Decentralized High-Dimensional Bayesian Optimization with Factor Graphs

Trong Nghia Hoang[§] and **Quang Minh Hoang**[†] and **Ruofei Ouyang**[†] and **Kian Hsiang Low**[†]

Laboratory of Information and Decision Systems, Massachusetts Institute of Technology, USA[§]

Department of Computer Science, National University of Singapore, Republic of Singapore[†]

nghiaht@mit.edu[§], {hqminh, ouyang, lowkh}@comp.nus.edu.sg[†]

Abstract

This paper presents a novel *decentralized high-dimensional Bayesian optimization* (DEC-HBO) algorithm that, in contrast to existing HBO algorithms, can exploit the interdependent effects of various input components on the output of the unknown objective function f for boosting the BO performance and still preserve scalability in the number of input dimensions without requiring prior knowledge or the existence of a low (effective) dimension of the input space. To realize this, we propose a sparse yet rich factor graph representation of f to be exploited for designing an acquisition function that can be similarly represented by a sparse factor graph and hence be efficiently optimized in a decentralized manner using distributed message passing. Despite richly characterizing the interdependent effects of the input components on the output of f with a factor graph, DEC-HBO can still guarantee no-regret performance asymptotically. Empirical evaluation on synthetic and real-world experiments (e.g., sparse Gaussian process model with 1811 hyperparameters) shows that DEC-HBO outperforms the state-of-the-art HBO algorithms.

1 Introduction

Many real-world applications/tasks often involve optimizing an unknown objective function f given a limited budget of costly function evaluations. Examples of such applications/tasks include automatic hyperparameter tuning for machine learning models (e.g., deep neural network) (Bergstra, Yamins, and Cox 2013; Snoek, Hugo, and Adams 2012) and parameter configuration for robotic control strategies. Whereas gradient-based methods fail to optimize a function without an analytic form/derivative, *Bayesian optimization* (BO) has established itself as a highly effective alternative. In particular, a BO algorithm maintains a *Gaussian process* (GP) belief of the unknown objective function f and alternates between selecting an input query to evaluate f and using its observed output to update the GP belief of f until the budget is exhausted. Every input query is selected by maximizing an acquisition function that is constructed from the GP belief of f. Intuitively, such an acquisition function has to trade off between optimizing f based on its current GP belief (exploitation) vs. improving its GP belief (exploration). Popular choices include improvement-based (Shahriari et al. 2016), information-based (Hennig and Schuler 2012), and upper confidence bound (Srinivas et al. 2010).

While BO has demonstrated to be an effective optimization strategy in general, it has mostly found success in the context of input spaces with low (effective) dimension (Djolonga, Krause, and Cevher 2013; Wang et al. 2013). However, several real-world application domains such as computer vision (Bergstra, Yamins, and Cox 2013), networking (Hornby et al. 2006), and computational biology (González et al. 2014) often require optimizing an objective function f over a high-dimensional input space without knowing if its low (effective) dimension even exists. This poses a grand challenge to the above conventional BO algorithms as the cost of maximizing an acquisition function grows exponentially with the number of input dimensions. To sidestep this issue, an extreme approach is to assume the effects of all input components on the output of f to be pairwise independent (Kandasamy, Schneider, and Póczos 2015; Wang and Jegelka 2017; Wang et al. 2017) (in the case of Li et al. (2016), after some affine projection of the input space). The acquisition function can then be maximized along each (projected) dimension separately, thus reducing its cost to linear in the number of input dimensions. Despite its simplicity, such a decoupling assumption can severely compromise the BO performance since it rarely holds in practice: The effects of different input components on the output of f are usually interdependent (Naveršnik and Rojnik 2012). In this paper, we argue and show that this highly restrictive assumption to gain scalability is an overkill: It is in fact possible to achieve the same scalability with a strong performance guarantee while still taking into account the interdependent effects of various input components on the output of f.

To achieve this, we first observe that the interdependent effects of many input components on the output of f tend to be indirect: The effect of one input component (on the output of f) can only directly influence that of some components in its immediate "neighborhood", which in turn may influence that of other components in the same manner. For example, in a multi-project company, the poor performance of one employee in a collaborative project only indirectly affects the performance of another employee in another project through those who work on both projects. This is also the case for many parameter tuning tasks with additive loss where different, overlapping subsets of parameters contribute to different additive factors of the loss function (Krähenbühl and Koltun 2011). The key challenge thus lies in investigating how the

unknown objective function f can be succinctly modeled to characterize such observed interdependent effects of the input components (on the output of f) and then exploited to design an acquisition function that can still be optimized scalably and yield a provable performance guarantee.

To address this challenge, this paper presents a novel *decentralized high-dimensional BO* (DEC-HBO) algorithm (Section 3) that, in contrast to some HBO algorithms (Kandasamy, Schneider, and Póczos 2015; Wang and Jegelka 2017; Wang et al. 2017), can exploit the interdependent effects of various input components (on the output of f) for boosting the BO performance and, perhaps surprisingly, still preserve scalability in the number of input dimensions without requiring the existence of a low (effective) dimension of the input space, unlike the other HBO algorithms (Djolonga, Krause, and Cevher 2013; Li et al. 2016; Wang et al. 2013). To realize this, we propose a sparse yet rich and highly expressive factor graph representation of the unknown objective function f by decomposing it into a linear sum of random factor functions, each of which involves only a small, *possibly overlapping* subset of input components and is assumed to be distributed by an independent GP prior (Section 3.1). As a result, the input components of the same factor function have direct interdependent effects on the output of f, while the input components that are distinct between any two factor functions have indirect interdependent effects on the output of f via their common input components; the latter is predominant due to sparsity of the factor graph representing f. We in turn exploit such a factor graph representation of f to design an acquisition function that, interestingly, can be similarly represented by a sparse factor graph (Section 3.2) and hence be efficiently optimized (Section 3.3) in a decentralized manner using a class of distributed message passing algorithms. The main novel contribution of our work here is to show that despite richly characterizing the interdependent effects of the input components on the output of f with a factor graph, our DEC-HBO algorithm not only preserves the scalability in the number of input dimensions but also guarantees the same trademark (asymptotic) no-regret performance. We empirically demonstrate the performance of DEC-HBO with synthetic and real-world experiments (e.g., sparse Gaussian process model with 1811 hyperparameters) (Section 5).

2 Background and Notations

This section first describes the zeroth-order optimization problem and its asymptotic optimality criterion which lay the groundwork for BO. Then, we review a class of well-studied BO algorithms (Kandasamy, Schneider, and Póczos 2015; Srinivas et al. 2010) and highlight their practical limitations when applied to high-dimensional optimization problems. We will discuss later in Sections 3 and 4 how our proposed DEC-HBO algorithm overcomes these limitations.

2.1 Zeroth-Order Optimization

Consider the problem of sequentially optimizing an unknown objective function $f : \mathcal{D} \to \mathbb{R}$ over a compact input domain $\mathcal{D} \subseteq \mathbb{R}^d$: In each iteration $t = 1, \ldots, n$, an input query $\mathbf{x}_t \in \mathcal{D}$ is selected for evaluating f to yield a noisy observed output $y_t \triangleq f(\mathbf{x}_t) + \epsilon$ with i.i.d. Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$ and noise variance σ_n^2. Since every evaluation of f is costly (Section 1), our goal is to strategically select input queries to approach the global maximizer $\mathbf{x}_* \triangleq \arg\max_{\mathbf{x} \in \mathcal{D}} f(\mathbf{x})$ as rapidly as possible. This can be achieved by minimizing a standard BO objective such as the *cumulative regret* R_n which sums the *instantaneous regret* $r_t \triangleq f(\mathbf{x}_*) - f(\mathbf{x}_t)$ incurred by selecting the input query $\mathbf{x}_t$ (instead of $\mathbf{x}_*$ due to not knowing $\mathbf{x}_*$ beforehand) to evaluate f over iteration $t = 1, \ldots, n$, that is, $R_n \triangleq \sum_{t=1}^{n} r_t$. A BO algorithm is said to be *asymptotically optimal* if it satisfies $\lim_{n \to \infty} R_n/n = 0$ which implies $\lim_{n \to \infty} (f(\mathbf{x}_*) - \max_{t=1}^{n} f(\mathbf{x}_t)) = 0$, thus guaranteeing *no-regret* performance asymptotically.

2.2 Bayesian Optimization with No Regret

A notable asymptotically optimal BO algorithm selects, in each iteration $t + 1$, an input query $\mathbf{x} \in \mathcal{D}$ to maximize an acquisition function called the *Gaussian process upper confidence bound* (GP-UCB) (Srinivas et al. 2010) that trades off between observing an expected maximum (i.e., with large GP posterior mean $\mu_t(\mathbf{x})$) given the current GP belief of f (i.e., exploitation) vs. that of high predictive uncertainty (i.e., with large GP posterior variance $\sigma_t(\mathbf{x})^2$) to improve the GP belief of f over $\mathcal{D}$ (i.e., exploration), that is, $\mathbf{x}_{t+1} \triangleq \arg\max_{\mathbf{x} \in \mathcal{D}} \mu_t(\mathbf{x}) + \beta_{t+1}^{1/2} \sigma_t(\mathbf{x})$ where the parameter $\beta_{t+1} > 0$ is set to trade off between exploitation vs. exploration for guaranteeing no-regret performance asymptotically with high probability and the GP posterior mean $\mu_t(\mathbf{x})$ and variance $\sigma_t(\mathbf{x})^2$ will be defined later in a similar manner to (1) (Section 3.2) to ease exposition.

Unfortunately, the GP-UCB algorithm does not scale well to high-dimensional optimization problems as its cost grows exponentially with the number of input dimensions. This prohibits its use in real-world application domains that require optimizing an objective function over a high-dimensional input space such as those mentioned in Section 1. To sidestep this issue, some HBO algorithms (Djolonga, Krause, and Cevher 2013; Li et al. 2016; Wang et al. 2013) assume the existence of a low-dimensional embedding of the input space which then allows them to operate in an exponentially smaller surrogate space and hence reduce their cost. But, these HBO algorithms impose strong assumptions (including prior knowledge of the dimension of the embedding) to guarantee that the global maximizer (or its affine projection) indeed lies within the surrogate space. In particular, one such precarious assumption is that the dimensionality of the low-rank surrogate space reflects the actual effective dimension of the input space.

A more practical alternative is to consider the effects of various input components on the output of f instead: The HBO algorithm of Kandasamy, Schneider, and Póczos (2015) assumes the unknown objective function f to be decomposable into a sum of independent, GP-distributed local functions $f_1, \ldots, f_d$, each of which involves only a single input dimension: $f(\mathbf{x}) \triangleq f_1(\mathbf{x}^{(1)}) + \ldots + f_d(\mathbf{x}^{(d)})$ where $\mathbf{x}^{(i)}$ and d denote component i and the dimension

of input $\mathbf{x}$, respectively. Interestingly, this in turn induces a similar decomposition of the above-mentioned GP-UCB acquisition function into a sum of independent local acquisition functions $\varphi_t^{(i)}(\mathbf{x}^{(i)}) \triangleq \mu_t^{(i)}(\mathbf{x}^{(i)}) + \beta_{t+1}^{1/2}\sigma_t^{(i)}(\mathbf{x}^{(i)})$ for $i = 1, \ldots, d$, that is, $\sum_{i=1}^d \varphi_t^{(i)}(\mathbf{x}^{(i)})$. As a result, each local acquisition function $\varphi_t^{(i)}(\mathbf{x}^{(i)})$ can be independently maximized along a separate input dimension, thus reducing the overall computational cost to linear in the number d of input dimensions. However, such a HBO algorithm and a few others (Wang and Jegelka 2017; Wang et al. 2017) preclude the interdependent effects of different input dimensions on the output of f (Naveršnik and Rojnik 2012), which can severely compromise their performance. We will describe in Sections 3 and 4 how our DEC-HBO algorithm can exploit the interdependent effects of various input components (on the output of f) for boosting the BO performance and still preserve scalability in the number of input dimensions as well as guarantee no-regret performance asymptotically.

3 Problem Formulation

This section first introduces the sparse factor representation of the unknown objective function f (Section 3.1). Then, we exploit it to reformulate the GP-UCB acquisition function to a form that can be similarly represented by a sparse factor graph (Section 3.2) and hence be efficiently optimized in a decentralized manner using distributed message passing (Section 3.3) to achieve the same trademark (asymptotic) no-regret performance guarantee (Section 4).

3.1 Sparse Factor Graph Representation

To scale up BO to high-dimensional optimization problems while still taking into account the interdependent effects of various input components on the output of f, we first state the following key structural assumption to represent f as a sparse yet rich and highly expressive factor graph:

Assumption 1. *The d-dimensional objective function f can be decomposed into a sum of $|\mathcal{U}|$ factor functions $\{f_\mathcal{I}\}_{\mathcal{I} \in \mathcal{U}}$, each of which depends on a $|\mathcal{I}|$-dimensional input $\mathbf{x}^\mathcal{I}$ comprising only a small,* possibly overlapping *subset $\mathcal{I} \subseteq \mathcal{S} \triangleq \{1, 2, \ldots, d\}$ of the input components of $\mathbf{x}$ (i.e., $|\mathcal{I}| \ll d$), that is, $f(\mathbf{x}) \triangleq \sum_{\mathcal{I} \in \mathcal{U}} f_\mathcal{I}(\mathbf{x}^\mathcal{I})$.*

Intuitively, Assumption 1 decomposes the high d-dimensional optimization problem into small sub-problems, each of which involves optimizing a single factor function $f_\mathcal{I}$ over a low $|\mathcal{I}|$-dimensional input space (and is hence much less costly) while succinctly encoding the compatibility of its selected input $\mathbf{x}^\mathcal{I}$ with that of other factor functions through their common input components; the latter is completely disregarded by the state-of-the-art HBO algorithms (Kandasamy, Schneider, and Póczos 2015; Wang and Jegelka 2017; Wang et al. 2017) due to their highly restrictive decoupling assumption (Section 2.2). In practice, our more relaxed assumption thus allows prior knowledge of the interdependent effects of different input components (on the output of f) to be explicitly and succinctly encoded into the sparse factor graph representation of f. As a result, the input components of the same factor

function have direct interdependent effects on the output of f, while the input components that are distinct between any two factor functions have indirect interdependent effects on the output of f via their common input components; the latter is predominant due to sparsity of the factor graph representing f. Interestingly, our assumption can be further coupled with Assumption 2 below to induce an additive GP model of f (Duvenaud, Nickisch, and Rasmussen 2011) with truncated ANOVA kernels which have shown to be highly expressive in characterizing the latent interaction between different input components.

Despite needing to maintain the compatibility of their selected inputs, these factor functions can still be optimized in a decentralized manner if a message passing protocol can be established between them to allow those with common input components to coordinate their optimization efforts without requiring any factor function to handle input components not of its own. To achieve this, two non-trivial research questions arise: Firstly, how can these factor functions (with compatibility constraints) without analytic expressions nor black-box generators be optimized (see Sections 3.2 and 3.3)? Secondly, even if it is possible to optimize each factor function, how can their coordinated optimization efforts be guaranteed to converge the selected input queries to the global maximizer of f (see Section 4)? Note that the second question has not been addressed by the previously established convergence guarantee for the GP-UCB algorithm (Srinivas et al. 2010) as it only applies to the centralized setting but not our decentralized BO setting.

3.2 Acquisition Function

To optimize each factor function without having direct access to its black-box generator, we need a mechanism that can draw inference on the output of the factor function $f_\mathcal{I}$ given only the noisy observed outputs of f. This is achieved with the following assumption:

Assumption 2. *Each factor function $f_\mathcal{I}$ in the decomposition of f in Assumption 1 is independently distributed by a GP $\mathcal{GP}(0, \sigma_0^\mathcal{I}(\mathbf{x}^\mathcal{I}, \mathbf{x}'^\mathcal{I}))$ with prior mean $\mu_0^\mathcal{I}(\mathbf{x}^\mathcal{I}) \triangleq 0$ and covariance $\sigma_0^\mathcal{I}(\mathbf{x}^\mathcal{I}, \mathbf{x}'^\mathcal{I})$.*

Assumption 2 implies that f is distributed by a GP $\mathcal{GP}(0, \sigma_0(\mathbf{x}, \mathbf{x}'))$ with prior mean 0 and covariance $\sigma_0(\mathbf{x}, \mathbf{x}') \triangleq \sum_{\mathcal{I} \in \mathcal{U}} \sigma_0^\mathcal{I}(\mathbf{x}^\mathcal{I}, \mathbf{x}'^\mathcal{I})$. It follows that for any subset $\mathcal{I} \subseteq \mathcal{S}$ of the input components of any input $\mathbf{x}$ and input queries $\mathbf{x}_1, \ldots, \mathbf{x}_t$, the prior distribution of $(f_\mathcal{I}(\mathbf{x}^\mathcal{I}), f(\mathbf{x}_1), \ldots, f(\mathbf{x}_t))^\top$ is a Gaussian. Then, given a column vector $\mathbf{y} \triangleq (y_i)_{i=1,\ldots,t}^\top$ of noisy outputs observed from evaluating f at the selected input queries $\mathbf{x}_1, \ldots, \mathbf{x}_t$ after t iterations, the posterior distribution of the output of the factor function $f_\mathcal{I}$ at some input $\mathbf{x}^\mathcal{I}$ in iteration $t+1$ is a Gaussian $\mathcal{N}(f_\mathcal{I}(\mathbf{x}^\mathcal{I})|\mu_t^\mathcal{I}(\mathbf{x}^\mathcal{I}), \sigma_t^\mathcal{I}(\mathbf{x}^\mathcal{I})^2)$ with the following posterior mean and variance:

$$\begin{aligned}
\mu_t^\mathcal{I}(\mathbf{x}^\mathcal{I}) &\triangleq \mathbf{k}_\mathbf{x}^{\mathcal{I}^\top}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{y}, \\
\sigma_t^\mathcal{I}(\mathbf{x}^\mathcal{I})^2 &\triangleq \sigma_0^\mathcal{I}(\mathbf{x}^\mathcal{I}, \mathbf{x}^\mathcal{I}) - \mathbf{k}_\mathbf{x}^{\mathcal{I}^\top}(\mathbf{K} + \sigma_n^2\mathbf{I})^{-1}\mathbf{k}_\mathbf{x}^\mathcal{I}
\end{aligned} \quad (1)$$

where $\mathbf{k}_\mathbf{x}^\mathcal{I} \triangleq (\sigma_0^\mathcal{I}(\mathbf{x}^\mathcal{I}, \mathbf{x}_i^\mathcal{I}))_{i=1,\ldots,t}^\top$ and $\mathbf{K} \triangleq (\sigma_0(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1,\ldots,t}$. Using (1), we can naively adopt

the HBO algorithm of Kandasamy, Schneider, and Póczos (2015) (Section 2.2) by independently maximizing a separate local acquisition function for every corresponding factor function $f_{\mathcal{I}}$. But, this does not guarantee compatibility of the inputs selected by independently maximizing each local acquisition function due to their common input components. So, we instead propose to jointly maximize them using the following additive acquisition function:

$$\sum_{\mathcal{I} \in \mathcal{U}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}), \quad \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \triangleq \mu_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) + \beta_{t+1}^{1/2} \sigma_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \quad (2)$$

which appears, with high probability, to bound the global maximum $f(\mathbf{x}_*) = \max_{\mathbf{x} \in \mathcal{X}} \sum_{\mathcal{I}} f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ from above, as shown later in Section 4. Intuitively, (2) is similar to the GP-UCB acquisition function (Srinivas et al. 2010) in the sense that both are exploiting the GP posterior mean of f to select the next input query since it can be shown that the sum of GP posterior means of the outputs of all factor functions is equal to the GP posterior mean of their sum (i.e., output of f). On the other hand, unlike GP-UCB, our proposed acquisition function (2) uses the sum of GP posterior variances of the outputs of all factor functions instead of the GP posterior variance of their sum (i.e., output of f) in order to construct an upper bound on the global maximum. This interestingly allows (2) to be optimized efficiently in a decentralized manner (Section 3.3) and at the same time preserves the asymptotic optimality of our DEC-HBO algorithm (Section 4).

3.3 Decentralized HBO (DEC-HBO)

When the interdependent effects of the majority of input components on the output of f are indirect, our proposed acquisition function (2) can in fact be represented by a sparse factor graph and hence be efficiently optimized in a decentralized manner. To do this, let (2) be represented by a factor graph with each factor and variable node denoting, respectively, a different local acquisition function and input component such that every edge connecting a factor node to some variable node implies a local acquisition function involving the participation of some input component. The following message passing protocol between the factor and variable nodes can then be used to optimize $\sum_{\mathcal{I} \in \mathcal{U}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ (2) via *dynamic programming* (DP):

Message Passing Protocol. In iteration $t + 1$, let $m_{\varphi_t^{\mathcal{I}} \to \mathbf{x}^{(i)}}(h)$ and $m_{\mathbf{x}^{(i)} \to \varphi_t^{\mathcal{I}}}(h)$ denote messages to be passed from a factor node $\varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ (i.e., a local acquisition function) to a variable node $\mathbf{x}^{(i)}$ (i.e., component $i \in \mathcal{I}$ of its input $\mathbf{x}^{\mathcal{I}}$) and from $\mathbf{x}^{(i)}$ back to $\varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$, respectively. Given $\mathbf{x}^{(i)} \triangleq h$,

$$m_{\varphi_t^{\mathcal{I}} \to \mathbf{x}^{(i)}}(h) \triangleq \max_{\mathbf{h}^{\mathcal{I} \backslash i} \in \mathcal{D}(\mathbf{x}^{\mathcal{I} \backslash i})} \Delta_{\varphi_t^{\mathcal{I}}}^{-i}(\mathbf{h}^{\mathcal{I} \backslash i}) + \varphi_t^{\mathcal{I}}(\mathbf{h}^{\mathcal{I} \backslash i}, h),$$

$$m_{\mathbf{x}^{(i)} \to \varphi_t^{\mathcal{I}}}(h) \triangleq \sum_{\mathcal{I}' \in \mathcal{A}(i) \backslash \{\mathcal{I}\}} m_{\varphi_t^{\mathcal{I}'} \to \mathbf{x}^{(i)}}(h)$$

$$(3)$$

where $\mathcal{I} \backslash i$ is used in place of $\mathcal{I} \backslash \{i\}$ to ease notations, $\Delta_{\varphi_t^{\mathcal{I}}}^{-i}(\mathbf{h}^{\mathcal{I} \backslash i}) \triangleq \sum_{j \in \mathcal{I} \backslash i} m_{\mathbf{x}^{(j)} \to \varphi_t^{\mathcal{I}}}(\mathbf{h}^{(j)})$, $\mathcal{D}(\mathbf{x}^{\mathcal{I} \backslash i})$ is the domain of input $\mathbf{x}^{\mathcal{I} \backslash i}$, $\mathbf{h}^{(j)}$ denotes component j of $\mathbf{h}^{\mathcal{I} \backslash i}$,

and $\mathcal{A}(i) \triangleq \{\mathcal{I}' | f_{\mathcal{I}'}(\mathbf{x}^{\mathcal{I}'})$ is a factor function $\wedge i \in \mathcal{I}'\}$. These message updates (3) can be performed simultaneously, which yields a fully decentralized optimization algorithm where full knowledge of the results of optimization are accessible to all nodes. This can be perceived as a concurrent learning process where each node tries to perfect its own DP perspective through exchanging information with its immediate neighbors. As the messages are passed back and forth simultaneously among nodes, their individual perspectives are updated and steadily converge to an equilibrium that maximizes $\sum_{\mathcal{I} \in \mathcal{U}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ (2). Our decentralized optimization algorithm yields a huge computational advantage in a sparse factor graph since the cost of evaluating each message in any iteration is only as expensive as iterating through the input domain of the local acquisition function involving the largest number of input components (i.e., maximum factor size), which is usually much smaller than the entire input domain. The overall computational cost at each node thus reduces at an exponential rate in the ratio between the sizes of the original input domain and that of such a local acquisition function.

Upon convergence[1], the final message $m_{\varphi_t^{\mathcal{I}'} \to \mathbf{x}^{(i)}}(h)$ from every factor node $\varphi_t^{\mathcal{I}'}(\mathbf{x}^{\mathcal{I}'})$ to a variable node $\mathbf{x}^{(i)}$ (i.e., component $i \in \mathcal{I}'$ of its input $\mathbf{x}^{\mathcal{I}'}$) is the maximum value achieved by optimizing the sum of all remaining factor nodes, except $\varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$, over the remaining variable nodes $\mathbf{x}^{\mathcal{S} \backslash i}$ while fixing $\mathbf{x}^{(i)} = h$. As such, component i of the maximizer $\mathbf{x}_{t+1} \triangleq \arg\max_{\mathbf{x} \in \mathcal{D}} \sum_{\mathcal{I}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ can be computed using an arbitrary variable-factor pair $(\mathbf{x}^{(i)}, \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}))$:

$$\mathbf{x}_{t+1}^{(i)} \triangleq \arg\max_{h \in \mathcal{D}(\mathbf{x}^{(i)})} \max_{\mathbf{h}^{\mathcal{I} \backslash i} \in \mathcal{D}(\mathbf{x}^{\mathcal{I} \backslash i})} \varphi_t^{\mathcal{I}}(\mathbf{h}^{\mathcal{I} \backslash i}, h) + m_{\mathbf{x}^{(i)} \to \varphi_t^{\mathcal{I}}}(h)$$

$$(4)$$

for all $i \in \mathcal{I}$ where $\mathcal{D}(\mathbf{x}^{(i)})$ denotes the domain of input component $\mathbf{x}^{(i)}$. Note that (4) only operates in the domains $\mathcal{D}(\mathbf{x}^{(i)})$ and $\mathcal{D}(\mathbf{x}^{\mathcal{I} \backslash i})$. Its time complexity is thus bounded by the cost of iterating through the input domain of the local acquisition function involving the largest number of input components (i.e., maximum factor size), as analyzed in Appendix A. Our decentralized optimization algorithm is similar in spirit to the max-sum algorithm for solving the well-known distributed constraint optimization problem (Leite, Enembreck, and Barthès 2014) operating in discrete input domains and in fact adapts it to maximize our additive acquisition function (2) over a continuous input domain. Such an adaptation is achieved by scheduling an iterative process of domain discretization with increasing granularity. Nevertheless, our DEC-HBO algorithm is guaranteed to be asymptotically optimal, as further detailed in Section 4. DEC-HBO requires a specification of the input partition $\mathcal{U} \subseteq 2^{\mathcal{S}}$ that underlies our additive acquisition function (2), which can be learned from data (Appendix B).

---

[1]Though the convergence of our message passing protocol is only guaranteed when the factor graph is a tree, it empirically converges to a competitive performance quickly (Section 5.2). To guarantee the performance for a general factor graph, a bounded variant of the max-sum algorithm (Rogers et al. 2011) can be considered.

4 Asymptotic Optimality

This section analyzes the asymptotic optimality of our proposed DEC-HBO algorithm (Section 3.3) that is powered by our additive acquisition function (2). We will first present a simplified version of our analysis in a simple setting with discrete input domains and then generalize it to handle a more realistic setting with continuous input domains.

Discrete Input Space. To guarantee the asymptotic optimality of DEC-HBO, it suffices to show that its average regret approaches zero in the limit (i.e., $\lim_{n \to \infty} R_n/n = 0$). To achieve this, we will first construct upper bounds for the instantaneous regrets $r_t = f(\mathbf{x}_*) - f(\mathbf{x}_t)$ (Theorem 1) and then combine these results to establish a sub-linear upper bound for the cumulative regret R_n (Theorem 2).

Theorem 1. *Given $\delta \in (0, 1)$, let $\beta_t \triangleq 2\log(|\mathcal{D}||\mathcal{U}|\pi_t/\delta)$ with $\pi_t \triangleq \pi^2 t^2/6$.*
$\Pr(\forall \mathbf{x} \in \mathcal{D}, t \in \mathbb{N} \ r_t \leq 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})) \geq 1 - \delta.$

Its proof is in Appendix C. Theorem 1 establishes a universal bound that holds simultaneously for all instantaneous regrets r_t with an arbitrarily high confidence and is adjustable via parameter β_t to trade off between exploitation vs. exploration. More importantly, Theorem 1 immediately implies the following bound on the cumulative regret R_n based on the notion of *maximum information gain* below:

Definition 1. *Let $\mathcal{A} \triangleq \{\mathbf{x}_t\}_{t=1,\ldots,n} \subseteq \mathcal{D}$ and $\mathbf{f}_{\mathcal{A}}^{\mathcal{I}} \triangleq (f_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}))_{t=1,\ldots,n}^{\top}$. Suppose that a column vector $\mathbf{y}_{\mathcal{A}}^{\mathcal{I}} \triangleq (y_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}))_{t=1,\ldots,n}^{\top}$ of noisy outputs $y_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) = f_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \epsilon$ can be observed from evaluating the latent factor function $f_{\mathcal{I}}$ at input queries $\mathbf{x}_1^{\mathcal{I}}, \ldots, \mathbf{x}_n^{\mathcal{I}}$, respectively. Then, the maximum information gain about $\mathbf{f}_{\mathcal{A}}^{\mathcal{I}}$ given $\mathbf{y}_{\mathcal{A}}^{\mathcal{I}}$ can be characterized in terms of their Shannon mutual information: $\gamma_n^{\mathcal{I}} \triangleq \max_{\mathcal{A}:\mathcal{A} \subseteq \mathcal{D}, |\mathcal{A}|=n} \mathbf{I}(\mathbf{f}_{\mathcal{A}}^{\mathcal{I}}, \mathbf{y}_{\mathcal{A}}^{\mathcal{I}})$. The total maximum information gain is $\gamma_n \triangleq \sum_{\mathcal{I} \in \mathcal{U}} \gamma_n^{\mathcal{I}}$.*

The total maximum information gain γ_n defined above can then be exploited to bound the cumulative regret R_n:

Theorem 2. *Given $\delta \in (0, 1)$, $\Pr(R_n \leq (Cn\beta_n\gamma_n)^{1/2} \in o(n)) \geq 1 - \delta$ under some mild condition on $\sigma_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})^2$ where C is some constant defined in Appendix D.*

Its proof is in Appendix D.

Remark Kandasamy, Schneider, and Póczos (2015) have attempted to bound the cumulative regret R_n in terms of the maximum information gain $\mathbf{I}(\mathbf{f}_{\mathcal{A}}, \mathbf{y}_{\mathcal{A}})$ of the objective function f directly (i.e., $\mathbf{f}_{\mathcal{A}} \triangleq (f(\mathbf{x}_t))_{t=1,\ldots,n}^{\top}$ and $\mathbf{y}_{\mathcal{A}} \triangleq (y_t)_{t=1,\ldots,n}^{\top}$) for an extreme special case of our work where the effects of all input components on the output of f are statistically independent. The validity of their proof appears to depend on the assumption that the sum of the GP posterior variances of the outputs of all factor functions is less than or equal to the GP posterior variance of the output of the objective function f, which is flawed as recently acknowledged by the authors in (Kandasamy, Schneider, and Póczos 2016). Our analysis (Appendix D) instead uses a different quantity (see Definition 1) to bound the cumulative regret R_n and therefore avoid making such a flawed assumption.

Theorem 2 implies $\lim_{n \to \infty} R_n/n = 0$ which guarantees the desired asymptotic optimality of DEC-HBO (hence, no regret) with an arbitrarily high confidence. However, when the input space is infinite, $\langle \beta_t \rangle_t$ tend to infinity and hence void the above analysis. To address this caveat, we extend our analysis to handle infinite, continuous input spaces by assuming Lipschitz continuity of the objective function f.

Continuous Input Space. To extend our previous analysis to the setting with infinite, continuous input spaces, we assume objective function f to be L-Lipschitz continuous:

Assumption 3. *There exist constants $a, b, L > 0$ such that $\Pr(\forall \mathbf{x}, \mathbf{x}' \in \mathcal{D} \ |f(\mathbf{x}) - f(\mathbf{x}')| \leq L\|\mathbf{x} - \mathbf{x}'\|_1) \geq 1 - a|\mathcal{U}|\exp(-L^2/b^2).$*

The Lipschitz continuity of f can be exploited to establish an upper bound on the cumulative regret R_n without relying on the finiteness of the input space. Intuitively, the key idea is to repeat the above finite-case analysis for a finite discretization of the input space by first establishing regret bounds for these discretized inputs. The resulting bounds can then be related to an arbitrary input by using the Lipschitz continuity of f in Assumption 3 to bound its output in terms of that of its closest discretized input with high probability. As such, it can be shown that if the discretization (especially its granularity) is carefully designed, then the cumulative regret R_n of our DEC-HBO algorithm can be bounded in terms of the above Lipschitz constants as well as the discretization granularity instead of the (infinite) size of the input space:

Theorem 3. *Given $\delta \in (0, 1)$, there exists a monotonically increasing sequence $\langle \beta_t \rangle_t$ such that $\beta_t \in \mathcal{O}(\log t)$ and $\Pr(R_n \leq (Cn\beta_n\gamma_n)^{1/2} + \pi^2/6 \in o(n)) \geq 1 - \delta$ under some mild condition on $\sigma_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})^2$.*

Its proof is in Appendix E. Theorem 3 concludes our analysis for the infinite case which is similar to Theorem 2 for the finite case: The upper bound on the cumulative regret R_n is sub-linear in n, which guarantees that its average regret approaches zero in the limit. So, DEC-HBO is asymptotically optimal with an arbitrarily high confidence.

5 Experiments and Discussion

This section empirically evaluates the performance of our DEC-HBO algorithm on an extensive benchmark comprising three synthetic functions: Shekel (4-dimensional), Hartmann (6-dimensional), Michalewicz (10-dimensional) (Section 5.1), and two high-dimensional optimization problems involving hyperparameter tuning for popular machine learning models such as sparse GP (Snelson and Ghahramani 2007) and convolutional neural network modeling two real-world datasets: Physicochemical properties of protein tertiary structure (Rana 2013) and CIFAR-10 (Section 5.2).

5.1 Optimizing Synthetic Functions

This section empirically compares the performance of DEC-HBO (with maximum factor size of 2 (MF2) or 3 (MF3)) with that of the state-of-the-art HBO algorithms like ADD-GP-UCB (Kandasamy, Schneider, and Póczos 2015), ADD-MES-G and ADD-MES-R (Wang and Jegelka 2017), and
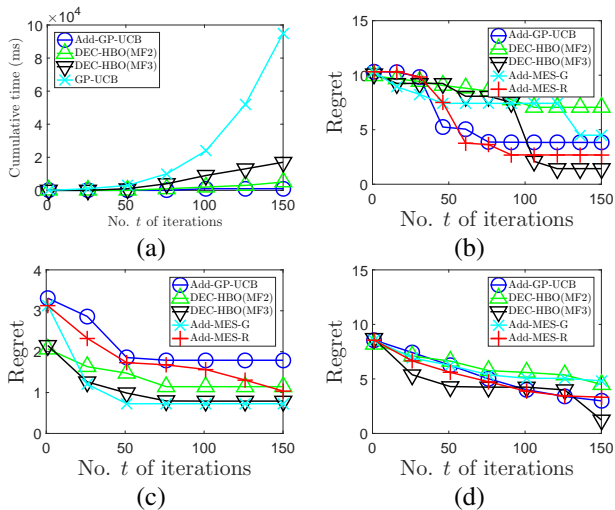
Figure 1: (a) Graphs of cumulative incurred time of tested algorithms vs. no. t of BO iterations for Shekel function, graphs of regret (i.e., $\min_{i=1}^{t} f(\mathbf{x}_i) - f(\mathbf{x}_*)$) achieved by tested HBO algorithms vs. no. t of BO iterations for (b) Shekel, (c) Hartmann, and (d) Michalewicz functions.

HBO	Hartmann	Shekel	Michalewicz
MF2	1.1436	7.0538	4.4944
MF3	**0.7904**	**1.4295**	**1.2367**
Add-GP-UCB	1.7898	3.8338	2.9870
Add-MES-G	**0.7268**	4.4951	4.8227
Add-MES-R	1.0372	2.6858	3.3296
REMBO	1.5843	5.1677	4.0524

Table 1: Regrets achieved by tested HBO algorithms for Hartmann, Shekel, and Michalewicz functions.

REMBO (Wang et al. 2013) in optimizing the Shekel, Hartmann, and Michalewicz functions (Appendix F).

Fig. 1 reports results of the regret (i.e., $\min_{i=1}^{t} f(\mathbf{x}_i) - f(\mathbf{x}_*)$) of the tested algorithms averaged over 5 runs, each of which comprises 150 iterations. For DEC-HBO (MF2) and DEC-HBO (MF3), each BO iteration involves 30 iterations of max-sum. For clarity, Table 1 further reports the final converged regrets achieved by the tested HBO algorithms including REMBO[2]. The results show that in general, our DEC-HBO variants perform competitively with the other state-of-the-art HBO algorithms for all synthetic functions. Interestingly, it can also be observed that DEC-HBO (MF3) consistently outperforms DEC-HBO (MF2) and Add-GP-UCB (corresponding to the DEC-HBO variant with max. factor size of 1) for all synthetic functions with input dimension $d \geq 4$. This highlights the importance of exploiting the interdependent effects of various input components on the output of f. In most cases, DEC-HBO (MF3) also outperforms Add-MES-G, Add-MES-R, and REMBO with the difference being most pronounced for the 10-dimensional Michalewicz function. This further indicates the efficacy of DEC-HBO when applied to higher-dimensional optimization problems and asserts that preserving the interdependent effects of various input components on the output of f is necessary. Fig. 1a also shows the cumulative incurred time of GP-UCB, Add-GP-UCB, DEC-HBO (MF2), and DEC-HBO (MF3) in optimizing the Shekel function. The results reveal that DEC-HBO performs competitively in terms of time efficiency with Add-GP-UCB with minimal increase in incurred time over 150 iterations in exchange for a signifi-

---

[2]The performance of REMBO is not plotted in Figs. 1 and 2 to ease clutter as it requires much more iterations for convergence using the authors' implementation: github.com/ziyuw/rembo.

cant improvement in terms of BO performance. In contrast, GP-UCB incurs much more time than our DEC-HBO variants, thus asserting the computational advantage of our decentralized optimization algorithm.

5.2 Optimizing Hyperparameters of ML Models

This section demonstrates the effectiveness of DEC-HBO in tuning the hyperparameters of two ML models like the sparse *partially independent conditional* (PIC) approximation of GP model (Snelson and Ghahramani 2007) and *convolutional neural network* (CNN). The goal is to find the optimal configuration of (a) kernel hyperparameters and inducing inputs for which PIC predicts well for the physicochemical properties of protein tertiary structure dataset (Rana 2013) and (b) network hyperparameters for which CNN classifies well for the CIFAR-10 dataset. These PIC and CNN hyperparameter tuning tasks are detailed as follows:

PIC. The PIC model is trained using the physicochemical properties of protein tertiary structure dataset (Rana 2013) which has 45730 instances, each of which contains $\kappa = 9$ attributes describing the physicochemical properties of a protein residue and its size (in armstrong) to be predicted. 95% and 5% of the dataset are used as training and test data, respectively. The training data is further divided into 5 equal folds. The goal is to find a hyperparameter configuration that minimizes the *root mean square error* (RMSE) of PIC's prediction on the test data. This is achieved via BO using the 5-fold validation performance as a noisy estimate of the real performance on the test data. Specifically, for every input query of hyperparameters, the corresponding PIC model separately predicts on each of these folds (validation data), having trained on the remaining folds (effective training data). The averaged prediction error over these 5 folds is then returned to the HBO algorithm to update the acquisition function for selecting the next input query of hyperparameters. Every such input query contains $2 + \kappa + \nu \times \kappa$ hyperparameters which include the signal and noise variances, κ length-scales of the squared exponential kernel, and $\nu = 200$ inducing inputs of dimension κ each.

CNN. The CNN model is trained using the CIFAR-10 object recognition dataset which has 50000 training images and 10000 test images, each of which belongs to one of the ten classes. 5000 training images are set aside as the validation data. Similar to PIC, the goal is to find a hyperparameter configuration that minimizes the classification error of CNN on the test data, which is likewise achieved via BO using the
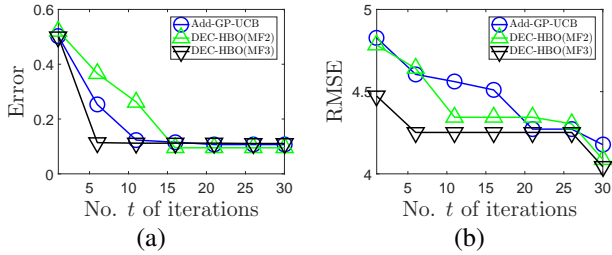
Figure 2: Graphs of (a) classification error of CNN and (b) RMSE of PIC's prediction vs. no. t of BO iterations.

HBO	MF2	MF3	Add-GP-UCB	REMBO
PIC	4.0879	4.0437	4.1774	9.9100
CNN	0.0950	0.1107	0.1068	-

Table 2: Minimum errors achieved by tested HBO algorithms for hyperparameter tuning of PIC and CNN.

performance on the validation data to estimate the real performance on the test data[3]. The six CNN hyperparameters to be optimized in our experiments include the learning rate of SGD in the range of $[10^{-5}, 1]$, three dropout rates in the range of $[0, 1]$, batch size in the range of $[100, 1000]$, and number of learning epochs in the range of $[100, 1000]$.

Fig. 2 shows results of the performance of DEC-HBO variants in comparison to that of ADD-GP-UCB for hyperparameter tuning of PIC and CNN trained with real-world datasets. Table 2 further reports the final converged RMSE achieved by the tested HBO algorithms including REMBO for PIC hyperparameter tuning[4]. It can be observed that in general, our DEC-HBO variants outperform ADD-GP-UCB and REMBO. Interestingly, for PIC hyperparameter tuning, the performance difference is also more pronounced in the early BO iterations, which suggests that DEC-HBO excels in time-constrained high-dimensional optimization problems and preserving the interdependent effects of various input components on the output of f boosts the performance of GP-UCB-based algorithms. This is consistent with our earlier observations in Section 5.1. The poor performance of REMBO as compared to DEC-HBO is expected since it only considers input hyperparameter queries generated from a random low-dimensional embedding of the input space, which severely restricts the expressiveness of PIC model.

6 Conclusion

This paper describes a novel DEC-HBO algorithm that, in contrast to existing HBO algorithms, can exploit the interdependent effects of various input components on the output of the unknown objective function f for boosting the BO performance and still preserve scalability in the number of input dimensions as well as guarantee no-regret performance asymptotically (see Remark in Section 4). To realize this, we propose a sparse yet rich factor graph representation of f to be exploited for designing an acquisition function that can be similarly represented by a sparse factor graph and hence be efficiently optimized in a decentralized manner using a class of distributed message passing algorithms. Empirical evaluation on both synthetic and real-world experiments show that our DEC-HBO algorithm performs competitively to the state-of-the-art centralized BO and HBO algorithms while providing a significant computational advantage for high-dimensional optimization problems. For future work, we plan to generalize DEC-HBO to batch mode (Daxberger and Low 2017) and the nonmyopic context by appealing to existing literature on nonmyopic BO (Ling, Low, and Jaillet 2016) and active learning (Cao, Low, and Dolan 2013; Hoang et al. 2014a; 2014b; Low, Dolan, and Khosla 2008; 2009; 2011) as well as to be performed by a multi-robot team to find hotspots in environmental sensing/monitoring by seeking inspiration from existing literature on multi-robot active sensing/learning (Chen, Low, and Tan 2013; Chen et al. 2012; 2015; Low et al. 2012; Ouyang et al. 2014). For applications with a huge budget of function evaluations, we like to couple DEC-HBO with the use of parallel/distributed (Chen et al. 2013; Hoang, Hoang, and Low 2016; Low et al. 2015) and online/stochastic (Hoang, Hoang, and Low 2015; 2017; Xu et al. 2014) sparse GP models to represent the belief of f efficiently.

References

Bergstra, J.; Yamins, D.; and Cox, D. D. 2013. Making a science of model search: Hyperparameter optimization and hundreds of dimensions for vision architectures. In *Proc. ICML*, 115–123.

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.; 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

---

[3]We use the same CNN structure as the example code of keras: github.com/fchollet/keras/ and replace the default optimizer in their code by *stochastic gradient descent* (SGD).

[4]The performance of DEC-HBO is not compared with that of REMBO (implemented in MATLAB) for CNN hyperparameter tuning as the CNN code in keras cannot be converted to MATLAB.

process-based decentralized data fusion and active sensing for mobility-on-demand system. In *Proc. RSS*.

Daxberger, E., and Low, K. H. 2017. Distributed batch Gaussian process optimization. In *Proc. ICML*, 951–960.

Djolonga, J.; Krause, A.; and Cevher, V. 2013. High-dimensional Gaussian process bandits. In *Proc. NIPS*.

Duvenaud, D. K.; Nickisch, H.; and Rasmussen, C. E. 2011. Additive Gaussian processes. In *Proc. NIPS*, 226–234.

Gardner, J. R.; Guo, C.; Weinberger, K. Q.; Garnett, R.; and Grosse, R. 2017. Discovering and exploiting additive structure for Bayesian optimization. In *Proc. AISTATS*.

González, J.; Longworth, J.; James, D.; and Lawrence, N. 2014. Bayesian optimization for synthetic gene design. In *NIPS Workshop on Bayesian Optimization in Academia and Industry*.

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

Hoang, T. N.; Low, K. H.; Jaillet, P.; and Kankanhalli, M. 2014a. Active learning is planning: Nonmyopic ϵ-Bayes-optimal active learning of Gaussian processes. In *Proc. ECML/PKDD Nectar Track*, 494–498.

Hoang, T. N.; Low, K. H.; Jaillet, P.; and Kankanhalli, M. 2014b. 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.

Hoang, T. N.; Hoang, Q. M.; and Low, K. H. 2016. A distributed variational inference framework for unifying parallel sparse Gaussian process regression models. In *Proc. ICML*, 382–391.

Hoang, Q. M.; Hoang, T. N.; and Low, K. H. 2017. A generalized stochastic variational Bayesian hyperparameter learning framework for sparse spectrum Gaussian process regression. In *Proc. AAAI*, 2007–2014.

Hornby, G. S.; Globus, A.; Linden, D. S.; and Lohn, J. D. 2006. Automated antenna design with evolutionary algorithms. In *Proc. AIAA Space Conference*.

Kandasamy, K.; Schneider, J.; and Póczos, B. 2015. High dimensional Bayesian optimization and bandits via additive models. In *Proc. ICML*.

Kandasamy, K.; Schneider, J.; and Póczos, B. 2016. High-dimensional Bayesian optimization and bandits via additive models. arXiv:1503.01673.

Krähenbühl, P., and Koltun, V. 2011. Efficient inference in fully connected CRFs with Gaussian edge potentials. In *Proc. NIPS*.

Leite, A. R.; Enembreck, F.; and Barthès, J.-P. A. 2014. Distributed constraint optimization problems: Review and perspectives. *Expert Systems with Applications* 41:5139–5157.

Li, C.-L.; Kandasamy, K.; Póczos, B.; and Schneider, J. 2016. High dimensional Bayesian optimization via restricted projection pursuit models. In *Proc. AISTATS*.

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. In *Proc. AAAI*, 1860–1866.

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.

Naveršnik, K., and Rojnik, K. 2012. Handling input correlations in pharmacoeconomic models. *Value in Health* 15:540–549.

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*.

Rana, P. S. 2013. Physicochemical properties of protein tertiary structure dataset. `http://archive.ics.uci.edu/ml/datasets/`.

Rogers, A.; Farinelli, A.; Stranders, R.; and Jennings, N. R. 2011. Bounded approximate decentralised coordination via the max-sum algorithm. *AIJ* 175(2):730–759.

Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.; and de Freitas, N. 2016. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE* 104(1):148–175.

Snelson, E. L., and Ghahramani, Z. 2007. Local and global sparse Gaussian process approximation. In *Proc. AISTATS*.

Snoek, J.; Hugo, L.; and Adams, R. P. 2012. Practical Bayesian optimization of machine learning algorithms. In *Proc. NIPS*, 2960–2968.

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.

Wang, Z., and Jegelka, S. 2017. Max-value entropy search for efficient Bayesian optimization. In *Proc. ICML*.

Wang, Z.; Zoghi, M.; Hutter, F.; Matheson, D.; and de Freitas, N. 2013. Bayesian optimization in high dimensions via random embeddings. In *Proc. IJCAI*, 1778–1784.

Wang, Z.; Li, C.; Jegelka, S.; and Kohli, P. 2017. Batched high-dimensional Bayesian optimization via structural kernel learning. In *Proc. ICML*, 3656–3664.

Xu, N.; Low, K. H.; Chen, J.; Lim, K. K.; and Ozgul, E. B. 2014. GP-Localize: Persistent mobile robot localization us-

ing online sparse Gaussian process observation model. In *Proc. AAAI*, 2585–2592.

A Time Complexity Analysis

Let the numbers of BO iterations and decentralized optimization iterations executed by our DEC-HBO algorithm be N_o and N_m, respectively. In BO iteration t, there are $|\mathcal{U}|$ factor nodes and d variable nodes operating independently. The time complexity per BO iteration for each type of node is detailed below:

Factor Node. In BO iteration t, the input space of factor $\mathcal{I}$ is discretized into a grid of size $\mathcal{D}_t^{|\mathcal{I}|}$. The corresponding factor function $\varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ (2) then needs to be evaluated over $\mathcal{D}_t^{|\mathcal{I}|}$ discretized inputs $\mathbf{x}^{\mathcal{I}}$. For each input $\mathbf{x}^{\mathcal{I}}$, the time complexity is $\mathcal{O}(t^3|\mathcal{I}|)$ since the number of past function evaluations is $\mathcal{O}(t)$ (i.e., $t_0 + t - 1$ where t_0 is a constant number of function evaluations prior to running the BO algorithm). Thus, precomputing $\varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ for all inputs $\mathbf{x}^{\mathcal{I}}$ incurs a total of $\mathcal{O}(\mathcal{D}_t^{|\mathcal{I}|}t^3|\mathcal{I}|)$ time. Once the precomputation is cached, computing each message $m_{\varphi_t^{\mathcal{I}} \to \mathbf{x}^{(i)}}(h)$ incurs $\mathcal{O}(|\mathcal{I}|\mathcal{D}_t^{|\mathcal{I}|-1})$ time (3). Since there are $|\mathcal{I}| \times \mathcal{D}_t^1$ such messages, the message computation (including factor precomputation) at this factor node over N_m max-sum iterations incurs a total of $\mathcal{O}(N_m|\mathcal{I}|^2\mathcal{D}_t^{|\mathcal{I}|} + \mathcal{D}_t^{|\mathcal{I}|}t^3|\mathcal{I}|) = \mathcal{O}(N_m\mathcal{D}_t^{|\mathcal{I}|}|\mathcal{I}|(t^3 + |\mathcal{I}|))$ time. At the end of the message iteration phase, each variable node sends its latest update of $m_{\mathbf{x}^{(i)} \to \varphi_t^{\mathcal{I}}}(h)$ to an arbitrary factor $\mathcal{I}$ in its neighborhood. The receiving factor then uses (4) to generate the optimal value for $\mathbf{x}^{(i)}$, which incurs $\mathcal{O}(\mathcal{D}_t^{|\mathcal{I}|})$ time per BO iteration. We assume the cost of sending and receiving messages between nodes are negligible and therefore omit them from the analysis for clarity.

Variable Node. For each decentralized optimization iteration, computing each message at a variable node incurs $\mathbf{x}^{(i)}$ is $\mathcal{O}(|\mathcal{A}(i)|)$ time. Since there are $|\mathcal{A}(i)| \times \mathcal{D}_t^1$ such messages, the message computation at this variable node over N_m iterations then incurs $\mathcal{O}(N_m|\mathcal{A}(i)|^2\mathcal{D}_t^1)$ time.

B Selecting Factor Graph Representation

The formulation of our distributed message passing algorithm in Section 3.3 requires a specification of the input partition $\mathcal{U} \subseteq 2^{\mathcal{S}}$ that underlies our additive acquisition function (2). This can be either specified manually by inspecting the data (Kandasamy, Schneider, and Póczos 2015) or learned from data (Gardner et al. 2017; Wang et al. 2017). We adopt the recent approach of Gardner et al. (2017) by associating each factor graph candidate with an equivalent kernel of the resulting additive GP (Duvenaud, Nickisch, and Rasmussen 2011). This allows (2) to be reformulated as a weighted average with respect to the posterior of $\mathcal{U}$ given the data $\mathfrak{D}_t \triangleq \{(\mathbf{x}_i, y_i)\}_{i=1,\ldots,t}$ of selected input queries and their noisy outputs observed from evaluating f after t

iterations:

$$\sum_{\mathcal{U}} p(\mathcal{U}|\mathfrak{D}_t) \sum_{\mathcal{I} \in \mathcal{U}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \approx k^{-1} \sum_{i=1}^{k} \sum_{\mathcal{I} \in \mathcal{U}_i} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \quad (5)$$

where $\mathcal{U}_1, \ldots, \mathcal{U}_k$ are i.i.d samples drawn from $p(\mathcal{U}|\mathfrak{D}_t)$ via MCMC sampling (Gardner et al. 2017). Interestingly, the RHS of (5) can be equivalently interpreted as a sum of augmented local acquisition functions $k^{-1}\varphi_t^{\mathcal{I}}(\mathbf{x})$ induced from subsets $\mathcal{I}$ of input components in the union of input partitions $\mathcal{U}_1 \cup \ldots \cup \mathcal{U}_k$. As such, the augmented additive acquisition function (5) can similarly be efficiently optimized in a decentralized manner using our distributed message passing algorithm in Section 3.3. It also allows the interdependent effects of different input components on the output of f to be discovered and simultaneously exploited to find the global maximizer (Gardner et al. 2017).

C Proof of Theorem 1

To prove Theorem 1, we first establish the following results:

Lemma 1. *Given* $\delta \in (0, 1)$*, let* $\beta_t \triangleq 2\log(|\mathcal{D}||\mathcal{U}|\pi_t/\delta)$ *with* $\pi_t = \pi^2 t^2/6$*. Then, with probability of at least* $1 - \delta$*,*

$$\left| f(\mathbf{x}) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$$

for all $\mathbf{x} \in \mathcal{D}$ *and* $t \in \mathbb{N}$ *where* $\mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ *and* $\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ *are previously defined in* (1).

Proof. For all $\mathbf{x} \in \mathcal{D}, \mathcal{I} \in \mathcal{U}$, and $t \in \mathbb{N}$,

$$f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \sim \mathcal{N}(\mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}), \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})^2) . \quad (6)$$

Let $r \triangleq (f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) - \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}))/\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$. Then, (6) implies $r \sim \mathcal{N}(0, 1)$ and hence, $\Pr(|r| \leq \beta_t^{1/2}) \geq 1 - \exp(-\beta_t/2)$. That is,

$$\Pr\left(|f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) - \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})| \leq \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})\right) \geq 1 - \exp(-\beta_t/2)$$

which, by applying the union bound over all tuples ($\mathbf{x} \in \mathcal{D}, \mathcal{I} \in \mathcal{U}, t \in \mathbb{N}$), implies

$$\Pr\left(\forall \mathbf{x}, \mathcal{I}, t \in \mathbb{N} \; |f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) - \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})| \leq \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})\right)$$
$$\geq 1 - |\mathcal{D}||\mathcal{U}| \sum_{t=1}^{\infty} \exp(-\beta_t/2) = 1 - \delta .$$

This means with probability of at least $1 - \delta$, the following inequalities hold simultaneously for all tuples $(\mathbf{x}, \mathcal{I}, t)$:

$$\begin{aligned} f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) &\leq \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) + \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) , \\ f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) &\geq \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) - \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) . \end{aligned} \quad (7)$$

Summing over $\mathcal{I} \in \mathcal{U}$ on both sides of the above inequalities yields

$$f(\mathbf{x}) = \sum_{\mathcal{I} \in \mathcal{U}} f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \leq \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) + \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) ,$$
$$f(\mathbf{x}) = \sum_{\mathcal{I} \in \mathcal{U}} f_{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \geq \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) - \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) .$$

That is, for all pairs of $(\mathbf{x}, t)$,

$$\left| f(\mathbf{x}) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) . \quad (8)$$

Since (7) holds simultaneously for all tuples $(\mathcal{I}, \mathbf{x}, t)$ with probability of at least $1 - \delta$, (8) also holds simultaneously for all pairs of $(\mathbf{x}, t)$ with probability of at least $1 - \delta$. $\square$

Lemma 2. *For all $t \in \mathbb{N}$, if*

$$\left| f(\mathbf{x}) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \quad (9)$$

for all $\mathbf{x} \in \mathcal{D}$, then $r_t \leq 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})$.

Proof. By definition, $\mathbf{x}_t = \arg\max_{\mathbf{x} \in \mathcal{D}} \sum_{\mathcal{I}} \varphi_t^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$ and hence, $\sum_{\mathcal{I}} \varphi_t^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \geq \sum_{\mathcal{I}} \varphi_t^{\mathcal{I}}(\mathbf{x}_*^{\mathcal{I}})$. This implies

$$\begin{aligned} &\sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \beta_t^{1/2} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \\ &\geq \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_*^{\mathcal{I}}) + \beta_t^{1/2} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_*^{\mathcal{I}}) \\ &\geq f(\mathbf{x}_*) \end{aligned}$$

where the second inequality follows directly from (9). Then,

$$\begin{aligned} r_t &= f(\mathbf{x}_*) - f(\mathbf{x}_t) \\ &\leq \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) - f(\mathbf{x}_t) . \end{aligned}$$
$$(10)$$

On the other hand, also by (9),

$$\sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) - f(\mathbf{x}_t) \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) . \quad (11)$$

Plugging (11) into (10) yields

$$r_t \leq 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) .$$

$\square$

Main Proof. Lemma 1 guarantees that (9) of Lemma 2 holds universally for all pairs of $(\mathbf{x}, t)$ with probability of at least $1 - \delta$. As such, Theorem 1 follows.

D Proof of Theorem 2

From Theorem 1,

$$\begin{aligned} \sum_{t=1}^{n} r_t^2 &\leq \sum_{t=1}^{n} 4\beta_t \left(\sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right)^2 \\ &\leq \sum_{t=1}^{n} \sum_{\mathcal{I} \in \mathcal{U}} 4\beta_t |\mathcal{U}| \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \end{aligned} \quad (12)$$

where the second inequality is due to the Cauchy-Schwarz inequality. To prove Theorem 2, we first introduce a mild assumption on the relationship between the posterior variance $\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$ conditioned on noisy outputs of f and its counterpart $\widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$ conditioned on noisy outputs of $f_{\mathcal{I}}$ (i.e.,

assuming hypothetically that they are available) which are perturbed by the same i.i.d. Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma_n^2)$, i.e.,

$$\widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \triangleq \sigma_0(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}}) - \mathbf{k}_{\mathbf{x}_t}^{\mathcal{I}^\top} (\mathbf{K}^{\mathcal{I}} + \sigma_n^2 \mathbf{I})^{-1} \mathbf{k}_{\mathbf{x}_t}^{\mathcal{I}}$$

where $\mathbf{k}_{\mathbf{x}_t}^{\mathcal{I}} \triangleq (\sigma_0^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_i^{\mathcal{I}}))_{i=1,\ldots,t-1}^\top$ and $\mathbf{K}^{\mathcal{I}} \triangleq (\sigma_0(\mathbf{x}_i^{\mathcal{I}}, \mathbf{x}_j^{\mathcal{I}}))_{i,j=1,\ldots,t-1}$.

Assumption 4. *For any sequence of input queries $\langle \mathbf{x}_t^{\mathcal{I}} \rangle_{t=1}^n$, there exists an arbitrary decreasing function $h : \mathbb{N} \to \mathbb{R}$ such that $\lim_{t \to \infty} h(t) > 0$ and*

$$h(t - 1) \, \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \leq \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$$

for $t \in \mathbb{N}$.

Assumption 4 allows us to bound the posterior variance $\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$ from the above by

$$\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \leq \frac{1}{h(t-1)} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 . \quad (13)$$

Plugging (13) into (12) yields

$$\begin{aligned} \sum_{t=1}^{n} r_t^2 &\leq \sum_{t=1}^{n} \sum_{\mathcal{I} \in \mathcal{U}} \frac{4\beta_t |\mathcal{U}|}{h(t-1)} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \\ &\leq F \sum_{t=1}^{n} \sum_{\mathcal{I} \in \mathcal{U}} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \end{aligned} \quad (14)$$

where $F \triangleq 4\beta_n |\mathcal{U}| / \lim_{t \to \infty} h(t)$ and the last inequality is due to the monotonic increase of β_t and $1/h(t)$ in t. Finally, to relate the total posterior variance $\sum_{t=1}^{n} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$ to the maximum information gain $\gamma_n^{\mathcal{I}}$ (Definition 1) for each factor function $f_{\mathcal{I}}$, we exploit the monotonically increasing property of the following function $g(s) = s / \log(1 + s)$ with $s = \sigma_n^{-2} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2$, as detailed below.

Specifically, since the function $g(s) = s / \log(1 + s)$ increases monotonically on $[0, \infty)$ and $\sigma_n^{-2} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \leq \sigma_n^{-2} \sigma_0^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}}) \leq \sigma_n^{-2} \sigma_s^2$ where[5] σ_s^2 denotes the signal variance, it follows that

$$\widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \leq \sigma_n^2 g(\sigma_n^{-2} \sigma_s^2) \log \left(1 + \sigma_n^{-2} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \right) .$$

Applying this result to (14) gives

$$\begin{aligned} \sum_{t=1}^{n} r_t^2 &\leq \beta_n C \sum_{\mathcal{I} \in \mathcal{U}} \frac{1}{2} \sum_{t=1}^{n} \log \left(1 + \sigma_n^{-2} \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})^2 \right) \\ &\leq \beta_n C \sum_{\mathcal{I} \in \mathcal{U}} \gamma_n^{\mathcal{I}} = C \beta_n \gamma_n \end{aligned}$$
$$(15)$$

where $C \triangleq 2\sigma_n^2 F g(\sigma_n^{-2} \sigma_s^2)$ and the second inequality follows directly from Lemma 5.3 of Srinivas et al. (2010). Finally, applying Cauchy-Schwarz to the LHS of (15) yields

$$R_n \triangleq \sum_{t=1}^{n} r_t \leq \sqrt{C n \beta_n \gamma_n} \leq \sqrt{C |\mathcal{U}|} \sqrt{n \beta_n \gamma_n^{\mathcal{I}_*}}$$

---

[5] The first inequality follows because $\widehat{\sigma}_0^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}}) = \sigma_0^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}})$ and the GP posterior variance is always non-increasing, i.e., $\widehat{\sigma}_0^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}}) \geq \widehat{\sigma}_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}, \mathbf{x}_t^{\mathcal{I}})$ for all $t \in \mathbb{N}$.

where $\gamma_n^{\mathcal{I}_*} \triangleq \max_{\mathcal{I}} \gamma_n^{\mathcal{I}}$. Since Theorem 5 of Srinivas et al. (2010) has shown a sublinear growth of the maximum information gain $\gamma_n^{\mathcal{I}_*}$ in n for the factor function $f_{\mathcal{I}_*}$, it follows that $\lim_{n \to \infty} \sqrt{n \beta_n \gamma_n^{\mathcal{I}_*}} / n = 0$. This consequently implies $\lim_{n \to \infty} R_n / n = 0$ or, equivalently, $R_n \leq (Cn \beta_n \gamma_n)^{1/2} \in o(n)$ when Theorem 1 holds. However, since Theorem 1 only holds with probability of at least $1 - \delta$, this implies $R_n \leq (Cn \beta_n \gamma_n)^{1/2} \in o(n)$ with probability of at least $1 - \delta$.

E   Proof of Theorem 3

As previously discussed in Section 4, Theorem 2 is unfortunately rendered void with infinite, continuous input spaces which cause $\langle \beta_t \rangle_t$ to tend to infinity. Fortunately, much of the proof of Lemma 1 can still be reused to bound $|f(\mathbf{x}) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})|$ for an arbitrary sequence of input queries selected by our DEC-HBO algorithm (Lemma 3) and an arbitrary finite discretization of the input space (Lemma 4). The real challenge is, however, to extend such an analysis to an arbitrary input not among the discretized inputs. As shall be elaborated later, this can be achieved by exploiting the Lipschitz continuity of the objective function f (see Assumption 3) to essentially bound the difference between outputs of f at two separate inputs in terms of their proximity, thus effectively "projecting" our bound beyond the discretization, which constitutes the central theme of this proof.

Lemma 3. *Given $\delta \in (0, 1)$, let $\beta_t \triangleq 2 \log(|\mathcal{U}| \pi_t / \delta)$, and $\langle \mathbf{x}_t \rangle_{t=1}^{\infty}$ denote an arbitrary sequence of input queries selected by our DEC-HBO algorithm. Then, with probability of at least $1 - \delta$,*

$$\left| f(\mathbf{x}_t) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \qquad (16)$$

for all $t \in \mathbb{N}$.

Proof. This result is similar to that of Lemma 1 but restricted to the (infinitely) countable sequence of input queries selected by our DEC-HBO algorithm. Most arguments established in the proof of Lemma 1 can be reused here, except that the union bound does not have to be applied to the entire input space, hence not causing $\langle \beta_t \rangle_t$ to blow up to infinity. In particular, using a similar argument to that of Lemma 1, for a given tuple $(\mathcal{I}, t)$,

$$\Pr\left(|f_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) - \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})| \leq \beta_t^{1/2} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right) \geq 1 - \exp(-\beta_t/2).$$

Applying the union bound over all tuples $(\mathcal{I}, t)$ yields

$$\Pr\left(\forall \mathcal{I}, t \in \mathbb{N} \; |f_{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) - \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}})| \leq \beta_t^{1/2} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right)$$
$$\geq 1 - |\mathcal{U}| \sum_{t=1}^{\infty} \exp(-\beta_t/2) = 1 - \delta \; .$$

$$(17)$$

Using a similar argument as that of Lemma 1, with probability of at least $1 - \delta$, the following inequality holds simultaneously for all $t \in \mathbb{N}$:

$$\left| f(\mathbf{x}_t) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \; .$$

Lemma 4. *Given $\delta \in (0, 1)$, let $\mathcal{D}_t$ denote an arbitrary finite discretisation of the input space, and $\beta_t \triangleq 2 \log(|\mathcal{D}_t| |\mathcal{U}| \pi_t / \delta)$. Then, with probability of at least $1 - \delta$,*

$$\left| f(\mathbf{x}) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}}) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}^{\mathcal{I}})$$

for all $\mathbf{x} \in \mathcal{D}_t$ and $t \in \mathbb{N}$.

Proof. This result follows directly by applying Lemma 1 to the finite discretization of the input space $\mathcal{D}_t$. $\square$

Putting together the above results of Lemmas 3 and 4, it is straightforward to see that the instantaneous regret $f(\mathbf{x}_*) - f(\mathbf{x}_t)$ can be bounded with high probability if $\mathbf{x}_* \in \mathcal{D}_t$. A tricky situation, however, arises when the global maximizer $\mathbf{x}_*$ is not among the discretized inputs. To resolve this, one possible approach is to relate the output of $\mathbf{x}_*$ to that of its closest discretized input. If this can be achieved, then we can exploit Lemma 4 to deliver a high-confidence bound on the instantaneous regret.

Specifically, suppose that we choose a finite discretization $\mathcal{D}_t$ of the original input space $\mathcal{D} = [0, r]^d$ in iteration t such that each dimension has τ_t uniformly-spaced discretized inputs; the exact value for τ_t will be determined later in Lemma 5. That is, $|\mathcal{D}_t| = \tau_t^d$ and $\|\mathbf{x} - [\mathbf{x}]_t\|_1 \leq rd/\tau_t$ for all $\mathbf{x} \in \mathcal{D}_t$ where $[\mathbf{x}]_t$ denotes the closest discretized input in $\mathcal{D}_t$ to $\mathbf{x}$. Under this setting, we are now ready to bound the true output of the global maximizer in terms of the predicted output (i.e., using the sum of GP posterior means of the outputs of all factor functions) of its closest discretized input with high confidence:

Lemma 5. *Given $\delta \in (0, 1)$, let $\beta_t \triangleq 2 \log(2|\mathcal{U}| \pi_t / \delta) + 2d \log(rdbt^2 \sqrt{\log(2|\mathcal{U}|a/\delta)})$ where a and b are the Lipschitz constants defined previously in Assumption 3. Then, with probability of at least $1 - \delta$,*

$$\left| f(\mathbf{x}_*) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) + \frac{1}{t^2}$$

for all $t \in \mathbb{N}$.

Proof. Setting $L \triangleq b\sqrt{\log(2|\mathcal{U}|a/\delta)}$ in Assumption 3 immediately implies that

$$|f(\mathbf{x}) - f(\mathbf{x}')| \leq b\sqrt{\log\left(\frac{2|\mathcal{U}|a}{\delta} \right)} \|\mathbf{x} - \mathbf{x}'\|_1$$

holds simultaneously for all $\mathbf{x}, \mathbf{x}' \in \mathcal{D}$ with probability of at least $1 - \delta/2$. Then,

$$|f(\mathbf{x}_*) - f([\mathbf{x}_*]_t)| \leq b\sqrt{\log\left(\frac{2|\mathcal{U}|a}{\delta} \right)} \|\mathbf{x}_* - [\mathbf{x}_*]_t\|_1$$
$$\leq \frac{rdb}{\tau_t} \sqrt{\log\left(\frac{2|\mathcal{U}|a}{\delta} \right)}$$

holds simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta/2$ such that the second inequality is due to the construction of $\mathcal{D}_t$ described earlier. Now, by choosing $\tau_t \triangleq rdbt^2\sqrt{\log(2|\mathcal{U}|a/\delta)}$,

$$|f(\mathbf{x}_*) - f([\mathbf{x}_*]_t)| \leq \frac{1}{t^2} \qquad (18)$$

holds simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta/2$. This also implies $|\mathcal{D}_t| = (rdbt^2\sqrt{\log(2|\mathcal{U}|a/\delta)})^d$. By applying this choice of $\mathcal{D}_t$ and $\delta/2$ to Lemma 4,

$$\left| f([\mathbf{x}_*]_t) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t)$$

$$(19)$$

holds simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta/2$ where $\beta_t = 2\log(2|\mathcal{D}_t||\mathcal{U}|\pi_t/\delta)$. By combining (18) and (19) and using the union bound, the following inequality holds simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta$:

$$\left| f([\mathbf{x}_*]_t) - \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) \right| \leq \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}([\mathbf{x}_*]_t^{\mathcal{I}}) + \frac{1}{t^2}$$

for $\beta_t = 2\log(2|\mathcal{D}_t||\mathcal{U}|\pi_t/\delta) = 2\log(2|\mathcal{U}|\pi_t/\delta) + 2d\log(rdbt^2\sqrt{\log(2|\mathcal{U}|a/\delta)})$. The second equality follows directly from the above choice of $\mathcal{D}_t$. $\qquad \square$

Using Lemma 5, we are now ready to bound the instantaneous regret $f(\mathbf{x}_*) - f(\mathbf{x}_t)$:

Lemma 6. *Given $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$r_t \leq 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \frac{1}{t^2}$$

for all $t \in \mathbb{N}$ where β_t is previously defined in Lemma 5.

Proof. By applying Lemma 5 for $\delta/2$, the following inequality

$$\begin{aligned} f(\mathbf{x}_*) &\leq \frac{1}{t^2} + \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) + \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}([\mathbf{x}_*^{\mathcal{I}}]_t) \\ &\leq \frac{1}{t^2} + \sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \end{aligned}$$

$$(20)$$

holds simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta/2$ where $\beta_t = 2\log(4|\mathcal{U}|\pi_t/\delta) + 2d\log(dt^2br\sqrt{\log(4|\mathcal{U}|a/\delta)})$. Note that the second inequality in (20) always hold with certainty due to the definition of $\mathbf{x}_t$. So,

$$\begin{aligned} &r_t \\ &= f(\mathbf{x}_*) - f(\mathbf{x}_t) \\ &\leq \sum_{\mathcal{I} \in \mathcal{U}} \left(\mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \beta_t^{1/2}\sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right) + \frac{1}{t^2} - f(\mathbf{x}_t) \\ &= \left(\sum_{\mathcal{I} \in \mathcal{U}} \mu_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) - f(\mathbf{x}_t) \right) + \beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \frac{1}{t^2} \\ &\leq 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}\left(\mathbf{x}_t^{\mathcal{I}}\right) + \frac{1}{t^2} \end{aligned}$$

$$(21)$$

where the first inequality holds with probability of at least $1-\delta/2$ due to (20) while the second inequality holds with the same probability of at least $1-\delta/2$ for $\beta_t \geq 2\log(2|\mathcal{U}|\pi_t/\delta)$ by Lemma 3.[6] As such, with $\beta_t = 2\log(4|\mathcal{U}|\pi_t/\delta) + 2d\log(dt^2br\sqrt{\log(4|\mathcal{U}|a/\delta)})$, we can conclude that both inequalities hold simultaneously for all $t \in \mathbb{N}$ with probability of at least $1 - \delta/2$ each. Applying union bound over them guarantees that (21) holds with probability of at least $1 - \delta$. $\qquad \square$

Main Proof. By replicating the arguments used in the proof of Theorem 2 in Appendix D (specifically, (12) to (15))[7], it can be derived that

$$4\sum_{t=1}^{n} \beta_t \left(\sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right)^2 \leq C\beta_n\gamma_n \qquad (22)$$

where C and γ_n are previously defined in Appendix D and Definition 1, respectively. By applying the Cauchy-Schwarz inequality on the LHS of (22),

$$\left(\sum_{t=1}^{n} 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) \right)^2 \leq Cn\beta_n\gamma_n$$

which immediately implies

$$\begin{aligned} \sum_{t=1}^{n} r_t &= \sum_{t=1}^{n} 2\beta_t^{1/2} \sum_{\mathcal{I} \in \mathcal{U}} \sigma_{t-1}^{\mathcal{I}}(\mathbf{x}_t^{\mathcal{I}}) + \sum_{t=1}^{n} \frac{1}{t^2} \\ &\leq \sqrt{Cn\beta_n\gamma_n} + \sum_{t=1}^{n} \frac{1}{t^2} \\ &\leq \sqrt{Cn\beta_n\gamma_n} + \frac{\pi^2}{6} \end{aligned}$$

where the last inequality follows from $\sum_{t=1}^{n}(1/t^2) \leq \sum_{t=1}^{\infty}(1/t^2) = \pi^2/6$. On the other hand, we have already established in Appendix D that $\lim_{n \to \infty} \sqrt{Cn\beta_n\gamma_n}/n = 0$, which means $\sqrt{Cn\beta_n\gamma_n} \in o(n)$. Since $\pi^2/6$ is a constant, $R_n = \sum_{t=1}^{n} r_t \leq \sqrt{Cn\beta_n\gamma_n} + \pi^2/6 \in o(n)$.

F   Synthetic Functions

The *Shekel* function is a 4-dimensional function over the hypercube $[0, 10]^4$ given by $f(\mathbf{x}) \triangleq -\sum_{i=1}^{10}(\beta_i + \sum_{j=1}^{4}(x_j - C_{ij})^2)^{-1}$ where $\mathbf{x} \triangleq (x_j)_{j=1,\dots,4}$, and $(\beta_i)_{i=1,\dots,10}$ and $(C_{ij})_{i=1,\dots,10,j=1,\dots,4}$ are given as constants. It has one global minimum $f(\mathbf{x}^*) = -10.5364$. The *Hartmann* function is a 6-dimensional function over the hypercube $[0, 1]^6$ given by $f(\mathbf{x}) \triangleq -\sum_{i=1}^{4} \alpha_i \exp(-\sum_{j=1}^{6} A_{ij}(x_j - P_{ij})^2)$ where $\mathbf{x} \triangleq (x_j)_{j=1,\dots,6}$, and $(\alpha_i)_{i=1,\dots,4}$, $(A_{ij})_{i=1,\dots,4,j=1,\dots,6}$, and $(P_{ij})_{i=1,\dots,4,j=1,\dots,6}$ are given as constants. It has one global minimum $f(\mathbf{x}^*) = -3.32237$. Lastly, the *Michalewicz* function is a 10-dimensional

---

[6]In Lemma 3, $\beta_t \geq 2\log(2|\mathcal{U}|\pi_t/\delta)$ is the minimum threshold for which (16) holds.

[7]Note that these arguments hold regardless of the choice of β_t. So, they can be reused in Theorem 3 even though our choice of β_t has changed.

function over the hypercube $[0, \pi]^{10}$ given by $f(\mathbf{x}) \triangleq -\sum_{i=1}^{10} \sin(x_i) \, \sin^{2m}(ix_i^2/\pi)$ where $\mathbf{x} \triangleq (x_i)_{i=1,\dots,10}$. It has one global minimum $f(\mathbf{x}^*) = -9.66015$.