water quality sensing. In: Proc. IEEE Aerospace Conference (2010)
27. Quiñonero-Candela, J., Rasmussen, C.E.: A unifying view of sparse approximate Gaussian process regression. JMLR 6, 1939–1959 (2005)
28. Sang, H., Huang, J.Z.: A full scale approximation of covariance functions for large spatial data sets. J. R. Statist. Soc. B 74(1), 111–132 (2012)
29. Schwaighofer, A., Tresp, V.: Transductive and inductive methods for approximate Gaussian process regression. In: Proc. NIPS. pp. 953–960 (2002)
30. Snelson, E.: Local and global sparse Gaussian process approximations. In: Proc. AISTATS (2007)
31. Wikle, C.K.: Low-rank representations for spatial processes. In: Gelfand, A.E., Diggle, P., Guttorp, P., Fuentes, M. (eds.) Handbook of Spatial Statistics, pp. 107–118 (2010)
32. Xu, N., Low, K.H., Chen, J., Lim, K.K., Özgül, E.B.: GP-Localize: Persistent mobile robot localization using online sparse Gaussian process observation model. In: Proc. AAAI. pp. 2585–2592 (2014)