Gaussian Process Planning with Lipschitz Continuous Reward Functions: Towards Unifying Bayesian Optimization, Active Learning, and Beyond

Chun Kai Ling[*] and **Kian Hsiang Low**[*] and **Patrick Jaillet**[†]
Department of Computer Science, National University of Singapore, Republic of Singapore[*]
Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, USA[†]
{chunkai, lowkh}@comp.nus.edu.sg[*], jaillet@mit.edu[†]

Abstract

This paper presents a novel nonmyopic adaptive *Gaussian process planning* (GPP) framework endowed with a general class of Lipschitz continuous reward functions that can unify some active learning/sensing and Bayesian optimization criteria and offer practitioners some flexibility to specify their desired choices for defining new tasks/problems. In particular, it utilizes a principled Bayesian sequential decision problem framework for jointly and naturally optimizing the exploration-exploitation trade-off. In general, the resulting induced GPP policy cannot be derived exactly due to an uncountable set of candidate observations. A key contribution of our work here thus lies in exploiting the Lipschitz continuity of the reward functions to solve for a nonmyopic adaptive *ϵ-optimal GPP* (ϵ-GPP) policy. To plan in real time, we further propose an asymptotically optimal, branch-and-bound anytime variant of ϵ-GPP with performance guarantee. We empirically demonstrate the effectiveness of our ϵ-GPP policy and its anytime variant in Bayesian optimization and an energy harvesting task.

1 Introduction

The fundamental challenge of integrated planning and learning is to design an autonomous agent that can plan its actions to maximize its expected total rewards while interacting with an unknown task environment. Recent research efforts tackling this challenge have progressed from the use of simple Markov models assuming discrete-valued, independent observations (e.g., in *Bayesian reinforcement learning* (BRL) (Poupart et al. 2006)) to that of a rich class of Bayesian nonparametric *Gaussian process* (GP) models characterizing continuous-valued, correlated observations in order to represent the latent structure of more complex, possibly noisy task environments with higher fidelity. Such a challenge is posed by the following important problems in machine learning, among others:

Active learning/sensing (AL). In the context of environmental sensing (e.g., adaptive sampling in oceanography (Leonard et al. 2007), traffic sensing (Chen et al. 2012; Chen, Low, and Tan 2013; Chen et al. 2015)), its objective is to select the most informative (possibly noisy) observations for predicting a spatially varying environmental field (i.e., task environment) modeled by a GP subject to some sampling budget constraints (e.g., number of sensors, energy consumption). The rewards of an AL agent are defined based on some formal measure of predictive uncertainty such as the entropy or mutual information criterion. To resolve the issue of sub-optimality (i.e., local maxima) faced by greedy algorithms (Krause, Singh, and Guestrin 2008; Low et al. 2012; Ouyang et al. 2014; Zhang et al. 2016), recent developments have made nonmyopic AL computationally tractable with provable performance guarantees (Cao, Low, and Dolan 2013; Hoang et al. 2014; Low, Dolan, and Khosla 2009; 2008; 2011), some of which have further investigated the performance advantage of adaptivity by proposing nonmyopic adaptive observation selection policies that depend on past observations.

Bayesian optimization (BO). Its objective is to select and gather the most informative (possibly noisy) observations for finding the global maximum of an unknown, highly complex (e.g., non-convex, no closed-form expression nor derivative) objective function (i.e., task environment) modeled by a GP given a sampling budget (e.g., number of costly function evaluations). The rewards of a BO agent are defined using an improvement-based (Brochu, Cora, and de Freitas 2010) (e.g., *probability of improvement* (PI) or *expected improvement* (EI) over currently found maximum), entropy-based (Hennig and Schuler 2012; Hernández-Lobato, Hoffman, and Ghahramani 2014), or *upper confidence bound* (UCB) acquisition function (Srinivas et al. 2010). A limitation of most BO algorithms is that they are myopic. To overcome this limitation, approximation algorithms for nonmyopic adaptive BO (Marchant, Ramos, and Sanner 2014; Osborne, Garnett, and Roberts 2009) have been proposed, but their performances are not theoretically guaranteed.

General tasks/problems. In practice, other types of rewards (e.g., logarithmic, unit step functions) need to be specified for an agent to plan and operate effectively in a given real-world task environment (e.g., natural phenomenon like wind or temperature) modeled by a GP, as detailed in Section 2.

As shall be elucidated later, similarities in the structure of the above problems motivate us to consider whether it is possible to tackle the overall challenge by devising a nonmyopic adaptive GP planning framework with a general class of reward functions unifying some AL and BO criteria and affording practitioners some flexibility to specify their desired choices for defining new tasks/problems. Such an integrated planning and learning framework has to address the exploration-exploitation trade-off common to the above problems: The agent faces a dilemma between gathering observations to maximize its expected total rewards given its current, possibly imprecise belief of the task environment (exploitation) vs. that to improve its belief to learn more about the environment (exploration).

This paper presents a novel nonmyopic adaptive *Gaussian process planning* (GPP) framework endowed with a general class of Lipschitz continuous reward functions that can unify some AL and BO criteria (e.g., UCB) discussed earlier and offer practitioners some flexibility to specify their desired choices for defining new tasks/problems (Section 2). In particular, it utilizes a principled Bayesian sequential decision problem framework for jointly and naturally optimizing the exploration-exploitation trade-off, consequently allowing planning and learning to be integrated seamlessly and performed simultaneously instead of separately (Deisenroth, Fox, and Rasmussen 2015). In general, the resulting induced GPP policy cannot be derived exactly due to an uncountable set of candidate observations. A key contribution of our work here thus lies in exploiting the Lipschitz continuity of the reward functions to solve for a nonmyopic adaptive *ϵ-optimal GPP* (ϵ-GPP) policy given an arbitrarily user-specified loss bound ϵ (Section 3). To plan in real time, we further propose an asymptotically optimal, branch-and-bound anytime variant of ϵ-GPP with performance guarantee. Finally, we empirically evaluate the performances of our ϵ-GPP policy and its anytime variant in BO and an energy harvesting task on simulated and real-world environmental fields (Section 4). To ease exposition, the rest of this paper will be described by assuming the task environment to be an environmental field and the agent to be a mobile robot, which coincide with the setup of our experiments.

2 Gaussian Process Planning (GPP)

Notations and Preliminaries. Let $\mathcal{S}$ be the domain of an environmental field corresponding to a set of sampling locations. At time step $t > 0$, a robot can deterministically move from its previous location s_{t-1} to visit location $s_t \in \mathcal{A}(s_{t-1})$ and observes it by taking a corresponding realized (random) field measurement z_t (Z_t) where $\mathcal{A}(s_{t-1}) \subseteq \mathcal{S}$ denotes a finite set of sampling locations reachable from its previous location s_{t-1} in a single time step. The state of the robot at its initial starting location s_0 is represented by prior observations/data $d_0 \triangleq \langle \mathbf{s}_0, \mathbf{z}_0 \rangle$ available before planning where $\mathbf{s}_0$ and $\mathbf{z}_0$ denote, respectively, vectors comprising locations visited/observed and corresponding field measurements taken by the robot prior to planning and s_0 is the last component of $\mathbf{s}_0$. Similarly, at time step $t > 0$, the state of the robot at its current location s_t is represented by observations/data $d_t \triangleq \langle \mathbf{s}_t, \mathbf{z}_t \rangle$ where $\mathbf{s}_t \triangleq \mathbf{s}_0 \oplus (s_1, \cdots s_t)$ and $\mathbf{z}_t \triangleq \mathbf{z}_0 \oplus (z_1, \cdots z_t)$ denote, respectively, vectors comprising locations visited/observed and corresponding field measurements taken by the robot up until time step t and '$\oplus$' denotes vector concatenation. At time step $t > 0$, the robot also receives a reward $R(z_t, \mathbf{s}_t)$ to be defined later.

Modeling Environmental Fields with Gaussian Processes (GPs). The GP can be used to model a spatially varying environmental field as follows: The field is assumed to be a realization of a GP. Each location $s \in \mathcal{S}$ is associated with a latent field measurement Y_s. Let $Y_{\mathcal{S}} \triangleq \{Y_s\}_{s \in \mathcal{S}}$ denote a GP, that is, every finite subset of $Y_{\mathcal{S}}$ has a multivariate Gaussian distribution (Rasmussen and Williams 2006). Then, the GP is fully specified by its *prior* mean $\mu_s \triangleq \mathbb{E}[Y_s]$ and covariance $k_{ss'} \triangleq \text{cov}[Y_s, Y_{s'}]$ for all $s, s' \in \mathcal{S}$, the latter of which characterizes the spatial correlation structure of the environment field and can be defined using a covariance function. A common choice is the squared exponential covariance function $k_{ss'} \triangleq \sigma_y^2 \exp\{-0.5(s-s')^\top M^{-2}(s-s')\}$ where σ_y^2 is the signal variance controlling the intensity of measurements and M is a diagonal matrix with length-scale components l_1 and l_2 governing the degree of spatial correlation or "similarity" between measurements in the respective horizontal and vertical directions of the 2D fields in our experiments.

The field measurements taken by the robot are assumed to be corrupted by Gaussian white noise, i.e., $Z_t \triangleq Y_{s_t} + \varepsilon$ where $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$ and σ_n^2 is the noise variance. Supposing the robot has gathered observations $d_t = \langle \mathbf{s}_t, \mathbf{z}_t \rangle$ from time steps 0 to t, the GP model can perform probabilistic regression by using d_t to predict the noisy measurement at any unobserved location $s_{t+1} \in \mathcal{A}(s_t)$ as well as provide its predictive uncertainty using a Gaussian predictive distribution $p(z_{t+1}|d_t, s_{t+1}) = \mathcal{N}(\mu_{s_{t+1}|d_t}, \sigma^2_{s_{t+1}|\mathbf{s}_t})$ with the following *posterior* mean and variance, respectively:

$$\mu_{s_{t+1}|d_t} \triangleq \mu_{s_{t+1}} + \Sigma_{s_{t+1}\mathbf{s}_t}\Gamma^{-1}_{\mathbf{s}_t\mathbf{s}_t}(\mathbf{z}_t - \mu_{\mathbf{s}_t})^\top$$
$$\sigma^2_{s_{t+1}|\mathbf{s}_t} \triangleq k_{s_{t+1}s_{t+1}} + \sigma_n^2 - \Sigma_{s_{t+1}\mathbf{s}_t}\Gamma^{-1}_{\mathbf{s}_t\mathbf{s}_t}\Sigma_{\mathbf{s}_t s_{t+1}}$$

where $\mu_{\mathbf{s}_t}$ is a row vector with mean components μ_s for every location s of $\mathbf{s}_t$, $\Sigma_{s_{t+1}\mathbf{s}_t}$ is a row vector with covariance components $k_{s_{t+1}s}$ for every location s of $\mathbf{s}_t$, $\Sigma_{\mathbf{s}_t s_{t+1}}$ is the transpose of $\Sigma_{s_{t+1}\mathbf{s}_t}$, and $\Gamma_{\mathbf{s}_t\mathbf{s}_t} \triangleq \Sigma_{\mathbf{s}_t\mathbf{s}_t} + \sigma_n^2 I$ such that $\Sigma_{\mathbf{s}_t\mathbf{s}_t}$ is a covariance matrix with components $k_{ss'}$ for every pair of locations s, s' of $\mathbf{s}_t$. An important property of the GP model is that, unlike $\mu_{s_{t+1}|d_t}$, $\sigma^2_{s_{t+1}|\mathbf{s}_t}$ is independent of $\mathbf{z}_t$.

Problem Formulation. To frame nonmyopic adaptive *Gaussian process planning* (GPP) as a Bayesian sequential decision problem, let an adaptive policy π be defined to sequentially decide the next location $\pi(d_t) \in \mathcal{A}(s_t)$ to be observed at each time step t using observations d_t over a finite planning horizon of H time steps/stages (i.e., sampling budget of H locations). The value $V_0^\pi(d_0)$ under an adaptive policy π is defined to be the expected total rewards achieved by its selected observations when starting with some prior observations d_0 and following π thereafter and can be computed using the following H-stage Bellman equations:

$$V_t^\pi(d_t) \triangleq Q_t^\pi(d_t, \pi(d_t))$$
$$Q_t^\pi(d_t, s_{t+1}) \triangleq \mathbb{E}[R(Z_{t+1}, \mathbf{s}_{t+1}) + V_{t+1}^\pi(\langle \mathbf{s}_{t+1}, \mathbf{z}_t \oplus Z_{t+1} \rangle)|d_t, s_{t+1}]$$

for stages $t = 0, \ldots, H-1$ where $V_H^\pi(d_H) \triangleq 0$. To solve the GPP problem, the notion of Bayes-optimalityis exploited for selecting observations to achieve the largest possible expected total rewards with respect to all possible induced sequences of future Gaussian posterior beliefs $p(z_{t+1}|d_t, s_{t+1})$ for $t = 0, \ldots, H-1$ to be discussed next.

Formally, this involves choosing an adaptive policy π to maximize $V_0^\pi(d_0)$, which we call the GPP policy π^*. That is, $V_0^*(d_0) \triangleq V_0^{\pi^*}(d_0) = \max_\pi V_0^\pi(d_0)$. By plugging π^* into $V_t^\pi(d_t)$ and $Q_t^\pi(d_t, s_{t+1})$ above,

$$\begin{aligned} V_t^*(d_t) &\triangleq \max_{s_{t+1}\in\mathcal{A}(s_t)} Q_t^*(d_t, s_{t+1}) \\ Q_t^*(d_t, s_{t+1}) &\triangleq \mathbb{E}[R(Z_{t+1}, \mathbf{s}_{t+1})|d_t, s_{t+1}] + \\ &\quad\ \mathbb{E}[V_{t+1}^*(\langle \mathbf{s}_{t+1}, \mathbf{z}_t \oplus Z_{t+1}\rangle)|d_t, s_{t+1}] \end{aligned} \tag{1}$$

for stages $t = 0, \ldots, H-1$ where $V_H^*(d_H) \triangleq 0$. To see how the GPP policy π^* jointly and naturally optimizes the exploration-exploitation trade-off, its selected location $\pi^*(d_t) = \arg\max_{s_{t+1}\in\mathcal{A}(s_t)} Q_t^*(d_t, s_{t+1})$ at each time step t affects both the immediate expected reward $\mathbb{E}[R(Z_{t+1}, \mathbf{s}_t \oplus \pi^*(d_t))|d_t, \pi^*(d_t)]$ given current belief $p(z_{t+1}|d_t, \pi^*(d_t))$ (i.e., exploitation) as well as the Gaussian posterior belief $p(z_{t+2}|\langle \mathbf{s}_t \oplus \pi^*(d_t), \mathbf{z}_t \oplus z_{t+1}\rangle, \pi^*(\langle \mathbf{s}_t \oplus \pi^*(d_t), \mathbf{z}_t \oplus z_{t+1}\rangle))$ at next time step $t+1$ (i.e., exploration), the latter of which influences expected future rewards $\mathbb{E}[V_{t+1}^*(\langle \mathbf{s}_t \oplus \pi^*(d_t), \mathbf{z}_t \oplus Z_{t+1}\rangle)|d_t, \pi^*(d_t)]$.

In general, the GPP policy π^* cannot be derived exactly because the expectation terms in (1) usually cannot be evaluated in closed form due to an uncountable set of candidate measurements (Section 1) except for degenerate cases like $R(z_{t+1}, \mathbf{s}_{t+1})$ being independent of z_{t+1} and $H \leq 2$. To overcome this difficulty, we will show in Section 3 later how the Lipschitz continuity of the reward functions can be exploited for theoretically guaranteeing the performance of our proposed nonmyopic adaptive ϵ-optimal GPP policy, that is, the expected total rewards achieved by its selected observations closely approximates that of π^* within an arbitrarily user-specified loss bound $\epsilon > 0$.

Lipschitz Continuous Reward Functions. $R(z_t, \mathbf{s}_t) \triangleq R_1(z_t) + R_2(z_t) + R_3(\mathbf{s}_t)$ where R_1, R_2, and R_3 are user-defined reward functions that satisfy the conditions below:

- $R_1(z_t)$ is Lipschitz continuous in z_t with Lipschitz constant ℓ_1. So, $h_\sigma(u) \triangleq (R_1 * \mathcal{N}(0, \sigma^2))(u)$ is Lipschitz continuous in u with ℓ_1 where '$*$' denotes convolution;
- $R_2(z_t)$: Define $g_\sigma(u) \triangleq (R_2 * \mathcal{N}(0, \sigma^2))(u)$ such that (a) $g_\sigma(u)$ is well-defined for all $u \in \mathbb{R}$, (b) $g_\sigma(u)$ can be evaluated in closed form or computed up to an arbitrary precision in reasonable time for all $u \in \mathbb{R}$, and (c) $g_\sigma(u)$ is Lipschitz continuous[1] in u with Lipschitz constant $\ell_2(\sigma)$;
- $R_3(\mathbf{s}_t)$ only depends on locations $\mathbf{s}_t$ visited/observed by the robot up until time step t and is independent of realized measurement z_t. It can be used to represent some sampling or motion costs or explicitly consider exploration by defining it as a function of $\sigma^2_{s_{t+1}|\mathbf{s}_t}$.

Using the above definition of $R(z_t, \mathbf{s}_t)$, the immediate expected reward in (1) evaluates to $\mathbb{E}[R(Z_{t+1}, \mathbf{s}_{t+1})|d_t, s_{t+1}] = (h_{\sigma_{s_{t+1}|\mathbf{s}_t}} + g_{\sigma_{s_{t+1}|\mathbf{s}_t}})(\mu_{s_{t+1}|d_t}) + R_3(\mathbf{s}_{t+1})$ which is Lipschitz continuous in the realized measurements $\mathbf{z}_t$:

Lemma 1 *Let* $\alpha(\mathbf{s}_{t+1}) \triangleq \|\Sigma_{s_{t+1}\mathbf{s}_t}\Gamma^{-1}_{\mathbf{s}_t\mathbf{s}_t}\|$ *and* $d'_t \triangleq \langle \mathbf{s}_t, \mathbf{z}'_t\rangle$. *Then,* $|\mathbb{E}[R(Z_{t+1},\mathbf{s}_{t+1})|d_t,s_{t+1}] - \mathbb{E}[R(Z_{t+1},\mathbf{s}_{t+1})|d'_t,s_{t+1}]| \leq \alpha(\mathbf{s}_{t+1})\left(\ell_1 + \ell_2(\sigma_{s_{t+1}|\mathbf{s}_t})\right)\|\mathbf{z}_t - \mathbf{z}'_t\|$.

Its proof is in (Ling, Low, and Jaillet 2016). Lemma 1 will be used to prove the Lipschitz continuity of V_t^* in (1) later. Before doing this, let us consider how the Lipschitz continuous reward functions defined above can unify some AL and BO criteria discussed in Section 1 and be used for defining new tasks/problems.

Active learning/sensing (AL). Setting $R(z_{t+1}, \mathbf{s}_{t+1}) = R_3(\mathbf{s}_{t+1}) = 0.5\log(2\pi e\sigma^2_{s_{t+1}|\mathbf{s}_t})$ yields the well-known nonmyopic AL algorithm called *maximum entropy sampling* (MES) (Shewry and Wynn 1987) which plans/decides locations with maximum entropy to be observed that minimize the posterior entropy remaining in the unobserved areas of the field. Since $R(z_{t+1}, \mathbf{s}_{t+1})$ is independent of z_{t+1}, the expectations in (1) go away, thus making MES non-adaptive and hence a straightforward search algorithm not plagued by the issue of uncountable set of candidate measurements. As such, we will not focus on such a degenerate case. This degeneracy vanishes when the environment field is instead a realization of log-Gaussian process. Then, MES becomes adaptive (Low, Dolan, and Khosla 2009) and its reward function can be represented by our Lipschitz continuous reward functions: By setting $R_1(z_{t+1}) = 0$, R_2 and $g_{\sigma_{s_{t+1}|\mathbf{s}_t}}$ as identity functions with $\ell_2(\sigma_{s_{t+1}|\mathbf{s}_t}) = 1$, and $R_3(\mathbf{s}_{t+1}) = 0.5\log(2\pi e\sigma^2_{s_{t+1}|\mathbf{s}_t})$, $\mathbb{E}[R(Z_{t+1}, \mathbf{s}_{t+1})|d_t, s_{t+1}] = \mu_{s_{t+1}|d_t} + 0.5\log(2\pi e\sigma^2_{s_{t+1}|\mathbf{s}_t})$.

Bayesian optimization (BO). The greedy BO algorithm of Srinivas et al. (2010) utilizes the UCB selection criterion $\mu_{s_{t+1}|d_t} + \beta\sigma_{s_{t+1}|\mathbf{s}_t}$ ($\beta \geq 0$) to approximately optimize the global BO objective of total field measurements $\sum_{t=1}^H z_t$ taken by the robot or, equivalently, minimize its total regret. UCB can be represented by our Lipschitz continuous reward functions: By setting $R_1(z_{t+1}) = 0$, R_2 and $g_{\sigma_{s_{t+1}|\mathbf{s}_t}}$ as identity functions with $\ell_2(\sigma_{s_{t+1}|\mathbf{s}_t}) = 1$, and $R_3(\mathbf{s}_{t+1}) = \beta\sigma_{s_{t+1}|\mathbf{s}_t}$, $\mathbb{E}[R(Z_{t+1}, \mathbf{s}_{t+1})|d_t, s_{t+1}] = \mu_{s_{t+1}|d_t} + \beta\sigma_{s_{t+1}|\mathbf{s}_t}$. In particular, when $\beta = 0$, it can be derived that our GPP policy π^* maximizes the *expected* total field measurements taken by the robot, hence optimizing the exact global BO objective of Srinivas et al. (2010) in the expected sense. So, unlike greedy UCB, our nonmyopic GPP framework does not have to explicitly consider an additional weighted exploration term (i.e., $\beta\sigma_{s_{t+1}|\mathbf{s}_t}$) in its reward function because it can jointly and naturally optimize the exploration-exploitation trade-off, as explained earlier. Nevertheless, if a stronger exploration behavior is desired (e.g., in online planning), then β has to be fine-tuned. Different from nonmyopic BO algorithm of Marchant, Ramos, and Sanner (2014) using UCB-based rewards, our proposed nonmyopic ϵ-optimal GPP policy (Section 3) does not need to impose an extreme assumption of maximum likelihood observations during planning and, more importantly, provides a performance guarantee, including for the extreme assumption made by nonmyopic UCB. Our GPP framework differs from nonmyopic BO algorithm of Osborne, Garnett, and Roberts (2009) in that every selected observation contributes to the total field measurements taken by the robot instead of considering just the expected improvement for the last observation. So, it usually does not have to expend all

[1] Unlike R_1, R_2 does not need to be Lipschitz continuous (or continuous); it must only be Lipschitz continuous after convolution with any Gaussian kernel. An example of R_2 is unit step function.

the given sampling budget to find the global maximum.

General tasks/problems. In practice, the necessary reward function can be more complex than the ones specified above that are formed from an identity function of the field measurement. For example, consider the problem of placing wind turbines in optimal locations to maximize the total power production. Though the average wind speed in a region can be modeled by a GP, the power output is not a linear function of the steady-state wind speed. In fact, power production requires a certain minimum speed known as the cut-in speed. After this threshold is met, power output increases and eventually plateaus. Assuming the cut-in speed is 1, this effect can be modeled with a logarithmic reward function[2]: $R(z_{t+1}, \mathbf{s}_{t+1}) = R_1(z_{t+1})$ gives a value of $\log(z_{t+1})$ if $z_{t+1} > 1$, and 0 otherwise where $\ell_1 = 1$. To the best of our knowledge, $h_{\sigma_{s_{t+1}|\mathbf{s}_t}}(u)$ has no closed-form expression. In (Ling, Low, and Jaillet 2016), we present other interesting reward functions like unit step function[1] and Gaussian distribution that can be represented by $R(z_{t+1}, \mathbf{s}_{t+1})$ and used in real-world tasks.

Theorem 1 below reveals that $V_t^*(d_t)$ (1) with Lipschitz continuous reward functions is Lipschitz continuous in $\mathbf{z}_t$ with Lipschitz constant $L_t(\mathbf{s}_t)$ defined below:

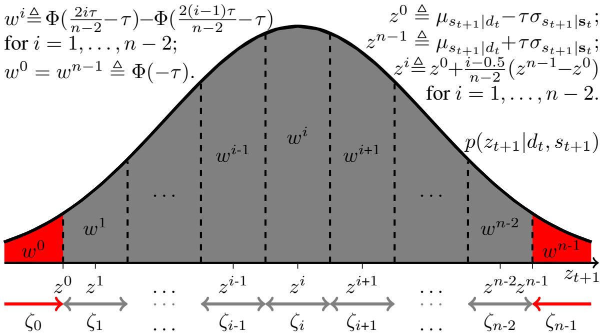
Definition 1 *Let* $L_H(\mathbf{s}_H) \triangleq 0$*. For* $t = 0, \ldots, H-1$*, define* $L_t(\mathbf{s}_t) \triangleq \max_{s_{t+1}\in\mathcal{A}(s_t)} \alpha(\mathbf{s}_{t+1})\left(\ell_1 + \ell_2(\sigma_{s_{t+1}|\mathbf{s}_t})\right) + L_{t+1}(\mathbf{s}_{t+1})\sqrt{1+\alpha(\mathbf{s}_{t+1})^2}$.

Theorem 1 (Lipschitz Continuity of V_t^*) *For* $t = 0, \ldots, H$, $|V_t^*(d_t) - V_t^*(d'_t)| \leq L_t(\mathbf{s}_t)\|\mathbf{z}_t - \mathbf{z}'_t\|$.

Its proof uses Lemma 1 and is in (Ling, Low, and Jaillet 2016). The result below is a direct consequence of Theorem 1 and will be used to theoretically guarantee the performance of our proposed nonmyopic adaptive ϵ-optimal GPP policy in Section 3:

Corollary 1 *For* $t = 0, \ldots, H$, $|V_t^*(\langle \mathbf{s}_t, \mathbf{z}_{t-1} \oplus z_t\rangle) - V_t^*(\langle \mathbf{s}_t, \mathbf{z}_{t-1} \oplus z'_t\rangle)| \leq L_t(\mathbf{s}_t)|z_t - z'_t|$.

3 ϵ-Optimal GPP (ϵ-GPP)



$w^i \triangleq \Phi(\frac{2i\tau}{n-2} - \tau) - \Phi(\frac{2(i-1)\tau}{n-2} - \tau)$
for $i = 1, \ldots, n-2$;
$w^0 = w^{n-1} \triangleq \Phi(-\tau)$.
$z^0 \triangleq \mu_{s_{t+1}|d_t} - \tau\sigma_{s_{t+1}|\mathbf{s}_t}$;
$z^{n-1} \triangleq \mu_{s_{t+1}|d_t} + \tau\sigma_{s_{t+1}|\mathbf{s}_t}$;
$z^i \triangleq z^0 + \frac{i-0.5}{n-2}(z^{n-1} - z^0)$
for $i = 1, \ldots, n-2$.
$p(z_{t+1}|d_t, s_{t+1})$
w^0
w^1
$\ldots$
$w^{i\text{-}1}$
w^i
w^{i+1}
$\ldots$
$w^{n\text{-}2}$
$w^{n\text{-}1}$
z_{t+1}
z^0
z^1
$\ldots$
$z^{i\text{-}1}$
z^i
z^{i+1}
$\ldots$
$z^{n\text{-}2}$
$z^{n\text{-}1}$
ζ_0
ζ_1
$\ldots$
$\zeta_{i\text{-}1}$
ζ_i
ζ_{i+1}
$\ldots$
$\zeta_{n\text{-}2}$
$\zeta_{n\text{-}1}$

The key idea of constructing our proposed nonmyopic adaptive ϵ-GPP policy is to approximate the expectation terms in (1) at every stage using a form of deterministic sampling, as illustrated in the figure below. Specifically, the measurement space of $p(z_{t+1}|d_t, s_{t+1})$ is first partitioned into $n \geq 2$ intervals $\zeta_0, \ldots, \zeta_{n-1}$ such that intervals $\zeta_1, \ldots, \zeta_{n-2}$ are equally spaced within the bounded gray region $[\mu_{s_{t+1}|d_t} - \tau\sigma_{s_{t+1}|\mathbf{s}_t}, \mu_{s_{t+1}|d_t} + \tau\sigma_{s_{t+1}|\mathbf{s}_t}]$ specified by a user-defined width parameter $\tau \geq 0$ while intervals ζ_0 and ζ_{n-1} span the two infinitely long red tails. Note that $\tau > 0$ requires $n > 2$ for the partition to be valid. The n sample measurements $z^0 \ldots z^{n-1}$ are then selected by setting z^0 as upper limit of red interval ζ_0, z^{n-1} as lower limit of red interval ζ_{n-1}, and $z^1, \ldots, z^{n-2}$ as centers of the respective gray intervals $\zeta_1, \ldots, \zeta_{n-2}$. Next, the weights $w^0 \ldots w^{n-1}$ for the corresponding sample measurements $z^0, \ldots, z^{n-1}$ are defined as the areas under their respective intervals $\zeta_0, \ldots, \zeta_{n-1}$ of the Gaussian predictive distribution $p(z_{t+1}|d_t, s_{t+1})$. So, $\sum_{i=0}^{n-1} w^i = 1$. An example of such a partition is given in (Ling, Low, and Jaillet 2016). The selected sample measurements and their corresponding weights can be exploited for approximating V_t^* with Lipschitz continuous reward functions (1) using the following H-stage Bellman equations:

$$
\begin{aligned}
V_t^\epsilon(d_t) &\triangleq \max\nolimits_{s_{t+1}\in\mathcal{A}(s_t)} Q_t^\epsilon(d_t, s_{t+1}) \\
Q_t^\epsilon(d_t, s_{t+1}) &\triangleq g_{\sigma_{s_{t+1}|\mathbf{s}_t}}(\mu_{s_{t+1}|d_t}) + R_3(\mathbf{s}_{t+1}) + \\
&\quad \sum_{i=0}^{n-1} w^i \big(R_1(z^i) + V_{t+1}^\epsilon(\langle \mathbf{s}_{t+1}, \mathbf{z}_t \oplus z^i\rangle)\big)
\end{aligned}
\qquad (2)
$$

for stages $t = 0, \ldots, H-1$ where $V_H^\epsilon(d_H) \triangleq 0$. The resulting induced ϵ-GPP policy π^ϵ jointly and naturally optimizes the exploration-exploitation trade-off in a similar manner as that of the GPP policy π^*, as explained in Section 2. It is interesting to note that setting $\tau = 0$ yields $z^0 = \ldots = z^{n-1} = \mu_{s_{t+1}|d_t}$, which is equivalent to selecting a single sample measurement of $\mu_{s_{t+1}|d_t}$ with corresponding weight of 1. This is identical to the special case of maximum likelihood observations during planning which is the extreme assumption used by nonmyopic UCB (Marchant, Ramos, and Sanner 2014) for sampling to gain time efficiency.

Performance Guarantee. The difficulty in theoretically guaranteeing the performance of our ϵ-GPP policy π^ϵ (i.e., relative to that of GPP policy π^*) lies in analyzing how the values of the width parameter τ and deterministic sampling size n can be chosen to satisfy the user-specified loss bound ϵ, as discussed below. The first step is to prove that V_t^ϵ in (2) approximates V_t^* in (1) closely for some chosen τ and n values, which relies on the Lipschitz continuity of V_t^* in Corollary 1. Define $\Lambda(n, \tau)$ to be equal to the value of $\sqrt{2/\pi}$ if $n \geq 2 \wedge \tau = 0$, and value of $\kappa(\tau) + \eta(n, \tau)$ if $n > 2 \wedge \tau > 0$ where $\kappa(\tau) \triangleq \sqrt{2/\pi}\exp(-0.5\tau^2) - 2\tau\Phi(-\tau)$, $\eta(n, \tau) \triangleq 2\tau(0.5 - \Phi(-\tau))/(n-2)$, and Φ is a standard normal CDF.

Theorem 2 *Suppose that* $\lambda > 0$ *is given. For all* d_t *and* $t = 0, \ldots, H$*, if*

$$
\lambda \geq \Lambda(n, \tau)\sigma_{s_{t+1}|\mathbf{s}_t}(\ell_1 + L_{t+1}(\mathbf{s}_{t+1})) \qquad (3)
$$

for all $s_{t+1} \in \mathcal{A}(s_t)$*, then* $|V_t^\epsilon(d_t) - V_t^*(d_t)| \leq \lambda(H-t)$.

Its proof uses Corollary 1 and is given in (Ling, Low, and Jaillet 2016).

Remark 1. From Theorem 2, a tighter bound on the error $|V_t^\epsilon(d_t) - V_t^*(d_t)|$ can be achieved by decreasing the sampling budget of H locations[3] and increasing the deterministic sampling size n; increasing n reduces $\eta(n, \tau)$ and hence

[2] In reality, the speed-power relationship is not exactly logarithmic, but this approximation suffices for the purpose of modeling.

[3] This changes ϵ-GPP by reducing its planning horizon though.

$\Lambda(n,\tau)$, which allows λ to be reduced as well. The width parameter τ has a mixed effect on this error bound: Note that $\kappa(\tau)$ ($\eta(n,\tau)$) is proportional to some upper bound on the error incurred by the extreme sample measurements z^0 and z^{n-1} ($z^1,\ldots,z^{n-2}$), as shown in (Ling, Low, and Jaillet 2016). Increasing τ reduces $\kappa(\tau)$ but unfortunately raises $\eta(n,\tau)$. So, in order to reduce $\Lambda(n,\tau)$ further by increasing τ, it has to be complemented by raising n fast enough to keep $\eta(n,\tau)$ from increasing. This allows λ to be reduced further as well.

Remark 2. A feasible choice of τ and n satisfying (3) can be expressed analytically in terms of the given λ and hence computed prior to planning, as shown in (Ling, Low, and Jaillet 2016).

Remark 3. $\sigma_{s_{t+1}|\mathbf{s}_t}$ and $L_{t+1}(\mathbf{s}_{t+1})$ for all $\mathbf{s}_{t+1}$ and $t = 0,\ldots,H-1$ can be computed prior to planning as they depend on $\mathbf{s}_0$ and all reachable locations from s_0 but not on their measurements.

Using Theorem 2, the next step is to bound the performance loss of our ϵ-GPP policy π^ϵ relative to that of GPP policy π^*, that is, policy π^ϵ is ϵ-optimal:

Theorem 3 *Given the user-specified loss bound* $\epsilon > 0$, $V_0^*(d_0) - V_0^{\pi^\epsilon}(d_0) \leq \epsilon$ *by substituting* $\lambda = \epsilon/(H(H+1))$ *into the choice of* τ *and* n *stated in Remark* 2 *above.*

Its proof is in (Ling, Low, and Jaillet 2016). It can be observed from Theorem 3 that a tighter bound ϵ on the error $V_0^*(d_0) - V_0^{\pi^\epsilon}(d_0)$ can be achieved by decreasing the sampling budget of H locations[3] and increasing the deterministic sampling size n. The effect of width parameter τ on this error bound ϵ is the same as that on the error bound of $|V_t^\epsilon(d_t) - V_t^*(d_t)|$, as explained in Remark 1 above.

Anytime ϵ-GPP. Unlike GPP policy π^*, our ϵ-GPP policy π^ϵ can be derived exactly since its incurred time is independent of the size of the uncountable set of candidate measurements. However, expanding the entire search tree of ϵ-GPP (2) incurs time containing a $\mathcal{O}(n^H)$ term and is not always necessary to achieve ϵ-optimality in practice. To mitigate this computational difficulty[4], we propose an anytime variant of ϵ-GPP that can produce a good policy fast and improve its approximation quality over time, as briefly discussed here and detailed with the pseudocode in (Ling, Low, and Jaillet 2016).

The key intuition is to expand the sub-trees rooted at "promising" nodes with the highest weighted uncertainty of their corresponding values $V_t^*(d_t)$ so as to improve their estimates. To represent such uncertainty at each encountered node, upper & lower heuristic bounds (respectively, $\overline{V}_t^*(d_t)$ and $\underline{V}_t^*(d_t)$) are maintained, like in (Smith and Simmons 2006). A partial construction of the entire tree is maintained and expanded incrementally in each iteration of anytime ϵ-GPP that incurs linear time in n and comprises 3 steps:

Node selection. Traverse down the partially constructed tree by repeatedly selecting nodes with largest difference between their upper and lower bounds (i.e., uncertainty) discounted by weight w^{i^*} of its preceding sample measurement z^{i^*} until an unexpanded node, denoted by d_t, is reached.

Expand tree. Construct a "minimal" sub-tree rooted at node d_t by sampling all possible next locations and only their median sample measurements $z^{\bar{i}}$ recursively up to full height H.

Backpropagation. Backpropagate bounds from the leaves of the newly constructed sub-tree to node d_t, during which the refined bounds of expanded nodes are used to inform the bounds of unexpanded siblings by exploiting the Lipschitz continuity of V_t^* (Corollary 1), as explained in (Ling, Low, and Jaillet 2016). Backpropagate bounds to the root of the partially constructed tree in a similar manner.

By using the lower heuristic bound to produce our anytime ϵ-GPP policy, its performance loss relative to that of GPP policy π^* can be bounded, as proven in (Ling, Low, and Jaillet 2016).

4 Experiments and Discussion

This section empirically evaluates the online planning performance and time efficiency of our ϵ-GPP policy π^ϵ and its anytime variant under limited sampling budget in an energy harvesting task on a simulated wind speed field and in BO on simulated plankton density (chl-a) field and real-world log-potassium (lg-K) concentration (mg l^{-1}) field (Ling, Low, and Jaillet 2016) of Broom's Barn farm (Webster and Oliver 2007). Each simulated (real-world lg-K) field is spatially distributed over a 0.95 km by 0.95 km (520 m by 440 m) region discretized into a 20×20 (14×12) grid of sampling locations. These fields are assumed to be realizations of GPs. The wind speed (chl-a) field is simulated using hyperparameters $\mu_s = 0$,[5] $l_1 = l_2 = 0.2236$ (0.2) km, $\sigma_n^2 = 10^{-5}$, and $\sigma_y^2 = 1$. The hyperparameters $\mu_s = 3.26$, $l_1 = 42.8$ m, $l_2 = 103.6$ m, $\sigma_n^2 = 0.0222$, and $\sigma_y^2 = 0.057$ of lg-K field are learned using maximum likelihood estimation (Rasmussen and Williams 2006). The robot's initial starting location is near to the center of each simulated field and randomly selected for lg-K field. It can move to any of its 4 adjacent grid locations at each time step and is tasked to maximize its total rewards over 20 time steps (i.e., sampling budget of 20 locations).

In BO, the performances of our ϵ-GPP policy π^ϵ and its anytime variant are compared with that of state-of-the-art *nonmyopic UCB* (Marchant, Ramos, and Sanner 2014) and *greedy PI, EI, UCB* (Brochu, Cora, and de Freitas 2010; Srinivas et al. 2010). Three performance metrics are used: (a) Total rewards achieved over the evolved time steps (i.e., higher total rewards imply less total regret in BO (Section 2)), (b) maximum reward achieved during experiment, and (c) search tree size in terms of no. of nodes (i.e., larger tree size implies higher incurred time). All experiments are run on a Linux machine with Intel Core i5 at 1.7 GHz.

Energy Harvesting Task on Simulated Wind Speed Field. A robotic rover equipped with a wind turbine is tasked to harvest energy/power from the wind while exploring a polar region (Chen et al. 2014). It is driven by the logarithmic reward function described under 'General tasks/problems' in Section 2. Fig. 1 shows results of performances of our ϵ-GPP

[4] The value of n is a bigger computational issue than that of H when ϵ is small and in online planning.

[5] Its actual prior mean is not zero; we have applied zero-mean GP to $Y_s - \mu_s$ for simplicity.

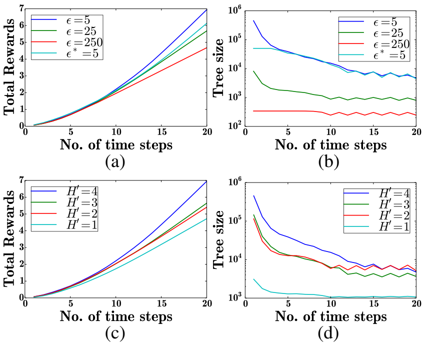


Figure 1: Graphs of total rewards and tree size of ϵ-GPP policies with (a-b) online planning horizon $H' = 4$ and varying ϵ and (c-d) varying $H' = 1, 2, 3, 4$ (respectively, $\epsilon = 0.002, 0.06, 0.8, 5$) vs. no. of time steps for energy harvesting task. The plot of $\epsilon^* = 5$ uses our anytime variant with a maximum tree size of 5×10^4 nodes while the plot of $\epsilon = 250$ effectively assumes maximum likelihood observations during planning like that of nonmyopic UCB (Marchant, Ramos, and Sanner 2014).

policy and its anytime variant averaged over 30 independent realizations of the wind speed field. It can be observed that the gradients of the achieved total rewards (i.e., power production) increase over time, which indicate a higher obtained reward with an increasing number of time steps as the robot can exploit the environment more effectively with the aid of exploration from previous time steps. The gradients eventually stop increasing when the robot enters a perceived high-reward region. Further exploration is deemed unnecessary as it is unlikely to find another preferable location within H' time steps; so, the robot remains near-stationary for the remaining time steps. It can also be observed that the incurred time is much higher in the first few time steps. This is expected because the posterior variance $\sigma_{s_{t+1}|\mathbf{s}_t}$ decreases with increasing time step t, thus requiring a decreasing deterministic sampling size n to satisfy (3).

Initially, all ϵ-GPP policies achieve similar total rewards as the robots begin from the same starting location. After some time, ϵ-GPP policies with lower user-specified loss bound ϵ and longer online planning horizon H' achieve considerably higher total rewards at the cost of more incurred time. In particular, it can be observed that a robot assuming maximum likelihood observations during planning (i.e., $\epsilon = 250$) like that of nonmyopic UCB or using a greedy policy (i.e., $H' = 1$) performs poorly very quickly. In the former case (Fig. 1a), the gradient of its total rewards stops increasing quite early (i.e., from time step 9 onwards), which indicates that its perceived local maximum is reached prematurely. Interestingly, it can be observed from Fig. 1d that the ϵ-GPP policy with $H' = 2$ and $\epsilon = 0.06$ incurs more time than that with $H' = 3$ and $\epsilon = 0.8$ despite the latter achieving higher total rewards. This suggests trading off tighter loss bound ϵ for longer online planning horizon H', especially when ϵ is too small that in turn requires a very large n and consequently incurs significantly more time[4].

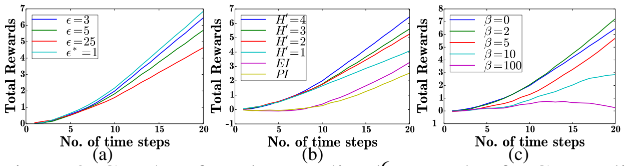


Figure 2: Graphs of total normalized[6] rewards of ϵ-GPP policies using UCB-based rewards with (a) $H' = 4$, $\beta = 0$, and varying ϵ, (b) varying $H' = 1, 2, 3, 4$ (respectively, $\epsilon = 0.002, 0.003, 0.4, 2$) and $\beta = 0$, and (c) $H' = 4$, $\epsilon = 1$, and varying β vs. no. of time steps for BO on real-world lg-K field. The plot of $\epsilon^* = 1$ uses our anytime variant with a maximum tree size of 3×10^4 nodes while the plot of $\epsilon = 25$ effectively assumes maximum likelihood observations during planning like that of nonmyopic UCB.

BO on Real-World Log-Potassium Concentration Field. An agricultural robot is tasked to find the peak lg-K measurement (i.e., possibly in an over-fertilized area) while exploring the Broom's Barn farm (Webster and Oliver 2007). It is driven by the UCB-based reward function described under 'BO' in Section 2. Fig. 2 shows results of performances of our ϵ-GPP policy and its anytime variant, nonmyopic UCB (i.e., $\epsilon = 25$), and greedy PI, EI, UCB (i.e., $H' = 1$) averaged over 25 randomly selected robot's initial starting location. It can be observed from Figs. 2a and 2b that the gradients of the achieved total normalized[6] rewards generally increase over time. In particular, from Fig. 2a, nonmyopic UCB assuming maximum likelihood observations during planning obtains much less total rewards than the other ϵ-GPP policies and the anytime variant after 20 time steps and finds a maximum lg-K measurement of 3.62 that is at least $0.4\sigma_y$ worse after 20 time steps. The performance of the anytime variant is comparable to that of our best-performing ϵ-GPP policy with $\epsilon = 3$. From Fig. 2b, the greedy policy (i.e., $H' = 1$) with $\beta = 0$ performs much more poorly than its nonmyopic ϵ-GPP counterparts and finds a maximum lg-K measurement of 3.56 that is lower than that of greedy PI and EI due to its lack of exploration. By increasing H' to 2-4, our ϵ-GPP policies with $\beta = 0$ outperform greedy PI and EI as they can naturally and jointly optimize the exploration-exploitation trade-off. Interestingly, Fig. 2c shows that our ϵ-GPP policy with $\beta = 2$ achieves the highest total rewards after 20 time steps, which indicates the need of a slightly stronger exploration behavior than that with $\beta = 0$. This may be explained by a small length-scale (i.e., spatial correlation) of the lg-K field, thus requiring some exploration to find the peak measurement. By increasing H' beyond 4 or with larger spatial correlation (Ling, Low, and Jaillet 2016), we expect a diminishing role of the $\beta\sigma_{s_{t+1}|\mathbf{s}_t}$ term. It can also be observed that aggressive exploration (i.e., $\beta \geq 10$) hurts the performance. Results of the tree size (i.e., incurred time) of our ϵ-GPP policy and its anytime variant are in (Ling, Low, and Jaillet 2016).

5 Conclusion

This paper describes a novel nonmyopic adaptive ϵ-GPP framework endowed with a general class of Lipschitz continuous reward functions that can unify some AL and BO criteria and be used for defining new tasks/problems. In particular, it can jointly and naturally optimize the exploration-exploitation trade-off. We theoretically guarantee the performances of our ϵ-GPP policy and its anytime variant and empirically demonstrate their effectiveness in BO and an en-

[6] To ease interpretation of the results, each reward is normalized by subtracting the prior mean from it.

ergy harvesting task. For our future work, we plan to scale up ϵ-GPP and its anytime variant for big data using parallelization (Chen et al. 2013; Low et al. 2015), online learning (Xu et al. 2014), and stochastic variational inference (Hoang, Hoang, and Low 2015) and extend them to handle unknown hyperparameters (Hoang et al. 2014).

Acknowledgments. This work was supported by Singapore-MIT Alliance for Research and Technology Subaward Agreement No. 52 R-252-000-550-592.

References

Brochu, E.; Cora, V. M.; and de Freitas, N. 2010. A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. arXiv:1012.2599.

Cao, N.; Low, K. H.; and Dolan, J. M. 2013. Multi-robot informative path planning for active sensing of environmental phenomena: A tale of two algorithms. In *Proc. AAMAS*.

Chen, J.; Low, K. H.; Tan, C. K.-Y.; Oran, A.; Jaillet, P.; Dolan, J. M.; and Sukhatme, G. S. 2012. Decentralized data fusion and active sensing with mobile sensors for modeling and predicting spatiotemporal traffic phenomena. In *Proc. UAI*, 163–173.

Chen, J.; Cao, N.; Low, K. H.; Ouyang, R.; Tan, C. K.-Y.; and Jaillet, P. 2013. Parallel Gaussian process regression with low-rank covariance matrix approximations. In *Proc. UAI*, 152–161.

Chen, J.; Liang, J.; Wang, T.; Zhang, T.; and Wu, Y. 2014. Design and power management of a wind-solar-powered polar rover. *Journal of Ocean and Wind Energy* 1(2):65–73.

Chen, J.; Low, K. H.; Jaillet, P.; and Yao, Y. 2015. Gaussian process decentralized data fusion and active sensing for spatiotemporal traffic modeling and prediction in mobility-on-demand systems. *IEEE Trans. Autom. Sci. Eng.* 12:901–921.

Chen, J.; Low, K. H.; and Tan, C. K.-Y. 2013. Gaussian process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*.

Deisenroth, M. P.; Fox, D.; and Rasmussen, C. E. 2015. Gaussian processes for data-efficient learning in robotics and control. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 37(2):408–423.

Hennig, P., and Schuler, C. J. 2012. Entropy search for information-efficient global optimization. *JMLR* 13:1809–1837.

Hernández-Lobato, J. M.; Hoffman, M. W.; and Ghahramani, Z. 2014. Predictive entropy search for efficient global optimization of black-box functions. In *Proc. NIPS*.

Hoang, T. N.; Low, K. H.; Jaillet, P.; and Kankanhalli, M. 2014. Nonmyopic ϵ-Bayes-optimal active learning of Gaussian processes. In *Proc. ICML*, 739–747.

Hoang, T. N.; Hoang, Q. M.; and Low, K. H. 2015. A unifying framework of anytime sparse Gaussian process regression models with stochastic variational inference for big data. In *Proc. ICML*, 569–578.

Krause, A.; Singh, A.; and Guestrin, C. 2008. Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *JMLR* 9:235–284.

Leonard, N. E.; Palley, D. A.; Lekien, F.; Sepulchre, R.; Fratantoni, D. M.; and Davis, R. E. 2007. Collective motion, sensor networks, and ocean sampling. *Proc. IEEE* 95:48–74.

Ling, C. K.; Low, K. H.; and Jaillet, P. 2016. Gaussian process planning with Lipschitz continuous reward functions: Towards unifying Bayesian optimization, active learning, and beyond. arXiv:1511.06890.

Low, K. H.; Chen, J.; Dolan, J. M.; Chien, S.; and Thompson, D. R. 2012. Decentralized active robotic exploration and mapping for probabilistic field classification in environmental sensing. In *Proc. AAMAS*, 105–112.

Low, K. H.; Yu, J.; Chen, J.; and Jaillet, P. 2015. Parallel Gaussian process regression for big data: Low-rank representation meets Markov approximation. In *Proc. AAAI*.

Low, K. H.; Dolan, J. M.; and Khosla, P. 2008. Adaptive multi-robot wide-area exploration and mapping. In *Proc. AAMAS*, 23–30.

Low, K. H.; Dolan, J. M.; and Khosla, P. 2009. Information-theoretic approach to efficient adaptive path planning for mobile robotic environmental sensing. In *Proc. ICAPS*.

Low, K. H.; Dolan, J. M.; and Khosla, P. 2011. Active Markov information-theoretic path planning for robotic environmental sensing. In *Proc. AAMAS*, 753–760.

Marchant, R.; Ramos, F.; and Sanner, S. 2014. Sequential Bayesian optimisation for spatial-temporal monitoring. In *Proc. UAI*.

Osborne, M. A.; Garnett, R.; and Roberts, S. J. 2009. Gaussian processes for global optimization. In *Proc. 3rd International Conference on Learning and Intelligent Optimization*.

Ouyang, R.; Low, K. H.; Chen, J.; and Jaillet, P. 2014. Multi-robot active sensing of non-stationary Gaussian process-based environmental phenomena. In *Proc. AAMAS*.

Poupart, P.; Vlassis, N.; Hoey, J.; and Regan, K. 2006. An analytic solution to discrete Bayesian reinforcement learning. In *Proc. ICML*, 697–704.

Rasmussen, C. E., and Williams, C. K. I. 2006. *Gaussian Processes for Machine Learning*. MIT Press.

Shewry, M. C., and Wynn, H. P. 1987. Maximum entropy sampling. *J. Applied Statistics* 14(2):165–170.

Smith, T., and Simmons, R. 2006. Focused real-time dynamic programming for MDPs: Squeezing more out of a heuristic. In *Proc. AAAI*, 1227–1232.

Srinivas, N.; Krause, A.; Kakade, S.; and Seeger, M. 2010. Gaussian process optimization in the bandit setting: No regret and experimental design. In *Proc. ICML*, 1015–1022.

Webster, R., and Oliver, M. 2007. *Geostatistics for Environmental Scientists*. NY: John Wiley & Sons, Inc., 2nd edition.

Xu, N.; Low, K. H.; Chen, J.; Lim, K. K.; and Ozgul, E. B. 2014. GP-Localize: Persistent mobile robot localization using online sparse Gaussian process observation model. In *Proc. AAAI*, 2585–2592.

Zhang, Y.; Hoang, T. N.; Low, K. H.; and Kankanhalli, M. 2016. Near-optimal active learning of multi-output Gaussian processes. In *Proc. AAAI*.